

**B.E. COMPUTER SCIENCE AND ENGINEERING
DEGREE PROGRAMME**

**SCHEDULING OF COURSES
&**

CURRICULUM AND DETAILED SYLLABI

FOR

FIRST SEMESTER TO EIGHTH SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2025-26 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

Department of Computer Science and Engineering

Vision

Excellence in Computer Science and Engineering education and research.

Mission

1. Strive for academic excellence in Computer Science and Engineering through a creative teaching learning process.
2. Transform students into technically competent, socially responsible and ethical Computer Science professionals.
3. Create Centres of Excellence in leading areas of Computer Science and Engineering.
4. Incubate, apply and spread innovative ideas by collaborating with relevant industries and R&D labs through focused research groups.
5. Attain these through continuous team work by a group of committed faculty, transforming the Computer Science and Engineering department as a leader in imparting Computer Science and Engineering education and research.

Program Educational Objectives (PEOs) for B.E. (CSE) Programme

- PEO1:** Graduates will be able to perform in technical/managerial roles ranging from design, development, problem solving to production support in software industries and R&D sectors.
- PEO2:** Graduates will be able to successfully pursue higher education in reputed institutions.
- PEO3:** Graduates will have the ability to adapt, contribute and innovate new technologies and systems in the key domains of Computer Science and Engineering.
- PEO4:** Graduates will be ethically and socially responsible solution providers and entrepreneurs in Computer Science and other engineering disciplines.

Program Outcomes (POs) for B.E. (CSE) Programme

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs) for B.E.(CSE) Programme

PSO1: (Cognitive Outcome)

Ability to solve complex Knowledge Engineering problems by building systems across domains including Systems Engineering, Software Development & Engineering, Networks & Security, Data Mining and Artificial Intelligence.

PSO2: (Skill Outcome)

Ability to apply technical and research based skills learnt through professional society events, certification programs, projects and lab exercises to provide sustainable solutions to Computer Science and Engineering problems related to the society and environment.

PSO3: (Attitudinal and Behavioral Outcome)

Ability to practice as an ethical Software Engineer and/or Researcher in the evolving disciplines of Computer Science and Engineering and its allied application domains by employing soft and project management skills learnt through internships, project work and/or collaborative projects with industry.

Thiagarajar College of Engineering, Madurai - 625015
Credit Distribution for B.E (CSE) Programme – 2025 – 2026 Batch

S.No	Category of Courses	Credits
		Regular Admission
A	Foundation Courses(FC)	54 - 66
	Humanities and Social Sciences including Management Courses (HSMC)	09 - 12
	Basic Science Courses (BSC)	24 - 27
	Engineering Science Courses (ESC)	21 - 27
B	Professional Core Courses (PCC)	55
C	Professional Elective Courses (PEC)	24 - 39
	Programme Specific Electives (PSE)	15 – 24
	Programme Electives for Expanded Scope (PEES)	9 - 15
D	Open Elective Courses (OEC)	6 - 12
	Interdisciplinary Elective (IE)	3 - 6
	Basic Science Elective (BSE)	3 - 6
E	Project work	12
E	Internship and Mandatory Audit as per the Regulatory Authorities	Non Credit (Not included for CGPA)
	Minimum credits to be earned for the award of the Degree	160 (from A to E) and the successful completion of Mandatory Courses

Thiagarajar College of Engineering, Madurai-625015
Department of Computer science and Engineering
Scheduling of Courses – for those join in the year 2025 – 2026

Sem	Theory / Theory cum Practical / Practical								9	Audit Courses (Mandatory Non- credit}	Credit
	1	2	3	4	5	6	7	8			
I	22MA112 Calculus for Engineers (BS-4)	23PH121 Physics (BS-3)	22CH131 Chemistry (BS-3)	22EG141 Technical English (HSS-2)	22CS151 Engineering Exploration (ES-2)	22CS162 Foundations of Computer Programming (PC -3)	22EG171 English Laboratory (HSS-1)	22PH182 Physics Laboratory (BS-1)	22CH191 Chemistry Laboratory (BS-1)		20
II	22CS211 Matrices and Linear Algebra (BS-4)	22CS221 Electronics and Digital Systems (PC -3)	22CS231 Computer Organization and Architecture (PC-3)	22CS241 Problem Solving and Programming (ES-3)	22CS251 System Programming (ES- 2)	24CS261 Extended Reality: Principles and Practice (ES-3)	22CS271 Computer Systems Lab (ES-1)	22CS281 Programming Lab (PC-1)		22CHAA1 Environmental Science	20
III	22CS311 Probability and Statistics (BS-4)	22CS321 Theory and Design of Programming Languages (ES-3)	22CS331 Object Oriented Programming (PC-3)	22CS342 Data Structures and Algorithms (PC-3)		22CS362 Operating Systems (PC-4)	22CS371 Data Structures Lab (PC-1)	22CS381 Object Oriented Programming Lab (PC-1)	22ES391 Design Thinking (ES-3)		22
IV	22CS411 Discrete Mathematics (BS-4)	22CS421 Design and Analysis of Algorithms (PC-3)	22CS431 Data Communication and Networks (PC-3)	22CS441 Database Management Systems (PC-3)	22CS451 Web Programming (PC-3)	22EG661 Professional Communication (HSS-2)	22CS471 Databases Lab (PC-1)	22CS481 Algorithms Lab (PC-1)	22CS491 Project Management (HSS-3)	23CHAD1 Indian Constitution and Knowledge Systems	23
V	22CS511 Modelling and Optimization (ES-3)	22CS521 Theory of Computation (ES-3)	22CS531 Artificial Intelligence (PC-3)	22CSPX1 Programme Elective (PE - 3)	22XXGX1 Interdisciplinary Elective (OE-3)	22CS562 Software Engineering (PC-4)	22CS571 Network Programming Lab (PC-1)	22CS581 Artificial Intelligence Lab (PC-1)	22CS591 Project – I (P-3)	23CHAE1 Universal Human Values and Ethics	24
VI	22CS611 Cryptography and Network Security (PC-3)	22CS621 Distributed Computing (PC-3)	22CS632 Compiler Design (PC-3)	22CSPX1 Programme Elective (PE - 3)	22XXFX1 Basic Science Elective (OE-3)	22CSPX1 Programme Elective (PE - 3)	22CS671 Distributed Computing Lab (PC-1)		22CS691 Project - II (P-3)		22
VII	22CS711 Engineering Economics (HSS-3)	22CS721 Human Computer Interaction (ES-2)	22CSPX1 Programme Elective (PE - 3)	22CSPX1 Programme Elective (PE - 3)	22CSPX1 Programme Elective (PE - 3)	22CSPX1 Programme Elective (PE - 3)			22CS791 Project - III (P-3)		20
VIII	22CSPX1 Programme Elective (PE - 3)	22CSPX1 Programme Elective (PE - 3)							22CS891 Project - IV (P-3)		9

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI- 625 015
(A Govt. Aided, Autonomous Institution affiliated to Anna University)
Department of Computer Science and Engineering
Categorization of Courses

List of Humanities and Social Sciences including Management Courses (9-12)

- Technical English (2)
- English Laboratory (1)
- Professional Communication (2)
- Project Management (3)
- Engineering Economics (3)

List of Basic Science Courses (24-27)

- Calculus for Engineers (4)
- Matrices and Linear Algebra (4)
- Probability and Statistics (4)
- Discrete Mathematics (4)
- Physics (3)
- Chemistry (3)
- Physics Laboratory (1)
- Chemistry Laboratory (1)

List of Engineering Science Courses (21-27)

- Engineering Exploration (2)
- Problem Solving and Programming (3)
- System Programming (2)
- Extended Reality: Principles and Practice (3)
- Computer Systems Lab (1)
- Theory and Design of Programming Languages (3)
- Design Thinking (3)
- Modelling and Optimization (3)
- Theory of Computation (3)
- Human Computer Interaction (2)

List of Core Courses (55)

- Foundations of Computer Programming (3)
- Electronics and Digital Systems (3)
- Computer Organization and Architecture (3)
- Programming Lab (1)
- Object Oriented Programming (3)
- Object Oriented Programming Lab (1)
- Data Structures and Algorithms (3)
- Data Structures Lab (1)
- Operating Systems (4)
- Design and Analysis of Algorithms (3)
- Algorithms Lab (1)
- Database Management Systems (3)
- Databases Lab (1)
- Data Communication and Networks (3)

- Web Programming (3)
- Network Programming Lab (1)
- Artificial Intelligence (3)
- Artificial Intelligence Lab (1)
- Software Engineering (4)
- Cryptography and Network Security (3)
- Distributed Computing (3)
- Compiler Design (3)
- Distributed Computing Lab (1)

Programme Elective Courses (24 – 39)

- Programme Electives (27)

Open Elective Courses (OEC): (6 – 12)

- Interdisciplinary Elective (3)
- Basic Science Elective (3)

Project (12)

- Project – I (3)
- Project - II (3)
- Project – III (3)
- Project – IV (3)

**B.E. COMPUTER SCIENCE AND ENGINEERING
DEGREE PROGRAMME**

CURRICULUM AND DETAILED SYLLABI

FOR

FIRST SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2025-26 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programmes

COURSES OF STUDY

(For the candidates admitted from 2025-26 onwards)

FIRST SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22MA112	Calculus for Engineers	BSC	3	1	-	4
23PH121	Physics	BSC	3	-	-	3
22CS251	System Programming	ESC	2	-	-	2
22EG141	Technical English	HSMC	2	-	-	2
22CS151	Engineering Exploration	ESC	1	1	-	2
THEORY CUM PRACTICAL						
22CS162	Foundations of Computer Programming	PC	2	-	2	3
PRACTICAL						
22EG171	English Laboratory	HSMC	-	-	2	1
22PH182	Physics Laboratory	BSC	-	-	2	1
AUDIT COURSES						
22CHAA1	Environmental Science	AC				-
22TAAA1	Heritage of Tamils(Tamizhar Marabhu)					-
Total			13	2	6	18

BSC : Basic Science Courses

HSMC : Humanities and Social Sciences including Management Courses

ESC : Engineering Science Courses

L : Lecture

T : Tutorial

P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit

1 Hour Tutorial is equivalent to 1 credit

2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programme

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2025-26 onwards)

FIRST SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22MA112	Calculus for Engineers	3	40	60	100	27	50
2	23PH121	Physics	3	40	60	100	27	50
3	22CS251	System Programming	3	40	60	100	27	50
4	22EG141	Technical English	3	40	60	100	27	50
5	22CS151	Engineering Exploration	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	22CS162	Foundations of Computer Programming	3 (Terminal Exam Type : Theory)	50	50	100	25	50
PRACTICAL								
7	22EG171	English Laboratory	3	60	40	100	18	50
8	22PH182	Physics Laboratory	3	60	40	100	18	50

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks

22CS151	ENGINEERING EXPLORATION	Category	L	T	P	Credit
		ESC	1	1	-	2

Preamble

The course Engineering Exploration provides an introduction to the engineering field. It is designed to help the student to learn about engineering and how it affects our everyday lives. On the successful completion of the course, students will be able to explain how engineering is different from science and technology and how science, mathematics and technology are an integral part of engineering design.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency	Expected Attainment
CO1	Explain technological & engineering development, change and impacts of engineering	TPS2	70	70
CO2	Draw a product in enough detail that others can accurately build it and write specification sheet for a given product	TPS3	70	70
CO3	Complete initial steps (Define a problem, list criteria and constraints, brainstorm potential solutions and document the ideas) in engineering design process	TPS3	70	70
CO4	Draw sketches to a design problem and provide a trade-off matrix	TPS3	70	70
CO5	Communicate possible solutions through drawings, modelling and Testing Final output	TPS3	70	70
CO6	Apply the concept of engineering fundamentals in Electrical and Computer Engineering	TPS3	70	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-	-	-	-	-	-	-	-	-	-	L	-	-
CO2	S	M	L	-	-	-	-	-	-	-	-	-	M	-	-
CO3	S	M	L	-	-	-	-	-	-	-	-	-	M	-	-
CO4	S	M	L	-	-	-	-	-	-	-	-	-	M	-	-
CO5	S	M	L	-	-	-	-	-	-	-	-	-	M	-	-

C0 6	S	M	L	-	-	-	-	-	-	-	-	-	M	-	-
---------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Worksheet-1			Worksheet-2			CAT			Terminal (Theory)
TPS Scale	1	2	3	1	2	3	1	2	3	
CO1	5	20							5	10
CO2		5	20						5	10
CO3				5	20				10	20
CO4					5	20			10	20
CO5									10	20
CO6									10	20

Syllabus

What is Engineering: Engineering Requirement, Knowledge within Engineering disciplines, Engineering advancements **Engineering Design:** Problem definition, idea generation through brainstorming and researching, solution creation through evaluating and communicating, text/analysis, final solution and design improvement. **Defining problems and Brainstorming:** Researching design, sketching problem solving **Communicating solution:** Dimensioning orthographic drawing, perspective drawing **Modeling and Testing final output:** Product evaluation, reverse engineering, final project report. **Electrical Engineering:** Reading analog multimeter, measuring current, voltage and resistance, electricity from chemicals, solar cells, magnets, Ohms law and watts law, circuit identification and circuit calculation, resistor color code, continuity **Computer Engineering:** Logic gates, algorithms, computer architecture, binary code

Reference Books

1. Ryan A.Brown, Joshua W.Brown and Michael Berkihiser: "Engineering Fundamentals: Design, Principles, and Careers", Goodheart-Willcox Publisher, Second Edition, 2014.
2. Saeed Moaveni, "Engineering Fundamentals: An Introduction to Engineering", Cengage learning, Fourth Edition, 2011.

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1.	What is Engineering	
1.1	Engineering Requirement	1
1.2	Knowledge within Engineering disciplines,	1
1.3	Engineering advancements	1
2	Engineering Design	
2.1	Problem definition,	1
2.2	idea generation through brainstorming and researching	1
2.3	solution creation through evaluating and communicating,	1
2.4	text/analysis	1
2.5	final solution and design improvement	1
3	Defining problems and Brainstorming:	
3.1	Researching design	1
3.2	sketching problem solving	2

No.	Topic	No. of Lectures
4	Communicating solution	
4.1	Dimensioning orthographic drawing	1
4.2	perspective drawing	1
5	Modeling and Testing final output	
5.1	Product evaluation	1
5.2	reverse engineering	1
5.3	final project report	1
6.	Electrical Engineering:	
6.1	Reading analog multimeter, measuring current, voltage and resistance	1
6.2	electricity from chemicals, solar cells, magnets	1
6.3	Ohms law and watts law, circuit identification and circuit calculation	2
6.4	resistor color code, continuity	1
7	Computer Engineering	
7.1	Logic gates, algorithms,	1
7.2	computer architecture,	1
7.3	binary code	1
	Total	24

Course Designers:

1. Mr. M.Sivakumar mskcse@tce.edu

22CS162**FOUNDATIONS OF COMPUTER
PROGRAMMING**

Category	L	T	P	Credit
PC	2	-	2	3

Terminal Exam: Theory**Preamble**

The course on computer programming is intended to introduce the students to computational thinking, Python programming and constructs of programming. The course aims to provide exposure to solve the problems through Python programming. It aims to train the student to the fundamental concepts of the Python programming language. This course has a lab component which is designed to give the student hands-on experience with the concepts.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO. No	Course Outcomes (COs)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Practice the basics of Procedural programming using Python	TPS3	B	85
CO2	Solve the problems using basic data types, operators and expressions.	TPS3	B	85
CO3	Appreciate the usage of collections, Array techniques in Python to solve problems.	TPS3	B	85
CO4	Make use of appropriate control structure to solve engineering problems.	TPS3	B	85
CO5	Demonstrate the Python functions, scoping, recursion and functions as modules to develop simple applications	TPS3	B	85
CO6	Take part in real world problem solving using Python programming.	TPS4	B	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	S	M	L	L	L			L	L	L	L	L	M	L	L
CO2.	S	M	L	L	L			L	L	L	L	L	M	L	L
CO3	S	M	L	L	L			L	L	L	L	L	M	L	L
CO4	S	M	L	L	L			L	L	L	L	L	M	L	L
CO5.	S	M	L	L	L			L	L	L	L	L	M	L	L
CO6	S	S	M	L	L	L	L	M	M	M	L	L	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

C O	CAT 1						CAT 2						OCR						Model Examination						Terminal (Theory Component)					
	100						100						100						100						100					
T P S S c a l e	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
C O 1	5	1	1										1						5	1					3	5	7			
C O 2	5	1	2										1						5	1					2	5	8			
C O 3	5	1	2										1						1	1					5	5	1			
C O 4							5	1	1				2						5	1					3	5	7			
C O 5							5	1	2				2						5	1					5	5	1			
C O 6							5	1	2				1						1	1					2	5	8			

Syllabus

Procedural Programming and Python Programming - Fundamentals of Computing-Problems-Pseudo-code and flowcharts- Memory and Programs-Python Programming Language –Alternate Implementations-Highlights- Setting Up Python Programming Environment- Python on Different Operating Systems -Creating and Running Python Program

Variables and simple Data Types- Identifiers and Keywords – Strings- Numbers-Comments- User Input-Operators and expressions- operator precedence – Boolean expression – expression evaluation

Collection Data Types and Array Techniques – Lists- Numerical List-Tuples - Sets and Frozen Sets -Mapping Types - Dictionaries - Changing, Adding, Removing, Slicing Elements-Avoiding Errors in collections- **Array Techniques** -Counting, reversal, partitioning and removal of duplicates in an array

Control Structures - Conditional Tests- if Statements Looping- for and while -Nesting-Control Statements – break, continue and pass- Iterating and Copying Collections

Functions and Modules -Defining a Function- Function Arguments-Return Values- Passing an Arbitrary Number of Arguments- scoping – specifications – recursion – Fibonacci numbers – palindromes – global Variables- Modules and Packages- Overview of Python's Standard Library

File Handling -Classes and objects -Projects - Writing and Reading Binary Data - Writing and Parsing Text Files- Writing and Parsing XML Files- Random Access Binary Files-Classes and objects -Projects

Text Books

1. Eric Matthes , “Python Crash Course A Hands-On , Project - Based Introduction to Programming”, No Starch Press, Inc. 2nd Edition,2019
2. Mark Summerfield, “Programming in Python 3: A Complete Introduction to the Python Language”, Pearson Education, 2nd Edition, 2018

Reference Books& web resources

1. Mark Lutz, “Python Pocket Reference”, O Reilly, Fifth Edition, 2014
2. Zed Shaw, “Learn Python the Hard Way” Addison-Wesley, 2013.
3. Martin C. Brown, “Python: The Complete Reference “,McGraw Hill,2018.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	Procedural Programming and Python Programming		
1.1	Fundamentals of Computing- Problems-Pseudo-code and flowcharts- Memory and Programs-	1	CO1
1.2	Python Programming Language –Alternate Implementations-Highlights	1	CO1
1.3	Setting Up Python Programming Environment-Python on Different Operating Systems -Creating and Running Python Program	1	CO1
2	Variables and simple Data Types		
2.1	Identifiers and Keywords – Strings- Numbers-Comments	1	CO2
2.2	User Input-Operators and expressions- operator precedence	1	CO2
2.3	Boolean expression – expression evaluation	1	CO2
3	Collection Data Types and Array Techniques		
3.1	Lists- Numerical List-Tuples - Sets and Frozen Sets	2	CO3
3.2	Mapping Types - Dictionaries - Changing, Adding, Removing, Slicing Elements-Avoiding Errors in collections	1	CO3
3.3	Array Techniques -Counting, reversal, partitioning and removal of duplicates in an array	1	CO3
4	Control Structures		
4.1	Conditional Tests- if Statements Looping- for and while -Nesting-	1	CO4
4.2	Control Statements – break, continue and pass-	1	CO4
4.3	Iterating and Copying Collections		
5	Functions and Modules		
5.1	Defining a Function- Function Arguments-Return Values- Passing an Arbitrary Number of Arguments	1	CO5
5.2	Scoping – abstractions – Recursion – Fibonacci numbers – palindromes	1	CO5

Module No.	Topic	No. of Lectures	Course Outcome
5.3	Modules and Packages- Overview of Python's Standard Library- global Variables	1	CO5
6	File Handling -Classes and objects -Projects		
6.1	Writing and Reading Binary Data - Writing and Parsing Text Files- Writing and Parsing XML Files- Random Access Binary Files-	2	CO6
6.2	Classes and objects -Projects	1	CO6
	Total Hours	24	

Course Contents and Lecture Schedule for Laboratory

Module No.	Topic	No. of Lectures	Course Outcome
1.	Write a Python program using basic datatypes	2	CO1
2.	Write a Python program to display multiple variables using operators and expressions	2	CO1
3.	Develop a Python Program for number conversions	2	CO2
4.	Demonstrate a Python Program for String manipulations	2	CO2
5.	Implement all the collections in Python	2	CO3
6.	Write a Python program to perform Array Techniques	2	CO3
7.	Write a Python Program using different types of function call to find Prime number, generate Fibonacci series, sorting and searching etc.	2	CO4
8.	Create a Python module for recursive functions.	2	CO4
9.	Write a program in Python for various file handling	2	CO5
10.	Write a Python Program for implementing classes and objects	2	CO5
11.	Mini project	4	CO1, CO2, CO3, CO4, CO5, CO6
	Total Hours	24	

Course Designers:

- | | | |
|----|-------------------|--|
| 1. | Dr.M.P.Ramkumar | ramkumar@tce.edu |
| 2. | Dr.M.Nirmala Devi | mnit@tce.edu |
| 3. | Dr.S.Prasanna | spcse@tce.edu |

22CS251 SYSTEM PROGRAMMING

Category L T P Credit

ESC 2 0 0 2

Preamble

This course teaches students how to think about, build, debug, and test computer programs. The course also emphasizes the understanding of how programs execute on today's computers and their interaction with hardware and software. The practical programming is in C on Unix/Linux systems to introduce students to the mapping of high-level language constructs to the system software components and the underlying machine.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the role and types of components significant in computing systems. (CO1)	TPS2	B	85
CO2	Identify and utilize the proper system software component for computer program execution.(CO2)	TPS3	B	80
CO3	Perform Unix/Linux basic commands to facilitate user interaction with the Operating System.(CO3)	TPS3	B	80
CO4	Illustrate the system programming tools and resources that assists in program development.(CO4)	TPS2	B	85
CO5	Develop C programs using Process Management and File Management System calls.(CO5)	TPS3	B	80
CO6	Construct bash scripts to solve problems using control statements and functions.(CO6)	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	M	L											L		
CO2.	M	L			L							L	L		
CO3	M	L			M					L		L	L	L	
CO4	M	L											L		

CO5	M	L			L					L		L	L		
CO6	S	M	L		M					M		L	M	L	

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Assignment 1	CAT2						Assignment 2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	10	10	-	-	-	-	20	-	-	-	-	-	-	-	3	10	-	-	-	-
CO2	10	10	20	-	-	-	30	-	-	-	-	-	-	-	3	5	6	-	-	-
CO3	10	10	20	-	-	-	50	-	-	-	-	-	-	-	3	5	12	-	-	-
CO4	-	-	-	-	-	-	-	5	15	-	-	-	-	20	3	10	-	-	-	-
CO5	-	-	-	-	-	-	-	5	10	25	-	-	-	40	3	5	12	-	-	-
CO6	-	-	-	-	-	-	-	5	10	25	-	-	-	40	3	5	12	-	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Fundamentals of Computer System: Functional components of a programming system: Machine Structure, Hardware and Software, Software: Application Software and System Software.

System Software: Operating System, Text Editor, Debugger, Language Translator, Linker and Loader, Programming Languages: Machine Language, Assembly Language and High level language, Programming language processing activities.

System Programming: Basics of Systems Programming, Tools of system programming: Shell, Editor, Debugger, IDE, Program Development and Management: Case Study “C” programming, Basics of C programming and C program execution model.

System Programming on UNIX/Linux Systems: Unix/Linux Operating System structure, Basic Unix/Linux commands, Input / Output – Streams, Buffers, Pipes, Files and Devices, Processes, Privileged instructions and Non-privileged instructions, Environments, Shells, Kernel System Resources, System Calls and System Libraries.

System Calls & Libraries: Basics of System Call and Operation of system calls, Process management system calls - fork(), exec(), wait() & exit(), File management system calls - fopen(), fclose(), fread(), fwrite() and fseek(), Libraries: purpose of libraries and example of C standard libraries.

Scripting Language: Basics of scripting languages, Shell scripting: Basic Shell commands, Variables, Conditionals, Loops and Functions.

Text Books

1. System programming with C and Unix, Adam Hoover, Addison-Wesley, 2010 Pearson Education.
2. Practical System Programming with C: Pragmatic Example Applications in Linux and Unix-Based Operating Systems, Sri Manikanta Palakollu, Apress Media.
3. Systems Programming, John. J.Donovan, Tata McGraw Hill Edition.
4. System Software – An Introduction to Systems Programming, Leland L. Beck, Third Edition, Addison-Wesley.
5. Systems Programming, DM Dhamdhare, McGraw-Hill Education (India) Pvt Limited, 2011.

Reference Books & web resources

1. System Software, Santanu Chattopadhyay, 2007, PHI Learning.
2. The Linux Programming Interface - A Linux and UNIX System Programming Handbook, Michael Kerrisk, 2010, No Starch Press.
3. <https://script.spoken-tutorial.org/index.php/BASH/C2/Introduction-to-BASH-Shell-Scripting/English-timed>
4. https://onlinecourses.swayam2.ac.in/aic20_sp05/preview

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Fundamentals of Computer System	
1.1	Functional components of a programming system: Machine Structure, Hardware and Software.	1
1.2	Software: Application Software and System Software.	1
2	System Software	
2.1	Operating System and Text Editor	1
2.2	Programming Languages: Machine Language, Assembly Language and High level language.	1
2.3	Debugger, Language Translator, Linker and Loader.	2
2.4	Programming language processing activities.	1
3	System Programming	
3.1	Basics of Systems Programming and Tools of system programming: Shell, Editor, Debugger, IDE.	1
3.2	Program Development and Management: Case Study "C" programming and Basics of C programming, C program execution model	2
4	System Programming on UNIX/Linux Systems	
4.1	Unix/Linux Operating System structure and Basic Unix/Linux commands.	1
4.2	Input / Output – Streams, Buffers, Pipes, Files and Devices.	1
4.3	Processes, Privileged instructions and Non-privileged instructions, Environments and Shells.	1
4.4	Kernel System Resources, System Calls and System Libraries.	1
5	System Calls & Libraries	
5.1	Basics of System Call and Operation of system calls.	1
5.2	Process management system calls - fork(), exec(), wait() and exit().	2

Module No.	Topic	No. of Periods
5.3	File management system calls - fopen(), fclose(), fread(), fwrite() and fseek().	2
5.4	Libraries: purpose of libraries and example of C standard libraries.	1
6	Scripting Language	
6.1	Basics of scripting languages and types of shells	2
6.2	Shell scripting: Basic Shell commands, Variables and Conditional control statements.	
6.3	Shell scripting using Loops and Functions.	2
	Total	24

Course Designer(s):

1. Dr.S.MercyShalinie, Professor,CSE shalinie@tce.edu
2. Dr.G.Madhupriya, Associate Professor,CSE gmadhupriya@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

SECOND SEMESTER

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programmes

COURSES OF STUDY

(For the candidates admitted from 2025 - 26 onwards)

SECOND SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CS211	Matrices and Linear Algebra	BSC	3	1	-	4
22CS221	Electronics and Digital Systems	PC	2	1	-	3
22CS231	Computer Organization and Architecture	PC	3	-	-	3
22CS241	Problem Solving and Programming	ESC	3	-	-	3
22CH131	Chemistry	BSC	3	-	-	3
THEORY CUM PRACTICAL						
24CS261	Extended Reality: Principles and Practice	ESC	2	-	2	3
PRACTICAL						
22CS271	Computer Systems Lab	ESC	-	-	2	1
22CS281	Programming Lab	PC	-	-	2	1
22CH191	Chemistry Laboratory	BSC	-	-	2	1
AUDIT COURSE						
22TAAB1	Tamils and Technology	AC				-
Total			16	2	8	22

BSC : Basic Science Courses

PC : Professional Core Courses

ESC : Engineering Science Courses

AC : Audit Course

L : Lecture

T : Tutorial

P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit

1 Hour Tutorial is equivalent to 1 credit

2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programme

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2025-26 onwards)

SECOND SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CS211	Matrices and Linear Algebra	3	40	60	100	27	50
2	22CS221	Electronics and Digital Systems	3	40	60	100	27	50
3	22CS231	Computer Organization and Architecture	3	40	60	100	27	50
4	22CS241	Problem Solving and Programming	3	40	60	100	27	50
5	22CH131	Chemistry	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	24CS261	Extended Reality: Principles and Practice (Terminal Exam: Theory)	3	50	50	100	22.5	50
PRACTICAL								
7	22CS271	Computer Systems Lab	3	60	40	100	18	50
8	22CS281	Programming Lab	3	60	40	100	18	50
9	22CH191	Chemistry Laboratory	3	60	40	100	18	50

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

22CS211	MATRICES AND LINEAR ALGEBRA	Category	L	T	P	Credit
		BSC	3	1	0	4

Preamble

This course introduces the concept of vector space, which is a unifying abstract framework for studying linear operations involving a diverse set of mathematical objects such as vector space, matrices, and functions. Students learn to use the concepts of basis and linear transformations to operate within and between vector spaces. They can perform approximations and orthogonal projections and construct an orthonormal basis for vector spaces. Moreover, this course demonstrates procedure to solve linear and non-linear system of equations.

Prerequisite

- Nil

Course Outcomes

On the successful completion of the course, students will be able to

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Evaluate basis and dimension of a vector space	TPS3	B	60
CO2	Compute the orthonormal basis of an inner product space given a basis	TPS3	B	60
CO3	Represent linear transformations by a matrix and hence verify their similarity	TPS3	B	60
CO4	Compute the Eigen value of a matrix and hence its singular value decomposition and generalised inverse	TPS3	B	60
CO5	Solve linear system of equations using matrices.	TPS3	B	60
CO6	Compute the solution of non-linear systems.	TPS3	B	60

Mapping with Programme Outcomes

COs	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L		M			L	M	L		M	M	L	L
CO2.	S	M	L		M			L	M	L		M	M	L	L
CO3.	S	M	L		M			L	M	L		M	M	L	L
CO4	S	M	L		M			L	M	L		M	M	L	L
CO5	S	M	L		M			L	M	L		M	M	L	L
CO6	S	M	L		M			L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assessment 1						Assessment 2						Terminal (%)		
	CAT 1 (%)			Assignment 1 (%)			CAT 2 (%)			Assignment 2 (%)					
TPS	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70
CO1	5	15	30	-	-	50	-	-	-	-	-	-		12	14
CO2	5	15	30	-	-	50	-	-	-	-	-	-		9	14
CO3	-	-	-	-	-	-	3	10	14	-	-	26		-	14
CO4	-	-	-	-	-	-	7	10	22	-	-	42		9	14
CO5	-	-	-	-	-	-		5	12	-	-	16		-	7
CO6	-	-	-	-	-	-		5	12	-	-	16		-	7
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Vector Spaces: Vector space - Definition and Examples, Subspaces, Linearly Independence, Basis and Dimension, Row space and Column spaces.

Orthogonality: The scalar Product in $\mathbf{R}^n$ - Orthogonal Subspaces- Inner Product spaces, Orthonormal sets - Gram-Schmidt process.

Linear Transformations: Linear Transformations-Definitions and Examples - Matrix Representations of Linear Transformations -Similarity.

Eigen systems: Eigen values of Matrices and Eigen vectors of Matrices–Diagonalization of Matrices – The Singular Value Decomposition (SVD) of Matrices –The eigen value problem of Matrices – Generalized inverse of Matrices.

Linear Systems: System of linear Equations-Row Echelon forms - Gauss Elimination. **Nonlinear systems:** System of Non-Linear Equations - Fixed point iteration- Newton Raphson Method - Nonlinear system of Equations-Gauss Jordan method.

Text Books

1. Steven J. Leon., "Linear Algebra with Application" Ninth Edition, Pearson, 2015.

2. Steven C. Chapra & Raymond P. Canale., Numerical Methods for Engineers. 7th Edition, McGraw Hill Publications. 2015.

Reference Books & web resources

1. David C. Lay., "Linear Algebra And Its Applications" 5th Edition, 2015.
2. Gilbert Strang., "Introduction to Linear Algebra" 5th Edition, 2016.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	VECTOR SPACES	
1.1	Vector space-Definition and Examples	1
1.2	Subspaces	2
	Tutorial	1
1.3	Linearly Independence	2
1.4	Basis and Dimension	2
	Tutorial	1
1.5	Row space and Column space	2
	Tutorial	1
	Program Using Python / MATLAB	
2	ORTHOGONALITY	
2.1	The scalar Product in $\mathbf{R}^n$	1
2.2	Orthogonal Subspaces	2
	Tutorial	1
2.3	Inner Product spaces	2
2.4	Orthonormal sets	2
	Tutorial	1
2.5	Gram-Schmidt process	2
	Tutorial	1
	Program Using Python / MATLAB	
3	LINEAR TRANSFORMATION	
3.1	Linear Transformation- Definition and Examples	1
3.2	Matrix Representations of Linear Transformations	2
	Tutorial	1
3.3	Similarity	1
	Tutorial	1
	Program Using Python / MATLAB	
4	EIGEN SYSTEMS	
4.1	Eigenvalues of Matrices and Eigen vectors of Matrices	1
4.2	Diagonalization of Matrices	2
	Tutorial	1
4.3	The Single Value Decomposition (SVD) of Matrices	2
4.4	The eigen value problem of Matrices	2
4.5	Generalized Inverse of Matrices	1
	Tutorial	1
	Program Using Python / Matlab	

5	LINEAR SYSTEMS	
5.1	System of linear equations	1
5.2	Row Echelon forms	1
5.3	Gauss Elimination	1
	Tutorial	1
	NONLINEAR SYSTEMS	
5.4	System of Non-Linear Equations, Fixed point iteration, Newton Raphson Method	1
5.5	Nonlinear system of Equations	1
5.6	Gauss Jordan Method	1
	Tutorial	1
	Program Using Python / Matlab	
	Total	48

Course Designer(s):

1. Dr.C.S.Senthilkumar kumarstays@tce.edu
2. Dr. P. Krishnapriya pkamat@tce.edu
3. Dr. P. Victor pvmat@tce.edu

22CS221**ELECTRONICS AND
DIGITAL SYSTEMS**

Category	L	T	P	Credit
PC	2	1	0	3

Preamble

This course is designed to enable the students to understand and apply the basic principles of analog operational amplifiers and digital circuits and systems. Operational amplifier principles and characteristics, their different configurations and applications are illustrated. Then introduction to number systems, binary arithmetic, Boolean algebra, digital logic gates, design and implementation of combinational logic circuits are discussed. HDL simulation of combinational logic is briefly introduced. The fundamental concepts of flip flops and their application in implementing sequential logic circuits are also illustrated. Principles of Mealy and Moore type of circuits and their design are briefly discussed.

Prerequisite

- 22CS151 - Engineering Exploration

Course Outcomes

On the successful completion of the course, students will be able to

CO		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the operation and characteristics of an operational amplifier (Understand)	TPS2	B	75
CO2	Design and implement linear and nonlinear circuits using operational amplifiers. (Apply)	TPS 3	B	70
CO3	Perform simplification of boolean logic functions by applying the theorems and postulates of Boolean algebra and Karnaugh map. (Apply)	TPS 3	B	70
CO4	Design combinational logic circuits for given specifications and implement them using logic gates, multiplexers and programmable logic devices. (Apply)	TPS 3	B	70
CO5	Explain the HDL simulation of combinational logic circuits. (Understand)	TPS 2	B	75
CO6	Design sequential logic circuits like counters and sequence detectors and implement them using given flip flops. (Apply)	TPS 3	B	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1.	M	L											L		
CO2.	S	M	L					L	M	M		L	M		L

CO3	S	M	L					L	M	M		L	M		L
CO4	S	M	L					L	M	M		L	M		L
CO5	M	L			L				L	L		L	L		L
CO6	S	M	L					L	M	M		L	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1/2						Terminal		
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3
CO1	6	10																	3	5	
CO2		6	24												34					3	12
CO3	4	4	46												66				2	2	22
CO4							2	4	28						40				1	2	14
CO5							6	10											3	5	
CO6							2	6	42						60				1	3	22

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Assignment 1 covers CO2 and CO3 only.

Assignment 2 covers CO4 and CO6 only.

Syllabus

Operational Amplifier (Op-Amp) Principles: Internal structure of an op-amp- Ideal op-amp characteristics and non-idealities in a practical op-amp- Idealized analysis of op-amp circuits.

Applications of Op-Amp: Inverting, non-inverting and differential amplifiers -Integrator, differentiator and comparator - Waveform generator -Zero crossing detector, peak detector and precision rectifier

Binary Number System and Boolean Algebra: Binary Number system and its conversion to hexadecimal number System-Complements, Signed Binary Numbers and Arithmetic-Binary Coded Decimal (BCD) and decimal Addition using BCD-Theorems and Properties of Boolean Algebra-Digital Logic Gates and Logic Operations-Simplification of logic functions using Karnaugh Map Method.

Combinational Logic Circuits: Design and Implementation

Binary Adder, Subtractor and Multiplier - Magnitude comparator, decoder and encoder-Multiplexers and their applications in implementing combinational logic functions -Organization of ROM, PLA and PAL and their application in implementing combinational logic circuits.

Simulation of Combinational Logic Circuits using HDL: Introduction to Hardware Description Language - Verilog model of simple combinational Circuits-Test benches, Boolean expressions and user defined primitives.

Sequential Logic Circuits: Introduction- Types of flip flops, their truth diagrams and conversions between them-Operation of shift Registers-Design of synchronous and ripple counters-. Principles and operation of Mealy and Moore type Circuits-Design of Mealy and Moore type simple sequence detectors.

Text Books

1. Ramakant A. Gayakwad, Op-Amps and Linear Integrated Circuits, Pearson Education; Fourth Edition, 2015.
2. M. Morris Mano, Micheal D. Ciletti, Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog, Pearson Education; Sixth edition, 2018.

Reference Books & web resources

1. R. P. Jain, Kishor Sarawadekar, Modern Digital Electronics, McGraw Hill; Standard Edition, 2022.
2. Leach, Malvino, Saha, Digital Principles and Applications, McGraw Hill Education; Eighth edition, 2014.
3. Jacob Millman, Christos Halkias, Chetan D. Parikh, Millman's Integrated Electronics- Analog and Digital Circuit and Systems, McGraw Hill Education, 2nd edition, 2017.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Operational Amplifier (Op-Amp) Principles:	
1.1	Internal structure of an op-amp.	1
1.2	Ideal op-amp characteristics and non-idealities in a practical op-amp.	1
1.3	Idealized analysis of op-amp circuits	1
2	Applications of Op-Amp	
2.1	Inverting, non-inverting and differential amplifier	1
2.2	Integrator, differentiator and comparator	1
2.3	Waveform generator	2
2.4	Zero crossing detector, peak detector and precision rectifier	1
3	Binary Number System and Boolean Algebra	
3.1	Binary Number system and its conversion to hexadecimal number system.	1
3.2	Complements, Signed Binary Numbers and arithmetic	2
3.3	Binary Coded Decimal (BCD) and decimal addition using BCD.	1
3.4	Theorems and Properties of Boolean Algebra	1
3.5	Digital Logic Gates and Logic Operations	1
3.6	Simplification of logic functions using Karnaugh Map Method	3

Module No.	Topic	No. of Periods
4	Combinational Logic Circuits: Design and Implementation	
4.1	Binary Adder, Subtractor and multiplier	1
4.2	Magnitude comparator, decoder and encoder	1
4.3	Multiplexers and their applications in implementing logic functions.	2
4.4	Organization of ROM, PLA and PAL and their application in implementing combinational logic circuits.	2
5	Simulation of Combinational Logic Circuits using HDL	
5.1	Introduction to Hardware Description Language	1
5.2	Verilog model of a simple combinational circuits	2
5.3	Test benches, Boolean expressions and user defined primitives	1
6	Sequential Logic Circuits	
6.1	Introduction- Types of flip flops, their truth diagrams and conversions between them.	2
6.2	Operation of shift registers	1
6.3	Design of synchronous and ripple counters	2
6.4	.Principles and operation of Mealy and Moore type circuits	1
6.5	Design of Mealy and Moore type simple sequence detectors	3
	Total	36

Course Designer(s):

1. C.Sridharan, Associate Professor of Computer Science and Engineering,
cscse@tce.edu
2. Dr.C.Senthil Kumar, Associate Professor of Computer Science and Engineering,
cskcse@tce.edu

22CS231	COMPUTER ORGANIZATION AND ARCHITECTURE
----------------	---

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

The purpose of the course is to introduce principles of computer organization and the basic architectural concepts. It gives a brief overview of the memory format and instruction execution. It explains the function of each element of memory hierarchy and different ways of communication with I/O devices. It helps to estimate the performance and trade-offs involved in designing.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Estimate the data transfer rate and length of the instruction cycle with bus interconnection and timing diagrams.	TPS3	70	70
CO2	Identify the impacts on cost/performance by making use of memory hierarchy	TPS3	70	70
CO3	Design the cache memory organization and replacement algorithms and evaluate the design trade-offs	TPS3	70	70
CO4	Explain the features of I/O and DMA transfer	TPS2	70	75
CO5	Perform ALU operations on binary numbers	TPS3	70	70
CO6	Identify the methods to deal with hazards in pipelining	TPS3	70	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L					L	L	L		L	M		L
CO2	S	M	L					L	L	L		L	M		L
CO3	S	M	L					L	L	L		L	M		L
CO4	M	L											L		
CO5	S	M	L						L	M		L	M		L
CO6	S	M	L						L	M		L	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assign ment		Terminal					
													1	2						
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	3	3	1	2	3	4	5	6
CO1	10	10											20		10					
CO2	10	10	20										40			10				
CO3	10		30										40			10	20			
CO4							20	10						30	10	10				
CO5							10	10	15					30			20			
CO6							10	10	15					40			10			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Introduction to IAS Computer Structure and Operation.

Computer Function and Interconnection: Top level view of components and functions, Instruction cycle and program execution, Interrupts and instruction cycles, Multiple interrupts, Interconnection structures, Bus interconnection, Multiple buses, Synchronous and asynchronous bus timings, GPU Architecture - Overview.

Memory and I/O: Characteristics and hierarchy of memory, Cache memory principles and operation, Cache design and mapping functions, replacement algorithms, main memory, DRAM and SRAM, Types of ROMs, Module organization, Introduction to magnetic disks, I/O transfer and disk performance, Interrupt driven and DMA transfers, Performance estimation and trade-offs in design.

Computer Arithmetic: Arithmetic and Logic Unit, Integer multiplication of unsigned and signed numbers, Booth's algorithm, Division of unsigned binary, Floating point arithmetic. **Pipelining:** Basic Concepts, Data Hazards, Instruction hazards, Influence on Instruction Sets, Data path and Control Considerations, Superscalar Operation, Performance Considerations.

Learning Resources

1. William Stallings, Computer Organization and Architecture Designing for Performance, Eleventh edition, Prentice Hall, 2013.
2. Andrew S Tanenbaum and Todd Austin, Structured Computer Organization, Sixth edition, Pearson, 2013.
3. Carl Hamacher, Computer Organization and Embedded Systems, Sixth edition, McGrawHill, 2012.
4. DodiyaTripti, Computer Organisation and Advanced Microprocessors, First edition, Cengage Learning India, 2012.
5. Barry B.Brey, The Intel Microprocessors Architecture Programming and Interfacing, Eighth edition, Pearson Prentice Hall, 2009.
6. N.Senthil Kumar, M.Saravanan and S. Jeevananthan, Microprocessors and Microcontrollers, First edition, Oxford University Press, 2010.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	Introduction (2)		
1.1	Introduction to IAS Computer Structure and Operation	2	CO1

Module No.	Topic	No. of Lectures	Course Outcome
2	Computer Function and Interconnection (9)		
2.1	Top level view of components and functions	1	CO1
2.2	Instruction cycle and program execution	1	
2.3	Interrupts and instruction cycles	1	
2.4	Multiple interrupts	1	
2.5	Interconnection structures	1	
2.6	Bus interconnection	1	
2.7	Multiple buses	1	
2.8	Synchronous and asynchronous bus timings	1	
2.9	GPU Architecture - Overview	1	
3	Memory and I/O (11)		
3.1	Characteristics and hierarchy of memory	2	CO2
3.2	Cache memory principles and operation		
3.3	Cache design and mapping functions		
3.4	Replacement algorithms	1	CO3
3.5	Main memory	1	
3.6	DRAM and SRAM	1	
3.7	Types of ROMs,	1	
3.8	Module organization	1	CO4
3.9	Introduction to magnetic disks	1	
3.10	I/O transfer and disk performance	1	CO4
3.11	Interrupt driven and DMA transfers	1	
3.12	Performance estimation and trade-offs in design	1	
4	Computer Arithmetic (7)		
4.1	Arithmetic and Logic Unit	1	CO5
4.2	Integer multiplication of unsigned and signed numbers	1	
4.3	Booth's algorithm	2	
4.4	Division of unsigned binary	2	
4.5	Floating point arithmetic	1	
5	Pipelining (7)		
5.1	Basic Concepts	1	CO6
5.2	Data Hazards	1	
5.3	Instruction hazards	1	
5.4	Influence on Instruction Sets	1	
5.5	Data path and Control Considerations	1	
5.6	Superscalar Operation	1	
5.7	Performance Considerations	1	
	Total Hours	36	

Course Designer(s):

1. Dr. K. NarasimhaMallikarjunan, Associate Professor, arjunkambaraj@tce.edu
CSE
2. Ms. C.Santhiya ,Assistant Professor, CSE csit@tce.edu

22CS241	PROBLEM SOLVING AND PROGRAMMING	Category	L	T	P	Credits
		ESC	3	0	0	3

Preamble

The main objective of this course is to learn problem solving methodologies and aspects of C programming. Programming is a fundamental task in finding solutions to real life problems. Aim of this course is to train the students to solve problems using the fundamentals of C programming.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcome 1 (CO1)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Interpret Mathematical problems using algorithms, flowchart and pseudocode.	TPS 2	B	85
CO2	Develop algorithms for solving engineering problems using appropriate conditional and control structure	TPS 3	B	80
CO3	Decompose a problem into functions for recursive solutions	TPS 3	B	80
CO4	Use arrays, pointers and structures to formulate algorithms and programs	TPS 3	B	80
CO5	Solve matrix addition, matrix multiplication, searching and sorting problems using C Programming	TPS 3	B	80
CO6	Implement file operations and pre-processor directives for a given application	TPS 3	B	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	M	L	-	-	L	-	-	L	L	L	L	L	L	-	L
CO2.	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L

CO3	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO4	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO5	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO6	S	M	L	-	L	L	L	L	L	M	L	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						CAT 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10	20	-	-	-	-	-	-	-	-	-	5	5	10	-	-	-
CO2	-	10	20	-	-	-	-	-	-	-	-	-	-	5	10	-	-	-
CO3	-	10	20	-	-	-	-	-	-	-	-	-	-	5	10	-	-	-
CO4	-	-	-	-	-	-	10	10	20	-	-	-	5	5	10	-	-	-
CO5	-	-	-	-	-	-		10	20	-	-	-	-	5	10	-	-	-
CO6	-	-	-	-	-	-		10	20	-	-	-	-	5	10	-	-	-

CO	Assignment 1						Assignment 2					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1	-	-	30	-	-	-	-	-	-	-	-	-
CO2	-	-	30	-	-	-	-	-	-	-	-	-
CO3	-	-	40	-	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	30	-	-	-
CO5	-	-	-	-	-	-	-	-	30	-	-	-

CO6	-	-	-	-	-	-	-	-	40	-	-	-
-----	---	---	---	---	---	---	---	---	----	---	---	---

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Problem Solving through programs, Algorithm, and Flowchart for problem solving with Sequential Logic Structure, Decisions and Loops, compilation process, Syntax and Semantic Errors-Character set – Constants – Keywords – Primitive data types – Declaration, Type Conversion

Conditional Branching and Iterative Loops-Sequential- Arithmetic Operators, Relational Operators, Logical Operators, Increment Decrement Operators, Bitwise Operators, Assignment Operators and Expressions, Precedence and Order of Evaluation, selective - If-Else-If, Switch- repetitive structures-for, while, do while, Nested loops, go to, break, continue –Finding maximum of 3 numbers, Unit converters, Interest calculators, multiplication tables, GCD and LCM, Prime number generation.

Arrays – Declaration, initialization and accessing array elements - 2-D arrays, Character Arrays and String operations -Matrix Operations-Searching- Linear search, Binary Search and Sorting-Bubble sort and Selection Sort - Minimum, Maximum and average of n numbers

Functions: Definition – call – prototypes - block structure -Storage Classes- call by Value-Variable Length Arguments-Call by Reference-Passing Arrays to Functions –recursion-iteration vs recursion- types of recursion- Simple recursive and non-recursive programs, Factorial and Fibonacci Generation-Towers of Hanoi problem

Pointers – Address and indirection operators, Pointer type declaration – Pointer arithmetic-Functions and pointers – Arrays and pointers -Strings and Pointers – Pointer to Pointers-Dynamic memory management

Structures – Variables, Accessing members, Assignment and nesting – Pointers to Structures -Structures and functions – Array of Structures – Unions- Pre-processor Directives-Command line arguments

Files: -Input and Output: Standard I/O, Formatted Output – printf, Formatted Input – scanf- file access including FILE structure, fopen, stdin, stdout and stderr, Error Handling including exit, perror and error.h, Line I/O, related miscellaneous functions.

Text Books

1. R.G.Dromey, "How to solve it by Computers", Reprint, PHI Publishers, 2011.
2. Kernighan, B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Publisher, 2015.
3. J.R.Hanly and E.B. Koffmann, "Problem Solving and Program design in C", Fifth Edition, Pearson Publisher, 2012

Reference Books & web resources

1. YashwantKanetkar, "Let us C", 18th Edition, BPB Publications, 2021.
2. ReemaThareja, "Programming in C", Second Edition,Oxford University Press,2016.
3. Byron Gottfried, "Programming with C", Fourth Edition, Tata McGraw Hill Education, 2018
4. Paul Deital and Harvey Deital, "C How to Program", Seventh Edition, Prentice Hall, 2012.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Problem Solving through programs	
1.1	Algorithm, and Flowchart for problem solving with Sequential Logic Structure, Decisions and Loops	1
1.2	compilation process, Syntax and Semantic Errors-Character set – Constants – Keywords	1
1.3	Primitive data types – Declaration, Type Conversion	1
2	Conditional Branching and Iterative Loops	
2.1	Sequential- Arithmetic Operators, Relational Operators, Logical Operators, Increment Decrement Operators, Bitwise Operators, Assignment Operators and Expressions, Precedence and Order of Evaluation	2
2.2	Selective - If-Else-If, Switch	1
2.3	Repetitive structures-for, while, do while, Nested loops, go to, break, continue	1
2.4	Finding maximum of 3 numbers, Unit converters, Interest calculators, multiplication tables, GCD and LCM, Prime number generation.	1
3	Arrays	
3.1	Declaration, initialization and accessing array elements - 2-D arrays	1
3.2	Character Arrays and String operations	1
3.3	Matrix operations	1
3.4	Searching- Linear search, Binary Search	1
3.5	Sorting-Bubble sort and Selection Sort	1
3.6	Minimum, Maximum and average of n numbers	1
4	Functions	
4.1	Definition – call – prototypes - block structure -Storage Classes	1
4.2	call by Value- Variable Length Arguments-Call by Reference	1
4.3	Passing Arrays to Functions	1
4.4	Recursion-iteration vs recursion- types of recursion	1

4.5	Simple recursive and non-recursive programs, Factorial and Fibonacci Generation	1
4.6	Towers of Hanoi problem	1
5	Pointers	
5.1	Address and indirection operators, Pointer type declaration – Pointer arithmetic	1
5.2	Functions and pointers	1
5.3	Arrays and pointers	1
5.4	Strings and pointers	1
5.5	Pointer to pointers	1
5.6	Dynamic memory management	1
6	Structures	
6.1	Variables, Accessing members, Assignment and nesting	1
6.2	Pointers to Structures	1
6.3	Structures and functions	1
6.4	Array of Structures, Unions	1
6.5	Pre-Processor Directives-Command line arguments	1
7	Files	
7.1	Input and Output: Standard I/O, Formatted Output – printf, Formatted Input – scanf	2
7.2	Variable length argument list	1
7.3	File access including FILE structure, fopen, stdin, stdout and stderr	1
7.4	Error Handling including exit, perror and error.h, Line I/O, related miscellaneous functions	1
	Total	36

Course Designer(s):

1. Dr.M.Nirmala Devi, AP,CSE,TCE
2. Ms.G.Bhavani, AP,CSE,TCE

mnit@tce.edu
gbicse@tce.edu

22CS271	COMPUTER SYSTEMS LAB
----------------	-----------------------------

Category	L	T	P	Credit
ESC	0	0	2	1

Preamble

The laboratory course is designed to enable the students to design and construct practically the combinational and sequential logic circuits for different applications. The list of experiments starts with the verification of Boolean theorems and truth table of gates. Then the design and construction of a variety of circuits using gates, flip flops and other devices are performed. The simulation of simple circuits using Hardware Description Language is also performed and students will get a chance to reinforce concepts in the working of a “real” operating system like Operating system installation procedure, shell Commands, file system management and process management techniques

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Design half adder, full adder and Verify the truth tables of Boolean logic gates and theorems of Boolean algebra (Apply)	TPS3	70	80
CO2	Design combinational logic circuits for given specifications and implement them using HDL, multiplexers and programmable logic devices. (Apply)	TPS3	70	80
CO3	Design sequential logic circuits like counters, Registers, flip-flops and sequence detectors using Logic gates. (Apply)	TPS3	70	80
CO4	Identify and utilize the proper system software component, for programming Systems. (Apply)	TPS3	70	80
CO5	Implement Process Management and File Management system calls in C program. (Apply)	TPS3	70	80
CO6	Create shell scripts to automate and capture common repetitive tasks using bash scripting. (Apply)	TPS3	70	80

Mapping with Programme Outcomes

C Os	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		L	-	-	M	M	M		M	M	L	M

CO2	S	M	L		L	-	-	M	M	M		M	M	L	M
CO3	S	M	L		L	-	-	M	M	M		M	M	L	M
CO4	S	M	L		L	-	-	M	M	M		M	M	L	M
CO5	S	M	L		L	-	-	M	M	M		M	M	L	M
CO6	S	M	L		L	-	-	M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyze		
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

1. Verification and interpretation of truth table for AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates
2. Construction of half / full adder using XOR and NAND gates and verification of its operation
3. Implementation of 4x1 multiplexer and 1x4 de-multiplexer using logic gates
4. Verification of Binary to Gray and Gray to Binary conversion using NAND gates only
5. Verification of the truth table of RS, JK, T and D flip-flops using NAND & NOR gates
6. Design and Verification of the 4-Bit Serial In - Parallel Out Shift Registers
7. Installation of Windows and Linux operating systems.
8. Perform operations using the text editors, Compilers and IDE in Windows and Linux OS.
9. Perform C programs using process management system calls.
10. Perform C programs using File system management system calls.
11. Perform shell scripts with the basic shell commands.
12. Perform shell scripts using conditional and looping constructs.

Learning Resources

1. Ramakant A. Gayakwad, Op-Amps and Linear Integrated Circuits, Pearson Education; Fourth Edition, 2015.
2. M. Morris Mano, Micheal D. Ciletti, Digital Design: With an Introduction to the Verilog HDL, VHDL, and SystemVerilog, Pearson Education; Sixth edition, 2018.
3. System programming with C and Unix, Adam Hoover, Addison-Wesley, 2010 Pearson Education.

Course Designers:

1. Dr.R. Leena Sri (rlsit@tce.edu)
2. Mr.D.Nagendra Kumar (dnkcse@tce.edu)

22CS281	PROGRAMMING LAB	Category	L	T	P	Credits
		PC	0	0	2	1

Preamble

This course objective is to provide practical experience on fundamentals of C programming as well as the design of simplified computer solutions to real-world problems. Programming concepts, data types, conditional and control structures, functions, arrays, recursion, file handling, and preprocessor directives are all projected in solving the engineering and real life problems.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcome 1 (CO1)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Develop Programs using various Data types and operators	TPS 3	B	80
CO2	Use appropriate conditional and control structure to solve Engineering problems	TPS 3	B	80
CO3	Implement modular programming using user defined and recursive functions	TPS 3	B	80
CO4	Perform sorting , searching and matrix operations using C Programming	TPS 3	B	80
CO5	Make use of pointers, structure and union to solve problems	TPS 3	B	80
CO6	Solve real life problems by file handling functions and preprocessor directives	TPS 3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	S	M	L	L	L	L	L	M	M	M	M	M	M	L	M
CO2.	S	M	L	L	L	L	L	M	M	M	M	M	M	L	M
CO3.	S	M	L	L	L	L	L	M	M	M	M	M	M	L	M
CO4.	S	M	L	L	L	L	L	M	M	M	M	M	M	L	M

CO5.	S	M	L	L	L	L	L	M	M	M	M	M	M	L	M
CO6.	S	M	L	L	L	L	L	M	M	M	M	M	M	L	M

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember	-	-
Understand	-	-
Apply	100	100
Analyse	-	-
Evaluate	-	-
Create	-	-

List of Experiments/Activities with CO Mapping

Ex.NO	Experiment	CO
1	Develop C Programs with various data types	CO1
2	Develop C Programs to illustrate operators and sequence control structure	CO1
3	Write a C Program for decision making problems	CO2
4	Implement C program to solve iterative problems using loops	CO2
5	Implement bubble and selection sorting using modular programming	CO3
6	Write C Programs for linear and binary searching techniques.	CO3,CO4
7	Implement matrix operations and recursive functions using C programs	CO3,CO4
8	Implement pointers to array and function	CO5
9	Develop a C Program for user defined Data Types: Structure and Union	CO5
10	Demonstrate file Handling operations using C program	CO6
11	Illustrate C preprocessor directives through programming	CO6
12	Simple Application Development	CO1,CO2,CO3,CO4,CO5,CO6

Learning Resources

- 1.R.G.Dromey, "How to solve it by Computers", Reprint, PHI Publishers, 2011.
2. Kernighan, B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Publisher, 2015.
3. J.R.Hanly and E.B. Koffmann, "Problem Solving and Program design in C", Fifth Edition, Pearson Publisher, 2012
4. YashwantKanetkar, "Let us C", 18th Edition, BPB Publications, 2021.
5. ReemaThareja, "Programming in C", Second Edition,Oxford University Press,2016.
6. Byron Gottfried, "Programming with C", Fourth Edition, Tata McGraw Hill Education, 2018
7. Paul Deital and Harvey Deital, "C How to Program", Seventh Edition, Prentice Hall, 2012

Course Designer(s):

- | | |
|----------------------------------|--|
| 1. Dr.M.Nirmala Devi, AP,CSE,TCE | mnit@tce.edu |
| 2. Ms.G.Bhavani, AP,CSE,TCE | gbicse@tce.edu |

24CS261**EXTENDED REALITY : PRINCIPLES
AND PRACTICE**

Category L T P Credit

ESC 2 0 2 3

Terminal Exam: Theory

Preamble

Extended Reality: Principles and Practice encapsulates various immersive technologies that can merge the physical and virtual worlds, including virtual reality (VR), and augmented reality (AR). This course is an introduction to graphics, objects design and interactions to the virtual worlds with practical applications of Extended Reality concepts in human perception, immersion and presence, virtual world modelling and 3D user interaction.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Define and explain the fundamental concepts in 2D,3D graphics, projections and immersive experiences.	TPS2	70	90
CO2	Understand the fundamental hardware and software for VR experiences.	TPS2	70	90
CO3	Apply VR tools to develop a scene for interaction and navigation with in virtual world.	TPS3	70	85
CO4	Design applications using any Game Engine for 2D, 3D, AR and VR for any target platform.	TPS3	70	85
CO5	Design applications for Augmented Reality experiences using mobile devices.	TPS3	70	85
CO6	Plan and implement a team-project to develop and build an AR/ VR applications or games.	TPS4	70	85

Mapping with Programme Outcomes

CO s	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O 1	PS O 2	PS O 3
CO 1	M	L						S	S	S	S	S	M	M	S
CO 2	M	L			S			S	S	S	S	S	M	M	S
CO 3	S	M	L		S			S	S	S	S	S	M	M	S
CO 4	S	M	L		S			S	S	S	S	S	M	M	S

CO 5	S	M	L					S	S	S	S	S	M	M	S
CO 6	S	S	M	L	S			S	S	S	S	S	S	M	S

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						CAT 2						Terminal					
	100						100						100					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	7	20											4	10				
CO2	7	20											4	10				
CO3	6	10	30					10	20				4	10	10			
CO4								10	20				4	10	10			
CO5								10	20	10			4	10	10			
CO6																		

CO	MODEL TEST						OCR					
	1	2	3	4	5	6	1	2	3	4	5	6
CO1												
CO2			10						10			
CO3			20						20			
CO4			25						25			
CO5			25						25			
CO6				20						20		

Syllabus

Extended Reality Principles- Graphics Pipeline, 2D and 3D Linear Transformations, Viewing and Projective Transformations and projections. Extended Reality, Virtual Reality system, Four Key Elements of Virtual Reality Experience, Virtual World, Augmented Reality, Tracking technologies, Augmenting displays, Position and orientation in 3d space.

Defining Virtual and Augmented Reality Exploring Virtual Reality, features, Controllers. Exploring Augmented Reality, Form Factors and Controllers. VR hardware, AR hardware
Creating Content for Virtual and Augmented Reality. Design Software, Capturing Real Life, Assessing Development Software. Distributing Your Content. 360 degree videos, AR VR Usecases

Development tools- Creating a sample scene for VR using sketchup. Navigating in a Virtual World Wayfinding Travel Navigation.

Creating AR/ VR Projects, getting started with Unity, Installing Unity Hub and Unity Editor, Editor interface, Creating Projects with templates, Asset store, creating a simple 2d game using game objects, Scripting in c#, Visual Scripting, Building/Publishing/Exporting to desktop OS, Mobile OS

Setting Up for AR Development Installing XR plugins for AR devices, Installing the AR Foundation package, Building the Simple AR scene in project, Importing the Sample assets, Using AR Session and AR Session Origin, AR Camera, Adding AR Raycast Manager, Building and running the scene. Placing an object on a plane, Writing the PlaceObjectOnPlane script, creating a prefab for placing on the scene.

A sample scene in AR/ VR Building and running the sample project using Unity for 3D games, Mobile tracking applications, Build the scene in to application to run on android / ios based mobile devices.

Text Books

1. Steve Marschner Peter Shirley, "Fundamentals of Computer graphics", Taylor & Francis, fourth edition, 2016
2. William Sherman, Alan Craig, "Understanding Virtual Reality – Interface, Application, Design", The Morgan Kaufmann Series, second edition, 2018.
3. Paul Mealy, Virtual & Augmented Reality for Dummies" John Wiley & Sons, Inc., 2018.
4. Kent Peterson, "Unity Game Development Script, Build, Customize Your Game from Scratch for Beginners and Experts" 2020.
5. Jonathan Linowes, "Augmented Reality with Unity AR Foundation", Packt Publishing, 2021

Reference Books & web resources

1. Alan B. Craig, William R. Sherman, Jeffrey D. Will, "Developing Virtual Reality Applications", 2017.
2. Matjaz Mihelj, Domen Novak, Samo Begus, "Virtual Reality Technology and Applications", 1st Edition, Springer Netherlands, 2014.
3. Grigore C. Burdea, Philip Coiffet, "Virtual Reality Technology", 2nd Edition, Wiley India, 2006.
4. John Vince, "Introduction in Virtual Reality", Springer, 2004.
5. Steve Aukstakalnis, "Practical Augmented Reality", Addison Wesley 2017.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods	CO
1	Extended Reality principles		
1.1	Graphics Pipeline, 2D and 3D Linear Transformations, Viewing and Projective Transformations and projections	1	CO1
1.2	Extended Reality, Virtual Reality system, Four Key Elements of Virtual Reality Experience, Virtual World, Augmented Reality	1	CO1
1.3	Tracking technologies, Augmenting displays Position and orientation in 3d space.	2	CO1
2	Defining Virtual and Augmented Reality		

Module No.	Topic	No. of Periods	CO
2.1	Exploring Virtual Reality, features, Controllers.	1	CO2
2.2	Exploring Augmented Reality, Form Factors and Controllers. VR hardware, AR hardware	1	CO2,
2.3	Creating Content for Virtual and Augmented Reality. Design Software, Capturing Real Life	2	CO2,
2.4	, Assessing Development Software. Distributing Your Content. 360 degree videos, AR VR Use cases	2	CO2,
3	Development tools-		
3.1	Creating a sample scene for VR using Sketchup.	2	CO3
3.2	Navigating in a Virtual World Wayfinding Travel Navigation.	1	CO3, CO6
Continuous Assessment Test – I			
4	Creating AR/ VR Projects		
4.1	Getting started with Unity, Installing Unity Hub and Unity Editor, Editor interface, Creating Projects with templates, Asset store,	1	CO4
4.2	Creating a simple 2d game using game objects, Scripting in c#,	1	CO4, CO6
4.3	Visual Scripting	1	CO4, CO6
4.4	Building/Publishing/Exporting to desktop OS, Mobile OS	1	CO4, CO6
	Assignment 1		CO6
5	Setting Up for AR Development		
5.1	Installing XR plugins for AR devices, Installing the AR Foundation package, Building the Simple AR scene in project,	2	CO5
5.2	Importing the Sample assets, Using AR Session and AR Session Origin, AR Camera, Building and running the scene.	1	CO5
5.3	Placing an object on a plane, Writing the script,	1	CO5, CO6
5.4	Creating a prefab for placing on the scene.	1	CO5, CO6
6	A sample scene in AR/ VR		CO5
6.1	Building and running the Sample project using Unity 3D games, Mobile tracking applications,	1	CO6
6.2	Build the scene in to application to run on android / ios based mobile devices	1	CO6
	Continuous Assessment Test – II		
	Total Hours	24	

Course Contents and Lecture Schedule for Laboratory

Module No.	Topic	No. of Lectures	Course Outcome
1.	Create a sample scene with 3d objects in SketchUp	2	CO3
2.	Integrating the model with VR devices	2	CO3
3.	Virtual Walkthrough using Jump and walk using fully immersive kits	2	CO3
4.	Creating a walkthrough on mobile devices for a model	2	CO4
5.	Unity installation and configuration	2	CO4
6.	Applying Physics, Rigid body, materials to game objects	2	CO4
7.	Creating an APK file to run on suitable platform.	2	CO6
8.	Creating a simple 2d game using scripts to get inputs from keyboard.		CO6
9.	Placing a 3d object on a plane in AR	4	CO5
10.	Tracking of images using mobile camera in AR using Unity	4	CO5, CO6
11.	Create a simple 3D game which runs on desktop computer/ mobile devices.	2	CO3, CO4, CO5, CO6
Total Hours		24	

Course Designer(s):

1. Dr. N. Shivakumar APCSE

shiva@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

THIRD SEMESTER

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programmes

COURSES OF STUDY

(For the candidates admitted from 2025-26 onwards)

THIRD SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CS311	Probability and Statistics	BSC	3	1	-	4
22CS321	Theory and Design of Programming Languages	ESC	3	-	-	3
22CS331	Object Oriented Programming	PC	3	-	-	3
22CS342	Data Structures and Algorithms	PC	3	-	-	3
THEORY CUM PRACTICAL						
22CS362	Operating Systems	PC	3	-	2	4
PRACTICAL						
22CS371	Data Structures Lab	PC	-	-	2	1
22CS381	Object Oriented Programming Lab	PC	-	-	2	1
22ES391	Design Thinking	ESC				3
Total			15	1	6	22

BSC : Basic Science Courses

PC : Professional Core Courses

ESC : Engineering Science Courses

L : Lecture

T : Tutorial

P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit

1 Hour Tutorial is equivalent to 1 credit

2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programme

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2025-26 onwards)

THIRD SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CS311	Probability and Statistics	3	40	60	100	27	50
2	22CS321	Theory and Design of Programming Languages	3	40	60	100	27	50
3	22CS331	Object Oriented Programming	3	40	60	100	27	50
4	22CS342	Data Structures and Algorithms	3	40	60	100	27	50
THEORY CUM PRACTICAL								
5	22CS362	Operating Systems (Terminal exam: Theory)	3	50	50	100	22.5	50
PRACTICAL								
6	22CS371	Data Structures Lab	3	60	40	100	18	50
7	22CS381	Object Oriented Programming Lab	3	60	40	100	18	50
8	22ES391	Design Thinking						

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks

22CS311 PROBABILITY AND STATISTICS

Category	L	T	P	Credit
BS	3	1	0	4

Preamble

A mathematical model of some phenomena is frequently impossible to construct due to inherent uncertainties. Many real-world problems, in fact, involve multiple variables (some of which may be unknown), making it difficult to define the problem and perform a precise mathematical analysis. Statistical inference can help in this situation by identifying and limiting the variability involved, as well as projecting outcomes with high probability. After constructing a probabilistic model, we can use correlation and regression to study the relationship between variables and solve the prediction problem. Furthermore, the significance of decisions based on experimental and observed data is validated using hypothesis testing. Discrete-transform concepts can also be used to solve difference equations, which can then be used to model discrete time systems.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the concept of probability, conditional probability to solve problems in the field of science and technology.	TPS3	B	85
CO2	Computing expectation, variance, moments and moment generating functions of random variables.	TPS3	B	85
CO3	Modelling and solving science and engineering problems using discrete probability distributions.	TPS3	B	85
CO4	Modelling and solving science and engineering problems using continuous probability distributions.	TPS3	B	85
CO5	Apply the test of hypothesis and ANOVA for small and large samples to make decisions based on observed and experimental data.	TPS3	B	85
CO6	Model discrete time system using difference equation and solve it by applying Discrete transform techniques.	TPS3	B	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	S	M	L		M			L	M	L		M	M	L	L
CO2.	S	M	L		M			L	M	L		M	M	L	L

CO3.	S	M	L		M	L		L	M	L		M	M	L	L
CO4	S	M	L		M	L		L	M	L		M	M	L	L
CO5	S	M	L		M	L		L	M	L		M	M	L	L
CO6	S	M	L		M			L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assessment 1						Assessment 2						Terminal (%)		
	CAT 1 (%)			Assignment 1 (%)			CAT 2 (%)			Assignment 2 (%)					
TPS	1 10	2 30	3 60	1 -	2 -	3 100	1 10	2 30	3 60	1 -	2 -	3 100	1 -	2 30	3 70
CO1	-	5	20	-	-	28	-	-	-	-	-	-		6	10
CO2	7	10	20	-	-	44	-	-	-	-	-	-		6	12
CO3	3	5	20	-	-	28	-	-	-	-	-	-		3	10
CO4	-	10	-	-	-	-	-	10				20		3	10
CO5	-	-	-	-	-	-	7	10	30			40		6	14
CO6	-	-	-	-	-	-	3	10	30			40		6	14
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Probability: Sample space - Events and their Probabilities-Rules of Probability-Combinatorics-Conditional probability and independence – Baye's theorem.

Random Variables and Their Distributions: Distribution of a random variable- Expectation- Expectation of a function-Variance and standard deviation – Probability density function- Expectation and variance-Moments-Moments generating functions.

Discrete Random Variables and Their Distributions: Bernoulli distribution -Binomial distribution - Negative Binomial distribution -Poisson distribution.

Continuous Random Variables and Their Distributions: Uniform distribution- Exponential distribution – Gamma distribution - Normal distribution.

Testing of Hypotheses: Hypotheses and test procedures – Z tests for Hypotheses about a population mean – The one sample t Test– Z tests for a difference between two population means – The two sample t Test – Analysis of paired data – inferences concerning two population variances- Single Factor ANOVA.

Discrete Transforms: The Z-transform – Properties of the Z-transform -The Inverse Z transform – Discrete Fourier transform - Discrete time systems and difference equations.

Text Book

1. Michael Baron., “Probability and Statistics for Computer Scientists” Second Edition, CRC Press, 2014.
2. Jay. L. Devore., “Probability and Statistics for Engineering and the Sciences”, Ninth Edition, Cengage Learning, 2014.
3. Glyn James and Phil Dyke., “Advanced Modern Engineering Mathematics”, Fifth Edition, Pearson Education Limited, 2018.

Reference Books

1. Douglas C. Montgomery., George C. Runger., “Applied Statistics and Probability for Engineers” Seventh Edition, 2018.
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers., “Probability & Statistics for Engineers & Scientists”, Ninth Edition, 2012.
3. Mendenhall, William, Robert J. Beaver, Barbara M. Beaver., “Introduction to probability and Statistics” Fifteenth Edition, 2019.
4. Introduction to Probability and Statistics for Engineers and Scientists, Sheldon M. Ross., USA,Elsevier, Fifth edition,2014.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	Introduction to Probability	
1.1	Sample space, Events and their Probabilities	1
1.2	Rules of Probability	1
1.3	Combinatorics	1
	Tutorial	1
1.4	Conditional probability and independence	1
1.5	Baye's theorem	1
	Program Using Python / Matlab	
2	Random Variables and Their Distributions	
2.1	Distribution of a random variable	1
	Tutorial	1
2.2	Expectation, Expectation of a function	1
2.3	Variance and standard deviation	1

2.4	Probability density function, Expectation and variance	1
	Tutorial	1
2.5	Moments	1
2.6	Moments generating functions	2
	Program Using Python / Matlab	
3	Discrete Random Variables and Their Distributions	
3.1	Bernoulli Distribution	1
	Tutorial	1
3.2	Binomial distribution	1
3.3	Negative Binomial distribution	1
3.4	Poisson distribution	1
	Tutorial	1
	Program Using Python / Matlab	
4	Continuous Random Variables and Their Distributions	
4.1	Uniform distribution	1
4.2	Exponential distribution	1
4.3	Gamma distribution	1
	Tutorial	1
4.4	Normal distribution	2
	Program Using Python / Matlab	
5	Testing of Hypotheses	
5.1	Hypotheses and test procedures	1
	Tutorial	1
5.2	Z tests for Hypotheses about a population mean	1
5.3	The one sample t test	1
5.4	Z tests for a difference between two population means	1
	Tutorial	1
5.5	The two-Sample t Test	1
5.6	Analysis of paired data	1
5.7	Inferences concerning two population Variances	1
5.8	Single factor ANOVA	1
	Tutorial	1
	Program Using Python / Matlab	
6.	Discrete Transforms	
6.1	The Z-transform	1
6.2	Properties of Z-transform	2
	Tutorial	1
6.3	The Inverse Z-transform	2
6.4	Discrete Fourier transform	1
	Tutorial	1
6.5	Discrete time systems and difference equations	2
6.6	Tutorial	1

	Program Using Python / Matlab	
	Total	48

Course Designer(s):

1. Dr.C.S.Senthilkumar kumarstays@tce.edu
2. Dr.P.Krishnapriya pkamat@tce.edu
3. Dr.P.Victor pvmat@tce.edu

22CS321 **THEORY AND DESIGN OF
PROGRAMMING LANGUAGES**

Category L T P Credit
ESC 3 0 0 3

Preamble

This course is a fundamental course on programming principles. It gives an overview of different programming paradigms. It deals with the concepts common in various imperative programming languages and declarative programming languages. The course also covers the use of parallelism in programming. The principles of programming are dealt with the case study of programming languages C, Java, Ada, Haskell, Prolog and JavaScript.

Prerequisite

22CS241 – Problem Solving and Programming

22CS251 – System Programming

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Solve engineering problems using Sequential Procedural Programming Paradigm (CO1)	TPS3	B	80
CO2	Solve engineering problems using Sequential Object Programming Paradigm (CO2)	TPS3	B	80
CO3	Solve engineering problems using Concurrent Programming Paradigm (CO3)	TPS3	B	80
CO4	Solve engineering problems using Functional Programming Paradigm (CO4)	TPS3	B	80
CO5	Solve engineering problems using Logic Programming Paradigm (CO5)	TPS3	B	80
CO6	Develop scripts for given specification using web scripting paradigm (CO6)	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L		M			L	L	L		L	M		
CO2.	S	M	L		M			L	L	L		L	M		
CO3.	S	M	L		M			L	L	M		L	M		L
CO4.	S	M	L		M			L	L	L		L	M		
CO5.	S	M	L		M			L	L	M		L	M		L
CO6.	S	M	L		M			L	L	M		L	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Assignment 1	CAT2						Assignment 2	Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	1	2	3	4	5	6
CO1	10	10	15	-	-	-	30	-	-	-	-	-	-	-	2	-	15	-	-	-
CO2	10	10	15	-	-	-	40	-	-	-	-	-	-	-	-	5	15	-	-	-
CO3	10	10	10	-	-	-	30	-	-	-	-	-	-	-	2	5	10	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	-	10	-	-	-
CO5	-	-	-	-	-	-	-	5	10	20	-	-	-	30	2	5	10	-	-	-
CO6	-	-	-	-	-	-	-	5	10	20	-	-	-	50	2	-	15	-	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

The Imperative Programming Paradigm: Basics of Programming Paradigms, Expression Semantics, Data Semantics, Imperative Design principles – Structured, Turing Complete, Modularity, Procedural abstraction, Structured Theorem - Sequencing, Selection, Iteration and Recursion.

Sequential Programming - Procedural Programming Paradigm: Procedures, Parameter passing, Procedural abstraction, Case Study - C Programming.

Sequential Programming - Object Oriented Programming Paradigm: OOP design principles - Groupings of Data and Operations, Encapsulation, Data Abstraction, Inheritance and Polymorphism, Exceptions, Case Study - Java Programming.

Concurrent Programming Paradigm: Concurrency design with interleaving of process, Safe Access to Shared Data – Semaphore, Liveness property of concurrent programs - Case Study - Concurrency in ADA.

Functional Programming Paradigm: Declarative programming design – Functional and logic paradigm, Basics of Functional programming paradigm, Expressions – Syntax, Parsing, Types and Values, Assigning names to expressions, Lambda Calculus fundamentals, Function Abstraction and Recursive Functions - Case Study Haskell.

The Logic Programming Paradigm: Clauses and Predicates, Operations and Arithmetic, List and Operations, Unification and Backtracking, Case Study – Prolog.

Scripting Paradigm: – Basics of Web design language and Scripting Language, Case Study: JavaScript – Syntax of writing JavaScript, Variables and Functions.

Text Book

1. Michael L. Scott, "Programming Language Pragmatics", Fourth Edition, Morgan Kaufmann, 2016.
2. Seyed Mohamed Buhari, "Principles of Programming Languages A Paradigm approach", Tata McGraw-Hill, 2011.
3. Ravi Sethi, "Programming Languages: Concepts and Constructs", AT&T Bell Laboratories, 2nd edition, Addison Wesley, 2007.
4. Programming Languages: Design and Implementation (4th Edition), by Terrence W. Pratt, Marvin V. Zelkowitz, Pearson, 2000.
5. David A.Watt, "Programming Language Concepts and Paradigms", Prentice Hall, 1990.

6. Bruce J. MacLennan, "Principles of Programming Languages: Design, Evaluation and Implementation", Second Edition.

Reference Books & web resources

1. Minh Quang Tran, "The Art of Functional Programming Paradigm", 2022.
2. Alejandro Serrano Mena, "Practical Haskell: A Real-World Guide to Functional Programming", Apress Media, 2019.
3. Axel Rauschmayer, "Speaking JavaScript", O'Reilly Media, 2015.
4. Max Bramer, "Logic Programming with Prolog", Springer 2005.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	The Imperative Programming Paradigm	
1.1	Basics of Programming Paradigms, Expression Semantics, Data Semantics, Imperative Design principles – Structured, Turing Complete, Modularity, Procedural abstraction	2
1.2	Structured Theorem - Sequencing, Selection, Iteration and Recursion.	1
2	Sequential Programming - Procedural Programming Paradigm	
2.1	Procedures, Parameter passing, Procedural abstraction	1
2.2	Case Study - C Programming.	2
3	Sequential Programming - Object Oriented Programming Paradigm	
3.1	OOP design principles - Groupings of Data and Operations, Encapsulation, Data Abstraction	1
3.2	OOP design principles - Inheritance and Polymorphism, Exceptions	2
3.3	Case Study - Java Programming	2
4	Concurrent Programming Paradigm	
4.1	Concurrency design with interleaving of process	1
4.2	Safe Access to Shared Data – Semaphore	2
4.3	Liveness property of concurrent programs	2
4.4	Case Study - Concurrency in ADA.	2
5	Functional Programming Paradigm	
5.1	Declarative programming design – Functional and logic paradigm, Basics of Functional programming paradigm, Expressions – Syntax, Parsing, Types and Values, Assigning names to expressions	2
5.2	Lambda Calculus fundamentals	2

Module No.	Topic	No. of Periods
5.3	Function Abstraction and Recursive Functions	1
5.4	Case Study Haskell.	2
6	The Logic Programming Paradigm	
6.1	Clauses and Predicates & Operations and Arithmetic	1
6.2	List and Operations, Unification and Backtracking,	2
6.3	Case Study – Prolog	2
7	Scripting Paradigm	
7.1	Basics of Web design language and Scripting Language	2
7.2	Case Study: JavaScript – Syntax of writing Java Script	2
7.3	Variables and Functions in JavaScript	2
	Total	36

Course Designer(s):

1. G.Madhupriya, Associate Professor,CSE gmadhupriya@tce.edu
2. M.Suguna, Assistant Professor, CSE mscse@tce.edu

22CS331**OBJECT ORIENTED
PROGRAMMING**

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

This syllabus is intended for Computer Science Engineering students and enables them to learn Object Oriented Programming and the design of computer solutions in a precise manner. The syllabus emphasizes OOP concepts, Functions, Polymorphism, Inheritance and I/O. The intention is to provide sufficient depth in these topics to enable students to apply Object Oriented approach to programming. The modules in the syllabus reflect solving general problems via programming solutions. Thus, modules collectively focus on programming concepts, strategies and techniques; and the application of these toward the development of programming solutions.

Prerequisite

Programming fundamentals

Course Outcomes

Upon the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct Object oriented programs using methods and passing arrays, objects, and array of objects to them.	TPS3	B	80
CO2	Demonstrate Compile-time and Run-time polymorphism using object oriented programs	TPS3	B	80
CO3	Illustrate the relationships between objects using inheritance hierarchies and aggregation	TPS3	B	80
CO4	Develop Object Oriented programs to handle data using Java collections, Files and Object Serialization	TPS3	B	80
CO5	Develop Object Oriented programs to handle exceptions	TPS3	B	80
CO6	Develop Object Oriented programs to demonstrate event driven programming, concurrent programming and network programming.	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		L	L		L	M	M		L	M	L	L
CO2	S	M	L		L	L		L	M	M		L	M	L	L

CO3	S	M	L		L	L		L	M	M		L	M	L	L
CO4	S	M	L		L	L		L	M	M		L	M	L	L
CO5	S	M	L		L	L		L	M	M		L	M	L	L
CO6	S	M	L		L	L		L	M	M		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment		Terminal					
													1	2						
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	3	3	1	2	3	4	5	6
CO1	10		10										20		2	4	10			
CO2	10	10	10										40		2	4	10			
CO3	10	20	20										40		4	4	10			
CO4							10		10					30	4	4	10			
CO5							10	10	10					30	4	4	10			
CO6							10	20	20					40	4		10			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Basics of Object oriented Programming Object oriented programming and its benefits - Object oriented programming concepts: Encapsulation, Information hiding and Abstraction – Generalization/Specialization and Polymorphism - Object oriented design: finding the Classes and their Responsibilities – Object oriented programming language: Java-

Classes and Objects Instance fields and Methods-Constructors–Passing Arguments to a Method – Returning Value from a Method – Method overloading –Constructor overloading-Passing Arrays as Arguments to Methods – Passing Objects to Methods- Returning Objects from Methods

Class collaborations and Polymorphism Object oriented Design: Class Collaborations – Aggregation –Composition –Chains of Inheritance – Overriding Super class methods – Abstract Classes and Abstract Methods – Interfaces

I/O Handling and Exception Handling – Binary files – Random-Access files- Object serialization – Exception handling

Collection Framework in Java – Introduction to java collections, Overview of java collection framework, commonly used collection classes- Array List, Vector, Hash table, Stack

Event-Driven Programming Concurrent Programming Network programming – Text-related GUI components – other GUI components – Handling mouse events and button events – Thread life cycle and methods – Runnable interface – Thread Synchronization – Basics of network programming

Text Book

1. Herbert Schildt: "Java: The Complete Reference", Twelfth Edition, McGraw-Hill, 2021.
2. Tony Gaddis, "Starting Out with Java: From Control Structures through Objects", Sixth edition, Pearson Education Limited, 2016.
3. Bart Baesens, Aimee Backiel, SeppevandenBroucke, "Beginning Java Programming: The Object-Oriented Approach", John Wiley & Sons, 2015.
4. Kenneth L. Calvert and Michael J. Donahoo, "TCP/IP Sockets in Java: Practical Guide for Programmers", 2nd Edition. Elsevier, 2011.
5. Grady Booch, Robert Maksimchuk, Michael Engel, Bobbi Young, Jim Conallen, Kelli Houston "Object Oriented Analysis and Design with Applications", Third Edition, 2012.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours
1	Basics of Object oriented Programming (6)	
1.1	Object oriented programming and its benefits	1
1.2	Object oriented programming concepts: Encapsulation, Information hiding and Abstraction, Generalization/Specialization and Polymorphism	1
1.3	Object oriented design: finding the Classes and their Responsibilities	2
1.4	Object oriented programming language: Java	1
1.5	Object oriented programming language: C++	1
2	Classes and Objects (6)	
2.1	Instance fields and Methods-Constructors	1
2.2	Passing Arguments to a Method – Returning Value from a Method - Method overloading	2
2.3	Constructor overloading	1
2.4	Passing Arrays As Arguments to Methods	1
2.5	Passing Objects to Methods, Returning Objects from Methods	1
3	Class collaborations and Polymorphism (6)	
3.1	Object oriented Design: Class Collaborations	1
3.2	Aggregation –Composition	1
3.3	Chains of Inheritance	1
3.4	Overriding Super class methods	1
3.5	Abstract Classes and Abstract Methods	1
3.6	Interfaces	1
4	I/O Handling and Exception Handling (5)	
4.1	Binary files	1
4.2	Random-Access files	1
4.3	Object serialization	1
4.4	Exception handling	2
5	Collection Framework (6)	

5.1	Introduction to java collections	1
5.2	Overview of java collection framework	1
5.3	Collection classes - Array List, Vector	2
5.4	Collection classes - Hash table, Stack	2
6	Event-Driven Programming Concurrent Programming Network programming (7)	
6.1	Frameworks	1
6.2	Text-related GUI components, other GUI components	1
6.3	Handling mouse events and button events	1
6.4	Thread life cycle and methods	1
6.5	Runnable interface	1
6.6	Thread Synchronization	1
6.7	Basics of network programming	1

Course Designer(s):

1. Dr M.Vijayalakshmi, Professor, CSE – mviji@tce.edu
2. Mr S.Santhana Hari, Assistant Professor, CSE – sshcse@tce.edu

22CS342**DATA STRUCTURES AND
ALGORITHMS**

Category L T P Credit

PC 3 0 0 3

Preamble

This course will cover various data structures and their operations for manipulating them. Students will learn how to organize the data so that, the data can be accessed and updated efficiently using computer programs.

Prerequisite

22CS241: Problem Solving and Programming

Course Outcomes

On the successful completion of the course, students will be able to

CO No.	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply various types of linked lists (singly, doubly, and circular) to solve basic computational problems.	TPS3	70	80
CO2	Implement stacks, linear queues, circular queues, and deques to solve computational problems such as expression evaluation, balancing symbols, and task scheduling.	TPS3	70	80
CO3	Construct binary trees and various types of heaps (binary heap, leftist heap, binomial queue) for efficient data organization, priority handling, and encoding applications.	TPS3	70	80
CO4	Analyze the structure and operations of binary search trees and balanced binary search trees (AVL trees, Splay trees, Red-Black trees) to optimize search efficiency and dynamic data management.	TPS4	70	80
CO5	Apply multiway search trees, tries (standard, compressed, suffix), and disjoint sets to solve problems in information retrieval, pattern matching, and union-find operations.	TPS3	70	80
CO6	Analyze the performance of internal and external sorting methods, searching techniques (linear, binary search), and hashing techniques including collision resolution and extendible hashing	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		L			M	M	M		M	M		M
CO2	S	M	L		L			M	M	M		M	M		M
CO3	S	M	L		L			M	M	M		M	M		M
CO4	S	S	M	L	L			M	M	M		M	M	L	M
CO5	S	M	L		L			M	M	M		M	M		M

CO6	S	S	M	L	L			M	M	M		M	M	L	M
-----	---	---	---	---	---	--	--	---	---	---	--	---	---	---	---

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Assignment-1	CAT2						Assignment-2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	5	10	20				40									4	12			
CO2		10	20				30									2	12			
CO3	5	10	20				30									4	12			
CO4								5	5		25			40		4		16		
CO5								5	5	20				30		2	12			
CO6								5	5		25			30		4		16		

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

INTRODUCTION TO ABSTRACT DATA TYPES AND ANALYSIS:

Abstract Data Types (ADT) - Basic concept of Data Structures - Performance measures for Data Structures - Time and Space Complexity Asymptotic Measures - Big-Oh, Omega, Theta.

LINKED LISTS:

Linked list implementation – Singly linked lists – Circularly linked lists – Doubly-linked lists – Operations - Applications of lists – Polynomial ADT – Radix Sort – Multilists – Josephus Problem – Palindrome Checking.

STACKS AND QUEUES:

Stack – Operations – Applications – Balancing Symbols – Evaluating arithmetic expressions - Infix to Postfix conversion – Function Calls – Queue – Operations – Circular Queue – DeQueue – Applications of Queues – Scheduling - Applications of DeQueue – clock-wise and anti-clockwise rotational operation.

BINARY TREES:

Binary Tree Terminologies – Tree Traversals – Tree representation and properties - Types of Binary Tree - Applications of Binary Tree - Expression trees – Priority Queue (Heaps) – Binary Heap – Leftist Heap – Binomial Queue – Applications of Heap – Huffman Coding, Heap Sort.

BINARY SEARCH TREES:

Binary Search Tree (BST) – Balanced BST - AVL Trees – Red Black Trees – Splay Trees - Operations

MULTIWAY SEARCH TREES:

Multiway Search Trees - B-Tree - Tries - Standard Tries - Compressed Tries - Suffix Trees – Linear Construction of Suffix Trees - Application on Tries - Longest Prefix – Binary Tries (PATRICIA Structure) - Pattern Searching

DISJOINT SETS:

Disjoint-set operations - Representation – Smart union algorithms – Path Compression

SEARCHING, SORTING AND HASHING TECHNIQUES

Sorting - Insertion sort – Shell sort – Merge Sort – External Sorting - The Simple Algorithm - Multiway Merge – Polyphase Merge - Replacement Selection. Searching – Linear Search – Binary Search - Hashing - Hash functions – Division Method, Multiplication Method, Mid-Square Method, Folding Method - Collision resolution and overflow handling techniques - Open Hashing

(Separate Chaining) - Open Addressing / Closed Hashing - Linear, Quadratic, Double, Rehashing
- Extendible Hashing

Text Books

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson Education, 2005.
2. Kamthane, Introduction to Data Structures in C, 1st Edition, Pearson Education, 2007

Reference Books & web resources

1. Langsam, Augenstein and Tanenbaum, Data Structures Using C and C++, 2nd Edition, Pearson Education, 2015.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms", Fourth Edition, Mcgraw Hill/ MIT Press, 2022.
3. Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft, Data Structures and Algorithms, 1st edition, Pearson, 2002.
4. Kruse, Data Structures and Program Design in C, 2nd Edition, Pearson Education, 2006.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
0	Syllabus Discussion (CO, CO vs PO, Assessment, Prerequisite, Concept Map, Text and Reference Books)	1
1	INTRODUCTION TO ABSTRACT DATA TYPES AND ANALYSIS	
1.1	Review of elementary data types and structures in C Abstract Data Types (ADT) - Basic concept of Data Structures	1
1.2	Performance measures for Data Structures - Time and Space Complexity Asymptotic Measures - Big-Oh, Omega, Theta	1
2	LINKED LISTS	
2.1	Abstract Data Types (ADTs) – List ADT – Array-based implementation- Linked list implementation	1
2.2	Singly linked lists	1
2.3	Circularly linked lists	1
2.4	Doubly-linked lists	1
2.5	Applications of lists – Polynomial ADT – Radix Sort	1
2.6	Applications of lists – Multilists – Josephus Problem – Palindrome Checking.	1
3	STACKS AND QUEUES	

Module No.	Topic	No. of Periods
3.1	Stack ADT – Operations	1
3.2	Applications – Balancing Symbols – Evaluating arithmetic expressions- Infix to Postfix conversion – Function Calls	1
3.3	Queue ADT – Operations	1
3.3	Circular Queue, DeQue	1
3.4	Applications of Queues- Scheduling- Applications of DeQue – clock-wise and anti-clockwise rotational operation	1
4	BINARY TREES	
4.1	Tree ADT – Tree Terminologies – Tree Traversals – Tree Representation and Properties	1
4.2	Binary Tree ADT – Types of Binary Tree Applications of Binary Tree – Expression trees	1
4.3	Binary Heap	1
4.4	Leftist Heap	1
4.5	Binomial Queue	1
4.6	Applications of Heap – Huffman Coding, Heap Sort.	1
5	BINARY SEARCH TREE	
5.1	Binary Search Tree ADT	1
5.2	AVL Trees	1
5.3	Red Black Trees	1
5.4	Splay Trees	1
6	MULTIWAY SEARCH TREE	
6.1	Multiway Search Trees - B-Tree	1
6.2	Tries - Standard Tries - Compressed Tries	1

Module No.	Topic	No. of Periods
6.3	Linear construction of Suffix Tries	1
6.4	Application on Tries - Longest Prefix – Binary Tries (PATRICIA Structure) - Pattern Searching	1
7	DISJOINT SETS	
7.1	Disjoint-set operations - Representation – Smart union algorithms – Path Compression	1
8	SEARCHING, SORTING AND HASHING TECHNIQUES	
8.1	Sorting –Insertion sort- Shell sort –. Merge Sort	1
8.2	External Sorting - The Simple Algorithm - Multiway Merge - Replacement Selection	1
8.3	Searching – Linear Search – Binary Search	1
8.4	Hashing - Hash functions – Division Method, Multiplication Method	1
8.5	Mid-Square Method, Folding Method	1
8.6	Collision resolution and overflow handling techniques - Open Hashing (Separate Chaining)	1
8.7	Open Addressing / Closed Hashing - Linear, Quadratic, Double, Rehashing - Extendible Hashing	1
	Total	36

Course Designer(s):

- | | |
|---|----------------|
| 1. Dr.M.K.Kavitha Devi, Professor, CSE | mkkdit@tce.edu |
| 2. Dr.Raja Lavanya, Assistant Professor, CSE | rlit@tce.edu |
| 3. Dr.R.Harine Rajashree, Assistant Professor,CSE | rhcse@tce.edu |

22CS362	OPERATING SYSTEMS
----------------	--------------------------

Category	L	T	P	Credit
PC	3	0	2	4

Preamble

This course has two components: a theory component is to teach the concepts and principles that underlie modern operating systems. Students will learn about processes and processor management, concurrency and synchronization, memory management schemes, file system, secondary storage management and security measures. The practical component of this course will relate theoretical principles with operating system implementation.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Practice the following terms in context of operating systems: Evolution, Types, Structure, Functions and components	TPS3	70	70
CO2	Choose the best design and evaluate the trade-offs of various modern operating systems	TPS3	70	70
CO3	Examine the given scenario to implement processor scheduling and synchronization algorithms	TPS4	70	70
CO4	Construct solutions for problems related to deadlocks in a multi-programmed operating system	TPS3	70	70
CO5	Develop appropriate solutions to solve problems related to primary and secondary memory management.	TPS3	70	70
CO6	Implement the disk allocation algorithms and identify the security measures for a given scenario	TPS3	70	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L			L		M	M	L	L	L	M	L	M
CO2	S	M	L			L		L	L	L	L	L	M	L	L
CO3	S	S	M	L	M		L	L	M	L	L	L	M	L	L
CO4	S	S	M	L	M		L	L	M	L	L	L	M	L	L
CO5	S	M	L		M		L	L	M	L	L	L	M	L	L
CO6	S	M	L		M		L	L	M	L	L	L	M	L	L

Passed in Adhoc Board of Studies meeting on 17.04.2026

Approved in 71st Academic Council meeting on 27.06.2026

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1(Theory Component)						CAT 2(Theory Component)						Model (Lab Component)				OCR				Terminal (Theory Component)					
	100						100						100				100				100					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	1	2	3	4	1	2	3	4	5	6
CO1	6	10	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3	5	5	-	-	-
CO2	6	10	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3	5	5	-	-	-
CO3	6	10	20	12	-	-	-	-	-	-	-	-	-	-	20	-	-	-	20	-	3	-	15	5	-	-
CO4	-	-	-	-	-	-	-	10	20	-	-	-	-	-	20	10	-	-	20	10	3	-	15	-	-	-
CO5	-	-	-	-	-	-	-	10	30	-	-	-	-	-	30	-	-	-	30	-	3	-	15	-	-	-
CO6	-	-	-	-	-	-	-	10	20	-	-	-	-	-	20	-	-	-	20	-	-	-	15	-	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Role of OS, Types of OS - Batch Systems, Multiprogramming, Time Sharing, Distributed & Real time OS, System Architecture, System components - OS Services, System Calls, System Programs, System Design, Design Trade-offs.

Process Management: Concepts - Process status, Process description, Process model, Process Scheduling - Scheduler organization, pre-emptive and non-pre-emptive scheduler strategies, Scheduling algorithms - FCFS, SJF, SRTF, Priority Scheduling, Round Robin Scheduling, Multiple Processor scheduling, Thread Concepts and Multiple threaded OS.

Process Synchronization and Deadlock: Process Co-operation, Inter-process communication, Process Synchronization, Synchronization Issues, Critical Section problem, Mutual exclusion, Process Synchronization with semaphores. Deadlock - Conditions, Prevention, Avoidance and Recovery.

Memory Management and File system: Paging, Segmentation and Contiguous memory allocation, Virtual Memory - Demand Paging, Page replacement and Frame Allocation policies, Thrashing, File System - Concepts, Access Method, Directory Structure and File System Management.

Disk management and Security: Disk management - Disk Structure and Scheduling, File systems and operating system support for distributed systems, Security- Protection Goals, Access Control, Program threats, System threats.

Advanced Operating Systems: Needs & Design Goals – Distributed OS, Multiprocessor OS, Mobile OS, MAC OS.

Learning Resources

1. Avi Silberschatz, Peter Baer Galvin and Greg Gagne: Operating System Concepts, Ninth edition, John Wiley and Sons, 2012.
2. Advanced Concepts in Operating Systems, MukeshSinghal, Niranjana G. Shivaratri, Tata McGraw-Hill Edition 2001.
3. Andrew S. Tanenbaum, Albert S.WoodHull: Operating Systems, Design and Implementation, Fourth Edition, Prentice Hall, 2008.
4. Gary Nutt, "Operating Systems", Third Edition, Addison Wesley, 2004

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	Introduction (7)		
1.1	Role of OS	1	CO1
1.2	Types of OS	2	CO1
1.3	System Architecture	1	CO1
1.4	System components	1	CO1
1.5	System Design	1	CO2
1.6	Design Trade-offs	1	CO2
2	Process Management (10)		
2.1	Concepts - Process status, Process description, Process model	2	CO3
2.2	Process Scheduling - Scheduler organization, pre-emptive and non-pre-emptive scheduler strategies	2	CO3
2.3	Scheduling algorithms- FCFS, SJF, SRTF, Priority Scheduling, Round Robin Scheduling, Multiple Processor scheduling	4	CO3
2.4	Thread Concepts and Multiple threaded OS	2	CO3
3	Process Synchronization and Deadlock (9)		
3.1	Process Co-operation	1	CO3
3.2	Inter-process communication	1	CO3
3.3	Process Synchronization	1	CO3
3.4	Synchronization Issues	1	CO3
3.5	Critical Section problem	1	CO3
3.6	Mutual exclusion	1	CO3
3.7	Semaphores	1	CO3
3.8	Deadlock - Conditions, Prevention, Avoidance and Recovery.	2	CO4
4	Memory Management and File system (4)		
4.1	Paging	1	CO5
4.2	Segmentation and Contiguous memory allocation	1	CO5
4.3	Virtual Memory - Demand Paging, Page replacement and Frame Allocation policies, Thrashing	1	CO5
4.4	File System - Concepts, Access Method, Directory Structure and File System Management	1	CO5
5	Disk management and Security (3)		
5.1	Disk Structure and Scheduling	1	CO6
5.2	File systems and operating system support for distributed systems	1	CO6
5.3	Security measures	1	CO6
6	Advanced Operating Systems (3)		
6.1	Needs & Design Goals – Distributed OS, Multiprocessor OS, Mobile OS, MAC OS.	3	CO2
	Total Hours	36	

List of Experiments

Module No.	Topic	No. of Lectures	Course Outcome
1.	Experiment to implement the following CPU scheduling algorithms a) FCFS b) SJF c) Round Robin d) Priority	4	CO3
2.	Program to implement producer – consumer problem using semaphores	2	CO3
3.	Program to implement Dining-philosophers problem	2	CO3
4.	Program to implement InterProcess Communication Using Pipes	2	CO3
5.	Program to implement the following contiguous memory allocation Techniques a) Worst fit b) Best fit c) First fit.	2	CO4
6.	Program to implement Bankers Algorithm for Dead Lock Avoidance.	2	CO4
7.	Program to implement Bankers Algorithm for Dead Lock Prevention	2	CO4
8.	Simulate all page replacement algorithms a) FIFO b) LRU c) OPTIMAL	2	CO5
9.	Simulate all File Organization Techniques a) Single level directory b) Two level directory	2	CO5
10.	Simulate all file allocation strategies a) Sequential b) Indexed c) Linked.	2	CO5
11.	Program to simulate disk scheduling algorithms. a) FCFS b) SCAN c) C-SCAN	2	CO6
Total Hours		24	

Course Designers:

1. Dr. K. NarasimhaMallikarjunan, Associate Professor, CSE arjunkambaraj@tce.edu
2. Ms. C.Santhiya ,Assistant Professor , CSE csit@tce.edu
3. Dr. G.Madhupriya, Associate Professor, CSE gmadhupriya@tce.edu

22CS371**DATA STRUCTURES LAB**

Category L T P Credit

PC 0 0 2 1

Preamble

With a dynamic learn-by-doing focus, this laboratory course encourages students to explore data structures by implementing them, a process through which students discover how data structures work and their applicability for toy problems. This course challenges students to exercise their creativity in both programming and analysis.

Prerequisite

22CS241 : Problem Solving using Computers

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct and Implement the stack and queue functionality for suitable applications.	TPS3	A	75
CO2	Implement the operations in linked list data structure for suitable applications	TPS3	A	75
CO3	Implement appropriate binary and multiway search tree for performing searching operations, with an understanding of the trade-off between the time and space complexity	TPS3	A	75
CO4	Implement heap tree for various applications	TPS3	A	75
CO5	Show the avoidance of collisions in the hash tables using collision resolution techniques including open and closed hashing techniques.	TPS3	A	75
CO6	Analyse various internal sorting techniques	TPS3	A	75
CO7	Implement of disjoint set and analyse their performance.	TPS3	A	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		M	L	L	L	L	L	L	M	M	L	L
CO2	S	M	L		M	L	L	L	L	L	L	M	M	L	L
CO3	S	M	L		M	L	L	L	L	L	L	M	M	L	L
CO4	S	M	L		M	L	L	L	L	L	L	M	M	L	L

CO 5	S	M	L		M	L	L	L	L	L	L	M	M	L	L
CO 6	S	M	L		M	L	L	L	L	L	L	M	M	L	L
CO 7	S	M	L		M	L	L	L	L	L	L	M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyse		
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

Module No.	Topic	CO
1	Array and List implementation of Stack, Queue and Circular Queue ADTs	CO1
2	Implementation of Evaluating Postfix Expressions, Infix to Postfix conversion using stack	CO1
3	Implementation of Josephus Problem using DLL	CO1
4	Implementation of Palindrome Checking using Stack and DLL	CO1
5	Implementation of Singly Linked List	CO2
6	Implementation of Polynomial Manipulation using Linked list	CO2
7	Implementation of Binary Search Trees-operations and traversal.	CO3
8	Implementation of AVL Trees	CO3
9	Implementation of Heaps using Priority Queues	CO4
10	Implementation of Open and Closed Hashing	CO5
11	Implementation internal sorting algorithms	CO6
12	Implementation of disjoint sets with smart union and path compression methods	CO7

Learning Resources

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson Education, 2005.
2. Kamthane, Introduction to Data Structures in C, 1st Edition, Pearson Education, 2007
3. Y. Langsam, M. J. Augenstein and A. N. Tanenbaum, "Data Structure Using C and C++", Pearson Education, 2nd Edition, 2015.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms", Fourth Edition, Mcgraw Hill/ MIT Press, 2022.

Course Designer(s):

- | | |
|---|----------------|
| 1. Dr.M.K.KavithaDevi, Professor,CSE | mkkdit@tce.edu |
| 2. Mrs.RajaLavanya, Assistant Professor,CSE | rlit@tce.edu |

22CS381	OBJECT ORIENTED PROGRAMMING LAB	Category	L	T	P	Credit
		PC	0	0	2	1

Preamble

This syllabus is intended for the Computer science students and enables them to learn Object Oriented Programming and the design of computer solutions in a precise manner. The experiments emphasize on OOP concepts, Functions, Polymorphism, Inheritance, I/O, event-driven, concurrent and network programming. The intention is to provide sufficient depth in these topics to enable candidates to apply Object Oriented Programming approach to programming.

Prerequisite

Programming Fundamentals

Course Outcomes

On the successful completion of the course students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Construct Object oriented programs using methods and passing arrays, objects, and array of objects to them	TPS3	B	80
CO2	Demonstrate Compile-time and Run-time polymorphism using object oriented programs	TPS3	B	80
CO3	Illustrate the relationships between objects using inheritance hierarchies and aggregation	TPS3	B	80
CO4	Develop Object Oriented programs to handle data using Java Collections, Files and Object Serialization	TPS3	B	80
CO5	Develop Object Oriented programs to handle exceptions	TPS3	B	80
CO6	Develop Object Oriented programs to demonstrate event driven programming, concurrent programming and network programming.	TPS3	B	80

Mapping with Programme Outcomes and Programme Specific Outcomes

Co s	P O 1	P O 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	P O 1	P O 1	P O 1	P S O	P S O	P S O
------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------

										0	1	2	1	2	3
CO1	S	M	L		L	L		L	M	M		L	M	L	L
CO2	S	M	L		L	L		L	M	M		L	M	L	L
CO3	S	M	L		L	L		L	M	M		L	M	L	L
CO4	S	M	L		L	L		L	M	M		L	M	L	L
CO5	S	M	L		L	L		L	M	M		L	M	L	L
CO6	S	M	L		L	L		L	M	M		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyse		
Evaluate		
Create		

List of Experiments with CO mapping

Ex. No	Experiment	CO
1.	Develop Object Oriented Program for passing arguments to a method and returning value from a method	CO1
2.	Develop Object Oriented Program for passing arrays and objects as arguments to method and returning objects from methods	CO1
3.	Construct Object Oriented Program for method overloading and constructor overloading	CO2
4.	Demonstrate aggregation and composition using object-oriented program	CO3
5.	Develop Object Oriented Program to demonstrate inheritance and overriding super class methods	CO2, CO3
6.	Develop Object Oriented Program to demonstrate abstract base classes abstract methods	CO3
7.	Construct Object Oriented Program to demonstrate File handling and Object Serialization	CO4
8.	Develop Object Oriented Program for manipulation of data using Collections in Java	CO4
9.	Construct Object Oriented Program to demonstrate exception	CO5

	handling	
10.	Develop event-driven programs using Java's delegation-based event model	CO6
11.	Develop concurrent programs using Java threads	CO6
12.	Develop network applications using Java sockets	CO6

Learning Resources

1. Herbert Schildt: "Java: The Complete Reference", Twelfth Edition, McGraw-Hill, 2021.
2. Tony Gaddis, "Starting Out with Java: From Control Structures through Objects", Sixth edition, Pearson Education Limited, 2016.
3. Bart Baesens, Aimee Backiel, SeppevandenBroucke, "Beginning Java Programming: The Object-Oriented Approach", John Wiley & Sons, 2015.
4. Kenneth L. Calvert and Michael J. Donahoo, "TCP/IP Sockets in Java: Practical Guide for Programmers", 2nd Edition. Elsevier, 2011.
5. Grady Booch, Robert Maksimchuk, Michael Engel, Bobbi Young, Jim Conallen, Kelli Houston "Object Oriented Analysis and Design with Applications", Third Edition, 2012

Course Designers:

1. DrM.Vijayalakshmi, Professor, CSE – mviji@tce.edu
2. MrS.Santhana Hari, Assistant Professor, CSE – sshcse@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

FOURTH SEMESTER

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programmes

COURSES OF STUDY

(For the candidates admitted from 2025-26 onwards)

FOURTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CS411	Discrete Mathematics	BSC	3	1	-	4
22CS421	Design and Analysis of Algorithms	PC	3	-	-	3
22CS431	Data Communication and Networks	PC	3	-	-	3
22CS441	Database Management Systems	PC	3	-	-	3
22CS491	Project Management	HSS	3	-	-	3
THEORY CUM PRACTICAL						
22CS451	Web Programming	PC	2	-	2	3
22EG661	Professional Communication	HSS	-	1	2	2
PRACTICAL						
22CS471	Databases Lab	PC	-	-	2	1
22CS481	Algorithms Lab	PC	-	-	2	1
AUDIT COURSE						
22CHAB1	Constitution of India	AC	-	-	-	-
Total			17	2	8	23

BSC : Basic Science Courses

PC : Professional Core Courses

ESC : Engineering Science Courses

L : Lecture

T : Tutorial

P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit

1 Hour Tutorial is equivalent to 1 credit

2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015**B.E. / B.Tech. Degree Programme****SCHEME OF EXAMINATIONS**

(For the candidates admitted from 2025-26 onwards)

FOURTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CS411	Discrete Mathematics	3	40	60	100	27	50
2	22CS421	Design and Analysis of Algorithms	3	40	60	100	27	50
3	22CS431	Data Communication and Networks	3	40	60	100	27	50
4	22CS441	Database Management Systems	3	40	60	100	27	50
5	22CS491	Project Management	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	22CS451	Web Programming	3	50	50	100	22.5	50
7	22EG661	Professional Communication	3	50	50	100	22.5	50
PRACTICAL								
8	22CS471	Databases Lab	3	60	40	100	18	50
9	22CS481	Algorithms Lab	3	60	40	100	18	50

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks

22CS411**DISCRETE MATHEMATICS**

Category L T P Credit

BSC 3 1 0 4

Preamble

A course in discrete mathematics elaborates discrete structures, which are the abstract mathematical structures used to represent discrete objects and relationships between these objects. These discrete structures include logic, predicate calculus, sets, functions, relations groups, rings and fields. An important problem-solving skill is the ability to count or enumerate objects. The discussion of enumeration in this course begins with basic techniques of counting. The general counting methods involve permutations and combinations and generating functions. These methods are very useful in constructing computer programs and in mastering many theoretical topics of computer science.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the concepts of sets, functions and relations to solve the problems in the field of science and technology.	TPS3	B	80
CO2	Translate statements in english into logical expressions and verify their equivalences.	TPS3	B	80
CO3	Introduce predicate calculus and apply logic rules of inference to check the validity of the propositional calculus, predicate calculus statements and to prove theorems.	TPS3	B	80
CO4	Apply the concept of recursion in recursive algorithms.	TPS3	B	80
CO5	Apply counting techniques to solve combinatorial problems	TPS3	B	80
CO6	Apply the concept of algebraic structures in coding theory.	TPS3	B	80

Mapping with Programme Outcomes

COs	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		M			L	M	L		M	M	L	L
CO2	S	M	L		M			L	M	L		M	M	L	L
CO3	S	M	L		M			L	M	L		M	M	L	L
CO4	S	M	L		M			L	M	L		M	M	L	L

CO5	S	M	L		M			L	M	L		M	M	L	L
CO6	S	M	L		M			L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assessment 1						Assessment 2						Terminal (%)		
	CAT 1 (%)			Assignment 1 (%)			CAT 2 (%)			Assignment 2 (%)					
TPS	1 10	2 30	3 60	1 -	2 -	3 100	1 10	2 30	3 60	1 -	2 -	3 100	-	30	70
CO1	3	10	20	-	-	40	-	-	-	-	-	-		6	12
CO2	-	10	20	-	-	30	-	-	-	-	-	-		3	12
CO3	-	10	20	-	-	30	-	-	-	-	-	-		3	12
CO4	7	-	-	-	-	-	3	-	14	-	-	20		3	10
CO5	-	-	-	-	-	-	7	20	30	-	-	60		12	12
CO6	-	-	-	-	-	-	-	10	16	-	-	20		3	12
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

Syllabus

Sets, Relations and Functions: Set- subsets- set operations – cardinality of sets – Relations and their properties – Representing relations – Equivalence relations. Functions-One-one function - Onto function – Bijective function –Inverse function.

Propositional calculus and Proof Techniques: Propositional logic – Applications of propositional logic – propositional equivalences – predicates and quantifiers – Nested quantifier – rules of inferences -Introduction to proofs.

Recursion: Recurrence relations, Recursion: Introduction - Recursively defined functions - Recursively defined sets and structures – Structural and general induction – Recursive algorithms.

Counting: The basics of counting – The Pigeonhole principle, the generalized Pigeonhole principle – Permutations and Combinations - Applications of recurrence relations - Solving linear recurrence relations: Homogeneous and non-homogeneous – Generating functions – Solving recurrence relations using generating functions.

Algebraic Structures: The structure of Algebras- Semigroups, Monoids and Groups, Abelian groups – Homomorphisms – Normal subgroups and congruence relations – Rings, Integral domains and Fields – Quotient and Product algebras – Coding theory – Polynomial codes.

Text Books

1. Kenneth H. Rosen., "Discrete Mathematics and Its Applications", 8th Edition, McGraw hill publications, 2019.

Reference Books & web resources

1. T.P. Tremblay and R. Manohar, "Discrete Mathematical Structures with application to Computer Science", Tata McGraw Hill, 2017.
2. Alan Tucker, "Applied Combinatorics", John Wiley & Sons, Incorporated, 2012.
3. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", Pearson, 2006.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Sets, Relations and Functions	
1.1	Set, Subsets set operations, cardinality of sets	1
1.2	Relations and their properties, Representing relations	1
	Tutorial	1
1.3	Equivalence relations	2
	Tutorial	1
1.4	Functions: one-one, onto, Bijective, inverse	2
2	Propositional Calculus and Proof Techniques	
2.1	Propositional logic	1
2.2	Applications of propositional logic	2
	Tutorial	1
2.3	Propositional equivalences	2
	Tutorial	1
2.4	Predicates and quantifiers	1
2.5	Nested quantifiers	1
2.6	Rules of inferences	2
2.7	Introduction to proofs	2
	Tutorial	1
3	Recursion	
3.1	Recurrence relations	1
3.2	Recursion: Introduction, recursively defined functions	1

Module No.	Topic	No. of Periods
	Tutorial	1
3.3	Recursively defined sets and structures, Structural and general induction	1
3.4	Recursive algorithms	1
	Tutorial	1
4	Counting	
4.1	The basics of counting	2
4.2	The pigeonhole principle, generalized pigeonhole principle, applications of pigeonhole principle	1
	Tutorial	1
4.3	Permutations and combinations	2
	Tutorial	1
4.4	Applications of recurrence relations	1
4.5	Solving linear recurrence relations	2
	Tutorial	1
4.6	Generating functions – Introduction, solve recurrence relations using generating functions.	2
5	Algebraic Structures	
5.1	The structure of algebras	1
5.2	Semigroups, Monoids and Groups, Abelian groups	1
	Tutorial	1
5.3	Homomorphisms, Normal subgroups, congruence relations	1
5.4	Rings, integral domains and fields, Quotient and product algebras	1
5.6	Coding theory and polynomial codes	1
	Tutorial	1
	Total	48

Course Designer(s):

1. Dr. C. S. Senthilkumar kumarstays@tce.edu
2. Dr. P. Victor pvmat@tce.edu
3. Dr. P. Krishnapriya pkamat@tce.edu

22CS421 **DESIGN AND ANALYSIS OF
ALGORITHMS**

Category L T P Credit
PC 3 0 0 3

Preamble

Algorithm design and analysis provide the theoretical backbone of computer science. On completion of this course students will be able to: i) Determine the asymptotic time complexity of algorithms ii) Write rigorous correctness proofs for algorithms iii) Use different paradigms of problem solving to illustrate efficient ways of solving a given problem

Prerequisite

- 22CS241: Problem Solving and Programming
- 22CS341: Data Structures and Algorithms

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify the problem and its complexity using frameworks like recurrences and amortized analysis.	TPS3	B	75
CO2	Apply graph algorithms to solve problems and analyze their efficiency.	TPS3	B	75
CO3	Analyze algorithm design techniques like divide and conquer, dynamic programming to solve problems.	TPS4	B	75
CO4	Apply design principles for developing solutions using greedy algorithm approaches.	TPS3	B	75
CO5	Use the state space tree method like backtracking and branch & bound techniques for solving problems.	TPS4	B	75
CO6	Determine the significance of NP complete problems and approximation algorithms.	TPS3	B	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L					L	M	M		M	M		M
CO2	S	M	L					L	M	M		M	M		M
CO3	S	S	M	L				L	M	M		M	M		M
CO4	S	M	L					L	M	M		M	M		M
CO5	S	S	M	L				L	M	M		M	M		M

CO 6	S	M	L					L	M	M		M	M		M
---------	---	---	---	--	--	--	--	---	---	---	--	---	---	--	---

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Assi g nme nt-1	CAT2						Assi g nme nt-2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	8	5	20				30								6		10			
CO2	8	5	20				30								2	5	10			
CO3	4	10		20			40								2	5		10		
CO4								8	5	20				30	2	5	10			
CO5								4	10		20			40	2	5		10		
CO6								8	5	20				30	6		10			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

INTRODUCTION:

Algorithm analysis: Time and space complexity - Asymptotic Notations and its properties Best case, Worst case and average case analysis – **Amortized Analysis:** Aggregate, Accounting and Potential Method – **Recurrence relation:** substitution method, recursion tree method, Master Theorem – **Searching and sorting algorithms analysis:** linear search, binary search and Insertion sort, heap sort.

GRAPH ALGORITHMS

Elementary Graph algorithms: Representations of graphs - Breadth-first search - Depth-first search - Topological sort - Strongly connected components - **Minimum spanning tree:** Kruskal's and Prim's algorithm, **Shortest path:** Bellman-Ford algorithm - Dijkstra's algorithm - Floyd-Warshall algorithm **Network flow:** Flow networks - Ford-Fulkerson method – **Matching:** Maximum bipartite matching

ALGORITHM DESIGN TECHNIQUES

Divide and Conquer methodology: Find maximum and minimum numbers - Merge sort - Quick sort – Strassen's Matrix Multiplication Algorithm – Karatsuba integer multiplication Algorithm. **Dynamic programming:** Elements of dynamic programming — Matrix-chain multiplication — Optimal Binary Search Trees – Knapsack Problem. **Greedy Technique:** Elements of the greedy strategy - Activity-selection problem — Huffman Trees – Knapsack Problem.

STATE SPACE SEARCH ALGORITHMS

Backtracking: n-Queens problem - Hamiltonian Circuit Problem - Subset Sum Problem – Graph coloring problem **Branch and Bound:** Solving 15-Puzzle problem - Assignment problem – Knapsack Problem - Travelling Salesman Problem

NP-COMPLETE ALGORITHMS

Tractable and intractable problems: Polynomial time algorithms – Polynomial time verification - Venn diagram representation – NP algorithms - NP-hardness and NP-completeness – Bin Packing problem - Problem reduction: TSP – 3- CNF problem

Text Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 3rd Edition, Prentice Hall of India, 2009.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran "Computer Algorithms/C++" Orient Blackswan, 2nd Edition, 2019.

Reference Books & web resources

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson Education, 2012.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Reprint Edition, Pearson Education, 2006.
3. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press, 2014.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	INTRODUCTION	
1.1	Time and space complexity- Asymptotic Notations and its properties Best case, Worst case and average case analysis	1
1.2	Amortized Analysis: Aggregate, Accounting and Potential Method with example	1
1.3	Recurrence relation: substitution method, Master Theorem	2
1.4	Searching: linear search, binary search and Interpolation Search	1
1.5	Pattern search: The naïve string matching algorithm	1
1.6	Rabin-Karp algorithm - Knuth-Morris-Pratt algorithm	1
1.7	Sorting: Insertion sort – heap sort	1
2	GRAPH ALGORITHMS	
2.1	Shortest path: Bellman-Ford algorithm	1
2.2	Dijkstra's algorithm - Floyd-Warshall algorithm	2
2.3	Network flow: Flow networks - Ford-Fulkerson method	1
2.4	Matching: Maximum bipartite matching	1

Module No.	Topic	No. of Periods
3	ALGORITHM DESIGN TECHNIQUES	
3.1	Divide and Conquer methodology: Finding maximum and minimum - Merge sort, Quick sort	2
3.2	Strassen's Matrix Multiplication Algorithm – Karatsuba integer multiplication Algorithm	2
3.3	Dynamic programming: Elements of dynamic programming — Matrix-chain multiplication	1
3.4	Optimal Binary Search Trees – Knapsack Problem	2
3.5	Greedy Technique: Elements of the greedy strategy - Activity-selection problem	1
3.6	Huffman Trees – Knapsack Problem	2
4	STATE SPACE SEARCH ALGORITHMS	
4.1	Backtracking: n-Queens problem - Hamiltonian Circuit Problem	1
4.2	Subset Sum Problem – Graph coloring problem	1
4.3	Branch and Bound: Solving 15-Puzzle problem - Assignment problem	2
4.4	Knapsack Problem - Travelling Salesman Problem	1
5	NP-COMPLETE AND APPROXIMATION ALGORITHMS	
5.1	Tractable and intractable problems: Polynomial time algorithms – Venn diagram representation	1
5.2	NP algorithms - NP-hardness and NP-completeness	1
5.3	Bin Packing problem - Problem reduction: TSP – 3-CNF problem	2
5.4	Approximation Algorithms: Travelling Salesman Problem	1
	Total	36

Course Designer(s):

- | | |
|--|----------------|
| 1. Dr.M.K.KavithaDevi, Professor,CSE | mkkdit@tce.edu |
| 2. Mrs.RajaLavanya,
Assistant Professor,CSE | rlit@tce.edu |

22CS431 DATA COMMUNICATION AND NETWORKS

Category L T P Credit
PC 3 0 0 3

Preamble

This course on Computer Network provides an introduction to the basic concepts in networks, reference models, layers, protocols, switching, routing and applications that use Computer Networks. This course introduces the concepts in Computer Networks with emphasis to different layers and the functionality of TCP/IP protocol suite. At the end of the course, the students should have an understanding of the basic principles and practice of Computer Networking.

Prerequisite

22CS221 Electronics and Digital Systems

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome Statement	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the operation of network applications with an understanding of the network models, switching techniques and layered architecture.	TPS2	70	85
CO2	Illustrate the concepts for transmission, and multiplexing techniques.	TPS3	70	80
CO3	Solve flow and error control issues in the data link layer, using appropriate techniques.	TPS3	70	80
CO4	Identify the performance implications of multiple access protocols.	TPS3	70	80
CO5	Construct routing and forwarding solutions for packet switching networks, with an understanding of the router architectures, algorithms and protocols.	TPS3	70	80
CO6	Identify the performance of transport layer protocols under given scenario.	TPS4	70	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-										M		
CO2	S	M	L		L			L	L	L		M	M		L
CO3	S	M	L		L			L	L	L		M	M		L
CO4	S	M	L		L			L	L	L		M	M		L
CO5	S	M	L		L			L	L	L		M	M		L
CO6	S	M	L	L	L			L	L	L		M	M	L	L

Assessment Pattern

CO	CAT1						CAT2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20						20					5	20				
CO2	10	10	20										5	5	10			
CO3		10	20											5	10			
CO4								20	20					10	10			
CO5									20						10			
CO6										20						10		
CO	Assignment 1						Assignment 2											
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6						
CO1																		
CO2																		
CO3			100															
CO4																		
CO5										50								
CO6												50						

Syllabus

Network Models - The OSI Model - TCP/IP Protocol Suite - OSI versus TCP/IP

Switching - Circuit-Switched Networks - Packet Switching, Structure of a Switch, Switching and TCP/IP Layers

Network Types: WAN, MAN, LAN, WLAN - Infrastructured and Ad hoc networks

Physical Layer – Transmission techniques, Multiplexing - TDM, FDM, WDM

Datalink Layer: Framing, Error Detection and Correction: Types of Errors - Error Detection, Cyclic Redundancy Check, Checksum, Forward Error Correction - Data link Control - Media Access Control (MAC): CSMA/CD, CSMA/CA– Channelization: FDMA, TDMA, CDMA

Network Layer - Connecting Devices: Hubs, Link-Layer Switches, Routers – Routing and Forwarding: Distance-Vector Routing, Link-State Routing - Performance Metrics: Delay, Throughput, Packet Loss, Congestion Control - Addressing: Internet Protocol, IPV6, IPV4 Addresses: Classful Addressing, Classless Addressing – Internet Control Message Protocol (ICMP) - Address Resolution Protocol (ARP) - RARP

Transport Layer: Transmission Control Protocol, User Datagram Protocol

Presentation Layer: Encryption/Decryption, Authentication, Translation

Application Layer: FTP - TELNET HTTP- Secure Shell (SSH) - Domain Name System (DNS)

Text Book

1. Data Communications and Networking, 5th Edition, BehrouzForouzan, Mc Graw Hill, 2017

Reference Books& web resources

1. Computer Networks: A Systems Approach, Larry L. Peterson, Bruce S. Davie, Elsevier, Mar 2011
2. Computer Networking: A Top-Down Approach featuring the Internet, 6th edition, James F. Kurose, Pearson Education India, 2013.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
------------	-------	--------------	----------------

1.	Network Models		
1.1	The OSI Model - TCP/IP Protocol Suite - OSI versus TCP/IP	2	CO1
2	Switching		
2.1	Circuit-Switched Networks	1	CO1
2,2	Packet Switching, Structure of a Switch, Switching and TCP/IP Layers	1	CO1
3	Network Types		
3.1	WAN, MAN, LAN	1	CO1
3.2	WLAN - Infrastructured and Ad hoc networks	1	CO1
	Physical Layer		
4.1	Transmission techniques	1	CO2
4.2	TDM, FDM, WDM	2	CO2
5	Datalink Layer		
5.1	Framing	1	CO3
5.2	Error Detection and Correction - Types of Errors, Error Detection, Cyclic Redundancy Check, Checksum, Forward Error Correction	2	CO3
5.3	Datalink Control	2	CO4
5.4	Media Access Control (MAC): CSMA/CD(Ethernet), CSMA/CA, Controlled Access	1	CO4
5.5	Channelization: FDMA, TDMA, CDMA	2	CO4
6	Network Layer		
6.1	Connecting Devices: Hubs, Link-Layer Switches, Routers	2	CO5
6.2	Routing and Forwarding-Distance-Vector and Link-State Routing	2	CO5
6.3	Performance Metrics : Delay, Throughput, Packet Loss, Congestion Control	2	CO5
6.4	Addressing: Internet Protocol –IPV6, IPV4 Addresses: Classful Addressing, Classless Addressing	2	CO5
6.5	Internet Control Message Protocol(ICMP), Address Resolution Protocol (ARP), RARP	2	CO5
7	Transport Layer		
7.1	Transmission Control Protocol	2	CO6
7.2	User Datagram Protocol	1	CO6
8	Presentation Layer		
8.1	Encryption/Decryption	1	CO1
8.2	Authentication, Translation	1	CO1
9	Application Layer		
9.1	FTP	1	CO1
9.2	TELNET	1	CO1
9.3	HTTP,Secure Shell (SSH)	1	CO1
9.4	Domain Name System (DNS)	1	CO1
	Total	36	

Course Designer(s):

1. Dr.C.Senthilkumar, Asso.Prof., CSE
2. Dr.G.S.R.EmilSelvan, APCSE

cskcse@tce.edu
emil@tce.edu

22CS441 **DATABASE MANAGEMENT
SYSTEMS**

Category L T P Credit
PC 3 0 0 3

Preamble

This course provides the students to understand the various concepts and functionalities of Database Management Systems. It includes various data models to store & retrieve data and the use of query languages, the effective designing of relational database, creating storage and access structures. Also, the students will learn about the concepts of transaction management, concurrency mechanisms and recovery procedures. This course also introduces about different forms of NoSQL databases.

Prerequisite

22CS341 – Data Structures and Algorithms

Course Outcomes

On the successful completion of the course, students will be able to

	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Develop data models such as Entity Relationship (ER) model and Relational model for a given application requirement. (Apply)	TPS3	B	85
CO2	Manipulate relational database using Relational languages and PL/SQL sub programs. (Apply)	TPS3	B	85
CO3	Design normalized databases for a given application by incorporating various constraints and normal forms. (Apply)	TPS3	B	85
CO4	Construct data structures like indexes and hash tables to support fast retrieval of data and explain query processing and optimization methods. (Apply)	TPS3	B	85
CO5	Make use of different forms of transactions, concurrency control and recovery mechanisms to maintain data consistency in a multi user environment. (Apply)	TPS3	B	85
CO6	Demonstrate NoSQL models for building the databases. (Apply)	TPS3	B	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	S	M	L		M	L		M	M	M		M	M	L	L
CO2.	S	M	L		M	L		M	M	M		M	M	L	L

CO3.	S	M	L		L	L		M	M	M		M	M	L	L
CO4.	S	M	L						M	M		M	M		L
CO5.	S	M	L		L				M	M		M	M		L
CO6.	S	M	L		L	L			M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1			Assignment 1			CAT2			Assignment 2			Terminal		
TPS Scale	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
CO1	5	10	15			20							1	5	10
CO2	5	10	25			60							2	5	15
CO3	5	10	15			20							2	5	10
CO4							5	10	25			45	2	5	10
CO5							5	10	25			45	2	5	10
CO6							5	5	10			10	1	5	5
Total	15	30	55				15	25	60				10	30	60

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to database and Data Models: Purpose of database system, System Architecture, Components of DBMS, Data Models – ER model, Relational model - Concept of relation, Constraints, Mapping ER model to Relational Model

Relational Languages - Relational algebra, Structured Query Language - DDL, DML – Set operations, Aggregate Functions., Nested Subqueries. Intermediate SQL – Joins, Views, Index, Transactions, Integrity Constraints, Authorization. Advanced SQL – Introduction to PL/SQL – Procedures, Functions and Triggers, Accessing SQL from Programming Languages, Recursive Queries.

Database design: Decomposition, Functional Dependencies - Closure set of FD's for key identification, Decomposition – Lossless decomposition, Normal forms – 1NF, 2NF, 3NF, Dependency Preserving – BCNF, Multivalued Dependencies and 4NF, Join dependencies and 5NF.

Transaction and Concurrency control - Transaction concepts, Concurrent Execution, Serializability, Concurrency Control - Lock based protocol, Deadlock handling. Recovery System - Log-based Recovery, Recovery with Concurrent Transaction.

Data Storage and structures: RAID levels, Database Compression, De-Duplication, File Organization - Indexing – Ordered Index - B+ tree – properties, insertion and deletion, Hashing - Static and dynamic hashing.

Query Processing and Optimization Basics of query processing and optimization.

No SQL databases: CAP Theorem – Document Based systems – Key value Stores – Column Based Systems – Graph Databases. **Database Security:** Security issues – Access control based on privileges – Role Based access control – SQL Injection

Text Book

1. Silberschatz, A, Henry F. Korth, and S. Sudharshan, “Database System Concepts”, 7th Edition, Tata McGraw Hill, 2019.

Reference Books & web resources

1. RamezElmasri and Shamkant B. Navathe, “Fundamentals of Database Systems”, 7th Edition, Pearson/Addison Wesley, 2016
2. C. J. Date, A. Kannan and S. Swamynathan, “An Introduction to Database Systems”, 8th ed, Pearson Education, 2006
3. Raghu Ramakrishnan, “Database Management Systems”, Third Edition, McGraw Hill, 2002.
4. Pramod J. Sadalage, Martin Fowler, “NoSQL Distilled - A Brief Guide to the Emerging World of Polyglot Persistence”, Pearson Education, 2012.
5. Andreas Meier, Michael Kaufmann, “ SQL & NoSQL Databases - Models, Languages, Consistency Options and Architectures for Big Data Management” , Springer Fachmedien Wiesbaden, 2019.
6. <https://nptel.ac.in/courses/106105175> (Database Management System, IIT Kharagpur)
7. <https://nptel.ac.in/courses/106/106/106106220>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to database and Data Models (5)	
1.1	Purpose of database system	1
1.2	System Architecture, Components of DBMS	
1.3	Data Models – ER model	1
1.4	Relational model - Concept of relation, Constraints	1
1.5	Mapping ER model to Relational Model	2
2	Relational Languages (9)	
2.1	Relational algebra	1
2.2	Structured Query Language - DDL, DML	1
2.3	Set operations, Aggregate Functions	1
2.4	Nested Subqueries	1

Module No.	Topic	No. of Periods
2.5	Intermediate SQL – Joins, Views, Index	1
2.6	Transactions, Integrity Constraints, Authorization	1
2.7	Advanced SQL – Introduction to PL/SQL – Procedures, Functions and Triggers	2
2.8	Accessing SQL from Programming Languages	1
2.9	Recursive Queries	
3	Database design (6)	
3.1	Decomposition – Lossless decomposition	1
3.2	Functional Dependencies - Closure set of FD's for key identification	1
3.3	Normal forms – 1NF, 2NF, 3NF	1
3.4	Dependency Preserving – BCNF, Multivalued Dependencies and 4NF	2
3.5	Join dependencies and 5NF	1
4	Transaction and Concurrency control (6)	
4.1	Transaction concepts	1
4.2	Concurrent Execution, Serializability	2
4.3	Concurrency Control - Lock based protocol, Deadlock handling	2
4.4	Recovery System - Log-based Recovery, Recovery with Concurrent Transaction	1
5	Data Storage and structures (6)	
5.1	RAID levels	1
5.2	Database Compression, De-Duplication	
5.3	File Organization - Indexing – Ordered Index	1
5.4	B+ tree – properties , insertion, deletion	2
5.5	Hashing - Static and dynamic hashing	1
5.6	Query Processing and Optimization - Basics of query processing and optimization	1
6	Introduction to No SQL databases (4)	
6.1	Introduction – CAP Theorem	1
6.2	Document Based systems – Key value Stores	1

Module No.	Topic	No. of Periods
6.3	Column Based Systems – Graph Databases	1
6.4	Database Security: Security issues – Access control based on privileges, Role Based access control – SQL Injection	1
	Total	36

Course Designer(s):

1. Dr.B.Subbulakshmi
2. Mr. M. Sivakumar

bscse@tce.edu
mskcse@tce.edu

22CS451	WEB PROGRAMMING
----------------	------------------------

Category	L	T	P	Credit
PC	2	0	2	3

Preamble

This course provides the basic principles and techniques used for developing Web based application. The objective of the course is to make students to study client side and server side programming language concepts. This course provides semi structured representation of data and transport data using XML related technologies. This course provides the techniques used for creating, publishing and consuming a web service in a web based application

Prerequisite

Programming Fundamentals

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome 1 (CO1)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Model web pages using HTML and DHTML (Apply)	TPS3	A	80
CO2	Demonstrate client side scripting which interact with users (Apply)	TPS3	A	80
CO3	Build user interfaces using React.(Apply)	TPS3	A	80
CO4	Illustrate multi-tier applications using JSP (Apply)	TPS3	A	80
CO5	Illustrate multi-tier applications using Node JS and Express framework (Apply)	TPS3	A	80
CO6	Develop web application using XML related technologies. (apply)	TPS3	A	80

Mapping with Programme Outcomes

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO1	S	M	L	-									M	L	L
CO2	S	M	L	-	M		L	L	L	L		L	M	L	L
CO3	S	M	L	-	M		L	L	L	L		L	M	L	L
CO4	S	M	L	-	M		L	L	L	L	L	L	M	L	L
CO5	S	M	L	-	M		L	L	L	L		L	M	L	L
CO6	S	M	L	-	M		L	L	L	L	L	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1(Theory Component)						CAT2(Theory Component)						Model (Lab Component)			OCR			Terminal(Theory Component)					
	100						100						100			100			100					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	1	2	3	1	2	3	4	5	6
CO1	10		10												10			10	2	4	10			
CO2	10	10	10												20			20	2	4	10			
CO3	10	20	20												20			20	4	4	10			
CO4							10		10						20			20	4	4	10			
CO5							10	10	10						20			20	4	4	10			
CO6							10	20	20						10			10	4		10			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Web page design: HTML5: headers, Linking, Images, Image map, meta elements, frameset, HTML forms, cascading style sheet., DHTML: object model, Event model, Two tier and Three tier architecture, J2EE architecture, HTTP request types.

Client-side programming: Java script: introduction, control statements, functions, objects. Event handling. JQuery: HTML and DOM manipulation, HTML event methods, AJAX.

Building User Interface: React –API, Classes, Arrow Function, Form, List, Sass Styling

Server-Side programming: JSP introduction, programming, Servlets - Introduction, Architecture, Programming with database connectivity.

JavaScript on Server: Node.js – Node JS HTTP Module, File system, URL Module, Events, Upload files, Introduction to the Express framework – Server-side rendering with Templating Engines – Static Files - async/await - Fetching JSON from Express.

XML: XML basics, DTD, XML Schema, XML Parser, XPATH, XMLQuery.

Text Book

1. David Flanagan, "Java Script: The Definitive Guide", O'Reilly Media, Inc, 7th Edition, 2020
2. Matt Frisbie, "Professional JavaScript for Web Developers", Wiley Publishing, Inc, 4th Edition, ISBN:978-1-119-36656-0, 2019
3. Deitel and Deitel, "Internet and World Wide Web How to Program", Prentice Hall of India, Fifth Edition, 2018.
4. Paul J.Deitel and Harvey M.Deitel, "AJAX, Rich Internet Applications, and Web Development for Programmers", Pearson Education, First Edition, 2009.
5. Alex Banks, Eve Porcello, "Learning React", O'Reilly Media, Inc, 2nd Edition, 2020

6. Marc Wandschneider, "Learning Node", Addison-Wesley Professional, 2nd Edition, 2016

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Web page design:	
1.1	HTML5: headers, Linking, Images, Image map, meta elements, frameset, HTML forms, cascading style sheet.,	1
1.2	DHTML: object model, Event model, Two tier and Three tier architecture, J2EE architecture, HTTP request types.	1
2	Client-side programming:	
2.1	Java script: introduction, control statements, functions, objects. Event handling.	1
2.2	JQuery: HTML and DOM manipulation, HTML event methods, AJAX,	2
3	Building User Interface:	
3.1	React –Classes, Arrow Function, Form,	2
3.2	List, Sass Styling	1
4	Server-Side programming	
4.1	JSP introduction, programming,	1
4.2	Servlets - Introduction, Architecture,	2
4.3	Programming with database connectivity.	2
5	Javascript on Server:	
5.1	Node.js – Node JS HTTP Module,	1
5.2	File system, URL Module,	1
5.3	Events, Upload files,	1
5.4	Introduction to the Express framework	1
5.5	Server-side rendering with Templating Engines	1
5.6	Static Files - async/await, Fetching JSON from Express	2
6	XML:	
6.1	XML basics, DTD, XML Schema,	2

Module No.	Topic	No. of Periods
6.2	XML Parser, XPATH, XMLQuery	2
	Total	24

List of Experiments:

S. No.	Experiments	CO mapping	No. of Periods
1.	Implement Client-side form design and validation.	CO1	2
2.	Create a Webpage to handle Events and Objects using Java Script.	CO2	2
3.	Develop application to demonstrate applications of JQuery and Ajax.	CO2	4
4.	Develop client-side application using React API	CO3	4
5.	Develop a multitier application using JSP	CO4	4
6.	Illustrate multi-tier applications using Node JS and Express framework	CO5	4
7.	Implement a program for DOM to Process XML File	CO6	4
		Total	24

Course Designer(s):

1. Mr. M. Siva Kumar, mscse@tce.edu
2. Dr. M. Vijayalakshmi, mviji@tce.edu

22CS471	DATABASES LAB
----------------	----------------------

Category	L	T	P	Credit
PC	0	0	2	1

Preamble

This course aims at facilitating the student to design database applications for the real world problems and perform operations such as creation, manipulation and maintenance of databases using RDBMS tools. It facilitates the students to access databases through high level languages using appropriate APIs. This course also enables students to work with NoSQL databases and execute basic query operations over it.

Prerequisite

Concepts of Databases

Course Outcomes

On the successful completion of the course, students will be able to

	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Create Entity Relationship and Relational models for the given real-world application.	TPS3	A	75
CO2	Design normalized database using various constraints.	TPS3	A	75
CO3	Construct relational database and manipulate them using simple and complex queries in SQL	TPS3	A	75
CO4	Create and use different database objects like Index, View, Sequence, abstract data types , Varray and Nested table using SQL	TPS3	A	75
CO5	Develop sub programs like Procedure, Functions, Triggers and Package using PL/SQL and manipulate the database through these programs	TPS3	A	75
CO6	Develop a complete database application using higher level language through JDBC /ODBC	TPS3	A	75
CO7	Develop NoSQL database and demonstrate execution of simple queries on it.	TPS3	A	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		M	M	M		L	M	L	M
CO2	S	M	L		M	L		M	M	M		L	M	L	M
CO3	S	M	L		S	L		M	M	M		L	M	M	M
CO4	S	M	L		S			M	M	M		L	M	M	M

CO5.	S	M	L		S	L		M	M	M		L	M	M	M
CO6.	S	M	L		S	L		M	M	M		L	M	M	M
CO7.	S	M	L		M	L		M	M	M		L	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyse		
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

Module No.	Topic	Course Outcomes
1.	Analyse the database application and create relational databases using normalization principles.	CO1
2.	Add the necessary integrity constraints to relational databases and practice with DCL commands.	CO2
3.	Manipulate the database using simple SQL Queries and practice with TCL commands such as COMMIT, ROLLBACK and SAVEPOINT commands	CO3
4.	Manipulate the database using Complex SQL Queries	CO3
5.	Practice with different database objects and complex data types such as BLOB, CLOB, NCLOB, BFILE	CO4
6.	Practice with PL/SQL blocks, programming constructs and composite data types in PL/SQL	CO5
7.	Develop sub programs such as procedures, functions with cursors and demonstrate exceptions using PL/SQL.	CO5
8.	Create Packages and Triggers using PL/SQL	CO5
9.	Develop a web application with database connectivity	CO6
10.	Demonstrate NoSQL database models and execute queries.	CO7
11.	Demonstration of a web application and Report submission	CO1..CO6

Students can take any real time application, create databases and execute queries over it. The application demonstration can be shown at the end of the lab. The sample applications can be:

1. College Management System
2. Railway Reservation System
3. Hospital Management System
4. Inventory Management System
5. Library Management System

Learning Resources

1. <https://www.w3schools.com/sql>
2. <https://www.tutorialspoint.com/sql/index.htm>
3. <https://practice.geeksforgeeks.org/tag-page.php?tag=SQL>
4. <https://www.hackerrank.com/domains/sql>
5. <https://www.hackerearth.com/practice/>

Course Designers:

- | | | |
|----|-------------------|----------------|
| 1. | Mr. M.Sivakumar | mskcse@tce.edu |
| 2. | Dr.B.Subbulakshmi | bscse@tce.edu |

22CS481**ALGORITHMS LAB**

Category	L	T	P	Credit
PC	0	0	2	1

Preamble

The objective of this laboratory course is to enable students to solve algorithmic problems by choosing and/or designing efficient data structures and algorithms to meet the problem constraints and implementing the algorithm in C/C++ and Python.

Prerequisite

- 22CS241: Problem Solving and Programming
- 22CS341: Data Structures and Algorithms
- 22CS371: Data Structures Lab

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Determine the time complexity of various sorting and searching techniques.	TPS3	A	70
CO2	Apply graph algorithms to solve problems and analyze their efficiency.	TPS3	A	70
CO3	Make use of divide and conquer algorithm design techniques like to solve problems	TPS4	A	70
CO4	Use of dynamic programming algorithm design techniques to solve problems	TPS3	A	70
CO5	Identify the problem and design the algorithm using greedy techniques.	TPS3	A	70
CO6	Use the state space tree method for solving problems	TPS4	A	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		M	M	M	M	S	M	M	M	M	M	M
CO2	S	M	L		M	M	M	M	S	M	M	M	M	M	M
CO3	S	S	M	L	M	M	M	M	S	M	M	M	M	M	M
CO4	S	M	L		M	M	M	M	S	M	M	M	M	M	M
CO5	S	M	L		M	M	M	M	S	M	M	M	M	M	M
CO6	S	S	M	L	M	M	M	M	S	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyse		
Evaluate		
Create		

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
	Searching and Sorting Algorithms	
1	Implement Linear Search and Binary search. Analyse their worst-case performance by determine the time required to search for an element. Repeat the experiment for different values of n (in the range 100 to 100000, which is stored in a separate file), the number of elements in the list to be searched and plot a graph of the time taken versus n.	CO1
2	Sort a given set of elements using the Insertion sort and Merge sort methods and determine the time required to sort the elements. Repeat the experiment for different values of n (in the range 100 to 100000, which is stored in a separate file), the number of elements in the list to be sorted and plot a graph of the time taken versus n.	CO1
	Graph Algorithms	
3	For a given unweighted / uniform weighted connected graph, implement (a) BFS to find single source shortest path (b) DFS to detect cycle	CO2
4	From a given vertex in a weighted connected graph, develop a program to find the shortest paths.	CO2
5	From a given vertex in a weighted connected graph, implement a program to find the minimum spanning tree.	CO2
6	Compute the transitive closure of a given directed graph using Warshall's algorithm.	CO2
	Algorithm Design Techniques	
7	Develop a program to find out the maximum and minimum numbers in a given list of n numbers using the divide and conquer technique.	CO3
8	Implement Merge sort and Quick sort methods to sort an array of elements and determine the time required to sort. Repeat the experiment for different values of n ((in the range 100 to 100000, which is stored in a separate file), the number of elements in the list to be sorted and plot a graph of the time taken versus n.	CO3
9	Implement matrix chain multiplication using dynamic programming	CO4

Module No.	Topic	No. of Periods
10	Implementation of 0/1 knapsack problem using knapsack problem	CO4
11	Implementation of fractional knapsack problem using greedy algorithm and analyse its performance for various item selection strategies	CO5
12	Develop Huffman codes using greedy approach.	CO5
	State Space Search Algorithms	
13	Implement N Queens / Hamiltonian Circuit Problem / Subset Sum Problem / Graph coloring problem using Backtracking approach.	CO6
14	Implement Assignment problem / Knapsack Problem / Travelling Salesman Problem using Branch and Bound technique	CO6

Learning Resources

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson Education, 2012.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Reprint Edition, Pearson Education, 2006.
3. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press, 2014.
4. <https://www.geeksforgeeks.org/fundamentals-of-algorithms/>
5. <https://www.hackerrank.com/domains/algorithms>
<https://www.codechef.com/wiki/tutorials>
6. Steven S. Skiena, The Algorithm Design Manual, Second Edition, Springer, 2010

Course Designer(s):

1. Dr.M.K.KavithaDevi, Professor,CSE mkkdit@tce.edu
2. Mrs.RajaLavanya, Assistant Professor,CSE rlit@tce.edu

22CS491**PROJECT MANAGEMENT**

Category	L	T	P	Credit
HSS	3	0	0	3

Preamble

This course develops students the principles underlying effective project management, providing the knowledge, skills, and framework necessary to manage a real project in the workplace.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Suggest an organizational structure for managing projects within the organisation	TPS3	70	85
CO2	Develop a project communication plan by defining its scope, priorities, and responsibility matrices	TPS3	70	85
CO3	Construct a work breakdown structure for a given business cases and derive a project network information.	TPS3	70	85
CO4	Develop a project schedule using critical path method and to develop a Gantt chart using any project management tool	TPS3	70	85
CO5	Develop a reschedule for a project based on constraints. Develop a suitable risk response based on the assessment.	TPS3	70	85
CO6	Plan and implement a team-project for developing a complete project schedule using Project management tools like Open Projects, MS project management.	TPS4	70	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L					S	S	S	S	S	M	M	S
CO2	S	M	L		S			S	S	S	S	S	M	M	S
CO3	S	M	L		S			S	S	S	S	S	M	M	S
CO4	S	M	L		S			S	S	S	S	S	M	M	S
CO5	S	M	L					S	S	S	S	S	M	M	S
CO6	S	S	M	L	S			S	S	S	S	S	S	M	S

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1			CAT 2			Assignment 1				Assignment 2				Terminal Examination		
TPS Scale	1	2	3	1	2	3	3	4	5	6	3	4	5	6	1	2	3
CO1	5	9		-	-	-	-	-	-	-	-	-	-	-	4	7	-
CO2	6	12	25	-	-	-	-	-	-	-	-	-	-	-	4	7	-
CO3	6	12	25	-	-	-	-	-	-	-	-	-	-	-	4	7	15
CO4	-	-	-	8	16	25	-	-	-	-		-	-	-	4	7	15
CO5	-	-	-	9	17	25	-	-	-	-		-	-	-	4	7	15
CO6	-	-	-				100	-	-	-	100	-	-	-	-	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**Modern Project Management**

The Project Management Body Of Knowledge (PMBOK), 10 Knowledge Areas, The Project Life Cycle, Organization Structure and Project Management, Organizing Projects within the functional organization, Matrix organisation and projectized organisation, Choosing the right project management structure.

Project Initiation

Project Scope, and Checklist, Project Priorities, Work Breakdown Structure Development, WBS coding, Responsibility Matrices in WBS, Project Communication Plan, Project Estimations

Agile Project Management, Differences between managing a project versus developing a product, Scrum - Defining the Product Vision and Product Roadmap, Planning Releases and Sprints.

Project Planning

Developing the Project Network, WBS to Project Network, Network Computation Process Networks Activity on Arrow, Activity-on-Node, Forward Pass, Earliest Times, Backward Pass—Latest Times, Use of Lags, Lag Relationships, Laddering

Resource optimization, Scheduling Resources and Costs, Resource Scheduling Problem, Time-Constrained Project, Resource-Constrained Projects. **Managing Risk**, Risk Management Process, Risk Identification, Assessment, and Response Development.

Conflict management, **Project Closure**, Types of Project Closure, Final Report, Post-Implementation Evaluation.

Text Books

1. Erik W. Larson, Clifford F. Gray, "Project Management The Managerial Process", McGraw-Hill/Irwin, Seventh Edition, 2018.
2. Mark C. Layton, Steven J Ostermiller, and Dean J. Kynaston "Agile Project Management For Dummies" John Wiley & Sons, Inc 3rd Edition 2020

Reference Books & web resources

1. Project Management Institute, "A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition and The Standard for Project Management", 2021
2. Jack R. Meredith, Samuel J. Mantel, Jr., "Project management A Managerial Approach"., John Wiley & Sons, Inc. Tenth Edition, 2017
3. Harold Kerzner, "Project Management A systems approach to Planning, scheduling, And controlling", Tenth edition, John Wiley & Sons, Inc. 2009
4. Harold Kerzner, "Project management best practices achieving global excellence", Fourth edition, John Wiley & Sons, Inc. 2018
5. A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Fifth Edition, Project Management Institute.
6. Harold Koontz, Heinz Weihrich "Essentials of Management", Tata McGraw-Hill Education, 2006 - Management

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods	CO
1	Modern Project Management		
1.1	The Project Management Body Of Knowledge (PMBOK), 10 Knowledge Areas,	1	CO1
1.2	The Project Life Cycle,	1	CO1
1.3	Organization Structure and Project Management, Organizing Projects within the functional organization,	2	CO1
1.4	Matrix organisation and projectized organisation,	2	CO1
1.5	Choosing the right project management structure,	2	CO1
2	Project Initiation		
2.1	Project Scope and Checklist, Project Priorities.	2	CO2
2.2	Work Breakdown Structure Development, WBS coding	1	CO3, CO6
2.3	,Responsibility Matrices in WBS, Project Estimations	2	CO2, CO6
2.4	Project Communication Plan.	1	CO2,
Assignment 1			CO6
3.1	Agile Project Management , Differences between managing a project versus developing a product,	1	CO3
3.2	Scrum -.Defining the Product Vision and Product Roadmap, Planning Releases and Sprints.	2	CO3, CO6
4	Project Planning		
4.1	Developing the Project Network, WBS to Project Network,	1	CO3, CO6

Module No.	Topic	No. of Periods	CO
	Continuous Assessment Test – I		
4.2	Network Computation Process Networks Activity on Arrow.	2	CO4
4.3	Activity-on-Node, Forward Pass, Earliest Times , Backward Pass—Latest Times	2	CO4
4.4	Use of Lags, Laddering. Lag Relationships.	1	CO4
4.5	Gantt chart using a project management tool	2	CO6
5	Resource optimization		
5.1	Scheduling Resources and Costs, Resource Scheduling Problem,	2	CO5
5.2	Time-Constrained Project, Resource-Constrained Projects.	1	CO5
5.3	Managing Risk, Risk Management Process.	2	CO5
5.4	Risk Identification, Assessment, and Response Development.	2	CO5
Assignment 2			CO6
6.1	Conflict management, Project Closure , Types of Project Closure,	2	CO5
6.2	Final Report, Post-Implementation Evaluation.	2	CO5
Continuous Assessment Test – 2			

Course Designer(s):

1. Dr. N. Shivakumar APCSE
2. Dr. S. Prasanna APCSE

shiva@tce.edu
sprcse@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

FIFTH SEMESTER

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programmes

COURSES OF STUDY

(For the candidates admitted from 2025-26 onwards)

FIFTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CS511	Modelling and Optimization	ES	3	-	-	3
22CS521	Theory of Computation	ES	3	-	-	3
22CS531	Artificial Intelligence	PC	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
22XXGX1	Interdisciplinary Elective	OE	3	-	-	3
THEORY CUM PRACTICAL						
22CS562	Software Engineering	PC	3	-	2	4
PRACTICAL						
22CS571	Network Programming Lab	PC	-	-	2	1
22CS581	Artificial Intelligence Lab	PC	-	-	2	1
PROJECT COURSE						
22CS591	Project - I	PW	-	-	6	3
Total			18	-	12	24

BS : Basic Science Courses
 PC : Professional Core Courses
 ES : Engineering Science Courses
 PW : Project Work

L : Lecture
 T : Tutorial
 P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit
 1 Hour Tutorial is equivalent to 1 credit
 2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programme

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2025-26 onwards)

FIFTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CS511	Modelling and Optimization	3	40	60	100	27	50
2	22CS521	Theory of Computation	3	40	60	100	27	50
3	22CS531	Artificial Intelligence	3	40	60	100	27	50
4	22CSPX1	Programme Elective	3	40	60	100	27	50
5	22XXGX1	Interdisciplinary Elective	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	22CS562	Software Engineering	3	50	50	100	22.5	50
PRACTICAL								
7	22CS571	Network Programming Lab	3	60	40	100	18	50
8	22CS581	Artificial Intelligence Lab	3	60	40	100	18	50
PROJECT COURSE								
9	22CS591	Project - I	3	40	60	100	27	50

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

22CS511	MODELLING AND OPTIMIZATION	Category	L	T	P	Credit
		ES	3	0	0	3

Preamble

Optimization is a scientific approach to decision making that seeks to best design and operate a system, usually under conditions requiring the allocation of scarce resources. Attention is given primarily to techniques applicable to problems in linear and non-linear programming. Modelling refers analysis tool and design tool for various systems. Mathematical models are constructed to solve some real life problem.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcome	TCE Proficiency Scale	Expected Proficiency	Expected Attainment Level %
CO1	Solve linear programming problems using graphical technique.	TPS3	B	65
CO2	Solve linear programming problems using simplex algorithm.	TPS3	B	65
CO3	Solve linear programming problem using semi definite optimization.	TPS3	B	65
CO4	Solve unconstrained Non-Linear Programming Problems (NLPP) using appropriate techniques.	TPS3	B	65
CO5	Solve constrained Non-Linear Programming Problems (NLPP) using appropriate techniques.	TPS3	B	65
CO6	Construct Mathematical models and solve problems related to science and technology	TPS3	B	65

S- Strong; M-Medium; L-Low

Mapping with Programme Outcomes

COs	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L			L		M	M	M		M	M	L	M
CO2.	S	M	L			L		M	M	M		M	M	L	M
CO3.	S	M	L			L		M	M	M		M	M	L	M
CO4	S	M	L			L		M	M	M		M	M	L	M

CO5	S	M	L			L		M	M	M		M	M	L	M
CO6	S	M	L			L		M	M	M		M	M	L	M

Assessment Pattern

CO	Assessment 1						Assessment 2						Terminal		
	CAT 1 (%)			Assignment 1			CAT 2 (%)			Assignment 2					
TPS	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
CO1	-	10	14	-	-	20	-	-	-	-	-	-	-	-	10
CO2	5	10	23	-	-	40	-	-	-	-	-	-	-	6	12
CO3	5	10	23	-	-	40	-	-	-	-	-	-	-	6	12
CO4	-	-	-	-	-		3	10	20	-	-	35	-	6	12
CO5	-	-	-	-	-		3	10	20	-	-	35	-	6	12
CO6	-	-	-	-	-		4	10	20	-	-	30	-	6	12
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

Syllabus

Linear programming: Introduction to LPP – Graphical LP solution (Extreme point solution method) – Special cases in Linear programming. The Simplex algorithm (maximization case only) - Unrestricted variables, Degeneracy- Alternative Optimal Solutions, Unbounded solutions, Infeasible solution- - Dual Simplex Method.

Semi definite optimization: Introduction to primal and dual LPP - From linear to semidefinite optimization – spectrahedra - Projected spectrahedra – Primal SDP formulation- optimality solutions for SDP - Properties of Spectrahedra - Conic Programming - Strong Duality.

Non-Linear Programming: Introduction - Unconstrained Optimization techniques: One-Dimensional minimization methods: Fibonacci method, Golden selection method – Multivariable optimization: Steepest Descent method and Broyden–Fletcher–Goldfarb–Shanno method(BFGS) – Constrained Optimization techniques: Multivariable Optimization with Equality constraints: Lagrange Multipliers Method - Multivariable optimization with inequality constraints: Kuhn-Tucker conditions - Graphical Solution Method.

Modelling: Introduction to mathematical modelling – Graph Based Models: Introduction to graphs and graph models. Matrix based model: Represent the data using curve of best fit using least square method. Linear Programming Models: Formulation of LPP model.

Text Books

1. J.K.Sharma., “Operations Research Theory and Applications”, 6th Edition, Trinity Press, India, 2017.
2. G.Blekherman, Pablo.A.Parrilo, R. Thomas., “Semidefinite Optimization and Convex Algebraic Geometry”, SIAM Publisher, 2012.
3. S.S.Rao., “Engineering Optimization Theory & Practice”, Fourth edition, John Wiley & sons publications, 2009.

4. Kenneth H. Rosen., "Discrete Mathematics and Its Applications", 8th Edition, McGraw hill publications, 2019.
5. Steven J. Leon., "Linear Algebra with Application" Ninth Edition, Pearson, 2015.

Reference Books & web resources

1. Frederick S.Hillier and Gerald J. Lieberman., " Introduction to Operations research" Tenth edition, Mc GrawHill Education, 2015.
2. Hamdy A. Taha, "Operations Research - An Introduction", Tenth Edition, Pearson, 2017.
3. J. Nocedal and S. Wright, "Numerical optimization", Second Edition, Springer, 2006.
4. Radhika Ranjan Roy, "Handbook of SDP for Multimedia Session Negotiations", Kindle Edition, (2018)

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Linear programming	
1.1	Introduction to LPP	1
1.2	Graphical LP solution (Extreme point solution method)	2
1.3	Special cases in Linear programming	1
1.4	The Simplex algorithm (maximization case only)	2
1.5	Unrestricted variables, Degeneracy	1
1.6	Alternative optimal Solutions, Unbounded solutions, Infeasible solution	2
1.7	Dual simplex method	2
2	Semi definite optimization	
2.1	Introduction to primal and dual LPP	1
2.2	From linear to semidefinite optimization – spectrahedra - Projected spectrahedra	1
2.3	Primal SDP formulation	1
2.4	optimality solutions for SDP	1
2.5	Properties of Spectrahedra - Conic Programming	1
2.6	Strong Duality	2
3	Non-Linear Programming	
3.1	Unconstrained Optimization One-Dimensional: Fibonacci method	1
3.2	Golden search method	1

Module No.	Topic	No. of Periods
3.3	Unconstrained Optimization Multi variable: Steepest Descent method	2
3.4	BFGS method, Introduction to LBFGS method	2
3.5	Constrained Optimization Equality Constrains: Lagrange Multiplier Method	2
3.6	Constrained Optimization inequality Constrains: Kuhn-Tucker conditions.	2
3.7	Constrained Optimization inequality Constrains: Graphical method.	2
4	Modelling	
4.1	Introduction to mathematical modelling	1
4.2	Linear programming model: Formulation of LPP model	1
4.3	Graphs and Graph based models: Introduction to graphs and graph models	2
4.4	Matrix based Models: Represent the data using curve of best fit using least square method	2
	Total	36

Course Designer(s):

- | | |
|---------------------------|--|
| 1. Dr. C. S. Senthilkumar | kumarstays@tce.edu |
| 2. Dr. P. Victor | pvmat@tce.edu |
| 3. Dr. P. Krishnapriya | pkamat@tce.edu |

22CS521	THEORY OF COMPUTATION	Category	L	T	P	C
		ES	3	0	0	3

Preamble

This syllabus on "Theory of Computations" is designed to introduce students to the fundamental concepts and applications of automata theory, formal languages, and computability. The syllabus emphasizes the basics of automata theory, including finite automata, regular expressions, and context-free grammars, and provides an overview of the fundamental concepts of computability theory, including Turing machines and the Church-Turing thesis. The modules in the syllabus reflect an approach to problem-solving via automata theory, and the syllabus focuses on the strategies, techniques, and limitations in solving computational problems. The goal is to equip students with the knowledge and skills required to work with automata theory and apply it in their respective fields, including computer science, mathematics, and related disciplines.

Prerequisite

- Nil

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Design Finite Automata machines for given problems	TPS3	B	80
CO2	Find regular expressions and language for any given Finite Automata machines	TPS3	B	80
CO3	Implement Pushdown Automata for given Context Free language and generate the strings of a given context-free languages using its grammar	TPS3	B	80
CO4	Construct Turing machine for any given computational problem	TPS3	B	80

CO5	Differentiate between decidable and undecidable problems based on the complexity of problems when solved using these machines	TPS4	B	80
CO6	Examine the hierarchy of classes of problems or formal languages (regular, context-free, context-sensitive, decidable, and undecidable)	TPS4	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		L			M	L	L		M	M		M
CO2	S	M	L		L			M	L	L		M	M		M
CO3	S	M	L		L			M	L	L		M	M		M
CO4	S	M	L		L			M	L	L		M	M		M
CO5	S	S	M	L	M	L		M	L	L		M	S	L	M
CO6	S	S	M	L	M	L		M	L	L		M	S	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	10	10										2	5	10			
CO2	5	10	10										2	5	10			
CO3	5	10	10										2	5	10			
CO4	5	10	10				5	10	20				2	5	10			
CO5							5	10		20			2	5		10		
CO6								10		20				5		10		
Total	20	40	40				10	30	20	40			10	30	40	20		

CO	Assignment 1						Assignment 2					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1			20									
CO2			30									
CO3			20									
CO4			30						40			
CO5										30		
CO6										30		
Total			100						40	60		

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Automata and Regular Expressions Need for automata theory - Introduction to formal proof Finite Automata (FA) – Deterministic Finite Automata (DFA) – Non-deterministic Finite Automata (NFA) – Equivalence between NFA and DFA – Finite Automata with Epsilon transitions – Equivalence of NFA and DFA- Equivalence of NFAs with and without ϵ -moves- Conversion of NFA into DFA – Minimization of DFAs

Regular Expressions and Languages Regular expression – Regular Languages- Equivalence of Finite Automata and regular expressions– Proving languages to be not regular (Pumping Lemma) – Closure properties of regular languages.

Context Free Grammar and Push Down Automata Types of Grammar - Push Down Automata (PDA): Definition – Moves - Instantaneous descriptions -Languages of pushdown automata – Equivalence of pushdown automata and CFG-CFG to PDA-PDA to CFG – Deterministic Pushdown Automata.

Normal Forms and Turing Machines Normal forms for CFG – Simplification of CFG- Chomsky Normal Form (CNF) and Greibach NormalForm (GNF) – Pumping lemma for CFL – Closure properties of Context Free Languages –Turing Machine: Basic model – definition and representation – Instantaneous Description – Language acceptance by TM

Undecidability Unsolvable Problems and Computable Functions –PCP-MPCP- Recursive and recursively enumerable languages – Properties - Universal Turing machine -Tractable and Intractable problems - P and NP completeness – Kruskal's algorithm – Travelling Salesman Problem- 3-CNF SAT problems.

Unconventional Computing - Membrane Computing, DNA Computing, Quantum Computing Models

Text Books

1. Hopcroft J.E., Motwani R. & Ullman J.D., "Introduction to Automata Theory, Languages and Computations", 3rd Edition, Pearson Education, 2008.
2. John C Martin , "Introduction to Languages and the Theory of Computation", 4th Edition, Tata McGraw Hill, 2011.

Reference Books

1. Harry R Lewis and Christos H Papadimitriou, "Elements of the Theory of Computation", 2nd Edition, Prentice Hall of India, 2015.
2. Peter Linz, "An Introduction to Formal Language and Automata", 6th Edition, Jones & Bartlett, 2016.
3. K.L.P.Mishra and N.Chandrasekaran, "Theory of Computer Science: Automata Languages and Computation", 3rd Edition, Prentice Hall of India, 2006.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	Automata and Regular Expressions	(6)
1.1	Need for automata theory - Introduction to formal proof – Finite Automata (FA)	1
1.2	Deterministic Finite Automata (DFA) – Non-deterministic Finite Automata (NFA)	1
1.3	Equivalence between NFA and DFA – Finite Automata with Epsilon transitions	1
1.4	Equivalence of NFA and DFA- Equivalence of NFAs with and without ϵ -moves	1
1.5	Conversion of NFA into DFA – Minimization of DFAs	2
2	Regular Expressions and Languages	(6)
2.1	Regular expression – Regular Languages	1
2.2	Equivalence of Finite Automata and regular expressions	2
2.3	Proving languages to be not regular (Pumping Lemma)	2
2.4	Closure properties of regular languages.	1
3	Context Free Grammar and Push Down Automata	(6)
3.1	Types of Grammar	2
3.2	Push Down Automata (PDA): Definition – Moves	1

3.3	Instantaneous descriptions -Languages of pushdown automata	1
3.4	Equivalence of pushdown automata and CFG-CFG to PDA-PDA to CFG - Deterministic Pushdown Automata	2
4	Normal Forms and Turing Machines	(6)
4.1	Normal forms for CFG – Simplification of CFG	1
4.2	Chomsky Normal Form (CNF) and Greibach Normal Form (GNF)	2
4.3	Pumping lemma for CFL – Closure properties of Context Free Languages	1
4.4	Turing Machine: Basic model – definition and representation	1
4.5	Instantaneous Description – Language acceptance by TM	1
5	Undecidability	(6)
5.1	Unsolvable Problems and Computable Functions –PCP-MPCP-Recursive and recursively enumerable languages	2
5.2	Properties - Universal Turing machine	1
5.3	Tractable and Intractable problems	1
5.4	P and NP completeness – Kruskal's algorithm – Travelling Salesman Problem- 3-CNF SAT problems.	2
6	Unconventional Computing	(6)
6.1	Membrane Computing, ,	2
6.2	DNA Computing	2
6.3	Quantum Computing Models	2

Course Designer(s):

1. Dr.K.Sundarakantham, Professor, CSE

kskcse@tce.edu

2. Mr. S.Santhana Hari, AP, CSE

sshcse@tce.edu

22CS531	ARTIFICIAL INTELLIGENCE	Category	L	T	P	C
		PC	3	0	0	3

Preamble

This course introduces the basic concepts and techniques of Artificial Intelligence. Artificial intelligence is the sub-area of computer science devoted to creating software and hardware to get computers to do things that would be considered intelligent as if people did them. This course will help the students to gain generic problem-solving skills that have applicability to a wide range of real-world problems. Students can learn how machines can engage in problem-solving, reasoning, learning, and interaction.

Prerequisite

Students are expected to have

- Basic Programming knowledge in Python
- Knowledge of search algorithms like BFS, DFS
- Graph data structures

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Demonstrate a fundamental understanding of the history of artificial intelligence (AI) and its foundations	TPS2	B	80
CO2	Interpret AI problems as a state space representation and solve using AI techniques like searching and game playing	TPS3	B	75
CO3	Construct knowledge representations using logic to facilitate inference in the given problem domain	TPS3	B	75
CO4	Formulate solutions for problems involving uncertain inputs or outcomes	TPS3	B	75
CO5	Apply basic principles of AI in solutions that require problem solving by learning	TPS3	B	75
CO6	Examine real-world problems and apply suitable AI techniques to develop intelligent systems	TPS4	B	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	M	L				L		L				M	M	L	L
CO2	S	M	L		L			M	L	L		M	M	M	L
CO3	S	M	L		L			M	L	L		M	M	M	L
CO4	S	M	L		L			M	L	L		M	M	M	L
CO5	S	M	L		L			M	L	L		M	M	M	L
CO6	S	S	M	L	M	L		M	L	L		M	S	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						Assignment 1			CAT 2						Assignment 2				Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6	1	2	3	4	1	2	3	4	5	6
CO 1	10	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	5		-	-	-
CO 2	10	10	25	-	-	-	-	-	50	-	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO 3	-	10	25	-	-	-	-	-	50	-	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO 4	-	-	-	-	-	-	-	-	-	10	10	15	-	-	-	-	-	30		2	5	10	-	-	-
CO 5	-	-	-	-	-	-	-	-	-	10	10	15	-	-	-	-	-	30		2	5	10	-	-	-
CO 6	-	-	-	-	-	-	-	-	-	10	-	-	20	-	-	-	-	-	40		5		20		

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Turing Test – Intelligent Agents, Characteristics of Intelligent Agents – Environments properties – Future of AI – Typical AI problems – Problem-Solving Approach to Typical AI problems.

Problem-Solving Methods: Search Strategies – Informed Search – Local Search – Searching with Partial Observations – Constraint Satisfaction Problems – Constraint Propagation,

Backtracking Search – Game Playing – Optimal Decisions in Games, Alpha - Beta Pruning – Stochastic games.

Knowledge Representation: First Order Predicate Logic – Unification – Forward Chaining – Backward Chaining – Resolution. Rule-Based Systems – Rete Graph. Classical planning.

Uncertain knowledge and Reasoning: Quantifying Uncertainty – Acting under Uncertainty – Inference using Full Joint Distributions – Bayes' rule. Probabilistic Reasoning – Bayesian Models – Relational and First-Order Probability Models – Time and Uncertainty.

Learning: Learning from Examples – Supervised Learning – Reinforcement Learning.

Case study: Large Language models - ChatGPT, Computer Vision, Automation.

Reference Books

1. Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach, 4th Edition, Prentice Hall, Feb 2020.
2. Deepak Khemani. A First Course in Artificial Intelligence, McGraw Hill Education (India), 2013.
3. Elain Rich and Kevin Knight, "Artificial Intelligence ", Tata McGraw Hill, Third Edition, 2008.
4. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem solving", Fourth Edition, Pearson Education, 2002.
5. Stefan Edelkamp and Stefan Schroedl. Heuristic Search: Theory and Applications, organ Kaufmann, 2011.
6. NPTEL Lectures

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	CO Mapped
1	Introduction (5)		
1.1	Turing Test – Intelligent Agents	1	CO1
1.2	Characteristics of Intelligent Agents – Environments properties – Future of AI	2	CO1
1.3	Typical AI problems – Problem-Solving Approach to Typical AI Problems	2	CO1
2	Problem-Solving Methods (9)		
2.1	Search Strategies – Informed Search	2	CO2
2.2	Local Search – Searching with Partial Observations	2	CO2
2.3	Constraint Satisfaction Problems	2	CO2

	Constraint Propagation, Backtracking Search		
2.4	Game Playing – Optimal Decisions in Games Alpha - Beta Pruning	2	CO2
2.5	Stochastic Games	1	CO2
3	Knowledge Representation (5)		
3.1	First-Order Predicate Logic	1	CO3
3.2	Unification – Forward Chaining Backward Chaining – Resolution	1	
3.3	Rule-Based Systems	1	CO3
3.4	Rete Graph	1	CO3
3.5	Classical planning	1	CO3
4	Uncertain Knowledge and Reasoning (8)		
4.1	Quantifying Uncertainty	1	CO4
4.2	Acting under Uncertainty	1	CO4
4.3	Inference using Full Joint Distributions	1	CO4
4.4	Bayes' rule	1	CO4
4.5	Probabilistic Reasoning	1	CO4
4.6	Bayesian models	1	CO4
4.7	Relational and First-Order Probability Models	1	CO4
4.8	Time and Uncertainty	1	CO4
5	Learning (5)		
5.1	Learning from Examples	1	CO5
5.2	Supervised Learning	2	CO5
5.3	Reinforcement Learning	2	CO5
	Mini Project Reviews	4	CO6

	Total Hours	36	
--	--------------------	-----------	--

Course Designer(s):

1. Dr. K.Sundarakantham

kskcse@tce.edu

2. Dr. M.Suguna

mscse@tce.edu

22CS562**SOFTWARE ENGINEERING**

Category	L	T	P	Credit
PC	3	0	2	4

Preamble

This course introduces software development processes, project management, and planning techniques. Students learn to model systems using activity, use case, sequence, class, and state diagrams, and apply cost estimation and scheduling models such as COCOMO model. It also covers component-based and distributed system design, SaaS, reliability engineering, and secure software development.

Prerequisite

Project management

Course Outcomes

On the successful completion of the course, students will be able to

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain various software development process and management techniques.	TPS2	70	90
CO2	Demonstrate system boundaries and interactions using Activity and Use Case diagrams, along with Sequence, Class, and State diagrams.	TPS3	70	85
CO3	Apply software pricing strategies and various software cost estimation models such as COCOMO, COCOMO II and parametric estimation approaches for effective project planning.	TPS3	70	85
CO4	Illustrate a system design considering the key issues in Components, architectural patterns for distributed systems and Software as a service .	TPS3	70	85
CO5	Use basic software testing methods like Black-box, White-box, Static, Dynamic, testing levels and the V-model to check and improve software quality.	TPS3	70	85
CO6	Analyse reliability engineering principles, including fault-tolerant architectures, programming techniques for reliability, secure system design practices, and methods for secure testing and assurance.	TPS4	70	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	M	L							L	L	L	L	M		L
CO2	S	M	L		S	M	M	M	M	M	L	L	M	L	M
CO3	S	M	L				M	M	M	M	L	L	M	L	M

CO4	S	M	L				M	M	M	M	L	L	M	L	M
CO5	S	M	L		S	M	M	M	M	S	S	S	M	L	M
CO6	S	S	M	L	S		S	S	S	S	S	S	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1				CAT 2				Terminal Examination			
TPS Scale	1	2	3	4	1	2	3	4	1	2	3	4
CO1	6	14							2	8		
CO2	6	14	20						2	5	10	
CO3	6	14	20						2	5	10	
CO4					5	10	15		2	5	10	
CO5					5	10	15		2	5	10	
CO6					5	10		25	2	5		15

CO	LAB MODEL TEST						OCR					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1		10						10				
CO2			20						20			
CO3			10						10			
CO4			20						20			
CO5			20						20			
CO6				20						20		

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Software process models

Professional software development- Software engineering ethics- Process activities - Coping with change- Process Improvement -Software Life cycle- Iterative, Spiral, Prototyping-Agile method- Agile developing techniques- Agile project management - Scaling Agile methods

System Modelling

Functional and non- functional requirements- Software Requirement Specification - Developing Context models, Interaction models, Structural models and Behavioural models- Architectural patterns, Application architectures- -Object oriented design- Design patterns.

Software Pricing and Project Planning

Software pricing, Software Cost Estimation Models - COCOMO, COCOMO II, Function Point Analysis-Use Case Point Method-Expert judgement, Delphi method - Parametric estimation models-Plan-driven development, Project scheduling, requirements to activities, activities to Gantt chart Agile planning Software Reuse- Risk Management

Component-Based and Distributed Software Engineering

Components models-principles of component-based development-CBSE identification, deployment and composition - Distributed systems and their characteristics- architectural patterns - middleware Communication-Software as a Service (SaaS).

Software testing

Testing – Meaning and purpose, testing methods - Black-box and White-box, Static and Dynamic testing, Level of testing, V-model.

Reliability Engineering

Fault tolerance Architectures- Programming for reliability- Secure systems design and programming- Secure testing and assurance.

Lab Content:

- Develop a mini project for a real world problem in which a software solution can be obtained (a team of 3 members) and do the following.
- Collect requirements for the chosen problem
- Model the system through interaction, structural diagrams and develop software architecture.
- Develop the system partially through Test Driven Development with unit test.

Sample case-studies:

- An embedded control system for a personal insulin pump

This case study discusses the control software for a personal insulin pump, which is used by diabetics to mimic the function of the pancreas and hence control the level of glucose (sugar) in their blood.

- The iLearn digital learning environment

The iLearn system is a digital learning environment used to support learning in schools with students from age 4 to 18. It is intended to replace an existing system (Glow) that was specially built for the purpose and which includes its own applications for e-mail, etc.

- The Mentcare system

This case study focuses on the requirements for a system that I have called the Mentcare system, which is a real system (although that is not its real name) which was used in a number of UK hospitals, including hospitals in Scotland.

- Wilderness weather station

This case study is based on the software for a wilderness weather station that collects weather information in remote areas that do not have local infrastructure (power, communications, roads, etc.).

Reference Books & web resources

1. Ian Sommerville , "Software Engineering" , 10th Edition, John Wiley and sons,2016.
2. Orit Hazzan, Yael Dubinsky, "Agile software engineering", Springer,2014
3. James Rumbaugh, Ivar Jacobson, Grady Booch," The Unified Modeling Language Reference Manual", 2nd Edition, Addison Wesley,2005.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	Software process models		
1.1	Professional software development- Software engineering ethics- Process activities- Coping with change- Process Improvement	2	CO1
1.2	Software Life cycle models- Iterative, Spiral and Prototyping models	2	CO1
1.3	Agile method- Agile developing techniques- Agile project management- Scaling Agile methods	2	CO1
2	System Modelling		
2.1	Functional and non- functional requirements.	1	CO2
2.2	Software Requirement Specification.	1	CO2
2.3	Developing Context models, Interaction models.	1	CO2
2.4	Structural models and Behavioral models.	1	CO2
2.5	Architectural patterns, Application architectures.	2	CO2
2.6	Object oriented design- Design patterns.	2	CO2
3	Software Project Planning Software Pricing and Project Planning		
3.1	Software pricing	1	CO3
3.2	Software Cost Estimation Models - COCOMO, COCOMO II	1	CO3
3.3	Function Point Analysis-Use Case Point Method	1	CO3
3.4	Expert judgement, Delphi method	1	CO3
3.5	Parametric estimation models	1	CO3
3.6	Plan-driven development	1	CO3
3.7	Project scheduling, requirements to activities	1	CO3
3.8	Activities to Gantt chart Agile planning Software Reuse	1	CO3
3.9	Risk Management	1	CO3
4	Component based Software Engineering and Distributed Software Engineering		
4.1	Component models	1	CO4
4.2	Principles of component-based development	1	CO4
4.3	CBSE identification, deployment and composition	1	CO4
4.4	Distributed systems and their characteristics	1	CO4
4.5	Architectural patterns	1	CO4
4.6	Middleware Communication-Software as a Service (SaaS)	1	CO4
5	Software testing		
5.1	Testing – Meaning and purpose	1	CO5
5.2	Testing methods - Black-box and White-box,	1	CO5

5.3	Static and Dynamic testing,	1	CO5
5.4	Level of testing and testing models	1	CO5
6	Reliability Engineering		
6.1	Fault tolerance Architectures- Programming for reliability	1	CO6
6.2	Secure systems design and programming	1	CO6
6.3	Secure testing and assurance	1	CO6
		36	
Module No.	Topic	No. of Lectures	Course Outcome
1	Develop a mini project for a real world problem in which a software solution can be obtained (a team of 3 members) and do the following: Collect requirements for the chosen problem and analyse the various fault tolerance architecture .	4	CO1 CO4 CO6
2	Develop Software Requirements specification document	4	CO2
3	Develop context level models for software architecture .	4	CO2
4	Model the system through structural diagrams	2	CO2
5	Model the system through interaction diagrams	2	CO2
6	Propose a project plan Using gantt chart and build a Java Swing application that takes project parameters as input and displays calculated effort and schedule in a graphical interface.	4	CO3
7	Develop a JUnit test case for a user login method that authenticates a user based on a username and password for an application of your choice	4	CO5

Course Designer(s):

- | | |
|----------------------------|--|
| 1. Dr. N. Shivakumar APCSE | shiva@tce.edu |
| 2. Dr. K.Indira APCSE | kiit@tce.edu |
| 3. Dr. Raja Lavanya APCSE | rlit@tce.edu |

22CS571	NETWORK PROGRAMMING LAB	Category	L	T	P	Credit
		PC	0	0	2	1

Preamble

Universal connectivity is realized through Computer Networks. It is important to gain knowledge on the hardware requirements and functioning of Computer Networks. This course provides insight into the working of network protocols and their characteristics.

Prerequisite

Object oriented programming

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Perform Configuration of networking components and installing device drivers and build a Local Area Network	TPS3	A	85
CO2	Perform port scanning and identify IP and MAC Address.	TPS3	A	85
CO3	Implement client server communication using socket programming	TPS3	A	85
CO4	Perform DNS server host name identification and resolve given host name	TPS3	A	85
CO5	Implement File transfer and RMI.	TPS3	A	85
CO6	Simulate a network topology using CISCO Packet tracer	TPS3	A	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	S	M	L		L	-	-	M	M	M		M	M	L	M
CO2.	S	M	L		L	-	-	M	M	M		M	M	L	M
CO3.	S	M	L		L	-	-	M	M	M		M	M	L	M
CO4.	S	M	L		L	-	-	M	M	M		M	M	L	M
CO5.	S	M	L		L	-	-	M	M	M		M	M	L	M
CO6.	S	M	L		L	-	-	M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		

Understand		
Apply	100	100
Analyze		
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

1. Establishment of a LAN: Preparation of network cables and installation and configuration of network.
2. Perform network commands like NETSTAT, NSLOOKUP, TRACERT, ARP, IPCONFIG and verify their need and usage
3. Develop a program to identify your machine's host name and IP address
4. Develop a program to locate the next hop router's IP address and MAC address
5. Develop a program to find which port is currently used, by scanning the port.
6. Develop a program to obtain local DNS server's host name and IP address and resolve a given host name.
7. Develop a program to illustrate a simple client/server communication and Time server.
8. Develop a program to implement ECHO and PING commands and time server.
9. Develop a program to implement a file transfer using TCP..
10. Develop a program to implement a file transfer using UDP.
11. Develop a program to implement Remote Method Invocation.
12. Simulate a network for the given specification using CISCO Packet tracer

Learning Resources

1. Data Communications and Networking, 5th Edition, BehrouzForouzan, Mc Graw Hill, 2017.
2. Computer Networks: A Systems Approach, Larry L. Peterson, Bruce S. Davie, Elsevier, Mar 2011
3. Computer Networking: A Top-Down Approach featuring the Internet, 6th edition, James F. Kurose, Pearson Education India, 2013.

Course Designers:

1. Dr.C.Senthilkumar ,Associate Professot, CSE, cskcse@tce.edu
2. Dr.G.S.R.EmilSelvan , Asslstant Professot, CSE emil@tce.edu

22CS581	ARTIFICIAL INTELLIGENCE LAB	Category	L	T	P	C
		PC	0	0	2	1

Preamble

The laboratory course will facilitate the Students to apply the concept of artificial intelligence for different problems like eight queens, travelling salesperson problem using machine learning libraries, Python, LISP and PROLOG. These experiments are aimed at imparting a practical exposure to the students to gain generic problem solving skills that have applicability to a wide range of real-world problems. Students can learn how machines can engage in problem solving, reasoning, learning, and interaction.

Prerequisite

- Fundamentals of Python programming

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Formulate the given problem as PEAS description	TPS3	A	75
CO2	Implement Heuristic search algorithms to solve given state space representable AI Problems	TPS3	A	75
CO3	Solve given 2-ply game using Min-max algorithm and optimize using alpha-beta pruning	TPS3	A	75
CO4	Represent the given Constraint Satisfaction Problem as Constraint graph and solve using Backtracking, forward checking or arc consistency	TPS3	A	75
CO5	Construct rule based systems for any application using logic programming language	TPS3	A	75
CO6	Examine the given use case and develop an intelligent system using suitable learning approach	TPS4	A	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L		M	L		S	S	S		M	M	L	M
CO 2	S	M	L		M	L		S	S	S		M	M	L	M
CO 3	S	M	L		M	L		S	S	S		M	M	L	M
CO 4	S	M	L		M	L		S	S	S		M	M	L	M
CO 5	S	M	L		M	L		S	S	S		M	M	L	M
CO 6	S	S	M	L	M	L		S	S	S	L	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	80	80
Analyze	20	20
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

Sl.No	Experiments	CO
1.	Select an AI problem and formulate it as PEAS description	CO1
2.	Implement A*search strategy to reach goal state in the given application	CO2
3.	Implement Local search strategy to reach goal state in the given application	CO2
4.	Solve 2-ply games like Alpha Go using APIs	CO3
5.	Solve Constraint Satisfaction Problems like Crypt arithmetic problem, Water jug problem, etc..	CO4
6.	Construct knowledge base for the given use case and apply inference in First Order Logic	CO5
7.	Solve classification problem using Bayesian models	CO6
8.	Implement Cart-Pole with Reinforcement learning	CO6
9.	Make use of suitable Large AI models in Language / Vision / Control with Prompt Engineering for the given application (Mini-Project)	CO6

Learning Resources

1. Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall, 2009.
2. Deepak Khemani. A First Course in Artificial Intelligence, McGraw Hill Education (India), 2013.
3. Elain Rich and Kevin Knight, "Artificial Intelligence ", Tata McGraw Hill, Third Edition, 2008

Course Designers:

1. Dr. K. Sundarakantham (kskcse@tce.edu)
2. Dr. M. Suguna (mscse@tce.edu)

22CS591	PROJECT – I
----------------	--------------------

Category	L	T	P	Credit
Project	-	-	6	3

GUIDELINES FOR PROJECT COURSES

In the curriculum of 2022 B.E./B.Tech. Programmes, 12 credits have been assigned for **Project work in the specific discipline**. The curriculum was designed based on around 4 focus/broad areas. This has been split into 4 project courses namely Project I, Project II, Project III and Project IV in semesters 5, 6, 7 and 8 respectively.

Choice of Focus Areas: HoDs are requested to provide choice for the students to choose one of the broad/focus areas to carry out project work in 5, 6, 7 and 8th semesters. The number of students in each broad/focus areas shall be based on the faculty and infrastructure availability.

Project Continuity and Switching: The students shall be given a big project in the chosen broad/focus area so that it can be spread over all four semesters with specific outcomes at each semester. In case, a student wants to switch from one focus area to other area. It has to be approved by the Head of the Department and project coordinator.

Internal Marks: Three reviews shall be conducted in each semester to monitor the progress of the project. Review 1; 10 Marks, Review 2: 15 Marks, Review 3: 15 Marks. Total = 40 Marks

Viva Voce Examinations: For external examinations, HoD shall appoint two examiners in each focus/broad area to conduct the Viva Voce examination in semesters 5, 6 and 7. Project guides are also one of the examiners, along with the two examiners appointed by the HoD. External exam will be conducted for 60 Marks.

Final Viva Voce Examination: In 8th Semester, Viva Voce will be conducted by an external examiner, HoD/HoD Nominee as internal examiner and Project Guides. External exam will be conducted for 60 Marks.

This structured approach ensures that students engage in a comprehensive project experience throughout their undergraduate studies, with regular monitoring of progress and formal evaluation through viva voce examinations. It also allows for flexibility by permitting students to switch focus areas with appropriate approvals.

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

SIXTH SEMESTER

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programmes

COURSES OF STUDY

(For the candidates admitted from 2025-26 onwards)

SIXTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CS611	Cryptography and Network Security	PC	3	-	-	3
22CS621	Distributed Computing	PC	3	-	-	3
22CS632	Compiler Design	PC	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
22XXGX1	Basic Science Elective	OE	3	-	-	3
PRACTICAL						
22CS671	Distributed Computing Lab	PC	-	-	2	1
PROJECT COURSE						
22CS691	Project - II	PW	-	-	6	- 3
Total			18	-	8	22

BSC : Basic Science Courses
 PC : Professional Core Courses
 ESC : Engineering Science Courses
 PW : Project work

L : Lecture
 T : Tutorial
 P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit
 1 Hour Tutorial is equivalent to 1 credit
 2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E. / B.Tech. Degree Programme

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2025-26 onwards)

SIXTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CS611	Cryptography and Network Security	3	40	60	100	27	50
2	22CS621	Distributed Computing	3	40	60	100	27	50
3	22CS632	Compiler Design	3	40	60	100	27	50
4	22CSPX1	Programme Elective	3	40	60	100	27	50
5	22CSPX1	Programme Elective	3	40	60	100	27	50
6	22XXGX1	Basic Science Elective	3	40	60	100	27	50
PRACTICAL								
7	22CS671	Distributed Computing Lab	3	60	40	100	18	50
PROJECT COURSE								
8	22CS691	Project - II	3	40	60	100	27	50

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

22CS611	CRYPTOGRAPHY AND NETWORK SECURITY	Category	L	T	P	C
		PC	3	0	0	3

Preamble

Cryptography is the science of information and communication security. This course will discuss common security weaknesses, vulnerabilities, attack methods and mitigation approaches in network security. The focus of the course is on confidentiality, data integrity and non-repudiation. Real time applications of cryptographic primitives are addressed.

Prerequisite

- Discrete Mathematics
- Basic knowledge in Computer Networks

Course Outcomes

On the successful completion of the course, students will be able to

CO. No	Course Outcomes (COs)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Demonstrate the fundamental theory of modern cryptography.	TPS3	B	85
CO2	Demonstrate the mathematical foundations of security mechanisms.	TPS3	B	85
CO3	Encrypt and Decrypt messages using Private key cryptosystems.	TPS3	B	85
CO4	Encrypt and Decrypt messages using Public key cryptosystems.	TPS3	B	85
CO5	Ensure authentication and data integrity security services through signing and hashing mechanisms.	TPS3	B	85
CO6	Examine the strength of any cryptographic algorithm and design a variant for given requirements.	TPS4	B	85
CO7	Describe the application of cryptographic algorithms in real world protocols	TPS2	B	90

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO1	S	M	L		L	L		L	L	L		M	M	L	L
CO2	S	M	L		L			L	L	L		M	M		L
CO3	S	M	L		L	L		L	L	L		M	M	L	L
CO4	S	M	L		L	L		L	L	L		M	M	L	L
CO5	S	M	L		L	L		L	L	L		M	M	L	L
CO6	S	S	M	L	L			M	M	M		M	M	L	M
CO7	M	L			M	L						M	L	L	

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						Assignment 1				CAT 2						Assignment 2				Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	1	2	3	4	5	6	1	2	3	4	1	2	3	4	5	6
CO1	5	5	15	-	-	-	-	-	20	-	-	-	-	-	-	-	-	-	-	-	2	-	10	-	-	-
CO2	5	5	15	-	-	-	-	-	30	-	5	-	-	-	-	-	-	-	10	-	2	5	10	-	-	-
CO3	-	5	25	-	-	-	-	-	30	-	-	-	-	-	-	-	-	-	-	-	-	-	15	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-	-	25	-	-	-	-	-	25	-	2	-	15	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-	25	-	-	-	-	-	25	-	-	-	15	-	-	-
CO6	-	5	-	15	-	-	-	-	20	-	-	-	20	-	-	-	-	-	10	-	5	-	10	-	-	-
CO7	-	-	-	-	-	-	-	-	-	-	5	20	-	-	-	-	-	30	-	-	4	5	-	-	-	-

Syllabus

Introduction: Security attacks and threats, OSI Security Architecture – mechanisms and services. Need for perfect secrecy, Shannon's theory - Classical encryption techniques, Modular Arithmetic – Hill cipher, Common attacks on Cryptosystems – CPA, CCA, Side channel attack. Semantic Security and Message Indistinguishability.

Block cipher Mechanisms: Feistel network - Data Encryption Standard, Introduction to Finite Fields: Groups - Rings and Fields - Galois Fields, Advanced Encryption Standard, Block cipher modes operation. **Stream Cipher Mechanisms:** One-time pad, Pseudo Random Number Generation algorithms, RC4 Stream Cipher.

Public Key Encryption: Introduction to Number Theory- Prime Numbers, Fermat's and Euler's Theorem, Testing for Primality. Public key ciphers - RSA Cryptosystem, Elliptic Curve Cryptography. **Key Management:** Diffie Hellman Key Exchange, Distribution of public keys.

Data Integrity: One-way function, Message Authentication Codes, Hash function - SHA algorithm. **Non-Repudiation:** Digital Signature – provably secure signature schemes, Zero knowledge protocols.

Real world applications: Authentication Application – Kerberos, Electronic Mail Security – PGP, IP Security - IP Security Architecture. Web Security- Secure Socket Layer and Transport layer, Secure Electronic Transaction, Blockchain architecture.

Reference Books & web resources

1. William Stallings, Cryptography and Network Security: Principles and Practice, Global Edition, Pearson Education, 2022.
2. Behrouz A. Forouzan and Debdeep Mukhopadhyay, Cryptography and Network Security, Mc Graw Hill, 2015.
3. Douglas R. Stinson and Maura B. Paterson, Cryptography: Theory and Practice, Fourth edition, CRC Press, Taylor and Francis Group, 2019.
4. <https://www.coursera.org/learn/crypto> offered by Dan Boneh - Professor, Computer Science, Stanford University.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	Introduction (8)	
1.1	Security attacks and threats	1
1.2	OSI Security Architecture - Mechanisms and services	1
1.3	Need for perfect secrecy - Shannon's theory	1
1.4	Classical encryption techniques	2
1.5	Modular Arithmetic - Hill cipher	1
1.6	Common attacks on Cryptosystems- CPA, CCA, Side channel attack	1
1.7	Semantic Security and Message Indistinguishability	1
2	Block cipher Mechanisms (5)	
2.1	Feistel network - Data Encryption Standard	1
2.2	Introduction to Finite Fields - Galois Fields	1
2.3	Advanced Encryption Standard	2
2.4	Block cipher modes of operation	1

Module No.	Topic	No. of Lectures
3	Stream Cipher Mechanisms (3)	
3.1	One-time pad	1
3.2	Pseudo Random Number Generation algorithms	1
3.3	RC4 Stream Cipher	1
4	Public Key Encryption (5)	
4.1	Introduction to Number Theory	1
4.2	Prime Numbers, Fermat's and Euler's Theorem, Testing for Primality	1
4.3	Public key ciphers - RSA Cryptosystem	1
4.4	Elliptic Curve Arithmetic	1
4.5	Elliptic Curve Cryptography	1
4.6	Key Management (3)	
4.6.1	Diffie Hellman Key Exchange - ECDSA	2
4.6.2	Distribution of public keys	1
5	Data Integrity (4)	
5.1	One-way function	1
5.2	Message Authentication Codes	1
5.3	Hash function - SHA algorithm	2
	Non-Repudiation (3)	
5.4	Digital Signature – provably secure signature schemes	2
5.5	Zero knowledge protocols	1
6	Real world applications (5)	
6.1	Authentication Application – Kerberos	1
6.2	Electronic Mail Security – PGP, IP Security - IP Security Architecture	1

Module No.	Topic	No. of Lectures
6.3	Web Security- Secure Socket Layer and Transport layer, Secure Electronic Transaction	2
6.4	Blockchain Architecture	1
	Total Hours	36

Course Designers:

1. Dr.M.Suguna mscse@tce.edu
2. Dr.M.P.Ramkumar ramkumar@tce.edu

22CS621	DISTRIBUTED COMPUTING	Category	L	T	P	C
		PC	3	0	0	3

Preamble

This course will cover both fundamental concepts in distributed computing and discuss system designs enabling distributed applications. The objectives of the course include: In-depth understanding of core concepts of distributed computing, including study of both abstract concepts and practical techniques for building system support for distributed applications

Prerequisite

- 22CS361 Operating Systems (Theory cum Practical)
- 22CS431 Data Communication and Networks
- 22CS571 Network Programming Lab

Course Outcomes

On the successful completion of the course, students will be able to

CO No.	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the design principles and architectures for distributed systems.	TPS2	B	85
CO2	Implement distributed algorithms for concurrent task termination (Apply)	TPS3	B	80
CO3	Practice Mutual Exclusion and deadlock detection algorithms that reveals concurrent programming error (Apply)	TPS3	B	80
CO4	Use appropriate protocols in a set of servers involving distributed Transaction (Apply)	TPS3	B	80
CO5	Apply, Recovery and Check pointing techniques for Fault tolerant distributed systems. (Apply)	TPS3	B	80
CO6	Examine distributed system issues to handle and process large data volumes through appropriate tools (Analyze)	TPS4	B	80

Mapping with Programme Outcomes

COs	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	M	L										L	L		
CO2.	S	M	L		L			M	M	M		M	M		M
CO3	S	M	L		L			M	M	M		M	M		M
CO4	S	M	L					M	M	M		M	M		M
CO5	S	M	L					M	M	M		M	M		M
CO6	S	S	M	L	M	L		M	M	M	L	M	S	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	10	10	-	-	-	-	-	-	-	-	-	-	-	-	-	15	-	-	-	-
CO2	5	5	30	-	-	-	50	-	-	-	-	-	-	-	-	5	12	-	-	-
CO3	5	5	30	-	-	-	50	-	-	-	-	-	-	-	-	5	12	-	-	-
CO4	-	-	-	-	-	-	-	5	5	-	-	-	-	40	-	5	12	-	-	-
CO5	-	-	-	-	-	-	-	5	5	50	-	-	-	40	-	5	12	-	-	-
CO6	-	-	-	-	-	-	-	-	10	-	20	-	-	20	-	5	-	12	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction – Distributed System Models and Enabling Technologies, Multi/Many-Core Computing, Memory System Parallelism for Data Intensive and Data-Driven Applications, Goals and Challenges in Distributed System

Logical Time and Global State – Clock Synchronization, Logical Clocks -Global State and Snapshot Recording Algorithm -Message ordering and group communication: Message ordering paradigms -- Asynchronous execution with synchronous communication – Synchronous program order on an asynchronous system.

Distributed Mutex And Deadlock – Distributed mutual exclusion algorithms - Ricart–Agrawala algorithm -- Quorum-based mutual exclusion algorithms -- Token-based algorithms -- Suzuki–Kasami’s broadcast algorithm; Deadlock detection in distributed systems: System model – Models of deadlocks -- Knapp’s classification of distributed deadlock detection algorithms -- Mitchell and Merritt’s algorithm for the single resource model -- Chandy–Misra–Haas algorithm –AND,OR Models

Distributed Transaction and Recovery – Flat and nested distributed Transactions-Atomic Commit protocols-Concurrency Control- Transaction with replicated data-- Consensus and agreement algorithms -- Checkpointing and rollback recovery: Issues in failure recovery -- Checkpoint-based recovery -- Log based rollback recovery - coordinated checkpointing algorithm -- asynchronous checkpointing and recovery.

Distributed File Systems – Name Services - File Models, File Accessing Models, File-sharing Semantics, File-caching Schemes, File Replication,- Sun’s network file system, Andrews file system-Naming- Directory services.- Security:Techniques **Case Study:** Peer to Peer Computing-Distributed Object Based System, Distributed Web Based System, Distributed Coordinated System

Text Book

1. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2011.
2. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, Distributed Systems: Concepts and Design, Fifth Edition, Pearson Education, 2017.

Reference Books & web resources

1. M. Van Steen, A.S. Tanenbaum, Distributed Systems, Third Edition, CreateSpace Independent Publishing Platform, 2017.
2. Garg VK. Elements of distributed computing. John Wiley & Sons, 2002.
3. Pradeep K Sinha, "Distributed Operating Systems: Concepts and Design", Prentice Hall of India, 2007
4. Liu M.L., —Distributed Computing, Principles and ApplicationsII, Pearson Education, 2004
5. Fokkink W. Distributed algorithms: an intuitive approach, Second Edition, MIT Press, 2018.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Distributed System Models and Enabling Technologies	1
1.2	Multi/Many-Core Computing	1
1.3	Memory System Parallelism for Data Intensive and Data-Driven Applications	2

Module No.	Topic	No. of Periods
1.4	Goals and Challenges in Distributed System	1
2	Logical Time and Global State	
2.1	Clock Synchronization, Logical Clocks	2
2.2	Global State and Snapshot Recording Algorithm	
2.3	Message ordering and group communication: Message ordering paradigms	2
2.4	Asynchronous execution with synchronous communication – Synchronous program order on an asynchronous system	2
3	Distributed Mutex And Deadlock	
3.1	Distributed mutual exclusion algorithms -Lamport's algorithm -- Ricart–Agrawala algorithm	2
3.2	Quorum-based mutual exclusion algorithms -- Maekawa's algorithm -- Token-based algorithms -- Suzuki–Kasami's broadcast algorithm	2
3.3	Deadlock detection in distributed systems: System model – Models of deadlocks	1
3.4	Knapp's classification of distributed deadlock detection algorithms	1
3.5	Mitchell and Merritt's algorithm for the single resource model -- Chandy–Misra–Haas algorithm – AND,OR Models	3
4	Distributed Transaction and Recovery	
4.1	Flat and nested distributed Transactions-Atomic Commit protocols-Concurrency Control	2
4.2	Transaction with replicated data-- Consensus and agreement algorithms	2
4.3	Checkpointing and rollback recovery: Issues in failure recovery --Checkpoint-based recovery -- Log based rollback recovery	1
4.4	Coordinated checkpointing algorithm -- asynchronous checkpointing and recovery.	2

Module No.	Topic	No. of Periods
5	Distributed File Systems	
5.1	File Models, File Accessing Models, File-sharing Semantics, File-caching Schemes	2
5.2	File Replication,- Sun's network file system, Andrews file system	2
5.3	Naming- Directory services.- Security:Techniques	2
6	Case Study	
6.1	Peer to Peer Computing,Distributed Object Based System, Distributed Web Based System, Distributed Coordinated System	2
	Total	36

Course Designer(s):

1. Dr.R. Leena Sri (rlsit@tce.edu)
2. Mr.D.Nagendra Kumar (dnkcse@tce.edu)

22CS632 COMPILER DESIGN

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

The objective of this course is to learn basic principles and advanced techniques of compiler design. It focuses on general phases of a compiler such as lexical analysis, syntactic analysis, semantic analysis, abstract syntax tree and code-generation as well as basic optimizations.

Prerequisite

Theory of Computation

Course Outcomes

On the successful completion of the course, students will be able to

No.	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the role of compilers for converting the high level language to machine level language and summarize the phases of compilers	TPS2	70	80
CO2	Apply lexical and syntax analysis technique for recognizing the tokens of a given language	TPS3	70	75
CO3	Demonstrate context-free grammars and construct parsing techniques (Top-Down and Bottom-Up) for language syntax verification.	TPS3	70	75
CO4	Develop syntax-directed translation schemes and generate intermediate code representations for arithmetic and control flow statements.	TPS3	70	75
CO5	Use optimization strategies for basic blocks and global data flow to enhance compiler efficiency.	TPS3	70	75
CO6	Analyse code generation algorithms including register allocation, instruction selection, and peephole optimization for efficient target language.	TPS4	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1.	M	L			L	L	M	M	M		M	M	L	M
CO2.	S	M	L		M	L	M	M	M		M	M	M	M

CO3.	S	M	L		M	L	M	M	M		M	M	M	M
CO4.	S	M	L		M	L	M	M	M		M	M	M	M
CO5.	S	M	L		M	L	M	M	M		M	M	M	M
CO6.	S	S	M	L	M	M	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2		Terminal					
	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	15	15		-	-	-		-	-	-	-	-	-	-			8		-	-	-
CO2	5	10	20	-	-	-	50	-	-	-	-	-	-	-			4	15	-	-	-
CO3	5	10	20		-	-	50	-	-	-	-	-	-	-			4	15		-	-
CO4	-	-	-	-	-	-	-	5	5	20		-	-	30			4	15		-	-
CO5	-	-	-	-	-	-	-	5	5	20		-	-	30			4	15		-	-
CO6	-	-	-	-	-	-	-	5	5	10	20	-	-		40		4	-	12	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Compiler structure: The analysis-synthesis model of compilation, various phases of a compiler, and tool-based approach to compiler construction.

Lexical analysis: interface with input, parser and symbol table, token, lexeme and patterns. Difficulties in lexical analysis. Error reporting. Implementation. Regular definition, Transition diagrams, NFA to DFA. **Case Study:** Lex/ Flex

Syntax analysis: Review on CFGs, ambiguity, associativity, precedence, top-down parsing, recursive descent parsing, transformation on the grammars, predictive parsing, bottom-up parsing, LR parsers (SLR, CLR, LALR). **Case Study :** Yacc / Bison

Syntax-directed Translation and Intermediate Code Generation: inherited and synthesized attributes, L- and S-attributed definitions, Intermediate representations, translation of declarations, assignments, control flow, boolean expressions, and procedure calls, Back patching. **Case Study:** LLVM IR, GIMPLE(GCC), MLIR

Code Optimization and Runtime Environments: Principal Sources of Optimization, Optimization of basic Blocks, DAG representation of Basic Blocks, Global Data Flow Analysis, Runtime Environments, Storage Organization, Storage Allocation strategies, Access to non-local names, Parameter Passing, Error detection and recovery. **Case Study: LLVM, GCC Optimizer**

Code generation and instruction selection: Issues in the design of code generator, basic blocks, flow graphs, register allocation and assignment, code generation, DAG representation of programs, code generation from DAGs, peephole optimization, **Case Study:** LLVM Code Generator, GCC Back end

Text Books

1. A.V. Aho, R. Sethi, J.D. Ullman, "Compilers: Principles, Techniques and Tools", Pearson Education, Second Edition, 2014.
2. AW Appel, M Ginsburg, Modern Compiler Implementation in C, Cambridge University Press, 2004.

Reference Books & Web resources

1. K Cooper, L Torczon, Engineering a Compiler, 2nd Ed., Morgan Kaufmann, 2011
2. Michael L Scott, Programming Language Pragmatics, 3rd Ed., Morgan Kaufmann, 2009
3. Santanu Chattopadhyay, "Compiler Design", PHI Learning Pvt. Ltd., 2015.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Compiler structure (2)	
1.1	Analysis-synthesis model of compilation, Phases of a compiler	1
1.2	Tool-based approach to compiler construction	1
2	Lexical analysis (4)	
2.1	Interface with input, parser, and symbol table	1
2.2	Token, lexeme and patterns, Difficulties in lexical analysis, Error reporting	1
2.3	Implementation. Regular definition, Transition diagrams. Case Study: Lex/ Flex	2
3	Syntax analysis (9)	
3.1	Review on CFGs, ambiguity, associativity	1
3.2	Precedence, top-down parsing	1
3.3	Recursive descent parsing	2
3.4	Transformation on the grammar, predictive parsing, bottom-up parsing	2
3.5	LR parsers (SLR, LALR, CLR). Case Study : Yacc / Bison	3
4	Syntax-directed definitions and Intermediate Code Generation(7)	
4.1	Inherited and synthesized attributes, L- and S-attributed definitions	2
4.2	Intermediate representations	2
4.3	Translation of declarations, assignments	1
4.4	Translation of control flow, boolean expressions, and procedure calls, Backpatching Case Study: LLVM IR, GIMPLE(GCC),MLIR	2
5	Code Optimization and Runtime Environments (8)	
5.1	Principal Sources of Optimization, Optimization of basic Blocks	2

Module No.	Topic	No. of Periods
5.2	DAG representation of Basic Blocks, Global Data Flow Analysis	2
5.3	Storage organization, activation tree, activation record	2
5.4	Stack allocation of activation records, parameter passing mechanisms Case Study: LLVM, GCC Optimizer	2
6	Code generation and instruction selection (6)	
6.1	Issues, basic blocks, and flow graphs,	2
6.2	Register allocation, code generation, DAG representation of programs, code generation from DAGs	2
6.3	Peephole optimization, code optimization for machine architectures, Case Study: LLVM Code Generator, GCC Back end	2
	Total	36

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, rlsit@tce.edu
Computer Science and Engineering,

22CS671	DISTRIBUTED COMPUTING LAB	Category	L	T	P	C
		PC	0	0	2	1

Preamble

This Practical course will enable students clearly understands broad range of concepts related to distributed computing including architecture, programming paradigms, algorithms and other applications of distributed computing.

Prerequisite

- 22CS361 Operating Systems (Theory cum Practical)
- 22CS431 Data Communication and Networks
- 22CS571 Network Programming Lab

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Design an Application programming Interface for multi-processing and multi programming environment. (Apply)	TPS3	A	85
CO2	Demonstrate a framework for a concurrent system using logical time and global state. (Apply)	TPS3	A	85
CO3	Implement Mutual Exclusion and deadlock detection algorithms in Distributed Systems. (Apply)	TPS3	A	85
CO4	Implement File sharing, caching and accessing mechanism in Distributed File systems. (Apply)	TPS3	A	85
CO5	Design distributed Object Based systems (Apply)	TPS3	A	85
CO6	Design distributed Web Based systems (Apply)	TPS3	A	85

Mapping with Programme Outcomes

C Os	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
C O1	S	M	L		M	M	L	S	S	S		M	M	M	M
C O2	S	M	L		M	M	L	S	S	S		M	M	M	M
C O3	S	M	L		M	M	L	S	S	S		M	M	M	M
C O4	S	M	L		M	M	L	S	S	S		M	M	M	M
C O5	S	M	L		M	M	L	S	S	S		M	M	M	M
C O6	S	M	L		M	M	L	S	S	S		M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cogniti ve Level s	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyz e		
Evalua te		
Create		

List of Experiments/Activities with CO Mapping

Sl.No	Experiments	CO
1.	Implementation of application programming interface with OpenMP programming	CO1
2.	Implementation of application programming interface with MPI programming	CO1
3.	Implementation of functioning of Logical and Vector clock.	CO2
4.	Implementations of Distributed Mutual exclusion algorithm	CO3
5.	Implementation of Non Token/ Token based algorithm in Distributed system	CO3
6.	Implementation of the Distributed Deadlock Detection algorithm	CO3
7.	Implementation of Hadoop –Map Reduce Programming	CO4
8.	Implementation of Implement 'RPC' mechanism for accessing methods of remote systems	CO5
9.	Implement CORBA mechanism by using C++ program at one end and Java Program on the other	CO5
10.	Implement Web Service for an web application	CO6

Learning Resources

1. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2011.
2. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, Distributed Systems: Concepts and Design, Fifth Edition, Pearson Education, 2017.
3. M. Van Steen, A.S. Tanenbaum, Distributed Systems, Third Edition, CreateSpace Independent Publishing Platform, 2017.
4. Garg VK. Elements of distributed computing. John Wiley & Sons, 2002.
5. Pradeep K Sinha, "Distributed Operating Systems: Concepts and Design", Prentice Hall of India, 2007

Course Designers:

1. Dr.R. Leena Sri (rlsit@tce.edu)
2. Mr.D.Nagendra Kumar (dnkcse@tce.edu)

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

SEVENTH and EIGHTH SEMESTER

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015**B.E. / B.Tech. Degree Programmes****COURSES OF STUDY**

(For the candidates admitted from 2025-26 onwards)

SEVENTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CS711	Engineering Economics	HSS	3	-	-	3
22CS721	Human Computer Interaction	ESC	2	-	-	2
22CSPX1	Programme Elective	PE	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
PROJECT COURSE						
22CS791	Project - III	PW	-	-	6	3
Total			17	-	6	20

EIGHTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22CSPX1	Programme Elective	PE	3	-	-	3
22CSPX1	Programme Elective	PE	3	-	-	3
PROJECT COURSE						
22CS891	Project - IV	PW	-	-	6	3
Total			6	-	6	9

BSC : Basic Science Courses

PC : Professional Core Courses

ESC : Engineering Science Courses

PW : Project Work

L : Lecture

T : Tutorial

P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit

1 Hour Tutorial is equivalent to 1 credit

2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015**B.E. / B.Tech. Degree Programme****SCHEME OF EXAMINATIONS**

(For the candidates admitted from 2025-26 onwards)

SEVENTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CS711	Engineering Economics	3	40	60	100	27	50
2	22CS721	Human Computer Interaction	3	40	60	100	27	50
3	22CSPX1	Programme Elective	3	40	60	100	27	50
4	22CSPX1	Programme Elective	3	40	60	100	27	50
5	22CSPX1	Programme Elective	3	40	60	100	27	50
6	22CSPX1	Programme Elective	3	40	60	100	27	50
PROJECT COURSE								
7	22CS791	Project - III	3	40	60	100	27	50

EIGHTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	22CSPX1	Programme Elective	3	40	60	100	27	50
2	22CSPX1	Programme Elective	3	40	60	100	27	50
PROJECT COURSE								
3	22CS891	Project - IV	3	40	60	100	27	50

* CA evaluation pattern will differ from course to course and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

		Category	L	T	P	Credit
22CS711	ENGINEERING ECONOMICS	HSS	3	0	0	3

Preamble

Efficient functioning of any business organization would enable it to provide goods/services at a lower price. In the process of managing organizations, the managers at different levels should take appropriate economic decisions which will help in minimizing investment, operating and maintenance expenditures besides increasing the revenue, savings and such other gains of the organization. These can be achieved through Engineering Economics which deals with the methods that enable one to make economic decisions towards minimizing costs and/or maximizing benefits to business organizations. The required techniques and methods are discussed in this course. Also, AI is becoming more and more important in making an impact on economic and finance theories. Hence, a few case studies are also discussed.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Determine the cut off production volume for a company to make profit by performing break even analysis with an understanding of various costs involved.	TPS3	A	70
CO2	Make investment decisions by applying interest formulas	TPS3	A	70
CO3	Choose the best project option from among a set of competing alternatives.	TPS3	A	70
CO4	Perform replacement / maintenance analysis to minimize the maintenance cost in an organization	TPS3	A	70
CO5	Perform depreciation accounting to obtain the book value of an asset.	TPS3	A	70
CO6	Build machine learning models for aiding economic policy design with a comprehension of the impact of AI in economic and finance theories.	TPS3	A	70

Mapping with Programme Outcomes

COs	P O 1	P O 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L					L	M	M	M	L	L		M
CO2.	S	M	L					L	M	M	M	L	L		M
CO3	S	M	L					L	M	M	M	L	L		M
CO4	S	M	L					L	M	M	M	L	L		M
CO5	S	M	L					L	M	M	M	L	L		M
CO6	S	M	L		L	L		L	M	M	M	M	L	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1			CAT2			ASSIGNMENT 1			ASSIGNMENT 2			TERMINAL		
TPS Scale	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
CO1	5		16						20				3		8
CO2	5	10	30						45				2	5	15
CO3		10	24						35					5	12
CO4				5		16						20		5	12
CO5				5	10	30						45	2	5	15
CO6					10	24						35	3		8

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Economics: Flow in an Economy- Law of Supply and Demand - Concept of Engineering Economics- Types of Efficiency- Definition and Scope of Engineering Economics - Elements of Costs- Other Costs/Revenues- Marginal Cost- Marginal Revenue- Sunk Cost- Opportunity Cost- Break-Even Analysis - Profit/Volume Ratio (P/V Ratio)- Application in make or buy decision.

Interest Formulas and Their Applications: Introduction -Time Value of Money- Interest Formulas - Single-Payment Compound Amount Single-Payment Present Worth Amount - Equal-Payment Series Compound Amount - Equal Payment Series Sinking Fund- Equal-Payment Series Present Worth Amount- Equal-Payment Series Capital Recovery Amount - Uniform Gradient Series Annual Equivalent Amount -Effective Interest Rate.

Comparison of Project Alternatives: Present worth method - Future worth method - Annual equivalent method - Rate of return method.

Replacement and Maintenance Analysis: Introduction - Types of Maintenance - Types of Replacement Problem - Determination of Economic Life of an Asset - Replacement of Existing Asset with a New Asset - Capital Recovery with Return- Concept of Challenger and Defender.

Depreciation: Introduction- Methods of Depreciation - Straight Line Method of Depreciation - Declining Balance Method of Depreciation - Sum-of-the-Years-Digits Method of Depreciation - Sinking Fund Method of Depreciation - Service Output Method of Depreciation.

Application of Machine Learning in Economics: Types of machine learning- Use Cases- Case Studies on prediction of real estate prices, market basket analysis and economic growth.

Text Book

Panneerselvam R, Engineering Economics, PHI Learning Private Limited; 2nd edition, 2013.

Reference Books & web resources

1. Chan S Park, Contemporary Engineering Economics, Pearson, 5th edition, 2015.
2. NPTEL course on Engineering Economic Analysis - <https://nptel.ac.in/courses/112107209>
3. Sendhil Mullainathan and Jann Spiess, Machine Learning: An applied Econometric Approach, *Journal of Economic Perspectives*-Volume 31, Number 2-Spring 2017- Pages 87–106

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Economics:	
1.1	Flow in an Economy- Law of Supply and Demand - Concept of Engineering Economics-Types of Efficiency- Definition and Scope of Engineering Economics	1
1.2	Elements of Costs- Other Costs/Revenues- Marginal Cost- Marginal Revenue- Sunk Cost- Opportunity Cost	1
1.3	Break-Even Analysis-Profit/Volume Ratio (P/V Ratio)	1
1.4	Application in make or buy decision.	1
2	Interest Formulas and Their Applications:	
2.1	Introduction -Time Value of Money- Interest Formulas - Single-Payment Compound Amount- Single-Payment Present Worth Amount .	2

Module No.	Topic	No. of Periods
2.2	Equal-Payment Series Compound Amount - Equal Payment Series Sinking Fund- Equal-Payment Series Present Worth Amount- Equal-Payment Series Capital Recovery Amount .	4
2.3	Uniform Gradient Series - Annual Equivalent Amount - Effective Interest Rate.	2
3	Comparison of Project Alternatives:	
3.1	Present worth method - Future worth method.	3
3.2	Annual equivalent method - Rate of return method.	3
4	Replacement and Maintenance Analysis:	
4.1	Introduction - Types of Maintenance - Types of Replacement Problem - Determination of Economic Life of an Asset	3
4.2	Replacement of Existing Asset with a New Asset - Capital Recovery with Return- Concept of Challenger and Defender.	3
5	Depreciation:	
5.1	Introduction- Methods of Depreciation - Straight Line Method of Depreciation - Declining Balance Method of Depreciation.	4
5.2	Sum-of-the-Years-Digits Method of Depreciation - Sinking Fund Method of Depreciation - Service Output Method of Depreciation.	4
6	Application of Machine Learning in Economics:	
6.1	Types of machine learning- Use Cases	1
6.2	Case Studies on prediction of real estate prices, market basket analysis and economic growth.	3
Total		36

Course Designer(s):C.Sridharan, Associate Professor of Computer Science and Engineering, cscse@tce.edu

22CS721	HUMAN COMPUTER INTERACTION
----------------	---------------------------------------

Category	L	T	P	Credit
ES	2	0	0	2

Preamble

The objective of this course is to introduce the basic theories and concepts of human-computer interaction (HCI). Through a thinking process, students will gain knowledge in components of human perception, cognition, and learning as they apply to the design, implementation, and evaluation of interfaces.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Acquire fundamental concepts of computer component functions regarding interaction with human and vice versa	TPS2	B	85
CO2	Choose the appropriate standards for HCI systems preferred by the user	TPS3	B	80
CO3	Categorize the given problem and design the interface using suitable interaction styles	TPS4	B	80
CO4	Using the interface to assess the extent and accessibility of the system functionality	TPS3	B	80
CO5	Apply evaluation strategies and validate the interfaces to meet ethical standards	TPS3	B	80
CO6	Design and implement usable and engaging interfaces	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L											L		
CO2	S	M	L									L	M		
CO3	S	S	M	L	M	L	L	M	L	L	L	L	M	M	M
CO4	S	M	L		L			M	L	L		L	M	L	L
CO5	S	M	L		L			M	L	L		L	M	L	L

CO6	S	M	L		L			M	L	L		L	M	L	L
-----	---	---	---	--	---	--	--	---	---	---	--	---	---	---	---

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
TP S Sca le	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO 1	10	5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO 2	5	10	20	-	-	-	-	-	-	-	-	-	-	15	-	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO 3	10	10	10	20	-	-	-	-	-	-	-	-	-	15	70	-	-	-	-	-	-	-	-	-	2	5	-	20	-	-
CO 4	-	-	-	-	-	-	10	10	15	-	-	-	-	-	-	-	-	-	-	15	-	-	-	-	2	5	10	-	-	-
CO 5	-	-	-	-	-	-	5	10	15	-	-	-	-	-	-	-	-	-	-	15	-	-	-	-	2	5	10	-	-	-
CO 6	-	-	-	-	-	-	10	10	15	-	-	-	-	-	-	-	-	-	-	70	-	-	-	-	5	10	-	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Foundations: History-Goals-Tools - User-oriented approach-Interface- Elements and methods - Relationship between HCI and AI.

Requirements: Components - Characteristics - Usability- Understandability – Correctness - Completeness- Scalability and Concurrency-Problems in Multi-Portal Systems - Error handling.

Design: Basics - User Focus – Scenarios – Navigation – Layout and Menu design - Usability Engineering - Design Rules – Interaction Models – Ergonomics - HCI Patterns - Perceptions and experiences of using AI – Cognitive Design Models using AR/VR.

Evaluation: Principles - Expert Reviews – Usability and Acceptance Testing – Evaluation of Spastic Devices Interaction Panels.

Visualization: Basics-Attention and overreliance on AI- Ambiguity in human intent and communication- Interpretability and Explainability - Interactive visualization – Understanding model behaviour.

Ethics and Safety: Ethics- Fairness and Equity- Privacy- Transparency-Bias- Reliability and Trustworthiness.

Case Study –Web Design Interface, Conversational AI and Explainable AI

Reference Books & web resources

1. Alan Dix, Janet E.Finlay, Gregory D.Abowd, Russell Beale , “Human- Computer Interaction” (3rd Edition), Prentice-Hall, Inc, 2009, ISBN : 0130461091.
2. B. Shneiderman; Designing the User Interface, Addison Wesley, 5th Edition, 2014.
3. John M.Carrol, “Human Computer Interaction in the New Millenium”, Pearson Education, 2002.
4. Don Norman, “The Design of Everyday Things”, First Edition, Basic Books, 2013.
5. Henry A.Kissinger, Eric Schmidt, Daniel Huttenlocher, “ The Age of A.I : and our Human Future” , John Murray, 2021, ISBN: 1529375975
6. <http://cs.brown.edu/courses/cs295-7/>

7. <http://www.cs.tufts.edu/~jacob/250bci/>
8. <https://courses.isds.tugraz.at/hci/hci.pdf>
9. <http://iitg.ac.in/uclab/courses.html>

Module No.	Topic	No. of Periods
1	Foundations (2)	
1.1	History	1
1.2	Goals	
1.3	Tools	
1.4	User-oriented approach	1
1.5	Interface	
1.6	Elements and methods	
1.7	Relationship between HCI and AI	
2	Requirements (3)	
2.1	Components	1
2.2	Characteristics	
2.3	Usability	1
2.4	Understandability	
2.5	Correctness	
2.6	Completeness	
2.7	Scalability and Concurrency	
2.8	Problems in Multi-portal Systems	1
2.9	Error handling	
3	Design (7)	
3.1	Basics	1
3.2	User Focus	
3.3	Scenarios	

Module No.	Topic	No. of Periods
3.4	Navigation	
3.5	Layout and Menu design	2
3.6	HCI in Software Process	
3.7	Usability Engineering	1
3.8	Design Rules	
3.9	Interaction Models	
3.10	Ergonomics	1
3.11	HCI Patterns	
3.12	Perceptions and experiences of using AI	
3.13	Cognitive Design Models using AR/VR	2
4	Evaluation (3)	
4.1	Principles	1
4.2	Expert Reviews	
4.3	Usability and Acceptance testing	2
4.4	Evaluation of Spastic Devices Interaction Panels	
5	Visualization (4)	
5.1	Attention and overreliance on AI	1
5.2	Ambiguity in human intent and communication	1
5.3	Interpretability and Explainability	1
5.4	Interactive visualization	
5.5	Understanding model behaviour	1
6	Ethics and Safety (2)	
6.1	Ethics	1
6.2	Fairness and Equity	
6.3	Privacy	

Module No.	Topic	No. of Periods
6.4	Transparency	1
6.5	Bias	
6.6	Reliability and Trustworthiness	
	Case Study –Web design Interface, Conversational AI and Explainable AI	3
	Total	24

Course Designer(s):

- | | | |
|----|---|---------------|
| 1. | Dr.M.Suguna, Assistant Professor,CSE | mscse@tce.edu |
| 2. | Ms.C.Santhiya, Assistant Professor, CSE | csit@tce.edu |

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

PROGRAMME ELECTIVES

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSPA1**DATA MINING****Category**
PSE**L**
3**T**
0**P**
0**Credit**
3**Preamble**

This course aims at facilitating the student to understand the concepts of data warehousing and various techniques involved in mining the data. It helps the students to learn about constructing and querying over data warehouse, preparing data for analysis, concepts and methodology of different data mining techniques. Students can also learn and practice the applications of data mining algorithms over different types of data.

Prerequisite

- Knowledge in Databases

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct a data warehouse for the given requirements and execute various OLAP operations on the data warehouse.	TPS3	B	80
CO2	Identify suitable data pre-processing methods to extract insights from the given data.	TPS3	B	80
CO3	Generate association rules using frequent pattern mining algorithms from the given dataset.	TPS3	B	80
CO4	Employ different classification and prediction algorithms to build analytical models for real-life problems.	TPS3	B	80
CO5	Develop clusters using appropriate clustering methods to handle outliers present in the given data set.	TPS3	B	80
CO6	Use suitable mining algorithms to extract insights from different types of data such as text, web, and sequence data.	TPS3	B	80
CO7	Experiment with various data mining techniques using modern tools such as Orange, Rapid Miner, Weka.	TPS4	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		M	M	M		M	M	L	M
CO2	S	M	L		M	L		M	M	M		M	M	L	M
CO3	S	M	L		M	L		M	M	M		M	M	L	M
CO4	S	M	L		M	L		M	M	M		M	M	L	M
CO5	S	M	L		M	L		M	M	M		M	M	L	M
CO6	S	M	L		M	L		M	M	M		M	M	L	M
CO7	S	S	M	L	M	M		M	M	M	M	M	S	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	10	20										2	5	10			
CO2	5	10	10											5	10			
CO3	10	10	20										2	5	10			
CO4							10	10	20				2	5	10			
CO5							5	10	20				2	5	10			
CO6							5	10	10				2	10	5			
CO7																		
Total	20	30	50				20	30	50				10	35	55			

CO	Assignment I						Assignment II					
	1	2	3	4	5	6	1	2	3	4	5	6
CO1			20									
CO2			30									
CO3			30									
CO4									30			
CO5									30			

CO6									20			
CO7				20						20		
Total			80	20					80	20		

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Data warehousing and OLAP: Data warehousing architecture and Implementation, ETL process, OLTP, OLAP, ROLAP, MOLAP, SQL support for OLAP, OLAP operations Multidimensional data model – Data cubes computation methods, Attribute oriented induction, Schema structures – star, snowflake, galaxy schemas, concept hierarchy.

Data Mining and KDD Process -: Data Mining basics, KDD process, Mining on different types of data and patterns, Related technologies, Applications and Issues in data mining. **Data Pre-processing:** Data objects and attribute types, Data Cleaning, Data Integration, Data Reduction, Data Transformation, and Discretization.

Mining Frequent Patterns, Associations, and Correlations: Concepts, Frequent pattern mining methods - Apriori and FP-Growth algorithm, mining using vertical data format, closed and maximal patterns - Pattern evaluation methods -Pattern mining in multilevel, multidimensional space.

Classification: Decision tree Induction - Bayesian classification–Rule-based classification, Support vector machines, K-NN classifier, Associative Classification - Model evaluation methods

Clustering: Cluster Analysis, measuring data similarity and dissimilarity, Partition based Methods: K-Means, K-Medoids, Hierarchical Methods: AGNES, DIANA, BIRCH, Density-based Methods: DBSCAN, Model-based clustering – COBWEB, Outlier Analysis - Outlier detection methods.

Other mining methods: Sequential Pattern Mining – SPADE, GSP, Text Mining – Pre-processing, Text classification, and clustering, Web Mining – Web Structure, Web content, and Web usage mining, Overview of data stream mining.

CO7 will be evaluated by the Mini project.

Guidelines for the Mini-project:

Group formation: Students are split into project groups with around 3 members in each group. A team can execute the project with various data mining algorithms and improve the efficiency of the algorithm by pre-processing methods using any of the data mining software like Orange, Weka and Rapid Miner etc. At the end of the semester, the team members have to present their project, submit their report and share their lessons learned/best practices with other teams.

Some of the activities may include: (but not limited to)

- ✓ Application identification and data set collection
- ✓ Select relevant data mining algorithms to extract knowledge from the data set.
- ✓ Design a diagram of knowledge extraction from raw data.
- ✓ Results and performance analysis for the chosen data mining technique.
- ✓ Documentation

Some of the Mini-project titles may include: (but are not limited to)

- ✓ e-governance
- ✓ Health care
- ✓ Banking
- ✓ University data analysis
- ✓ Social media

Text Books

1. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining Concepts and Techniques", Third Edition, Elsevier, 2011.
2. Arun K. Pujari, "Data Mining Techniques", Second Edition, University Press, 2013.

Reference Books

1. Alex Berson and Stephen J. Smith "Data Warehousing, Data Mining & OLAP", Tata McGraw – Hill Edition, Tenth Reprint 2007.
2. Ian H. Witten, Eibe Frank, Mark A. Hall, "Data Mining: Practical Machine Learning Tools and Techniques", Elsevier, Fourth Edition, 2016.
3. G. K. Gupta, "Introduction to Data Mining with Case Studies", Prentice Hall India Learning Private Limited, Third Edition, 2014.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lecture Hours
1	Data warehousing and OLAP (6)	
1.1	Data warehousing concepts and architecture, Implementation	1
1.2	ETL process, OLTP, OLAP, ROLAP, MOLAP, SQL support for OLAP, OLAP operations	2
1.3	Multidimensional data model – Data cube computation methods, Attribute oriented induction	2
1.4	Schema structures – star, snowflake, galaxy schemas, concept hierarchy	1
2	Data Mining and KDD Process, Data Pre-processing (6)	
2.1	Data Mining Basics, KDD process, Mining on different types of data and patterns	1
2.2	Related technologies, Applications and issues in data mining	1
2.3	Data Pre-processing: Data objects and attribute types, Data Cleaning	1
2.4	Data Integration and Data Reduction	2
2.5	Data Transformation, Data discretization	1
3	Mining Frequent Patterns, Associations, and Correlations(6)	
3.1	Concepts, Frequent pattern mining methods - Apriori algorithm	2
3.2	FP-Growth Algorithm, Mining using vertical data format, closed and maximal patterns	2
3.3	Pattern evaluation methods	1
3.4	Pattern mining in multilevel, multidimensional space	1
4	Classification (7)	

4.1	Decision tree Induction	2
4.2	Bayesian classification	1
4.3	Rule-based classification, Support vector machines	2
4.4	K-NN classifier , Associative Classification	1
4.5	Model evaluation methods	1
5	Clustering(7)	
5.1	Cluster Analysis, measuring data similarity and dissimilarity, Partition-based Methods: K-Means, K-Medoids	2
5.2	Hierarchical Methods: AGNES, DIANA, BIRCH	1
5.3	Density-based Methods: DBSCAN	1
5.4	Model-based clustering – COBWEB	2
5.5	Outlier Analysis, Outlier detection methods	1
6	Other mining methods (4)	
6.1	Sequential Pattern Mining – SPADE, GSP	1
6.2	Text Mining – Pre-processing, Text classification, and clustering	1
6.3	Web Mining – Web Structure, Web content, and Web usage mining	1
6.4	Overview of data stream mining	1
	Total Hours	36

Course Designer(s):

1. Dr.B.Subbulakshmi, AP,CSE
2. Dr.M.Nirmala Devi, AP,CSE

bscse@tce.edu

mnit@tce.edu

22CSPB1 INTERACTIONS DESIGN FOR XR

Category	L	T	P	Credit
PSE	3	0	0	3

Preamble

Students will be able to create a virtual environment and interact with it using devices, including the concepts and technologies for VR interaction.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO Number	Course Outcome Statement	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Create 3D graphics objects and to lay them out to create an environment.	TPS3	B	75
CO2	Interact with a VR world, including the concepts and technologies of VR interaction.	TPS3	B	70
CO3	Build simple, interactive mobile applications with augmented reality functions.	TPS3	B	70
CO4	Develop a walkthrough for a model and view it on a mobile.	TPS3	B	70
CO5	Discuss the applications of VR in Business, Manufacturing, Architecture and Construction and Entertainment	TPS3	B	70
CO6	Evaluate current trends of AR and VR media delivery to propose options to potential clients, and discuss the benefits, challenges and misconceptions involved with working in AR and VR.	TPS4	B	70

Mapping with Programme Outcomes

CO s	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L		S	M			S	S		M	M	M	L
CO 2	S	M	L		S	M			S	S		M	M	M	L
CO 3	S	M	L		S	M			S	S		M	M	M	L
CO 4	S	M	L		S	M			S	S		M	M	M	L
CO 5	S	M	L		S	M			S	S		M	M	M	L
CO 6	S	S	M	L	S	M			S	S		S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20											4	10				
CO2		20	10										4	10				
CO3		20	20										4	10				
CO4							10	20	15				4		15			
CO5							10	20	15				4		15			
CO6										10						10		

CO	ASSIGNMENT 1						ASSIGNMENT2					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1												
CO2		50										
CO3			50									
CO4									50			
CO5												
CO6										50		

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

AR VR, XR ENVIRONMENTS Virtual environments, Requirements for VR, benefits of Virtual reality, Augmented Reality, Computer-Generated Worlds, Defining Position and Orientation, The Three I's of Virtual Reality, WEB3, 5G and other technologies. Metaverse - 3D virtual space where humans can experience.

Representing the Virtual World Understanding the Human Senses and Their Relationship to Output / Input Devices. Component Technologies of Head-Mounted Displays, Augmenting Displays, Fully Immersive Displays, Sensors for Tracking Position, Orientation, and Motion, Devices to Enable Navigation

Visual Perception & Rendering - Creating a 3D Model, Drawing Lines, Shapes, and 3D Objects, drawing and design models, anchoring, scaling, and rotating of models, Immersive Walking through a model, jump/ walkthrough the models. Mobile Walkthrough.

AR applications – Building AR experience and build as android application, transform the objects in your scene, Modify the virtual environment, Load and display 3D models, Load and play audio and video.

Business and Manufacturing Using Augmented Reality for advertising, virtual prototyping system, the cave to examine the aesthetics of an automobile interior. **science and technology virtual** wind tunnel, telepresence. **Education Applications**, Tangible Skills development in Education, Knowledge Acquisition, and Concept Formation.

Case study:

Create components, integrate color or textures, use the 3D Warehouse to import images. create a VR walk-through of your Sketch Up Model an AR application to move a character off its marker and control it's animation.

Text Books

1. Alan B. Craig, William R. Sherman, Jeffrey D. Will, "Developing Virtual Reality Applications", 2017.
2. Matjaz Mihelj, Domen Novak, Samo Begus, "Virtual Reality Technology and Applications", 1st Edition, Springer Netherlands, 2014.
3. Grigore C. Burdea, Philip Coiffet, "Virtual Reality Technology", 2nd Edition, Wiley India, 2006

Reference Books & web resources

1. William R.Sherman, Alan B.Craig, "Understanding Virtual Reality – Interface, Application, Design", The Morgan Kaufmann Series, 2003.
2. John Vince, "Introduction in Virtual Reality", Springer, 2004.
3. Gerard Jounghyun Kim, "Designing Virtual Reality Systems, the Structured Approach" Springer London, 2005. Course Contents and Lecture Schedule
4. Steve Aukstakalnis, "Practical Augmented Reality", Addison Wesley 2017.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours
1	AR VR, XR ENVIRONMENTS	
1.1	Virtual environments, Requirements for VR,	1
1.2	benefits of Virtual reality, Augmented Reality, Computer-Generated Worlds,	1
1.3	Defining Position and Orientation, The Three I's of Virtual Reality, A Short History of Early Virtual Reality,	1
1.4	WEB3, 5G and other technologies.	1
1.5	Metaverse - 3D virtual space where humans can experience.	1
2	Representing the Virtual World	
2.1	Understanding the Human Senses and Their Relationship to Output / Input Devices.	1
2.2	Component Technologies of Head-Mounted Displays,	1
2.3	Augmenting Displays, Fully Immersive Displays,	1
2.4	Sensors for Tracking Position,	1
2.5	Orientation, and Motion, Devices to Enable Navigation	1
3	Visual Perception & Rendering	

3.1	Creating a 3D Model, Drawing Lines, Shapes,	2
3.2	3D Objects, drawing and design models,	2
3.3	anchoring , scaling , and rotating of models,	1
3.4	Immersive Walking through a model, jump/ walkthrough the models	1
4.1	AR applications Building AR experience and build as android application	2
4.2	transform the objects in your scene, Modify the virtual environment,	2
4.3	Load and display 3D models,	2
4.4	Design an Augmented Reality experience that can be used to visualize products and narratives, at any time from your mobile phone	2
5.1	Business and Manufacturing Using Augmented Reality for advertising	1
5.2	virtual prototyping system, the cave to examine the aesthetics of an automobile interior.	1
5.3	science and technology virtual wind tunnel, telepresence.	1
5.4	Education Applications , Tangible Skills development in Education, Knowledge Acquisition, and Concept Formation	1
	Lab activities	
1	Create 3D objects and transforms to lay out a 3D scene	2
2	Create a VR walk-through of your SketchUp Model	2
3	An AR application to move a character off its marker and control its animation.	4
	Total	36

Course Designer(s)

1. Dr.N.Shivakumar AP/CSE shiva@tce.edu
2. Mrs.G.Bhavani AP/CSE gbicse@tce.edu

22CSPC1	CLOUD COMPUTING
----------------	------------------------

Category	L	T	P	Credit
PSE	3	0	0	3

Preamble

This course provides a strong foundation of knowledge on Cloud Computing concepts and services, facilitating the usage of Cloud based services and tools in application development and deployment. Students will be able to learn how to build, deploy and manage container-based applications on cloud.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Utilize the main concepts, key technologies, strengths and limitations to understand cloud computing environment	TPS3	B	80
CO2	Choose suitable cloud architectures, delivery and deployment models for the given scenario	TPS3	B	80
CO3	Identify the virtualization technology that enhance cloud computing	TPS3	B	80
CO4	Categorize and demonstrate the concept of containerization using Docker	TPS3	B	80
CO5	Illustrate the data availability, data replication, and data footprint reduction techniques of cloud storage services	TPS3	B	80
CO6	Examine cloud environment using cloudsim	TPS4	B	80

Mapping with Programme Outcomes

COs	P01	P02	P03	P04	P05	P06	P07	P08	P09	PO10	PO11	PO12	PS01	PS02	PS03

CO 1	S	M	L									L	M		
CO 2	S	M	L						L	L		L	M		L
CO 3	S	M	L		M			L	L	L		L	M	L	L
CO 4	S	M	L		M			L	L	L		L	M	L	L
CO 5	S	M	L						L	L		L	M		
CO 6	S	S	M	L	M	L		L	L	L	L	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

C O	CAT1						CAT2						Assignment1						Assignment2						Terminal					
TP S Sc al e	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO 1	10	5	10	-	-	-	-	-	-	-	-	-	-	-	30	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO 2	5	10	20	-	-	-	-	-	-	-	-	-	-	-	30	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO 3	10	10	20	-	-	-	-	-	-	-	-	-	-	-	40	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO 4	-	-	-	-	-	-	10	10	15	-	-	-	-	-	-	-	-	-	5	5	-	-	-	-	2	5	10	-	-	-
CO 5	-	-	-	-	-	-	15	10	15	-	-	-	-	-	-	-	-	-	15	30	-	-	-	-	2	5	10	-	-	-
CO 6	-	-	-	-	-	-	-	10	5	10	-	-	-	-	-	-	-	-	15	05	25	-	-	-	5	-	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Cloud components - Essential characteristics - Architectural and Technological Influences – Benefits – Limitations - Policies - Comparing cloud providers with traditional IT service providers.

Architecture, Services and Models: Architecture - NIST Cloud Computing Reference Model, Utility Computing and Federated Computing - Services: Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS) - Challenges and risks in cloud adoption- Cloud deployment model: Public clouds, Private clouds, Community clouds, Hybrid clouds-Capex Vs OpEx in cloud.

Data Center and Virtualization: IT Service management - Design considerations – Scalability, On-demand services – Energy efficiency- Components and Benefits of Virtualization - Types of Virtualizations (Storage and Desktop) – VM Provisioning and Migration.

Containerization: Overview – Virtualization Vs Containers - Features -Components-Creating Containerized Services - Managing Containers - Orchestration in docker-Cluster management in Kubernetes.

Cloud Security and Storage: Security concerns in cloud computing- Cloud data security- Security patterns and Architectural elements- Cloud storage services and functionalities- Storage system architectures- Data Footprint Reduction Techniques.

Cloud Applications: Simulating a cloud environment using cloudsim, Case Study - Use cases of Cloud in various domains.

Text Book

1. Thomas Erl, Zaigham Mahmood, Ricardo Puttini, "Cloud Computing: Concepts, Technology & Architecture", Prentice Hall Service Technology Series, 2013.
2. Nelson Ruest, Danielle Ruest, "Virtualization, A Beginner's Guide", McGraw-Hill Companies, 2009.

Reference Books & web resources

1. John Rittinghouse, James Ransome, "Cloud Computing: Implementation, Management and Security", CRC Press 2010.
2. Greg Schulz, "Cloud and Virtual Data Storage Networking", CRC Press, 2012.
3. Nelson Ruest, Danielle Ruest, "Virtualization, A Beginner's Guide", McGraw-Hill Companies, 2009.
4. <https://www.edx.org/learn/cloud-computing>.
5. <https://www.coursera.org/browse/information-technology/cloud-computing>
6. <https://docs.docker.com/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction (3)	
1.1	Cloud components	

1.2	Essential characteristics	1
1.3	Architectural and Technological Influences	
1.4	Benefits	1
1.5	Limitations	
1.6	Policies	
1.7	Comparing cloud providers with traditional IT service providers.	1
2	Architecture, Services and Models (6)	
2.1	NIST Cloud Computing Reference Model	1
2.2	Utility and Federated Computing Architecture	1
2.3	Services-SaaS, PaaS and IaaS	1
2.4	Challenges and risks in cloud adoption	1
2.5	Cloud deployment models- Public clouds, Private clouds, Community clouds, Hybrid clouds	1
2.6	Capex Vs OpEx in cloud	1
3	Data Center and Virtualization (9)	
3.1	IT Service management	1
3.2	Design considerations	1
3.3	Scalability	1
3.4	On-demand services	1
3.5	Energy efficiency	
3.6	Components and Benefits of Virtualization	1
3.7	Types of Virtualizations	2
3.8	VM Provisioning and Migration	2
4	Containerization (9)	
4.1	Overview	1

4.2	Virtualization Vs Containers	1
4.3	Features	1
4.4	Components	
4.5	Creating Containerized Services	1
4.6	Managing Containers	1
4.7	Orchestration in docker	2
4.8	Cluster management in Kubernetes	2
5	Cloud Security and Storage (5)	
5.1	Security concerns in cloud computing	1
5.2	Cloud data security	
5.3	Security patterns and Architectural elements	1
5.4	Cloud storage services and functionalities	1
5.5	Storage system architectures	1
5.6	Data Footprint Reduction Techniques	1
6	Cloud Applications (4)	
6.1	Simulating a cloud environment using cloudsim	2
6.2	Case Study - Use cases of Cloud in various domains	2
	Total	36

Course Designer(s):

1. C.Santhiya, Assistant Professor, CSE csit@tce.edu

22CSPD1**KERNEL PROGRAMMING**

Category	L	T	P	Credit
PSE	3	0	0	3

Preamble

This course introduces the fundamental design of kernel components in structuring the operating system. The course is structured on a widely used open source operating system Linux. The students will get a chance to reinforce concepts in the working of a “real” operating system. The idea is to learn and explore a full-fledged operating system and to use it for kernel-based modifications.

Prerequisite

- Basic Knowledge of Operating Systems

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Show the role of kernel in the booting process and initialization of operational computer system	TPS3	B	80
CO2	Configure and tune the kernel daemon and networking service	TPS3	B	80
CO3	Utilize the system calls and libraries of Linux kernel to perform system related tasks	TPS3	B	80
CO4	Construct Synchronization solutions to achieve consistent access to shared resource	TPS3	B	80
CO5	Construct and Configure the interrupt handlers of Linux kernel	TPS3	B	80
CO6	Examine and infer the file system organization and design of kernel modules	TPS4	B	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	S	M	L		M			L	M	M		L	M	L	L
CO2	S	M	L		M	L		L	M	M		L	M	L	L
CO3	S	M	L		M	L		L	M	M		L	M	L	L
CO4	S	M	L		M	L		L	M	M		L	M	L	L
CO5	S	M	L		M	L		L	M	M		L	M	L	L

CO6	S	S	M	L	M	L		L	M	M		L	S	L	L
-----	---	---	---	---	---	---	--	---	---	---	--	---	---	---	---

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
TP S Sca le	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO 1	5	1 0	1 5	-	-	-	-	-	-	-	-	-	-	1 0	1 0	-	-	-	-	-	-	-	-	-	2	5	6	-	-	-
CO 2	5	1 0	1 5	-	-	-	-	-	-	-	-	-	-	1 5	2 5	-	-	-	-	-	-	-	-	-	2	5	1 0	-	-	-
CO 3	1 0	1 0	2 0	-	-	-	-	-	-	-	-	-	-	1 5	2 5	-	-	-	-	-	-	-	-	-	2	5	1 0	-	-	-
CO 4	-	-	-	-	-	-	10	1 0	1 5	-	-	-	-	-	-	-	-	-	-	5	5	-	-	-	2	5	1 0	-	-	-
CO 5	-	-	-	-	-	-	5	1 0	1 5	-	-	-	-	-	-	-	-	-	-	1 5	3 0	-	-	-	2	5	1 0	-	-	-
CO 6	-	-	-	-	-	-	10	1 0	1 5	-	-	-	-	-	-	-	-	-	-	1 5	3 0	-	-	-	2	5	1 2	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Basics of Kernel: Overview of Kernel - Linux kernel design and modes of operation, Boot Process – Boot Loaders and Loading of Kernel – Case study: Initialization of the Linux Kernel.

Linux Kernel services and networking: Daemon Services, Service Configuration and tuning, Networking devices - The TCP/IP protocol stack communication- Obtaining System IP address.

Linux Kernel in Process and File management: Process, Threads, Process Address Space, Processes Scheduling – System Calls and Libraries: APIs, POSIX, and the C Library, Syscalls, System Call Handler, System call context.

Linux Kernel Synchronization: Basics of Kernel Synchronization, Kernel Synchronization Methods – Kernel Locking mechanism, Atomic Operations, Spin Locks, Semaphore.

Device Management: Device Types and Device Drivers, Interrupts and Interrupt handlers, Timers and Time management: Jiffies, Hardware Clocks and Timers.

Linux Kernel in Virtual File System and Kernel Modules: File Systems Interface and Abstraction, VFS objects: Superblock object, Inode object, Dentry object and File object, Kernel modules – Dynamic Loading and unloading of kernel modules, Case Study Building Kernel module.

Text Book

1. Kaiwan N Billimoria, "Linux Kernel Programming", Packt Publisher, 2021.
2. Robert Love, "Linux Kernel Development: A thorough guide to the design and implementation of the Linux Kernel", Third Edition, Addison-Wesley.

Reference Books & web resources

1. Daniel P.Bovet, Marco Cesati, "Understanding the Linux Kernel", , O'Reilly, Third Edition, 2006.
2. Gary Nutt , "Kernel projects for Linux", Addison Wesley, First edition, 2001.
3. Peter Jay Salzman Michael Burian Ori Pomerantz, "The Linux Kernel Module Programming Guide", 2001.
4. Linux with Operating System Concepts, Richard Fox, CRC Press, Taylor & Francis, A Chapman & Hall Book, 2015.
5. Linux Kernel Development, Robert Love, Pearson Education, Third Edition, 2010.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Basics of Kernel	
1.1	Overview of Kernel	1
1.2	Linux kernel design and modes of operation	1
1.3	Boot Process – Boot Loaders and Loading of Kernel	1
1.4	Case study: Initialization of the Linux Kernel	1
2	Linux Kernel services and networking	
2.1	Daemon Services	1
2.2	Service Configuration and tuning	2
2.3	Networking devices	1
2.4	The TCP/IP protocol stack communication	1
2.5	Obtaining System IP address	1
3	Linux Kernel in Process and File management	
3.1	Process, Threads and Process Address Space	2
3.2	Linux Processes Scheduling	2
3.3	System Calls and Libraries: APIs, POSIX, and the C Library	1
3.4	Syscalls and System Call Handler	2
3.5	System call context	1
4	Linux Kernel Synchronization	

4.1	Basics of Kernel Synchronization	2
4.2	Kernel Synchronization Methods – Kernel Locking mechanism	1
4.3	Atomic Operations	1
4.4	Spin Locks	1
4.5	Semaphore	2
5	Device Management	
5.1	Device Types and Device Drivers	1
5.2	Interrupts and Interrupt handlers	2
5.3	Timers and Time management: Jiffies, Hardware Clocks and Timers	2
6	Linux Kernel in Virtual File System and Kernel Modules	
6.1	File Systems Interface and Abstraction	1
6.2	VFS objects: Superblock object, Inode object, Dentry object and File object	2
6.3	Kernel modules – Dynamic Loading and unloading of kernel modules	1
6.4	Case Study Building Kernel module	2
	Total	36

Course Designer(s):

1. Dr. G.Madhupriya,
Associate Professor, CSE

gmadhupriya
@tce.edu

22CSPE1	WIRELESS NETWORKS
----------------	--------------------------

Category	L	T	P	Credit
PSE	3	0	0	3

Preamble

This course on Wireless Networks provides an introduction to the basic concepts of wireless networks, architecture and topologies. The objective of this course is to introduce the fundamental concepts and also discuss the main issues in wireless networks such as mobility management, capacity expansion and security. At the end of the course, the students will have a firm understanding of the basic principles of wireless Networks and the issues involved.

Prerequisite

Data communication and Networks

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the generations of wireless networks, principles of IEEE 802.11 standards for WLAN, GPRS, wireless local loops and operation of Bluetooth.	TPS2	B	70
CO2	Construct a suitable multiplexing/duplexing technique for a given specification.	TPS3	B	65
CO3	Determine the capacity of a CDMA system, along with its output based on a spread spectrum technique.	TPS3	B	65
CO4	Develop a suitable mechanism to increase the capacity of a cellular network.	TPS3	B	65
CO5	Determine the handoff transitions in a cellular network based on different algorithms.	TPS4	B	65
CO6	Examine the security mechanisms in IEEE802.11 wireless LAN and GSM.	TPS3	B	65

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-	-	-	-	-	-	-	-	-	-	L	-	-
CO2	S	M	L	-	-	L	-	L	L	L	-	L	M	-	L
CO3	S	M	L	-	-	L	-	L	L	L	-	L	M	-	L
CO4	S	M	L	-	-	L	-	L	L	L	-	L	M	-	L
CO5	S	M	L	L	-	L	-	L	L	L	-	L	M	-	L
CO6	S	M	L	-	-	L	-	L	L	L	-	L	M	-	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment		Terminal					
													1	2						
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	3	3	1	2	3	4	5	6
CO1	20	20													10	15				
CO2	5	10	20													5	10			
CO3	5	10	10													10	5			
CO4							10	10	20							5	10			
CO5							10	10	10	10						5	5	5		
CO6								10	10							10	5			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Generations of wireless networks- 1G wireless systems- FDMA, 2G wireless systems-TDMA, GSM architecture- Mechanism to support Mobile Environment-GPRS – Reference Architecture-Mobility Support in GPRS- Spread spectrum –FHSS, DSSS, CDMA, 3G wireless systems,4G and beyond

Multiple Access Techniques-Comparison of TDMA,FDMA and CDMA-Capacity of CDMA - Duplexing techniques FDD and TDD and comparison

IEEE 802.11 LAN standard- IEEE 802.11 Protocol architecture and services, OFDM and MAC.- Bluetooth overview, Protocol architecture =, Wireless Local loop

Infrastructure Network Topology- Cellular topology ,Cell fundamentals, Signal to interference ratio calculation, Capacity expansion techniques – Cell splitting, sectoring and overlaid methods

Issues in mobility management- Location management- location update algorithms- Paging Scheme- Location information Disseminations-Hand off Management – Architecture- algorithms- Handoff management process- Mobile IP -Security mechanisms – Cryptographic protocols in IEEE802.11 WLAN and GSM

Learning Resources

1. William Stallings, “Wireless Communications and Networks”, Pearson education, 2003
2. Kaveh Pahlavan and Prashant Krishnamurthy, “Principles of Wireless Networks – A unified approach”, Pearson Education, Fourth Edition, 2003
3. Bernard Menezes , “Network Security and Cryptography”, Cengage Learning India, Third impression, 2014.
4. Jochen Schiller, “Mobile Communications”, Pearson Education, Second Edition, 2003
5. <https://www.coursera.org/courses?query=wireless>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
------------	-------	--------------	----------------

1	Generations of wireless networks		
1.1	1G wireless systems- FDMA	1	CO1
1.2	2G wireless TDMA systems	1	CO1
1.3	GSM Architecture and GPRS	2	CO1
1.4	Spread spectrum –FHSS,DSSS,CDMA,	3	CO3
1.5	Capacity of CDMA network	2	CO3
1.6	3G wireless systems, 4G and beyond	1	CO1
1.7	Comparison of TDMA,FDMA and CDMA	1	CO2
1.8	Comparison of FDD and TDD	1	CO2
1.9	Wireless Local loop	1	CO1
2	IEEE 802.11 LAN standard		
2.1	IEEE 802.11 Protocol architecture and services,	1	CO1
2.2	OFDM and MAC	2	CO1
2.3	Bluetooth overview	1	CO1
2.4	Bluetooth protocol architecture	1	CO1
3	Infrastructure Network Topology		
3.1	Cellular topology and cell fundamentals	2	CO4
3.2	Signal to interference ratio calculation,	2	CO4
3.3	Capacity expansion techniques- cell splitting	2	CO4
3.4	Cell sectoring and Overlaid methods	2	CO4
4	Issues in mobility management and security		
4.1	Mobility management- location management	2	CO5
4.2	Hand off management	3	CO5
4.3	Security mechanisms	2	CO6
4.4	Cryptographic protocols in IEEE802.11 WLAN and GSM	3	CO6
	Total	36	

Course Designers:

1. Dr.C.Sridharan,,Associate Professor, CSE,cscse@tce.edu
2. Dr.C.Senthilkumar, Associate Professor, CScskcse@tce.edu

22CSPF1 INTERNET OF THINGS AND ITS APPLICATIONS

Category L T P Credit
PSE 3 0 0 3

Preamble

This course aims at providing a basic understanding of Internet of Things, exemplifying the application areas where Internet of Things can be applied and enables designing prototypes of Internet-connected products using appropriate tools.

Prerequisite

- Basic knowledge in Networking and Programming

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the general IoT architecture and connected domains. (Understand)	TPS3	B	85
CO2	Analyze the requirements to figure out the suitable communication technology and protocols required for an IoT application. (Analyze)	TPS4	B	85
CO3	Develop an IoT management System using network management protocol. (Apply)	TPS3	B	85
CO4	Design a step by step Model Specifications for an IoT System based on IoT – A reference model. (Apply)	TPS3	B	85
CO5	Develop an IoT application using Raspberry Pi for the given specification applying the IoT technologies. (Apply)	TPS3	B	85
CO6	Design a Web API for transaction of data and services in cloud. (Apply)	TPS3	B	85

Mapping with Programme Outcomes

Cos	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L	-								M	M		
CO2	S	S	M	L	-	L	L	L	M	L		M	M	L	L

CO3	S	M	L	-	M	L	L	L	M	L		M	M	L	L
CO4	S	M	L	-	-	L	L	L	M	L		M	M	L	L
CO5	S	M	L	-	M	L	L	L	M	L		M	M	L	L
CO6	S	M	L	-	M	L	L	L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
TPS Scale																														
CO1	1 0	1 0													3 0										5	1 0				
CO2	1 0	1 0		1 0												3 0									5			1 2		
CO3	1 0	2 0	2 0												4 0												1 2			
CO4							1 0		1 0											3 0					1 0	1 2				
CO5							1 0	1 0	2 0											3 0				5		1 2				
CO6							1 0	2 0	1 0											4 0				5		1 2				

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to IoT: Introduction, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, Types of Sensors, Types of Actuators, IoT Networking, IoT Levels and Deployment Templates, IoT and M2M, SDN and NFV for IoT. Wireless Protocol for IoT, Communication Technologies - NFC, Bluetooth, Wi-Fi, ZigBee, Z-wave, 6LoWPAN, HTTP, AMQP, Features & Functions of CoAP, MQTT, OAuth2, XMPP, CoAP vs HTTP, Digital Twins, Federated Computing.

IoT System Management with NETCONF: Need for IoT System Management, SNMP, Network Operator Requirements, NETCONF, YANG, NETOPEER, Managing Home Intrusion Detection IoT System with NETCONF-YANG

IoT Platforms Design Methodology: IoT Design Methodology – Purpose and Requirement Specification, Process Model Specification, Domain Model Specification, Information Model Specification, Service Specification, IoT Level Specification, Functional and Operational View Specification, Device and Component Integration, Application Development. IoT Systems – Logical Design using Python - Introduction to Python, Python Classes, Packages of Interest for IoT- JSON, XML, HTTPLib, URLLib, SMTPLib

IoT Physical Devices and Endpoint: Basic Building Block of an IoT Device, Exemplary Device: Raspberry Pi, About the board, Linux on Raspberry Pi, Raspberry Pi Interfaces, Programming Raspberry Pi with Python, Other IoT Devices- pcDuino, BeagleBone Black, Cubieboard.

IoT Physical Server and Cloud: Cloud Storage Models and Communication API, WAMP, Xively Cloud, Designing RESTful Web API, AWS for IoT, SkyNet IoT Messaging platform.

Case Studies Illustrating IoT Design and Deployment: Home Automation, Smart city, Environment, Agriculture, Healthcare, IIoT, Multi-Tier Deployment.

Text Book

1. Sudip Misra, Anandarup Mukherjee, Arijit Roy, "Introduction to IoT", Cambridge University Press, 2022.
2. Arshdeep Bahga, Vijay Madiseti, Internet of Things: A Hands on Approach, 2014
3. Uckelmann, Dieter, Mark Harrison, and Florian Michahelles, Architecting the Internet of Things. Springer Science & Business Media, 2011.
4. Jean-Philippe Vasseur, Adam Dunkels, Interconnecting Smart Objects with IP: The Next Internet, Morgan Kuffmann, 2010
5. Jonathan L. Zittrain, The Future of the Internet, Yale University Press & Penguin UK 2008.
6. Samuel Greengard, The Internet of Things (The MIT Press Essential Knowledge series), MIT Press, 2015

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to IoT	
1.1	Introduction, Physical Design of IoT, Logical Design of IoT,	1
1.2	IoT Enabling Technologies, Types of Sensors, Types of Actuators, IoT Networking,	1
1.3	IoT Levels and Deployment Templates, IoT and M2M, SDN and NFV for IoT.	1
1.4	Wireless Protocol for IoT, Communication Technologies - NFC, Bluetooth, Wi-Fi, ZigBee, Z-wave,	1
1.5	6LoWPAN, HTTP, AMQP, Features & Functions of CoAP, MQTT, OAuth2, XMPP, CoAP vs HTTP.	1
1.6	Digital Twins, Federated Computing.	1
2	IoT System Management with NETCONF	
2.1	Need for IoT System Management, SNMP	1
2.2	Network Operator Requirements	1

Module No.	Topic	No. of Periods
2.3	NETCONF, YANG	1
2.4	NETOPEER	1
2.5	Managing Home Intrusion Detection IoT System with NETCONF-YANG	2
3	IoT Platforms Design Methodology	
3.1	IoT Design Methodology	1
3.2	Purpose and Requirement Specification, Process Model Specification, Domain Model Specification, Information Model Specification	1
3.3	Service Specification, IoT Level Specification, Functional and Operational View Specification,	1
3.4	Device and Component Integration, Application Development.	1
3.5	IoT Systems – Logical Design using Python -	1
3.6	Introduction to Python, Python Classes,	1
3.7	Packages of Interest for IoT- JSON, XML, HTTPLib, URLLib, SMTPLib	1
4	IoT Physical Devices and Endpoint	
4.1	Basic Building Block of an IoT Device, Exemplary Device: Raspberry Pi, About the board, Linux on Raspberry Pi, Raspberry Pi Interfaces.	1
4.2	Programming Raspberry Pi with Python	4
4.3	Other IoT Devices- pcDuino, Beagle Bone Black, Cubieboard	1
5	IoT Physical Server and Cloud:	
5.1	Cloud Storage Models	1
5.2	Communication API	1
5.3	Xively Cloud	1

Module No.	Topic	No. of Periods
5.4	Designing RESTful Web API	2
5.5	AWS for IoT	1
5.6	SkyNet IoT Messaging platform	1
6	Case Studies Illustrating IoT Design and Deployment:.	
6.1	Home Automation, Smart city	1
6.2	Environment	1
6.3	Agriculture, Multi-Tier Deployment	1
6.4	Healthcare, IIoT	1
	Total	36

Course Designers:

1. M.Vijayalakshmi mviji@tce.edu
2. G.S.R.Emil Selvan emil@tce.edu

22CSPG1**RECOMMENDER SYSTEMS**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course provides an in-depth exploration of recommender systems, a critical component of many modern engineering applications. Students will learn the principles, algorithms, and practical implementation of recommendation systems in various domains.

Prerequisite

- Basic knowledge of machine learning and data analysis
- Programming skills in Python

Course Outcomes

On the successful completion of the course, students will be able to

CO#	Course Outcome 1 (CO1)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Develop user-based and item-based collaborative filtering methods by knowing the differences between memory-based and model-based collaborative filtering, and implement matrix factorization techniques	TPS3	B	70
CO2	Apply content-based filtering techniques by extracting features from items and use TF-IDF and cosine similarity to generate personalized recommendations.	TPS3	B	70
CO3	Integrate collaborative filtering and content-based filtering into hybrid recommender systems by applying weighting strategies or switching mechanisms.	TPS3	B	70
CO4	Use neural networks for recommendations, including embedding layers, matrix factorization with neural networks, neural collaborative filtering, and the application of auto encoders.	TPS3	B	70
CO5	Develop a recommender system for a specific domain or user context by considering the factors such as user preferences, item characteristics, and potential challenges like the attacks, cold start problem and scalability.	TPS3	B	70
CO6	Design, implement, and evaluate a recommender system for a specific use case, considering practical challenges and constraints	TPS4	B	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		L	L	L	M	M	M		M	M	L	M
CO2	S	M	L		L	L	L	M	M	M		M	M	L	M
CO3	S	M	L		L	L	L	M	M	M		M	M	L	M
CO4	S	M	L		L	L	L	M	M	M		M	M	L	M
CO5	S	M	L		L	L	L	M	M	M		M	M	L	M
CO6	S	S	M	L	L	L	L		S	S	S	M	M	L	S

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	10	20											40											4	4	12			
CO2	5	10	20											30											4	4	12			
CO3	5	5	20											30											4	4	12			
CO4							10	10	20											20					4	4	12			
CO5							10	10	20											20					4	4	12			
CO6										20														60						

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Recommender Systems - Overview of recommendation systems, Types of recommendation systems: collaborative filtering, content-based filtering, hybrid methods, Importance and applications of recommendation systems, Evaluation metrics for recommendation systems

Collaborative Filtering - User-based collaborative filtering, Item-based collaborative filtering, Memory-based vs. model-based collaborative filtering, Matrix factorization techniques (Singular Value Decomposition, Alternating Least Squares) - Non-negative matrix factorization

Content-Based Filtering - Feature extraction and representation of items, Term Frequency-Inverse Document Frequency (TF-IDF), Cosine similarity and other similarity measures, Building content-based recommendation models

Hybrid Recommender Systems - Integration of collaborative and content-based filtering, Weighted and switching hybrid methods, Advantages and challenges of hybrid systems

Deep Learning for Recommender Systems - Introduction to neural networks for recommendations, Embedding layers and matrix factorization with neural networks, Neural collaborative filtering, Variational Autoencoders for recommendation

Advanced Topics - Context-aware recommendation systems, Session-based recommendation, Attack-resistant recommender systems, Cold start problem and its solutions, Scalability and efficiency in recommendation systems

Real-world Applications and Case Studies - Industry use cases of recommender systems, Ethical considerations in recommendation systems, Challenges and future trends in recommendation systems

Mini Project

Text Books

1. Charu C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016.
2. Dietmar Jannach , Markus Zanker , Alexander Felfernig and Gerhard Friedrich , Recommender Systems: An Introduction, Cambridge University Press, 1st ed, 2011.
3. Francesco Ricci , Lior Rokach , Bracha Shapira , Recommender Systems Handbook, 1st ed, Springer, 2011.

Reference Books & web resources

1. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, Mining of massive datasets, 3rd edition, Cambridge University Press, 2020.
2. Kim Falk, Practical Recommender Systems, Manning Publications, 2019
3. Movielens Dataset: A popular dataset for movie recommendations, commonly used in research and academia. It includes various versions with different sizes.
4. Kaggle Datasets: Kaggle is a platform that hosts machine learning competitions. It has a variety of datasets, some of which are related to recommender systems.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Recommender Systems	
1.1	Overview of recommendation systems, Types of recommendation systems: collaborative filtering, content-based filtering, hybrid methods, Importance and applications of recommendation systems	1
1.2	Evaluation metrics for recommendation systems	1
2	Collaborative Filtering	
2.1	User-based collaborative filtering	1
2.2	Item-based collaborative filtering	1
2.3	Memory-based vs. model-based collaborative filtering	1
2.4	Matrix factorization techniques (Singular Value Decomposition, Alternating Least Squares)	2
2.5	Non-negative matrix factorization	2
3	Content-Based Filtering	
3.1	Feature extraction and representation of items	1
3.2	Frequency-Inverse Document Frequency (TF-IDF)	1

Module No.	Topic	No. of Periods
3.3	Cosine similarity and other similarity measures	1
3.4	Building content-based recommendation models	1
4	Hybrid Recommender Systems	
4.1	Integration of collaborative and content-based filtering	1
4.2	Weighted hybrid methods	1
4.3	Switching hybrid methods	1
4.4	Advantages and challenges of hybrid systems	1
5	Deep Learning for Recommender Systems	
5.1	Introduction to neural networks for recommendations	2
5.2	Embedding layers and matrix factorization with neural networks	2
5.3	Neural collaborative filtering	2
5.4	Variational Autoencoders for recommendation	2
6	Advanced Topics	
6.1	Context-aware recommendation systems	2
6.2	Session-based recommendation	2
6.3	Attack-resistant recommender systems	2
6.4	Cold start problem and its solutions	1
6.5	Scalability and efficiency in recommendation systems	1
7	Real-world Applications and Case Studies	
7.1	Industry use cases of recommender systems	1
7.2	Ethical considerations in recommendation systems	1
7.3	Challenges and future trends in recommendation systems	1
	Total	36

Course Designer(s):

1. Dr M K Kavitha Devi, Professor, Dept. of Computer Science and Engineering mkkdit@tce.edu

22CSPH1**CROSS PLATFORM MOBILE
APPLICATION DEVELOPMENT**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course provides an exploration of cross-platform mobile application development, equipping students with the skills and knowledge to create mobile apps that can run on multiple platforms using frameworks such as React Native, Flutter, or Xamarin. This course mainly focus the Dart Programming based Mobile Application development in Flutter and this will incorporate the Data connectivity, API Consuming and Application performance.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the fundamentals of Native and Hybrid Mobile application requirements for a chosen application.	TPS3	70	80
CO2	Adapt the Dart Programming functionalities such as OOPs and Collections for handling user interface in Flutter Applications.	TPS3	70	80
CO3	Apply the Flutter User Interface such as Widgets, Layout and Gestures with each stage of App state Management.	TPS3	70	80
CO4	Build an Application that incorporates the RESTAPI Service, Routing and Google Maps.	TPS3	70	80
CO5	Adapt the Database connectivity CRUD operation for chosen application and infer the performance metrics.	TPS3	70	80
CO6	Analyse the different Mobile application framework for cross platform development and conclude with the best use case.	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO2.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO3.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO4.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO5.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO6.	S	M	L	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2		Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	5	5	15	-	-	-	30	-	-	-	-	-	-	-		2	5	10	-	-	-
CO2	5	5	30	-	-	-	35	-	-	-	-	-	-	-		2	5	10	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-		2	-	10	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20		2	-	15	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20		2	5	15	-	-	-
CO6	-	-	-	-	-	-	-	5	5	10	20	-	-	30	30	-	5	-	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Introduction- Mobile app Development-Event Driven Model-Android App Development- Native App, Hybrid App, Cross-platform App-One Code Base-Multiplatform- Flutter Vs React Native-Flutter Architecture-Environmental Setup

Dart Programming-Dart Programming Basics-Datatypes, Variables-Operators-Functions-Interfaces-OOPs-Collection-Libraries.

Flutter User Interface: IDE-Widgets-Layout-Scaffold-Container-Design and Theme-Interactivity-Gestures-Animation-Calendar-App State Management-GitHub Repos

Connectivity: Packages-Accessing REST API-Navigation and Routing-Splash Screen-Google Maps-Silvers-Internalization.

Data store and Testing – SQLite-Firebase –Testing-Deployment-Development Tools-

Non-Functional Requirements: Performance in App Size-Deferred Components-Rendering-Profiling.Case study: Flutter vs Xamarinn vs Kotlin vs Ionic.

Text Book

1. Building Cross-Platform Apps with Flutter and Dart: Build scalable apps for Android, iOS, and web from a single codebase (English Edition). India, BPB PUBLICATIONS, 2023.
2. Bailey, Thomas, and Biessek, Alessandro. Flutter for Beginners: An Introductory Guide to Building Cross-platform Mobile Applications with Flutter 2.5 and Dart, 2nd Edition. United Kingdom, Packt Publishing, 2021.

Reference Books & web resources

1. Payne, Rap. Beginning App Development with Flutter: Create Cross-Platform Mobile Apps. United States, Apress, 2019.
2. Napoli, Marco L.. Beginning Flutter: A Hands On Guide to App Development. United States, Wiley, 2019.

3. Biessek, Alessandro. Flutter for Beginners: An Introductory Guide to Building Cross-platform Mobile Applications with Flutter and Dart 2. United Kingdom, Packt Publishing, 2019.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Mobile app Development-Event Driven Model	1
1.2	Android App Development	1
1.3	Native App, Hybrid App	1
1.4	Cross-platform App	1
1.5	One Code Base	2
1.6	Multiplatform	1
1.7	Flutter Vs React Native	1
1.8	Flutter Architecture-	1
1.9	Environmental Setup	2
2	Dart Programming	
2.1	Basics-Datatypes	1
2.2	Variables-Operators-Functions	1
2.3	Interfaces-OOPs	1
2.4	Collection-Libraries	1
3	Flutter User Interface	
3.1	IDE-Widgets-Layout	2
3.2	Scaffold-Container	1
3.3	Design and Theme	1
3.4	Interactivity-	1
3.5	Gestures-Animation	1
3.6	Calendar- App State Management	1
3.7	GitHub Repos	1
4	Connectivity	
4.1	Packages	1
4.2	Accessing REST API	2
4.3	Navigation and Routing	2

Module No.	Topic	No. of Periods
4.4	Splash Screen	1
4.5	Google Maps	1
5	Data store and Testing	
5.1	SQLite-Firebase	2
5.2	Testing-Deployment	1
6.	Non-Functional Requirements	
6.1	Performance in App Size	1
6.2	Deferred Components	1
6.3	Rendering-Profiling	1
6.4	Case study: Flutter vs Xamarinn vs Kotlin vs Ionic.	1
	Total	36

Course Designer(s):

1. Dr.K.Indira, Assistant Professor, Computer Science and Engineering, kiit@tce.edu
2. Dr.RajaLavanya, Assistant Professor, Computer Science and Engineering, rlit@tce.edu

22CSPJ1**SECURITY AND PRIVACY
IN CLOUD**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course provides an in-depth study of the security and privacy of cloud computing systems. It also addresses the development of an audit to ensure operational integrity and protection of customer data in cloud-based resources.

Prerequisite

Knowledge on Cloud Computing, Cryptography

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome Statement	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Determine the Cloud Computing terminology, & concepts	TPS2	B	85
CO2	Construct the security design and architectural considerations for Cloud	TPS3	B	85
CO3	Determine the Data Protection strategies in Cloud	TPS3	B	85
CO4	Explore the Identity, Access control in Cloud	TPS3	B	85
CO5	Illustrate the best practices for Cloud security using various design patterns	TPS3	B	85
CO6	Investigate the monitoring and auditing cloud applications for security	TPS3	B	85

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L						L	L	L		L	M		L
CO2	S	M	L	L	L			L	L	L	L	L	M	L	L
CO3	S	M	L	L	L			L	L	L	L	L	M	L	L
CO4	S	M	L	L	L			L	L	L	L	L	M	L	L
CO5	S	M	L	L	L			L	L	L	L	L	M	L	L
CO6	S	S	M	L	L	L	L	M	M	M	L	L	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10																							2	5				
CO2	10	10	20											100											2	5	10			
CO3	10	10	20																						2	5	10			

CO4							10	10	10														2	5	15			
CO5							10	10	10										100					2	5	10		
CO6							10	10	20																5	15		

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Fundamentals of Cloud Security- Overview of cloud security- Security Services -Confidentiality, Integrity, Authentication, Non-repudiation, Access Control - Basic of cryptography - Conventional and public-key cryptography, hash functions, authentication, and digital signatures

Security Design and Architecture for Cloud- Security design principles for Cloud Computing - End-to-End access control - Common attacks and threats - Network and Storage - Secure Isolation Strategies - Virtualization strategies - Inter-tenant network segmentation strategies

Data Protection strategies- Comprehensive data protection, Data retention, deletion and archiving procedures for tenant data, Encryption, Data Redaction, Tokenization, Obfuscation, PKI and Key, Business Continuity and Disaster Recovery (BCDR), Principles of General Data Protection Regulation (GDPR)

Access Control and Identity Management - Access control requirements for Cloud infrastructure - User Identification - Authentication and Authorization - Roles-based Access Control - Multi-factor authentication - Single Sign-on, Identity Federation - Storage and network access control options - Intruder Detection and prevention

Cloud Security Design Patterns- Introduction to Design Patterns, Cloud bursting, Geo-tagging, Secure Cloud Interfaces, Cloud Resource Access Control, Secure On-Premise Internet Access, Secure External Cloud

Monitoring, Auditing and Management- Proactive activity monitoring - Incident Response, Monitoring for unauthorized access, malicious traffic, abuse of system privileges - Events and alerts - Auditing – Record generation, Reporting and Management, Tamper-proofing audit logs, Quality of Services, Secure Management, User management, Identity management, Security Information and Event Management

Text Book

1. Roger Mchaney, "Cloud Technologies", Wiley 2021
2. Dave Shackelford, "Virtualization Security", Wiley 2013
3. Muhammad Imran Tariq, Shahzadi Tayyaba, Valentina Emilia Balas, "Security and Privacy Trends in Cloud Computing and Big Data", Taylor & Francis, 2022

Reference Books & web resources

1. Mark C. Chu-Carroll, "Code in the Cloud", Pragmatic Bookshelf, 2011.
2. RajkumarBuyya, Christian Vechhiola, "Mastering Cloud Computing Foundations and Applications Programming", Elsevier Science, 2013.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Fundamentals of Cloud Security	

Module No.	Topic	No. of Periods
1.1	Overview of Cloud Security- Security Services - Confidentiality, Integrity, Authentication, Non-repudiation, Access Control	1
1.2	Basics of Cryptography - Conventional and public-key cryptography	2
1.3	Hash functions, Authentication, and Digital Signatures	2
2	Security Design and Architecture for Cloud	
2.1	Security design principles for Cloud Computing - End-to-End access control	2
2.2	Common attacks and threats - Network and Storage - Secure Isolation Strategies	2
2.3	Virtualization strategies - Inter-tenant network segmentation strategies	2
3	Data Protection strategies	
3.1	Comprehensive data protection, Data retention, deletion and archiving procedures for tenant data	2
3.2	Data Redaction, Tokenization, Obfuscation, PKI and Key, Business Continuity and Disaster Recovery (BCDR), Principles of General Data Protection Regulation (GDPR)	3
4	Access Control and Identity Management	
4.1	Access control requirements for Cloud infrastructure - User Identification - Authentication and Authorization	2
4.2	Roles-based Access Control - Multi-factor authentication - Single Sign-on, Identity Federation	3
4.3	Storage and network access control options - Intruder Detection and prevention	2
5	Cloud Security Design Patterns	
5.1	Introduction to Design Patterns, Cloud bursting	1
5.2	Geo-tagging, Secure Cloud Interfaces	1
5.3	Secure Cloud Interfaces, Cloud Resource Access Control	1
5.4	Secure On-Premise Internet Access, Secure External Cloud	1
6	Monitoring, Auditing and Management	
6.1	Proactive activity monitoring - Incident Response, Monitoring for unauthorized access, malicious traffic, abuse of system privileges	3
6.2	Events and alerts - Auditing – Record generation, Reporting and Management	2
6.3	Tamper-proofing audit logs, Quality of Services	1
6.4	Secure Management, User management, Identity management, Security Information and Event Management	3

Module No.	Topic	No. of Periods
	Total	36

Course Designer(s):

1. Dr.M.P.Ramkumar
Associate Professor
Computer Science and Engineering ramkumar@tce.edu

22CSPK1**AI BASED RESOURCE
MANAGEMENT FOR CLOUD**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course will introduce students to the fundamental concepts of resource management in cloud. Students will learn about various AI based resource management techniques to optimize resource utilization. The course will also cover the ethical and societal implications of resource management in cloud environment.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify the fundamental concepts and techniques of resource management in cloud environment	TPS3	70	80
CO2	Acquire proficiency in different AI based techniques for cloud resource management	TPS3	70	80
CO3	Analyze AI based techniques to optimize resource prediction, allocation, scheduling, and demand analysis.	TPS4	70	80
CO4	Apply predictive modelling to forecasting	TPS3	70	80
CO5	choose the ethical and societal implications of resource management in cloud environment.	TPS3	70	80
CO6	Apply AI based cloud cost optimization strategy by selecting appropriate tool	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L		M	L		M	L	L		M	M	M	L
CO 2	S	M	L		L			M	L	L		M	M	M	L
CO 3	S	S	M	L	M	L		M	L	L		M	S	M	L

CO 4	S	M	L		M	L		M	L	L		M	M	M	L
CO 5	S	M	L		M	L		M	L	L		M	M	M	L
CO 6	S	M	L	L	L			M	L	L		M	M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1		CAT2						Ass2	Terminal					
	1	2	3	4	5	6	3	4	1	2	3	4	5	6	3	1	2	3	4	5	6
CO1	5	10	20	-	-	-	35	-	-	-	-	-	-	-	-	1	4	12	-	-	-
CO2	5	10	20	-	-	-	30	-	-	-	-	-	-	-	-	1	4	12	-	-	-
CO3	5	10	10	-	-	-	10	25	-	-	-	-	-	-	-	1	4	-	12	-	-
CO4	-	-	-	-	-	-	-	-	5	-	25	-	-	-	10	1	4	12	-	-	-
CO5	-	-	-	-	-	-	-	-	5	5	20	-	-	-	30	1	4	12	-	-	-
CO6	-	-	-	-	-	-	-	-	5	10	25	-	-	-	60	1	4	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Motivation - Benefits of AI in cloud resource management - Challenges and opportunities of using AI for cloud resource management.

Resource Provisioning and Scaling: Demand prediction and forecasting techniques for cloud resources- AI-driven resource allocation and scaling algorithms- Dynamic provisioning of cloud resources.

Workload Placement and Optimization: Workload characterization and classification- AI-based workload placement algorithms- Optimization of workload performance, cost, and energy efficiency.

Predictive Resource Management: Predictive modelling – resource optimization -resource allocation.

Ethical Considerations and Social Impact of AI in Resource Management: Bias and fairness - Privacy and security concerns - Social impact of AI on resource access.

AI-driven Cloud Cost Optimization: Cost analysis and profiling - Optimization of cloud resource allocation for cost efficiency-AI based cloud cost tool

Learning Resources

1. Janusz Kacprzyk , Pedro Y. Pinero Perez , Rafael E. Bello Perez, “Artificial Intelligence in Project Management and making decisions”, Springer, 2022.
2. Dr. Wenhong Tian and Yong Dr. Zhao, “Optimized Cloud Resource Management and Scheduling: Theories and Practices”, 1st Edition ,Morgan, 2014.
3. Chebiyyam Siva Murthy , “Resource Management in Real-Time Systems & Networks”, MIT Press, 2001.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Motivation	2
1.2	Benefits of AI in resource management	2
1.3	Challenges and opportunities of using AI for resource management	2
2	Resource Provisioning and Scaling	
2.1	Demand prediction and forecasting	2
2.2	AI-driven resource allocation and scaling algorithms	2
2.3	Dynamic provisioning of cloud resources	2
3	Workload Placement and Optimization	
3.1	Workload characterization and classification	2
3.2	AI-based workload placement algorithms	2
3.3	Optimization of workload performance, cost, and energy efficiency	2
4	Predictive Resource Management	
4.1	Predictive modelling	2
4.2	Resource optimization	2
4.3	Resource allocation	3
5	Ethical Considerations and Social Impact	
5.1	Bias and fairness	2
5.2	Privacy and security concerns	1
5.3	Social impact of AI on resource access and distribution	2
6	AI-driven Cloud Cost Optimization	

Module No.	Topic	No. of Periods
6.1	Cost analysis and profiling	2
6.2	Optimization of cloud resource allocation for cost efficiency	2
6.3	Cloud Cost tool	2
	Total	36

Course Designer(s):

1. Ms. C. Santhiya, Assistant Professor, CSE

csit@tce.edu

**22CSPL1 GENERATIVE ADVERSARIAL
NETWORKS**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

Generative Adversarial Networks (GANs) are considered as one of the most modern fascinating technologies within the field of Deep Learning and Computer Vision. GANs are gaining a lot of attention because they can create fake content and are revolutionizing the field of Artificial Intelligence in producing high-quality results. In this course, we will focus on GAN components, and modern architectures of GANs, and ultimately explore applications from photorealistic imagery of faces, products, and 3D scenes to applications for data augmentation and training of other neural networks.

Prerequisite

Fundamentals of Machine Learning

Vanilla Recurrent Neural Networks

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Demonstrate Generative Adversarial Networks for generative modeling in images and videos	TPS3	B	80
CO2	Use GANs for Adversarial training, Image Enhancement, and Intrusion Detection	TPS3	B	80
CO3	Illustrate the progress in GANs by the inclusion of gaming effects using deep learning methods	TPS3	B	80
CO4	Employ different GANs like Semi-Supervised GANs, Conditional GANs for optimal solutions	TPS3	B	80
CO5	Develop Image to Image translations with Cycle GANs along with developing GANs with Object-oriented design.	TPS3	B	80
CO6	Compare the significant technological advancements in GAN and analyze their real-world applications	TPS4	B	80

Mapping with Programme Outcomes

CO s	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L	L	L	L		M	M	M		M	M	L	M
CO 2	S	M	L	L	L	L		M	M	M		M	M	L	M
CO 3	S	M	L	L	L	L		M	M	M		M	M	L	M
CO 4	S	M	L	L	L	L		M	M	M		M	M	L	M
CO 5	S	M	L	L	L	L		M	M	M		M	M	L	M
CO 6	S	S	M	L	M			M	M	L			S	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2		Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	5	5	20	-	-	-	30	-	-	-	-	-	-	-	-	1	4	12	-	-	-
CO2	5	5	25	-	-	-	35	-	-	-	-	-	-	-	-	1	4	12	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-	-	1	4	12	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	3 5	-	1	4	12	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	3 5	-	1	4	12	-	-	-
CO6	-	-	-	-	-	-	-	5	5	20	10	-	-	-	3 0	1	4	-	10	-	-

Syllabus

INTRODUCTION -Motivation – Generative Adversarial Networks (GANs)- GANs in action- GAN training -Reaching equilibrium- use of GANs-Introduction to generative modeling – autoencoders function on a high level- autoencoders to GAN-autoencoder components-Usage of autoencoders -Unsupervised learning

GENERATING HANDWRITTEN DIGITS - GAN frameworks: Adversarial training -The Generator and the Discriminator -GAN training algorithm -Generating handwritten digits- Deep Convolutional GAN (DCGAN) -Convolutional neural networks - Batch normalization- Generating handwritten digits with DCGAN

TRAINING AND COMMON CHALLENGES: GANing for success - Evaluation -Training challenges - Game setups -Training hacks - Progressing with GANs -Latent space interpolation – Key innovations –Tensor Flow Hub and hands-on

SEMI-SUPERVISED GAN - Implementing a Semi-Supervised GAN - Comparison to a fully supervised classifier - Conditional GAN –Motivation

CYCLEGAN - Image-to-image translation -Cycle-consistency loss -Adversarial loss - Identity loss -Architecture -Object-oriented design of GANs -Building CycleGAN

ADVERSARIAL EXAMPLES -Context of adversarial examples - Lies, damned lies, and distributions - Adversaries to GANs-Practical Applications -GANs in medicine-GANs in fashion

Text Books

1. Jakub Langr, Vladimir Bok, "GANs in action- Deep learning with Generative Adversarial Networks", Manning Publishers, 2019

Reference Books

1. Roshani Raut, Pranav Deepank Pathak, Sachin R Sakhare, Sonali Patil, "Generative Adversarial Networks and Deep Learning, Theory and Applications", CRC Press, 2023
2. Andrew Glassner, "Deep Learning: A Visual Approach", No Starch Press, 2021.
3. Josh Patterson, Adam Gibson, "Deep Learning: A Practitioner's Approach", O'Reilly Media, 2017.
4. <https://cs236g.stanford.edu/> Generative Adversarial Networks(GANs) Lecture Videos
5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014a. Generative adversarial nets. Adv. Neural Inf.Process. Syst. 27.

6. <https://www.coursera.org/learn/build-basic-generative-adversarial-networks-gans/>
Course era - Course on building basic Generative adversarial networks

NPTEL –NOC IITM Generative adversarial networks
<http://www.nitttrc.edu.in/nptel/courses/video/106105215/lec60.pdf>

7. Generative Models- University of Washington, Lecture Notes
<https://courses.cs.washington.edu/courses/cse599i/20au/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	INTRODUCTION – (6)	

1.2	Generative Adversarial Networks (GANs)	1
1.4	GANs in action - GAN training	1
1.5	Reaching equilibrium - Use of GANs	
1.6	Introduction to generative modeling	1
1.7	Autoencoders function on a high level	1
1.8	Autoencoders to GAN	2
2	GENERATING HANDWRITTEN DIGITS – (6)	
2.1	Foundations of GANs - Adversarial training	1
2.2	The Generator and the Discriminator	1
2.3	GAN training algorithm	1
2.4	Generating handwritten digits	1
2.5	Deep convolutional GAN - Batch normalization	1
2.6	Generating handwritten digits with DCGAN	1
3	TRAINING AND COMMON CHALLENGES - (6)	
3.1	GANing for success - Evaluation	1
3.2	Training challenges - Progressing with GANs	1
3.3	Latent space interpolation	1
3.4	Key innovations	1
3.5	TensorFlow Hub	1
3.6	Hands on	1
4	SEMI-SUPERVISED GAN - (6)	
4.1	Introducing the Semi-Supervised GAN	1
4.2	Implementation	2
4.3	Comparison to a fully supervised classifier	1
4.4	Conditional GAN	2
5	CYCLEGAN - (6)	
5.1	Image-to-image translation	1
5.2	Cycle-consistency loss	1
5.3	Adversarial loss - Identity loss	1
5.4	Object-oriented design of GANs	1
5.5	CycleGAN	2
6	ADVERSARIAL EXAMPLES - (6)	
6.1	Context of adversarial examples	1
6.2	Lies, damned lies, and distributions	1
6.3	Adversaries to GANs - Practical Applications of GANs	1
6.4	GANs in medicine, fashion	2
6.5	GAN innovations	1
	Total	36 Hours

Course Designer(s):

1. Dr. K.Sundarakantham, Professor, CSE

kskcse@tce.edu

2. Dr. S.Mercy Shalinie, Professor and Head, CSE

shalinie@tce.edu

22CSPM1**MOBILE AR DESIGN AND
DEVELOPMENT**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course objective is to underrated the fundamental of Augmented Reality in the modern application and provide the hands-on practice on immersive and interactive AR experiences for mobile devices which involves a combination of creative design, programming, unique capabilities and business domain in AR Application.

Prerequisite

22CS261 – Engineering Graphics and Extended Reality

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the fundamental of Mobile AR concept such as Marker and Marker less in chosen application.	TPS3	70	80
CO2	Adapt the AR functionalities such as Pose Tracking, Detection and Ray casting in Mobile Application.	TPS3	70	80
CO3	Apply the AR Interface such as Placing an Object-Prefabs and adapt the C# scripting language for using AR kit.	TPS3	70	80
CO4	Build an Application that incorporates the UI Controller and UX assets.	TPS3	70	80
CO5	Adapt the AR user Framework to Place Object and Hiding tracked objects for chosen application.	TPS3	70	80
CO6	Apply the AR Concept and Algorithms for an Art Gallery Project or similar	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO7.	S	M	L	L	L	L		M	M	M		M	M	L	M
CO8.	S	M	L		L	L		M	M	M		M	M	L	M
CO9.	S	M	L		L	L		M	M	M		M	M	L	M
CO10.	S	M	L		L	L		M	M	M		M	M	L	M
CO11.	S	M	L		L	L		M	M	M		M	M	L	M
CO12.	S	M	L	L	M	M	L	M	M	M	L	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	1	2	3	4	5	6
CO1	5	5	15	-	-	-	30	-	-	-	-	-	-	-	2	3	-	-	-	-

CO2	5	5	30	-	-	-	35	-	-	-	-	-	-	2	3	15	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	2	3	15	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	15	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	15	-	-
CO6	-	-	-	-	-	-	-	5	5	30	-	-	-	60	2	3	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Mobile AR-Historical Development-Types of Mobile AR- Marker Based AR- Marker Less AR- Virtual Reality- Mixed Reality.AR Foundation – Detection of surfaces-identifying feature points-track virtual objects-face and object tracking-AR Algorithms.

AR concepts: Pose Tracking-Environmental Detection-Ray casting and physics for AR-Light Estimation-Occlusion.

Interface-Simple AR Scene-Placing an Object-Prefabs-C# Scripting: Variables and Data Types-Operators-Conditionals-Arrays-Functions-Classes and Objects- AR Core and AR Kit-set up.

Reusable AR Framework: AR User Framework-UI canvas and panels-UI controller- Interaction Controller mode- interaction modes behaviour-UX assets.

AR User Framework: AR Framework Scene Template-Adding Place Object-mode with instructional UI-Building and Run-Hiding tracked objects.

Building an AR App: Art Gallery project UX-Collecting Image Data-Framed photo prefab-Hanging a virtual photo-Editing Virtual Objects- Edit mode - Edit menu - Object from Edit mode - Avoiding intersecting objects - Deleting a picture – Replacing image-Replacing the frame-Interacting to edit a picture - finger to move the picture - Pinching to resize the picture-Tracking Images.

Text Books

1. Linowes, Jonathan. Augmented Reality with Unity AR Foundation: A Practical Guide to Cross-platform AR Development with Unity 2020 and Later Versions. N.p., Packt Publishing, 2021.
2. Glover, Jesse, and Linowes, Jonathan. Complete Virtual Reality and Augmented Reality Development with Unity: Leverage the Power of Unity and Become a Pro at Creating Mixed Reality Applications. United Kingdom, Packt Publishing, 2019.

Reference Books & web resources

1. Peddie, Jon. Augmented Reality: Where We Will All Live. Germany, Springer International Publishing, 2017.
2. Grubert, Jens, and Grasset, Raphael. Augmented Reality for Android Application Development. United Kingdom, Packt Publishing, 2013.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Mobile AR	1
1.2	Historical Development	
1.3	Types of Mobile AR- Marker Based AR- Marker Less AR	1
1.4	Virtual Reality- Mixed Reality	1
1.5	AR Foundation	1
1.6	Detection of surfaces	1
1.7	identifying feature points-track virtual objects	1

Module No.	Topic	No. of Periods
1.8	AR Algorithms	1
2	AR concepts	
2.1	Pose Tracking	1
2.2	Environmental Detection	1
2.3	Ray casting and physics for AR	1
2.4	Light Estimation-Occlusion.	1
3	Interface	
3.1	Simple AR Scene	2
3.2	Placing an Object	1
3.3	Prefabs	1
3.4	C# Scripting: Variables and Data Types-Operators-Conditionals	1
3.5	Arrays-Functions-	1
3.6	Classes and Objects	1
3.7	AR Core and AR Kit-set up	1
4	Reusable AR Framework:	
4.1	AR User Framework	1
4.2	UI canvas and panels-UI controller	2
4.3	Interaction Controller mode	2
4.4	interaction modes behaviour- UX assets.	1
4.5	AR User Framework: AR Framework Scene Template	1
4.6	Adding Place Object	1
4.7	Mode with instructional UI	1
4.8	Building and Run-Hiding tracked objects.	1
5	Building an AR App:	
5.1	Art Gallery project UX-Collecting Image Data	2
5.2	Framed photo prefab- Hanging a virtual photo-Editing Virtual Objects	1
5.3	Edit mode - Edit menu - Object from Edit mode	1
5.4	Avoiding intersecting objects - Deleting a picture	1
5.5	Replacing image - Replacing the frame	

Module No.	Topic	No. of Periods
5.6	Interacting to edit a picture	1
5.7	Finger to move the picture - Pinching to resize the picture-	
5.8	Tracking Images	1
	Total	36

Course Designer(s):

1. Dr.N.Shivakumar, Assistant Professor, Computer Science and Engineering, shiva@tce.edu
2. Dr.K.Indira, Assistant Professor, Computer Science and Engineering, kiit@tce.edu

22CSPN1 HIGH PERFORMANCE COMPUTING

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course teaches students with strong foundation in parallel computing and high performance computing systems. Students will explore shared and distributed-memory parallel programming model along with parallel programming using OpenMP and MPI. It also describes the principles and structures of high performance computing. This course also gives the practical hands on experience to students in GPU programming using CUDA.

Prerequisite

- Basics of Computer Architecture and Organization

Course Outcomes

On the successful completion of the course, students will be able to

Cos	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Show the features of High Performance Computing Systems from traditional computing.	TPS3	B	75
CO2	Construct parallel programs using shared memory OpenMP.	TPS3	B	75
CO3	Construct parallel programs using distributed memory MPI.	TPS3	B	75
CO4	Identify the HPC components suitable for the given specification.	TPS3	B	75
CO5	Solve the complex data applications by choosing the suitable HPC model.	TPS3	B	75
CO6	Perform basic GPU programming using CUDA.	TPS3	B	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L					L	M			M	M		L
CO 2	S	M	L	L	M			L	M	L		M	M	L	L
CO 3	S	M	L	L	M			L	M	L		M	M	L	L
CO 4	S	M	L		M			L	M			M	M	L	L
CO 5	S	M	L	M	M	L		L	M	M		M	M	L	L
CO 6	S	M	L		M			L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10	10	-	-	-	-	-	-	-	-	-	5	5	10	-	-	-	-	-	-	-	-	-	2	3	10	-	-	-
CO2	10	10	15	-	-	-	-	-	-	-	-	-	5	10	25	-	-	-	-	-	-	-	-	-	2	3	10	-	-	-
CO3	10	10	15	-	-	-	-	-	-	-	-	-	5	10	25	-	-	-	-	-	-	-	-	-	2	3	10	-	-	-
CO4	-	-	-	-	-	-	10	10	10	-	-	-	-	-	-	-	-	-	5	5	20	-	-	-	2	3	10	-	-	-
CO5	-	-	-	-	-	-	10	10	10	-	-	-	-	-	-	-	-	-	5	10	10	-	-	-	2	3	10	-	-	-
CO6	-	-	-	-	-	-	10	10	20	-	-	-	-	-	-	-	-	-	5	15	25	-	-	-	2	3	10	-	-	-

Syllabus

Basics of High Performance Computing (HPC): Key Properties of HPC Architecture - Speed, Parallelism, Efficiency, Power, Reliability and Programmability. Structure of HPC systems, Applications of HPC.

Parallel systems: Flynn's taxonomy, Von Neumann Sequential Processors, Vector and Pipelining - Pipeline Parallelism and Vector Processing, Multiprocessors - Shared-Memory Multiprocessors - Massively Parallel Processors and Commodity Clusters.

Parallel computing: shared memory computers, distributed memory computers and hybrid systems, parallel programming using OpenMP and MPI.

HPC Architecture components: HPC Clusters, Computing Capabilities – multi-core CPU, GPU, FPGA and ASICs, Storage - File Storage, Object Storage, Block Storage and Memory-Centric Storage. Network Interconnects and Protocols.

HPC Models and Frameworks: Basics of complex data applications – volume, velocity, variety and veracity, Examples - Batch Processing-Apache Spark, Stream Processing, Query Processing-MySQL, Graph Processing and AI processing applications-PyTorch.

Accelerator Architecture: Basics of Graphics Processing Units (GPU), GPU Architecture, Heterogeneous systems, CUDA programming – basics of parallel programming and threads in GPUs.

Text Books

1. High-Performance Big Data Computing, Dhabaleswar K. Panda, Xiaoyi Lu, and Dipti Shankar, The MIT Press, Massachusetts Institute of Technology, Cambridge, 2022.
2. Introduction to High Performance Computing for Scientists and Engineers, Georg Hager Gerhard Wellein, Chapman & Hall/CRC Computational Science Series, CRC Press, 2011.
3. High Performance Computing Modern Systems and Practices, Thomas Sterling, Matthew Anderson, Maciej Brodowicz, Morgan Kaufmann Publisher, 2018.

4. Programming in Parallel with CUDA - A Practical Guide, Richard Ansorge, Cambridge University Press, 2022.

Reference Books & web resources

1. Introduction to Parallel Computing, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Second Edition, Addison-Wesley, Pearson Education Limited 2003.
2. CUDA by Example: An Introduction to General-Purpose GPU Programming, Jason Sanders, Edward Kandrot, Addison-Wesley Professional, 1st Edition, 2010.
3. The OpenMP Common Core - Making OpenMP Simple Again, Timothy G. Mattson, Yun (Helen) He, and Alice E. Koniges, The MIT Press, 2019.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Basics of High Performance Computing (HPC)	
1.1	Key Properties of HPC Architecture - Speed, Parallelism, Efficiency, Power, Reliability and Programmability.	1
1.2	Structure of HPC systems	2
1.3	Applications of HPC	2
2	Parallel systems	
2.1	Flynn's taxonomy	1
2.2	Von Neumann Sequential Processors	1
2.3	Vector and Pipelining - Pipeline Parallelism and Vector Processing	2
2.4	Multiprocessors - Shared-Memory Multiprocessors - Massively Parallel Processors and Commodity Clusters	2
3	Parallel computing	
3.1	shared memory computers, distributed memory computers and hybrid systems	1
3.2	Parallel programming using OpenMP	3
3.3	Parallel programming using MPI	3
4	HPC Architecture Components	
4.1	HPC Clusters	1
4.2	Computing Capabilities – multi-core CPU, GPU, FPGA and ASICs	2
4.3	Storage - File Storage, Object Storage, Block Storage and Memory-Centric Storage	2

Module No.	Topic	No. of Periods
4.4	Network Interconnects and Protocols	2
5	HPC Models and Frameworks	
5.1	Basics of complex data applications – volume, velocity, variety and veracity	1
5.2	Examples - Batch Processing, Stream Processing, Query Processing, Graph Processing and AI processing applications	2
5.3	Apache Spark, MySQL and PyTorch	2
6	Accelerator Architecture	
6.1	Basics of Graphics Processing Units (GPU) and GPU Architecture	2
6.2	Heterogeneous systems	1
6.3	CUDA programming – basics of parallel programming and threads in GPUs	3
	Total	36

Course Designer(s):

1. Dr.G.Madhupriya, Associate Professor,CSE gmadhupriya@tce.edu

**22CSPP1 APPLIED LINEAR ALGEBRA IN
AI AND ML**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

Linear algebra, optimization techniques and statistical methods together form essential tools for most of the algorithms in Artificial Intelligence and Machine Learning. The main objective of this course is to provide students a solid foundation in mathematics and prepare students to pursue advanced study or research in the field of AI and ML. The outcome of this course is to train students with the important concepts and computational techniques in linear algebra which are useful for AI and ML applications.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Implement matrix operations to compute QR factorization and pseudo inverse	TPS3	B	70
CO2	Develop System of linear equations and functions using Taylor approximation and Regression model.	TPS3	B	70
CO3	Demonstrate multi-class least squares classifiers, Polynomial classifiers using Linear least squares	TPS3	B	70
CO4	Build Multi-objective, Constrained and Multi-collinearity least squares for AI and ML problems	TPS3	B	70
CO5	Illustrate Nonlinear least squares using Penalty and approximation algorithms	TPS3	B	70
CO6	Use LRA and SLRA to model order selection in time series analysis and collaborative filtering.	TPS3	B	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		M	M	M		M	M	L	M
CO2	S	M	L		M	L		M	M	M		M	M	L	M
CO3	S	M	L		M	L		M	M	M		M	M	L	M
CO4	S	M	L		M	L		M	M	M		M	M	L	M
CO5	S	M	L		M	L		M	M	M		M	M	L	M
CO6	S	M	L		M	L		M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment I						Assignment II						Terminal						
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	
CO1	4	10	20												50												5	15			
CO2	4	10	20												30												5	15			
CO3	2	10	20				2	10	15						20						20						5	15			
CO4							4	10	15												30						5	15			
CO5							2	5	10												25							10			
CO6							2	5	10												25							10			
	10	30	60				10	30	60						100						100						20	80			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Matrices: Zero and Identity matrices, Transpose, Addition, Norm, Matrix-Vector multiplication, Geometric transformations, Selectors, Incidence matrix, Convolution, Matrix-Matrix multiplication, QR factorization, left and right inverses, pseudo inverse, Lagrangian Matrices and Graph.

Linear functions and Equations: Taylor approximation, Regression model, Linear and affine functions, System of linear equations, Composition of linear functions, Understanding Linear dynamical systems.

Linear least squares: Geometrical interpretation, data fitting, feature engineering, fitting with continuous and discontinuous piecewise linear functions, least squares to classification, multi-class least squares classifiers, Polynomial classifiers. **Case Study:** Application to MNIST data set.

Multi-objective least squares: Estimation and regularized inversion and data fitting, Multi-objective least squares application to image de-blurring, constrained least squares application to portfolio optimization, multi-collinearity least squares applications to Principal Component Analysis (PCA) and application to Google page ranking algorithm.

Nonlinear Model and least squares: Nonlinear model fitting, Nonlinear least squares classification, Constrained nonlinear least squares: Penalty algorithm, Augmented Lagrangian algorithm, Low rank approximation (LRA) and structured low rank approximation problem (SLRA) and solution. **Case study:** LRA and SLRA application to model order selection in time series, Tensors decomposition based sparse learning in deep networks and application to collaborative filtering.

Text Books

1. Gilbert Strang, "Linear Algebra and Learning from Data", Wellesley-Cambridge Press, First Edition, 2019
2. Stephen Boyd and Lieven Vandenberghe, "Introduction to Applied Linear Algebra- Vectors, Matrices, and Least Squares" Cambridge University Press, 1st edition, 2018

Reference Books & web resources

1. Crista Arangala, "Linear Algebra with Machine Learning and Data", Chapman and Hall/CRC; First Edition, 2023.
2. Charu C. Aggarwal, "Linear Algebra and Optimization for Machine Learning", Springer. First Edition, 2020

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lecture Hours
------------	-------	----------------------

1	Matrices(8)	
1.1	Zero and identity matrices, Transpose, addition, norm, Matrix-vector multiplication	2
1.2	Geometric transformations, Selectors, Incidence matrix	2
1.3	Convolution, Matrix-matrix multiplication, QR factorization	2
1.4	Left and right inverses, pseudo inverse, Lagrangian Matrices and Graph	2
2	Linear functions and Equations(6)	
2.1	Taylor approximation, Regression model, Linear and affine functions	2
2.2	System of linear equations, Composition of linear functions	2
2.3	Understanding Linear dynamical systems	2
3	Linear least squares(7)	
3.1	Geometrical interpretation, data fitting, feature engineering	2
3.2	Fitting with continuous and discontinuous piecewise linear functions, least squares to classification	2
3.3	Multi-class least squares classifiers, Polynomial classifiers.	2
3.4	Case Study: Application to MNIST data set.	1
4	Multi-objective least squares (7)	
4.1	Estimation and regularized inversion and data fitting	2
4.2	Multi-objective least squares application to image de-blurring	1
4.3	Constrained least squares application to portfolio optimization	2
4.4	Multi-collinearity least squares applications to Principal Component Analysis (PCA) and application to Google page ranking algorithm.	2
5	Nonlinear Model and least squares (8)	
5.1	Nonlinear model fitting, Nonlinear least squares classification	2
5.2	Constrained nonlinear least squares: Penalty algorithm, Augmented Lagrangian algorithm	2
5.3	Low rank approximation (LRA) and structured low rank approximation problem (SLRA) and solution.	2
5.4	Case study: LRA and SLRA application to model order selection in time series, Tensors decomposition based sparse learning in deep networks and application to collaborative filtering.	2
	Total Hours	36

Course Designer(s):

1. Dr C S Senthilkumar, AP, Maths
2. Dr. M.Nirmala Devi, AP,CSE
3. Dr.B.Subbulakshmi, AP,CSE

kumarstays@tce.edu
mnit@tce.edu
bscse@tce.edu

22CSPQ1**IOT SECURITY**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course will introduce students to the need for IoT security. It provides an overview of the architecture of IoT and emphasizes the value of security protection and privacy measures to the IoT components and applications.

Prerequisite

Knowledge on Wireless networks, IoT and Security.

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Establish the needs and requirements for security in an IoT system. (Understand)	TPS2	70	80
CO2	Interpret the security architectures for IoT. (Apply)	TPS3	70	75
CO3	Deduce the significance of cryptographic algorithms for IoT. (Apply)	TPS3	70	75
CO4	Design a layered mechanism for IoT security. (Apply)	TPS3	70	75
CO5	Develop an mechanism to ensure privacy of IoT applications. (Apply)	TPS3	70	75
CO6	Design a mechanism to guarantee security measures for IoT. (Apply)	TPS3	70	75

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	L										L		
CO2	S	M	L	-	-	L	L	L	M	L		M	M	L	L
CO3	S	M	L	-	M	L	L	L	M	L		M	M	L	L
CO4	S	M	L	-	-	L	L	L	M	L		M	M	L	L
CO5	S	M	L	-	M	L	L	L	M	L		M	M	L	L
CO6	S	M	L	-	M	L	L	L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10																							4	10				
CO2	10	10	10												50										4	4	10			
CO3	10	20	20												40										4	4	10			
CO4							10		10												30				4	4	10			
CO5							10	10	20												30				4	4	10			
CO6							10	20	10												40				4		10			

Syllabus

Introduction: Physical Design of IoT, Logical Design of IoT - Components of an IoT System - Architectures of the Internet of Things - IoT Industries and Their Characteristics - Enabling Technologies - Trends in IoT - M2M, IIoT, Digital Twin, Internet of Robotic Things (IoRT)

Security Architectures for IoT: Layered IoT Security Architecture - Security Threats and Methods of Protection for IoT Devices - Security Threats and Methods of Security Protection for Up-Stream Data - Security Threats and Methods of Security Protection for Down-Stream Data - IoT Perception Layer Security - IoT Network Layer Security Mechanisms - Typical Security Techniques for IoT Network Layer - IoT Processing Layer Security Mechanisms - IoT Application Layer Security Mechanisms

Cryptographic Fundamentals for IoT: Cryptographic Algorithms and their Security Services – Deterministic and Probabilistic Encryptions - Cryptographic Hash Functions and Message Authentication Codes - Trust Mechanism and Key Management in IoT - Types of Identities in IoT Systems - Identity Authentication Protocols - Key Management - Lightweight PKI for IoT Applications Identity & Access Management Solutions For IoT

IoT Security - IoT Perception Layer Security - Security Threats in IoT Perception Layer - IoT Network Layer Security - Security Threats in IoT Network Layer - Security Techniques in Mobile Communication Networks - Security Techniques in LPWAN - IoT Processing Layer Security - Security Threats in IoT Processing Layer - Database for IoT Processing Layer - Access Control Policies Applicable to IoT Processing Layer - Security Mechanisms in Cloud Computing

Privacy Protection in IoT Applications - Privacy challenges introduced by the IoT - Privacy Protection by Identity Anonymity - Privacy Protection Based on Data Linkage - Location Privacy Protection - Graded Privacy Protection - RFID System Security - Security and Privacy Issues in RFID Systems - IT Security Versus OT Security - Characteristics of Network Attacks Targeting at OT Security - Characteristics of OT Security - The Limitations of IoT Attackers - Intrusion Tolerance Mechanisms in IoT Systems - Architectural Construction of Intrusion Tolerance Scheme for IoT

Security Measures for IoT - IoT Security Versus Traditional IT System Security - The Secure Development Life Cycle (SDLC) - The challenge of secure IoT development - Secure design goals - Setting Up an IoT Compliance Monitoring - IoT Incident Response and Forensic Analysis - Defining, planning, and executing an IoT incident response - IoT forensics

Text Books

1. Chuan-Kun Wu, Internet of Things Security Architectures and Security Measures, Springer, 2021.

2. Brian Russell, Drew Van Duren, Practical Internet of Things Security, Second Edition, Packt Publishing, 2018.

Reference Books & web resources

1. Anupama C. Raman and Pethuru Raj, The Internet of Things: Enabling technologies, platforms, and Use Cases, CRC Press, 2017.
2. Fei Hu, Security and Privacy in Internet of Things (IoTs) - Models, Algorithms, and Implementations, CRC Press, 2016.

Course Contents and Lecture Schedule

Module No	Topic	No of Lectures	Course Outcome
1	Introduction		
1.1	Physical Design of IoT, Logical Design of IoT	1	CO1
1.2	Components of an IoT System	1	CO1
1.3	Architectures of the Internet of Things	1	CO1
1.4	IoT Industries and Their Characteristics	1	CO1
1.5	Enabling Technologies - Trends in IoT	1	CO1
2	Security Architectures for IoT		
2.1	Layered IoT Security Architecture	1	CO2
2.2	Threats and Methods of Protection for IoT Devices	1	CO2
2.3	Security Threats and Methods of Security Protection for Up-Stream, Down-Stream Data	1	CO2
2.4	IoT Perception Layer & Network Layer Security Mechanisms	1	CO2
2.5	IoT Processing Layer & Application Layer Security Mechanisms	1	CO2
3	Cryptographic Fundamentals for IoT		
3.1	Cryptographic Algorithms and their Security Services	1	CO3
3.2	Deterministic and Probabilistic Encryptions	1	CO3
3.3	Cryptographic Hash Functions and Message Authentication Codes	1	CO3
3.4	Trust Mechanism and Key Management in IoT	1	CO3
3.5	Types of Identities in IoT Systems - Identity Authentication Protocols	1	CO3
3.6	Key Management, Identity & Access Management Solutions For IoT	1	CO3
4	IoT Security		
4.1	IoT Perception Layer Security, Security Threats in IoT Perception Layer	2	CO4
4.2	IoT Network Layer Security, Security Threats in IoT Network Layer	2	CO4
4.3	Security Techniques in Mobile Communication Networks, LPWAN	1	CO4
4.4	IoT Processing Layer Security - Security Threats in IoT Processing Layer	1	CO4
4.5	Access Control Policies Applicable to IoT Processing Layer	1	CO4
4.6	Security Mechanisms in Cloud Computing	1	CO4
5	Privacy Protection in IoT Applications		
5.1	Privacy challenges introduced by the IoT	1	CO5
5.2	Privacy Protection by Identity Anonymity, Data Linkage, Location, Graded Privacy Protection	1	CO5
5.3	RFID System Security - Security and Privacy Issues in RFID Systems	1	CO5
5.4	IT Security Versus OT Security	1	CO5
5.5	The Limitations of IoT Attackers	1	CO5

5.6	Intrusion Tolerance Mechanisms in IoT Systems - Architectural Construction of Intrusion Tolerance Scheme for IoT	1	CO5
6	Security Measures for IoT		
6.1	IoT Security Versus Traditional IT System Security	1	CO6
6.2	The Secure Development Life Cycle (SDLC)	1	CO6
6.3	The challenge of secure IoT development, design goals	1	CO6
6.4	Setting Up an IoT Compliance Monitoring	1	CO6
6.5	Defining, planning, and executing an IoT incident response	1	CO6
6.6	IoT forensics	1	CO6
	Total	36	

Course Designer(s):

1. Dr.G.S.R.Emil Selvan, Associate Professor, CSE email@tce.edu

22CSPR1 REAL TIME OPERATING SYSTEMS

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This course provides the basic core technologies used in developing real time systems. The evolution of System on Chip and its ported real time operating systems are discussed with practical hands on with free RTOS.

Prerequisite

- Operating Systems

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify Core, Platform, Real Time Operating Systems supported in System on Chip	TPS3	B	80
CO2	Experiment with Time Management basic Service in Real Time Systems	TPS3	B	80
CO3	Implement Task Scheduling Algorithm in Real Time Systems based on problem	TPS3	B	80
CO4	Choose Inter Task Communication and appropriate Synchronization Mechanism	TPS3	B	80
CO5	Apply appropriate Deadlock algorithm for the given real time scenario	TPS3	B	80
CO6	Examine Real time System basic functions in the embedded support Emulator, QEMU	TPS4	B	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L	L	L	L		M	M	M		M	M	L	M
CO2.	S	M	L		L	L		M	M	M		M	M	L	M
CO3.	S	M	L		L	L		M	M	M		M	M	L	M
CO4.	S	M	L		L	L		M	M	M		M	M	L	M
CO5.	S	M	L		L	L		M	M	M		M	M	L	M
CO6.	S	S	M	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal						
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	5	5	20	-	-	-	30	-	-	-	-	-	-	-		1	4	10	-	-	-
CO2	5	5	25	-	-	-	35	-	-	-	-	-	-	-		1	4	10	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-		1	4	10	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20		1	4	10	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20		1	4	10	-	-	-
CO6	-	-	-	-	-	-	-	5	5	20	10	-	-	40	20	1	4	10	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

System on a Chip: Embedded Systems-Basics of Chips and SoC ICs – Intellectual Property Cores - System on Programmable Chips (SoPC) – Network on Chip – Bus based Interconnection – CPU/IP Cores - Arm – DSP Case Study: Board Specific Packages

RTOS: Real-Time Systems and OS –Types - Requirement – Basic Functions – Micro-Kernel Architecture- Kernel mode decomposing and Programming- Performance Metrics-Time Management- Clock Synchronization (SCL)

Task Management: RTOS Task Model- Task Control Block –Scheduling Problems – Aperiodic Task Scheduling: Jackson Algorithm, Horn's Algorithm, Non –Pre-emptive Scheduling –Periodic Task Scheduling: Timeline Scheduling, Rate Monotonic Scheduling, Earliest Deadline First – Interrupt handling – Interrupt Service Routines

Concurrency and Synchronization: Message Queue – Mail boxes – Resource Sharing –Inter Task Access Synchronization: Semaphore, Mutex- Inter Task Event Synchronization-Event Flags-Dead Lock - Priority Inheritance Protocol-Highest Locker Protocol – Priority Ceiling Protocol- Memory management

Real Time Kernel: QEMU Emulator – Free RTOS – Task Management – Queue Management – Software Timer management – Case Study: Azure RTOS, GreenHills RTOS

Text Books

1. C.Krishna and K.Shin, Real Time Systems, McGraw Hill, 2000
2. Philip A. Laplante, Seppo J. Ovaska, Real-Time Systems Design and Analysis, Wiley, 2012.
3. Giorgio C. Buttazzo. Hard Real Time Computing Systems: Predictable Scheduling Algorithms and Applications, 3rd Edition. Springer 2011

Reference Books & web resources

1. Jane W.s.Liu, Real Time Systems, Pearson, 2000
2. Michael J. Flynn, Wayne Luk, Computer System Design: System on Chip, John Wiley and Sons Inc. 2011
3. <https://archive.nptel.ac.in/courses/106/105/106105172/>, Prof.Rajib Mall, IIT Kharagpur

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	System on a Chip	
1.1	Embedded Systems-Basics of Chips and SoC ICs	1
1.2	Intellectual Property Cores- System on Programmable Chips (SoPC)	1
1.3	Network on Chip – Bus based Interconnection	1
1.4	CPU/IP Cores - Arm – DSP	1
1.5	Case Study : Board Specific Packages	2
2	Real Time Operating Systems	
2.1	Real-Time Systems and OS	1
2.2	Types - Requirement	1
2.3	Basic Functions – Micro-Kernel Architecture- Kernel mode decomposing and Programming	2
2.4	Performance Metrics -Time Management - Clock Synchronization (SCL)	2
3	Task Management	
3.1	RTOS Task Model- Task Control Block	1
3.2	Scheduling Problems – Aperiodic Task Scheduling: Jackson Algorithm, Horn's Algorithm, Non –Pre-emptive Scheduling	3
3.3	Periodic Task Scheduling: Timeline Scheduling, Rate Monotonic Scheduling, Earliest Deadline First	3
3.4	Interrupt handling – Interrupt Service Routines	2
4	Concurrency and Synchronization	
4.1	Message Queue – Mail boxes – Resource Sharing	1
4.2	Inter Task Access Synchronization: Semaphore, Mutex	2
4.3	Inter Task Event Synchronization-Event Flags	2
4.4	Dead Lock - Priority Inheritance Protocol-Highest Locker Protocol – Priority Ceiling Protocol	3
4.5	Memory management	

Module No.	Topic	No. of Periods
5	Real Time Kernel	
5.1	QEMU Emulator	1
5.2	Free RTOS	1
5.3	Task Management	1
5.4	Queue Management	1
5.5	Software Timer management1	1
5.6	Case Study: Azure RTOS, GreenHills RTOS	2
	Total	36

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, Computer Science and Engineering, rlsit@tce.edu

22CSPS1**AGENT BASED SYSTEMS**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This Agent-Based Systems course is a primer for aspiring engineers, offering hands-on training in distributed computing and intelligent automation. Designed to equip participants with essential skills, the program anticipates and aligns with the evolving landscape of agent-based systems, especially in industrial applications.

Prerequisite

- Distributed Systems
- Artificial Intelligence
- Deep Learning
- Probability and Statistics

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the core principles and characteristics of multi-agent systems, including agent architectures, communication, and coordination mechanisms.	TPS 3	70	80
CO2	Utilize programming languages to design and implement agent architectures, communication protocols, and coordination mechanisms.	TPS 3	70	80
CO3	Employ formal methods to verify and validate agent-based systems, including the use of model checking and specification languages.	TPS 3	70	80
CO4	Apply linear programming techniques to optimize resource allocation and decision-making processes within agent-based systems	TPS 3	70	80
CO5	Analyze and evaluate the emergence of self-organizing behaviors within multi-agent systems and their impact on overall system performance.	TPS 4	70	80
CO6	Analyze real-world case studies, such as smart traffic management, supply chain optimization, and collaborative robotics in manufacturing, to evaluate the effectiveness and adaptability of multi-agent systems.	TPS 4	70	80

Mapping with Programme Outcomes

COs	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L	L	L	L		M	M	M		M	M	L	M
CO2.	S	M	L		L	L		M	M	M		M	M	L	M
CO3.	S	M	L		L	L		M	M	M		M	M	L	M
CO4.	S	M	L		L	L		M	M	M	L	M	M	L	M
CO5.	S	M	L		L	L		M	M	M	L	M	M	L	M
CO6.	S	M	L	L	M	M	L	M	M	M	M	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal						
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	10		10	-	-	-	20	-	-	-	-	-	-	-		2	3	15	-	-	-
CO2	10	10	10	-	-	-	40	-	-	-	-	-	-	-		2	3	15	-	-	-
CO3	10	20	20	-	-	-	40	-	-	-	-	-	-	-		2	3	10	-	-	-
CO4	-	-	-	-	-	-	-	10	-	10	-	-	-	20		2	3	10	-	-	-
CO5	-	-	-	-	-	-	-	10	10	-	15	-	-	20	20	2	3	-	10	-	-
CO6	-	-	-	-	-	-	-	10	20	-	15	-	-	20	20	2	3	-	10	-	-

Syllabus

Foundations of Multi-Agent Systems - Introduction to Artificial Intelligence Programming - Fundamentals of Artificial Intelligence, Machine Learning and Deep Learning - Basics of

Distributed Systems - Overview of Multi-Agent Systems - **Agent Architectures and**

Communication - Agent Architectures - Agent Communication - Agent Coordination Mechanisms -

Learning and Decision-Making in Multi-Agent Systems - Agent Learning - Formal Methods for Agent-Based Systems - Game Theory in Multi-Agent Decision-Making - Linear Programming in Agent Decision-Making - **Advanced Topics in Multi-Agent Systems**

- Swarm Intelligence in Multi-Agent Systems - Behaviour Trees in Agent Architectures - Ontologies in Multi-Agent Systems - Temporal Logic and Model Checking - **Applications and Specialized Areas** - Self-Organization in Multi-Agent Systems - Security and Trust in Multi-Agent Systems - Human-Agent Interaction - Bio-Inspired Computing in Multi-Agent Systems

- Multi-Robot Systems and Physical Deployments - Emergent Properties in Multi-Agent Systems - **Case Studies** - Smart Traffic Management - Supply Chain Optimization - Collaborative Robotics in Manufacturing – Drones

Text Books

1. Multiagent Systems – A Modern Approach to Distributed Artificial Intelligence (Intelligent Robotics & Autonomous Agents Series), Gerhard Weiss, 2000.
2. Swarm Intelligence From Natural to Artificial Systems, Eric Bonabeau, Marco Dorigo, and Guy Theraulaz, 1999 by Oxford University Press.

3. Behavior Trees in Robotics and AI: An Introduction, Michele Colledanchise (Author), Petter Ögren (Author), Chapman & Hall/CRC Artificial Intelligence and Robotics Series) 1st Edition, 2017
4. Reinforcement Learning: An Introduction second edition, Richard S. Sutton and Andrew G. Barto The MIT Press Cambridge, Massachusetts London, England, 2020.

Reference Books & web resources

1. Ontology-Based Multi-Agent Systems - <https://link.springer.com/book/10.1007/978-3-642-01904-3>
2. Temporal Logic and State Systems (Texts in Theoretical Computer Science. An EATCS Series) - <https://www.amazon.com/Temporal-Systems-Theoretical-Computer-Science/dp/3540674012>
3. <https://library.iitmandi.ac.in/os/onlineresources/E-Books/springer-CSE.html>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Foundations of Multi-Agent Systems	(6)
1.1	Introduction to Artificial Intelligence Programming	1
1.2	Fundamentals of Artificial Intelligence, Machine Learning and Deep Learning	2
1.3	Basics of Distributed Systems	1
1.4	Overview of Multi-Agent Systems	2
2	Agent Architectures and Communication	(6)
2.1	Agent Architectures	2
2.2	Agent Communication	2
2.3	Agent Coordination Mechanisms	2
3	Learning and Decision-Making in Multi-Agent Systems	(6)
3.1	Agent Learning	1
3.2	Formal Methods for Agent-Based Systems	1
3.3	Game Theory in Multi-Agent Decision-Making	2
3.4	Linear Programming in Agent Decision-Making	2
4	Advanced Topics in Multi-Agent Systems	(6)
4.1	Swarm Intelligence in Multi-Agent Systems	1
4.2	Behaviour Trees in Agent Architectures	1
4.3	Ontologies in Multi-Agent Systems	2
4.4	Temporal Logic and Model Checking	2
5	Applications and Specialized Areas	(9)
5.1	Self-Organization in Multi-Agent Systems	1
5.2	Security and Trust in Multi-Agent Systems	2
5.3	Human-Agent Interaction	1
5.4	Bio-Inspired Computing in Multi-Agent Systems	2
5.5	Multi-Robot Systems and Physical Deployments	2
5.6	Emergent Properties in Multi-Agent Systems	1
6	Case Studies	(3)
6.1	Smart Traffic Management	1
6.2	Supply Chain Optimization	1
6.3	Collaborative Robotics in Manufacturing - Drones	1
	Total	36

Course Designer(s):

1. Dr. M.Sivakumar, AP/CSE
2. Mr.S.Santhana Hari, AP/CSE

mskcse@tce.edu
sshcse@tce.edu

22CSPT1

**BLOCKCHAIN SECURITY AND
PRIVACY**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

The security and privacy of blockchain can be viewed in two ways. In one, blockchain is used as a security solution to several applications such as finance, supply chain, and healthcare to name a few. The other fantabulous way of looking at it is the security and privacy techniques that is there in the blockchain and that can be tailored in it in the future. This course familiarizes the students about blockchain security and privacy in the second way.

Prerequisite

A basic course on Blockchain Technology and its Applications

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the fundamental concepts of blockchain technology	TPS3	70	80
CO2	Illustrate the security risks existing in blockchain platform	TPS3	70	80
CO3	Illustrate the vulnerabilities existing in smart contracts and methods to address the security issues.	TPS3	70	80
CO4	Illustrate the advanced cryptographic protocols for blockchain technology.	TPS3	70	80
CO5	Apply the various privacy-preserving primitives to blockchain.	TPS3	70	80
CO6	Analyze the use of security techniques for the various blockchain platforms.	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1.	S	M	L		L	M	M	L	M	M		M	M	L	L
CO2.	S	M	L		M	M	M	L	M	M		M	M	M	L
CO3	S	M	L		L	M	M	L	M	M		M	M	L	L
CO4	S	M	L		M	M	M	L	M	M		M	M	M	L
CO5	S	M	L		L	M	M	L	M	M		M	M	L	L
CO6	S	S	M	L	M	S	M	L	M	M		M	M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass2	Terminal						
TPS Sca le	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO 1	5	10	20	-	-	-	30	-	-	-	-	-	-	-		-	5	10	-	-	-
CO 2	5	10	20	-	-	-	30	-	-	-	-	-	-	-		2	5	10	-	-	-
CO 3	-	10	20	-	-	-	40	-	-	-	-	-	-	-		2	5	10	-	-	-
CO 4	-	-	-	-	-	-	-	5	10	15	-	-	-	20		2	5	10	-	-	-
CO 5	-	-	-	-	-	-	-	5	10	15	-	-	-	20		2	5	10	-	-	-
CO 6	-	-	-	-	-	-	-	-	10	10	20	-	-	30	30	2	-	-	15	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Blockchain Revisited: Basics, Architecture, Distributed Consensus, Smart Contracts, Cryptographic Hash, Merkle Tree, Digital Signature - ECDSA.

Security Challenges in Blockchain: Common vulnerabilities and attacks, Front End Risks, Back End Risks, Hyperledger Fabric Risks.

Smart Contract Security: Ethereum, Deploying smart contracts in Solidity, Case study: The DAO Attack, The Party MultiSig Hack, and Smart Contract Security Practices.

Advanced Cryptographic Primitives in Blockchain: Commitment Schemes – Hash-based, Encryption-based, Discrete-log based, Accumulators – Basics, Types, Dynamic Universal Accumulator Scheme, Proof systems for Blockchain – ZKP, NIZK, ZK-SNARK.

Privacy-Preserving Authentication Techniques: Anonymous Credentials, Blind Signature in Blockchain, Ring signature schemes.

Blockchain Security in Action: Bulletproofs, Identity Mixer in Hyperledger Fabric, Monero.

Text Books

1. Imran Bashir, 'Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained', Packt Publishers, 2nd Edition, 2018.

2. S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, 'Blockchain Technology: Cryptocurrency and Applications', Oxford University Press, 2019.

Reference Books & web resources

1. Boneh, D, Bu'inz, B & Fisch, B, 'Batching techniques for accumulators with applications to iops and stateless blockchains', Annual International Cryptology Conference, Springer LNCS, vol. 11692, pp. 561-586, 2019.
2. Androulaki, E, Barger, A, Bortnikov, V, Cachin, C, Christidis, K, De Caro, A, Enyeart, D, Ferris, C, Laventman, G & Manevich, Y, 'Hyperledger Fabric: A distributed operating system for permissioned blockchains', Proceedings of the Thirteenth EuroSys Conference, DOI. 10.1145/3190508.3190538, Article No. 30, pp. 1-15, 2018.
3. Bünz, Benedikt, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. "Bulletproofs: Short proofs for confidential transactions and more." In 2018 IEEE symposium on security and privacy (SP), pp. 315-334. IEEE, 2018.
4. Sun, Shi-Feng, Man Ho Au, Joseph K. Liu, and Tsz Hon Yuen. "Ringct 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency monero." In Computer Security–ESORICS 2017: 22nd European Symposium on Research in Computer Security, Oslo, Norway, September 11-15, 2017, Proceedings, Part II 22, pp. 456-474. Springer International Publishing, 2017.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Blockchain Fundamentals	
1.1	Basics, Architecture	1
1.2	Distributed Consensus	1
1.3	Smart Contracts	1
1.4	Cryptographic Hash, Merkle Tree	1
1.5	Digital Signature - ECDSA	2
2	Security Challenges in Blockchain, ..	
2.1	Common vulnerabilities and attacks,	1
2.2	Front End Risks	2
2.3	Back End Risks	1
2.4	Hypeledger Fabric Risks	2
3	Smart Contract Security	
3.1	Ethereum	1
3.2	Deploying smart contracts in Solidity	2
3.3	The DAO Attack	1
3.4	The Party MultiSig Hack	1
3.5	Smart Contract Security Practices	1
4	Advanced Cryptographic Primitives in Blockchain	
4.1	Commitment Schemes	2
4.2	Accumulators – Basics and Types	1
4.3	Dynamic Universal Accumulator	1
4.4	Proof systems for blockchain	2
5	Privacy-Preserving Authentication Techniques	
5.1	Anonymous Credentials	2
5.2	Blind Signature in Blockchain	2
5.3	Ring signature schemes	2
6	Blockchain Security in Action	
6.1	Bulletproofs	2
6.2	Identity Mixer in Hyperledger Fabric	2

Module No.	Topic	No. of Periods
6.3	Monero	2
	Total	36

Course Designer(s):

1. Dharani J, Assistant Professor, CSE jdicse@tce.edu

22CSPU1**TRUSTWORTHY AI**

Category	L	T	P	Credit
PE	3	0	0	3

Preamble

This syllabus emphasizes on the basics of Trust, including Fairness. Robustness, Safety, security and Privacy in the context of Artificial Intelligence. This also covers the steps required to implement Trustworthy AI in practice.

Prerequisite

- Artificial Intelligence

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome 1	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Develop Fairness and Robustness in AI	TPS3	B	75
CO2	Develop Transparent and explainable AI	TPS3	B	75
CO3	Build Safe and secure AI	TPS3	B	75
CO4	Apply the concept of privacy and laws in AI	TPS3	B	75
CO5	Impart accountability and Responsibility in AI	TPS3	B	75
CO6	Experiment Trustworthy AI in practice	TPS4	B	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1.	S	M	L		L			L	L	L		L	M		L
CO2.	S	M	L		L			L	L	L		L	M		L
CO3	S	M	L		L			L	L	L		L	M		L
CO4	S	M	L		L			L	L	L		L	M		L
CO5	S	M	L		L			L	L	L		L	M		L
CO6	S	S	M	L	L			L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	5	10										20												5	10				
CO2	10	10	20										40												2	5	10			
CO3	10	10	20										40												2	5	10			
CO4							10	10											20						2	5	10			

CO5						1 0	1 0	2 0									4 0			2	5	1 0			
CO6						1 0	1 0	2 0										4 0		2	5		1 0		

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Fair and Robust

Bias in AI-Selection Bias-Confirmation Bias-Explicit and Implicit Bias-Institutional Bias-Tradeoffs in Fairness- Leading Practices in Promoting Fairness-Fairness Toolkits Robust vs Brittle AI- Developing Reliable AI-The Challenge of Generalizable Deep Learning-Factors Influencing AI Reliability - Robustness and Bad Actors-Leading Practices in Building Robust and Reliable AI-Driving Toward Robust and Reliable Tools

Transparent and Explainable

Defining the Nature of Transparency in AI-The Limits of Transparency-Weighing the Impact on the Stakeholders-Taking Steps into Transparency The Components of Understanding AI Function -The Value in Explainable AI -Driving Innovation and Application- Improving Model Performance- Encouraging Use Through Trust-Satisfying Regulatory Inquiries-Factors in Explainability- AI models for Explainability- Technical Approaches to Fostering Explainability-Leading Practices in Process- The Explainable Imperative-

Secure and Safe

AI Compromise-Unsecure AI -The Consequences from Compromised AI-Leading Practices for Shoring-Up AI Security-Securing the Future with AI Understanding Safety and Harm in AI-Aligning Human Values and AI Objectives-Technical Safety Leading Practices-Seeking a Safer Future with AI

Privacy

Consent, Control, Access, and Privacy-The Friction Between AI Power and Privacy-Beyond Anonymization or Pseudonymization-Privacy Laws and Regulations-Methods for preserving privacy-Leading Practices in Data and AI Privacy

Accountable and Responsible

Accountability-Balancing Innovation and Accountability-Laws, Lawsuits, and Liability-Leading Practices in Accountable AI-Accounting for Trust in AI Corporate Responsibility in the AI Era-Motivating Responsible AI Use-Balancing Good, Better, and Best-Leading Practices in the Responsible Use of AI-Trust Emerging from Responsibility

Trustworthy AI in practice

Steps – Identify the Relevant Dimensions of Trust-- Cultivating Trust Through People, Processes, and Technologies -Guidelines for Action on Trustworthy AI

Text Book

1. Beena Ammanath "Trustworthy AI_ A Business Guide for Navigating Trust and Ethics in AI" First Edition-Wiley -2022

Reference Books & web resources

1. Yada Pruksachatkun, Matthew Mcateer, Subho Majumdar "Practicing Trustworthy Machine Learning: Consistent, Transparent, and Fair AI Pipelines" O'REILLY -2023
2. <https://www.coursera.org/learn/trustworthy-generative-a>
3. <https://www.coursera.org/learn/trustworthy-ai-for-healthcare-management>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Fair and Robust	
1.1	Bias in AI-Selection Bias-Confirmation Bias-Explicit and Implicit Bias	1
1.2	Institutional Bias-Tradeoffs in Fairness- Leading Practices in Promoting Fairness	1
1.3	Robust vs Brittle AI- Developing Reliable AI	1
1.4	The Challenge of Generalizable Deep Learning-Factors Influencing AI Reliability	1
1.5	Robustness and Bad Actors-Leading Practices in Building Robust and Reliable AI-	1
1.6	Driving Toward Robust and Reliable Tools	1
2	Transparent and Explainable	
2.1	Defining the Nature of Transparency in AI-The Limits of Transparency-Weighing the Impact on the Stakeholders-	1
2.2	Taking Steps into Transparency -The Components of Understanding AI Function	1
2.3	The Value in Explainable AI -Driving Innovation and Application- Improving Model Performance- Encouraging Use Through Trust-	1
2.4	Satisfying Regulatory Inquiries-Factors in Explainability	1
2.5	Technical Approaches to Fostering Explainability-Leading Practices in Process- The Explainable Imperative	2
3	Secure and Safe	
3.1	AI Compromise-Unsecure AI -The Consequences from Compromised AI	1
3.2	Leading Practices for Shoring-Up AI Security-Securing the Future with AI	2

Module No.	Topic	No. of Periods
3.3	Understanding Safety and Harm in AI-Aligning Human Values and AI Objectives-	1
3.4	Technical Safety Leading Practices-Seeking a Safer Future with AI	2
4	Privacy	
4.1	Consent, Control, Access, and Privacy	1
4.2	The Friction Between AI Power and Privacy-Beyond Anonymization or Pseudonymization-	2
4.3	Privacy Laws and Regulations	1
4.4	Leading Practices in Data and AI Privacy	2
5	Accountable and Responsible	
5.1	Accountability-Balancing Innovation and Accountability	1
5.2	Laws, Lawsuits, and Liability	1
5.3	Leading Practices in Accountable AI-Accounting for Trust in AI	1
5.4	Corporate Responsibility in the AI Era	1
5.5	Motivating Responsible AI Use--Balancing Good, Better, and Best	1
5.6	Leading Practices in the Responsible Use of AI-Trust Emerging from Responsibility	1
6	Trustworthy AI in practice	
6.1	Steps – Identify the Relevant Dimensions of Trust	2
6.2	Cultivating Trust Through People, Processes, and Technologies -	2
6.3	Guidelines for Action on Trustworthy AI	2
	Total	36

Course Designer(s):

1. Dr.C.Senthilkumar, Associate Professor, CSE, cskcse@tce.edu

22CSPV1	AUTONOMOUS SYSTEMS	Category	L	T	P	C
		PE	3	0	0	3

Preamble

This course will cover every aspect of autonomous systems, from basic ideas and workings to their wide range of uses and latest developments. Learn the concepts that underpin these robots' autonomous operation and decision-making abilities as you explore their complexities. Become an expert in the human-machine interface, control systems, and computational methods that will influence autonomous systems in the future.

Prerequisite

- Python Programming
- Linear Algebra and Calculus
- Basic Electronics
- Artificial Intelligence

Course Outcomes

On the successful completion of the course, students will be able to

CO No.	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Demonstrate a deep and comprehensive understanding of autonomous systems and its applications.	TPS2	70	80
CO2	Acquire proficiency in the kinematics and dynamics of robotic systems, sensor and actuator integration, and control architectures for locomotion and manipulation	TPS3	70	75
CO3	Apply computational techniques, and demonstrating competence for implementing autonomous systems.	TPS3	70	75
CO4	Analyze historical aspects of UAVs and gain comprehensive knowledge of their components.	TPS4	70	75
CO5	Design and Analyze feedback control systems to ensure the stability and performance in autonomous systems.	TPS4	70	75
CO6	Design safe, reliable, and verifiable user interfaces for autonomous systems, with an importance of ethical practices in technology and explore advanced and emerging trends in autonomous systems.	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1.	M	L										L	L		
CO2.	S	M	L		L					L		M	M		M
CO3	S	M	L		L					L		M	M		M
CO4	S	S	M	L	M		M	M	M			M	M		M
CO5	S	S	M	L			M	M	M	L		M	M		M
CO6	S	M	L		M			M	M	L	L	M	S	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1		CAT2						Ass2		Terminal					
	1	2	3	4	5	6	2	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	-	30	-	-	-	-	20		-	-	-	-	-	-	-		-	15	-	-	-	-
CO2	-	10	25	-	-	-	-	40	-	-	-	-	-	-	-		-	5	12	-	-	-
CO3	-	10	25	-	-	-	-	40	-	-	-	-	-	-	-		-	5	12	-	-	-
CO4	-	-	-	-	-	-	-	-	-	15	-	20	-	-		40	-	5	-	12	-	-
CO5	-	-	-	-	-	-	-	-	-	15	-	20	-	-		40	-	5	-	12	-	-
CO6	-	-	-	-	-	-	-	-	-	20	10	-	-	-	20		-	5	12	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus **Introduction to Autonomous Systems** - Definition and significance of autonomous systems - Historical context and evolution of autonomous technologies – Functional Architecture – Software and Hardware Architecture - Major Functions - Applications across Industries

Mechanisms – Locomotion and Manipulation - Kinematics and dynamics of robotic systems - Sensors and actuators in robotics - Robot programming languages and control architectures

Computation for Autonomous Systems - Vision - Feature extraction and object recognition - Localization techniques - Visual Simultaneous Localization and Mapping (SLAM) - Path planning algorithms - Collision avoidance strategies

Unmanned Autonomous Vehicles - Historical aspects - Electro-Optical Sensors – Radio Frequency, and Electronic Components – Electrical Sources - Navigation System

Control Systems in Autonomous Systems - Feedback control systems - PID controllers and their applications - State-space representation for control

Human-Machine Interaction in Autonomous Systems - User interfaces for autonomous systems - Ethical considerations in autonomous technology - Safe, Reliable, and Verifiable Autonomous Systems.

Advanced/Emerging Trends in Autonomous Systems - Swarm robotics – Nano Robots - Bio-inspired autonomous systems - Edge computing in autonomous systems

Text Book

1. Nikolaus Correll, Bradley Hayes, Christoffer Heckman, Alessandro Roncone, "Introduction to Autonomous Robots: Mechanisms, Sensors, Actuators, and Algorithms" MIT Press, 2022.
2. R.Jha "Theory, Design, and Applications of Unmanned Aerial Vehicles" CRC Press, Taylor & Francis 2017
3. Katsuhiko Ogata, "Modern Control Engineering", Pearson Education, Fifth Edition, 2010

Reference Books & web resources

1. Mhamed Itmi & Alain Cardon, "New Autonomous Systems", John Wiley & Sons Wiley-ISTE, 2016.
2. Mikell P Groover & Nicholas G Odrey, Mitchel Weiss, Roger N Nagel, Ashish Dutta, "Industrial Robotics, Technology programming and Applications", McGraw Hill, 2012.
3. Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer, 2010.
4. De Gyurky, S.M., & Tarbell, M.A., "The Autonomous System: A Foundational Synthesis of the Sciences of the Mind", John Wiley & Sons, 2013.
5. S.R. Deb, "Robotics Technology and flexible automation", Tata McGraw-Hill Education., 2009.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Autonomous Systems	
1.1	Definition and significance of autonomous systems	1
1.2	Historical context and evolution of autonomous technologies	1
1.3	Functional Architecture, Software and Hardware Architecture	1
1.4	Major Functions, Applications across industries	1
2	Mechanisms	
2.1	Locomotion and Manipulation	2
2.2	Kinematics and dynamics of robotic systems	2
2.3	Sensors and actuators in robotics	2
2.4	Robot programming languages and control architectures	2
3	Computation for Autonomous Systems	

3.1	Vision - Feature extraction and object recognition	1
3.2	Localization techniques	1
3.3	Visual Simultaneous Localization and Mapping (SLAM)	2
3.4	Path planning algorithms, Collision avoidance strategies	2
4	Unmanned Autonomous Vehicles	
4.1	Historical aspects - Electro-Optical Sensors	2
4.2	Radio Frequency, and Electronic Components	2
4.3	Propulsion Systems, Electrical Sources	2
4.4	Navigation System	1
5	Control Systems in Autonomous Systems	
5.1	Feedback control systems	1
5.2	PID controllers and their applications	2
5.3	State-space representation for control	2
6	Human-Machine Interaction in Autonomous Systems	
6.1	User interfaces for autonomous systems	1
6.2	Ethical considerations in autonomous technology	1
6.3	Safe, Reliable, and Verifiable Autonomous Systems.	1
7	Advanced/Emerging Trends in Autonomous Systems	
7.1	Swarm robotics, Nano Robots	1
7.2	Bio-inspired autonomous systems, Edge computing in autonomous systems	2
	Total	36

Course Designer(s):

1. Mr.D.Nagendra Kumar (dnkcse@tce.edu)

22CSRA1	DEEP LEARNING	Category	L	T	P	C
		PEES	3	0	0	3

Preamble

This syllabus on "Deep Learning" is intended for students with a background in computer science and provides an in-depth exploration of Artificial Neural Networks, Convolutional Neural Networks (ConvNets), and Recurrent Neural Networks (RNNs). The syllabus emphasizes on the practical aspects of Deep Learning, including applications of ConvNets in image recognition, object detection, and segmentation, as well as the use of RNNs in natural language processing (NLP) tasks.

Prerequisite

- Python Programming

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct training algorithms for Neural Networks using the Perceptron learning algorithm and Feed-Forward Networks	TPS 3	B	70
CO2	Illustrate optimization methods for Deep Learning Algorithms	TPS 3	B	70
CO3	Illustrate Convolutional Neural Networks algorithms and solve real-world problems	TPS 3	B	70
CO4	Analyse the Deep Learning algorithms to solve complex problems with an understanding of the trade-offs involved	TPS 4	B	70
CO5	Construct learning algorithms for Recurrent Neural Networks	TPS 3	B	70
CO6	Analyse the Deep Learning algorithms which are more appropriate for various types of learning tasks in domains like Vision and NLP	TPS 4	B	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		M	L	L		M	M	M	L
CO2	S	M	L		M	L		M	L	L		M	M	M	L

CO3	S	M	L		M	L		M	L	L		M	M	M	L
CO4	S	S	M	L	M	L		M	L	L		M	S	M	L
CO5	S	M	L		M	L		M	L	L		M	M	M	L
CO6	S	S	M	L	M	L		M	L	L		M	S	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						Assignment 1			CAT 2						Assignment 2			Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6
CO1	-	10	20	-	-	-	-	-	20	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	-	10	20	-	-	-	-	-	20	-	-	-	-	-	-	-	-	-	2	5	12	-	-	-
CO3	-	20	20	-	-	-	-	-	60	-	-	-	-	-	-	-	-	-	2	5	12	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	10	20	-	-	-	-	-	10	2	5	-	12	-	-
CO5	-	-	-	-	-	-	-	-	-	-	10	20	-	-	-	-	-	30	2	5	12	-	-	-
CO6	-	-	-	-	-	-	-	-	-	-	20	20	-	-	-	-	-	60	-	5	-	12	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Artificial Neural Networks: Basic concepts of Artificial Neurons, single and multi-layer perceptron, perceptron learning algorithm, its convergence proof, different Activation Functions, softmax cross entropy loss function.

Optimization: Types of errors, bias-variance trade-off, overfitting-underfitting, brief review of concepts from optimization, variants of gradient descent, momentum-based methods

ConvNets, ConvNet Architectures and Dropout/Regularization/BatchnormConvNets: Basic concepts of Convolutional Neural Networks starting from filtering. Convolution and pooling operation and arithmetics of these. ConvNet Architectures: Discussions on famous ConvNet architectures - AlexNet, GoogLeNet, ResNet, MobileNet-v2 Discussion on regularization, Dropout, Batch normalization – Introduction to GCNN (Graph Convolutional Neural Network) - Multiclass Classification Model for Computer Vision (CV)

CNN Detection and Segmentation Applications: Basics on detection, segmentation problem definition, challenges, Evaluation, Datasets and Localization by regression. Discussion on detection as classification, region proposals, RCNN and YOLO architectures, fully convolutional segmentations, Mask-RCNNs

Recurrent Architectures and NLP Applications: Discussion on Recurrent Neural Networks (RNNs) – Gated Neural Network (GRU), Long-Short Term Memory (LSTM) architectures – Auto Encoders - Generative Adversarial Network (GAN)

Transformers: Vision Transformers, Attention Model types, Transformer (BERT)

CO6 will be evaluated by Mini project - Guidelines for the Mini-project:

Group formation: Students are split into project groups with around 3 members in each group. A team can execute the project using DL algorithms and improve the efficiency of the algorithm by pre-processing methods using any of the DL frameworks like Keras, TensorFlow. At the end of the semester, the team members have to present their project, submit their report and share their lessons learnt/best practices with other teams.

Some of the activities may include: (but not limited to)

- ✓ Image Classification with different DL namely, Multilayer perceptron, Convolutional Neural Networks, Recurrent Neural Networks and Transformers
- ✓ Language Translation with Sequence-to-Sequence Models, Auto encoders and GPT2 model

Some of the Mini-project titles may include: (but not limited to)

- ✓ Image recognition for autonomous vehicles
- ✓ Speech recognition for virtual assistants
- ✓ Real-time emotion detection in videos for marketing analysis

Text Books

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep learning, In preparation for MIT Press, Available online: <http://www.deeplearningbook.org>, 2016

Reference Books

1. S. Haykin, Neural Networks and Learning Machines, Prentice Hall of India, 2010
2. Satish Kumar, Neural Networks - A Class Room Approach, Second Edition, Tata McGraw-Hill, 2013
3. B. Yegnanarayana, Artificial Neural Networks, Prentice- Hall of India, 1999
4. C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	Artificial Neural Networks	(6)
1.1	Basic concepts of Artificial Neurons	1
1.2	Single and multi-layer perceptron's	2
1.3	Perceptron learning algorithm, its convergence proof	1
1.4	Activation Function - different Activation Functions	1
1.5	Softmax cross entropy loss function.	1
2	Optimization	(6)
2.1	Types of errors	1
2.2	Bias-variance trade-off	1
2.3	Overfitting–Underfitting	1
2.4	Brief review of concepts from optimization	1
2.5	Variants of gradient descent	1
2.6	Momentum based methods	1

3	ConvNets, ConvNet Architectures and Dropout/Regularization/Batchnorm.	(6)
3.1	ConvNets: Basic concepts of Convolutional Neural Networks starting from filtering	1
3.2	Convolution and pooling operation and arithmetic's of these	1
3.3	ConvNet Architectures: Discussions on famous ConvNet architectures - AlexNet, GoogLeNet, ResNet, MobileNet-v2	1
3.4	Discussion on regularization, Dropout, Batch normalization etc.	1
3.5	Introduction to Graph Convolutional Neural Network (GCNN)	1
3.6	Multiclass Classification Model for Computer Vision (CV)	1
4	Applications: Detection and Segmentation	(6)
4.1	Discussion on detection, Segmentation problem definition, challenges	1
4.2	Evaluation, Datasets and Localization by regression	1
4.3	Discussion on detection as classification, region proposals	1
4.4	RCNN and YOLO architectures	2
4.5	Fully convolutional segmentations, Mask-RCNNs	1
5	Recurrent Architectures and NLP Applications	(6)
5.1	Introduction to NLP	1
5.2	Discussion on Recurrent Neural Networks (RNNs)	1
5.3	Gated Neural Network (GRU) Architecture	1
5.4	Long-Short Term Memory (LSTM) architectures	1
5.5	Auto Encoders - Generative Adversarial Networks (GAN)	2
6	Transformers	(6)
6.1	Introduction to Vision Transformers	1
6.2	Discussion on Attention Model types	2
6.3	Discussion on Transformers (BERT)	2
	Mini Project on various Deep Learning Algorithms/ Applications	1
Total Hours		36

Course Designer(s):

1. Dr.K. Sundarakantham, Professor, CSE,
2. Mr. S.Santhana Hari, AP,CSE

kskcse@tce.edu
sshcse@tce.edu

22CSRB1**WEB 3.0 AND METAVERSE**

Category L T P Credit

PEES 3 0 0 3

Preamble

The cutting-edge technologies that are reshaping the internet are examined in this course. The word "Metaverse" is causing a stir in the constantly changing computer industry and grabbing the interest of tech enthusiasts. The purpose of this course is to familiarise students with the idea of the Metaverse and its relevance in the modern world. It will discover about the functions of blockchain technology and non-fungible tokens (NFTs) in these domains.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the fundamentals of blockchain technology and its role in Web3	TPS3	70	80
CO2	Identify and explain the key technologies that enable the metaverse, such as virtual reality, augmented reality, blockchain, and decentralized systems.	TPS3	70	80
CO3	Explain the key components and technologies that contribute to the creation of the metaverse.	TPS3	70	80
CO4	Adapt various NFT standards and their use cases for decentralised organisations.	TPS3	70	80
CO5	Design and build decentralized applications using appropriate development frameworks.	TPS3	70	80
CO6	Analyze industry-specific applications of the metaverse like gaming, education, healthcare.	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	S	M	L		M	M	L	M	M	M		M	M	L	M
CO2	S	M	L		M	M	L	M	M	M		M	M	L	M
CO3	S	M	L		M	M	L	M	M	M		M	M	L	M
CO4	S	M	L		M	M	L	M	M	M		M	M	L	M

Passed in Adhoc Board of Studies meeting on 17.04.2026

Approved in 71st Academic Council meeting on 27.06.2026

CO5	S	M	L		M	M	L	M	M	M		M	M	L	M
CO6	S	M	L	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
	1	2	3	4	5	6	3	1	2	3	4	5	6		1	2	3	4	5	6
CO1	5	5	15	-	-	-	30	-	-	-	-	-	-	-	2	5	10	-	-	-
CO2	5	5	30	-	-	-	35	-	-	-	-	-	-	-	2	5	10	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-	2	-	10	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	5	15	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	5	15	-	-	-
CO6	-	-	-	-	-	-	-	5	5	10	20	-	-	30	30	-	-	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Web 3.0

Building a Web 3.0 Community-Web 3.0: The Road Ahead- Design elements for Web 3.0 Websites – Structure- Human Computer Interaction perspective- Key Blockchain Components: DLT and Smart Contracts -Blockchain-Based Digital Assets-Stablecoins-Valuation Metrics for Digital Assets-dApps

Fundamentals of Metaverse

The history of the Metaverse- Evolution of metaverse-Overview to metaverse functional blocks- Cases on metaverse applications in different industries-Understanding Extended Reality- Overview to AR/VR for metaverse-Technical architecture of XR applications-Understanding VR ecosystems

Building the metaverse technology:

Networking -Computing -Virtual World Engines - Interoperability - Hardware - Payment Rails - Artificial Intelligence, Cloud, Data Analytics-Digital Twins, Mirror World and Manufacturing-Importance of Avatars to the Metaverse and Digital Identity-Challenges and trends

Non-Fungible Tokens (NFTs)

Four Types of NFTs-Three Types of NFT Platforms-DAOs- Decentralised Finance (DeFi): types of market makers, designs and applications-Decentralised Autonomous Organisations (DAO): Communities, governance and use cases

DeFi,GameFi and SocialFi:

Introduction to DeFi, Smart contracts and how it works-DeFi Vs Traditional Finance-Five layers of DeFi-Both sides of DeFi-Use Cases of DeFi in Metaverse-More use cases of DeFi, - Metaverse and GameFI, GameFI Ecosystem in Metaverse- Changing gaming landscape GameFI and Metaverse- Metaverse and SocialFI

Case study: Social Media in Meta, Education in Metaverse, Gaming in Meta, Facebook's Meta

Text Book

1. Evan Karnoupakis ,”NFTs, the Metaverse, and Everything Web 3.0” O'Reilly Media, Inc., January 2023,
2. Matthew Ball ,The Metaverse: And How It Will Revolutionize Everything
3. Cathy Hackl, Dirk Lueth, Tommaso Di Bartolo, John Arkontaky (Editor), Yat Siu_ “Navigating the Metaverse: A Guide to Limitless Possibilities in a Web 3.0 World”,Wiley,2022

Reference Books & web resources

1. Metaverse Web 3.0 and DeFi: A Fintech Masterclass | Udemy
2. Web3 and Strategy: Blockchain, Metaverse, and NFT Essentials (15.562) | MIT Orbit
3. IIT Delhi Advanced Certification in Web 3.0, Social Media, and Metaverse (timespro.com)
4. Metaverse and Web3 Online Course with Certificate | 3.0 University (3university.io)

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Web 3.0	
1.1	Building a Web 3.0 Community-Web 3.0: The Road Ahead	1
1.2	Design elements for Web 3.0 Websites – Structure- Human Computer Interaction perspective	1
1.3	Key Blockchain Components: DLT and Smart Contracts	1
1.4	Blockchain-Based Digital Assets	1
1.5	Stablecoins-Valuation Metrics for Digital Assets-	1
1.6	dApps	1
2	Fundamentals of Metaverse	
2.1	The history of the Metaverse- Evolution of metaverse	1

Module No.	Topic	No. of Periods
2.2	Overview to metaverse functional blocks	1
2.3	Cases on metaverse applications in different industries	1
2.4	Understanding Extended Reality-Overview to AR/VR for metaverse	1
2.5	Technical architecture of XR applications	1
2.6	Understanding VR ecosystems	1
3	Building the metaverse technology	
3.1	Networking	1
3.2	Computing -Virtual World Engines	1
3.3	Interoperability - Hardware - Payment Rails	1
3.4	Artificial Intelligence, Cloud, Data Analytics	2
3.5	Digital Twins, Mirror World and Manufacturing	1
3.6	Importance of Avatars to the Metaverse and Digital Identity - Challenges and trends	2
4	Non-Fungible Tokens (NFTs)	
4.1	Four Types of NFTs-Three Types of NFT Platforms	1
4.2	DAOs- Decentralised Finance (DeFi	2
4.3	Types of market makers, designs and applications	1
4.4	Decentralised Autonomous Organisations (DAO): Communities, governance and use cases	2
5	DeFi,GameFi and SocialFi	
5.1	Introduction to DeFi, Smart contracts and how it works	1
5.2	DeFi Vs Traditional Finance-Five layers of DeFi-Both sides of DeFi	1
5.3	Use Cases of DeFi in Metaverse	1
5.4	Metaverse and GameFI, GameFI Ecosystem in Metaverse	2
5.5	Changing gaming landscape GameFI and Metaverse	2

Module No.	Topic	No. of Periods
5.6	Metaverse and Socialfi	2
6	Case study	
6.1	Social Media in Meta, Education in Metaverse	1
6.2	Gaming in Meta	1
6.3	Facebook's Meta	1
	Total	36

Course Designer(s):

- | | |
|--|----------------|
| 1. Dr.S.Prasanna,Associate Professor,CSE | sprcse@tce.edu |
| 2. Dr.Raja Lavanya,Assistant Professor,CSE | rlit@tce.edu |

22CSRC1**5G NETWORK SECURITY**

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

This course will introduce students to security for 5G networks. It provides an overview of the standard 5G networks, as well as emphasizes the importance of monitoring, privacy and security in the 5G architecture.

Prerequisite

Knowledge on Computer networks, Mobile Communication and Security.

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Determine the requirements for security in 5G (Understand)	TPS2	70	75
CO2	Interpret the recommendations and challenges for 5G security. (Apply)	TPS3	70	75
CO3	Deduce the importance of 5G Security landscape. (Apply)	TPS3	70	75
CO4	Design a step by step model for 5G device and user security. (Apply)	TPS3	70	75
CO5	Develop a mechanism to ensure privacy and security of operators and 5G positioning. (Apply)	TPS3	70	75
CO6	Design a mechanism to ensure 5G Cloud and Virtual Network Security. (Apply)	TPS3	70	75

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	L										L		
CO2	S	M	L	-	-	L	L	L	M	L		M	M	L	L
CO3	S	M	L	-	M	L	L	L	M	L		M	M	L	L
CO4	S	M	L	-	-	L	L	L	M	L		M	M	L	L
CO5	S	M	L	-	M	L	L	L	M	L		M	M	L	L
CO6	S	M	L	-	M	L	L	L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	1 0	1 0																							4	1 0				
CO2	1 0	1 0	1 0												5 0										4	4	1 0			
CO3	1 0	2 0	2 0												5 0										4	4	1 0			
CO4							1 0		1 0											3 0					4	4	1 0			
CO5							1 0	1 0	2 0											3 0					4	4	1 0			
CO6							1 0	2 0	1 0											4 0					4		1 0			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

5G Mobile Networks - 5G Requirements - 5G Enabling Technologies - Drivers for 5G - New Use Cases - 5G Architecture overview - Security requirements and security services of the 5G system - Security services - Security domains - Network access security - Network domain security - User domain security - Application domain security - SBA domain security - Visibility and configurability of security

Mobile Networks Security Landscape - Mobile Security Lifecycle Functions - Design Principles for 5G Security - Security Recommendations and Challenges - Novel Technologies for 5G Security - Security in SDN based Mobile Networks - Cyber Security Business Models in 5G - The Business Case of Cyber Security in the Era of 5G - Business Model Options in 5G Cyber Security

5G Network Security - Physical Layer Security - WBPLSec Applied to 5G networks - 5G-WLAN Security - Network Architecture for WiFi-5G Networks interoperability - Security Consideration for Architectural Design of WiFi-5G Networks - Safety of 5G Network Physical Infrastructures - Customer Edge Switching - CES Security Framework - Software Defined Security Monitoring in 5G Networks - Software-Defined Monitoring Architecture

5G Device and User Security - IoT Security - Distributed Security Platform - Mobile Cloud Robot Security Scenarios - User Privacy, Identity and Trust in 5G - 5G Positioning - Outdoor versus Indoor Positioning Technologies - Passive versus Active Positioning - Security Threats and Privacy Issues in 5G Positioning - Physical-Layer Based Security Enhancements Mechanisms for Positioning in 5G - Cryptographic Techniques for Security and Privacy of Positioning

5G Cloud and Virtual Network Security - Mobile Virtual Network Operators (MVNO) Security - Cloudification of the Network Operators - TaaS Deployment Security - 5G, NFV, SDN and a Telco Cloud Security -- NFV-based Security Services - Cloud and MEC Security - Cloud Computing in 5G Networks - MEC in 5G Networks - Security Challenges in 5G Cloud and 5G MEC - Security Architectures for 5G Cloud and MEC - Security and Privacy Issues in New 5G Technologies

Text Books

1. Pramod Nair, Securing 5G and Evolving Architectures, Addison-Wesley Professional, 2021.

2. Madhusanka Liyanage, Ijaz Ahmad, Ahmed Bux Abro, A Comprehensive Guide to 5G Security, Wiley, 2018.

Reference Books & web resources

1. Ahmed A. Abd El-Latif · Bassem Abd-El-Atty, Security and Privacy Preserving for IoT and 5G Networks: Techniques, Challenges, and New Directions, Springer, 2022.
2. Jyrki T. J. Penttinen, 5G Explained - Security and Deployment of Advanced Mobile Communications, Wiley, 2019.

Course Contents and Lecture Schedule

Module No	Topic	No of Lectures	Course Outcome
1	5G Mobile Networks		
1.1	5G Requirements - 5G Enabling Technologies	1	CO1
1.2	Drivers for 5G - New Use Cases	1	CO1
1.3	5G Architecture overview	1	CO1
1.4	Security requirements and security services of the 5G system	1	CO1
1.5	Security domains	1	CO1
2	Mobile Networks Security Landscape		
2.1	Mobile Security Lifecycle Functions	1	CO2
2.2	Design Principles for 5G Security	1	CO2
2.3	Security Recommendations and Challenges	1	CO2
2.4	Novel Technologies for 5G Security	1	CO2
2.5	Security in SDN based Mobile Networks	1	CO3
2.6	Cyber Security Business Models in 5G	1	CO3
2.7	The Business Case of Cyber Security in the Era of 5G	1	CO3
2.8	Business Model Options in 5G Cyber Security	1	CO3
3	5G Network Security		
3.1	WBPLSec Applied to 5G networks	1	CO3
3.2	5G-WLAN Security - Network Architecture for WiFi-5G Networks interoperability	1	CO3
3.3	Security Consideration for Architectural Design of WiFi-5G Networks	1	CO3
3.4	Safety of 5G Network Physical Infrastructures	1	CO3
3.5	Customer Edge Switching, CES Security Framework	1	CO4
3.6	Software Defined Security Monitoring in 5G Networks	1	CO4
3.7	Software-Defined Monitoring Architecture	1	CO4
4	5G Device and User Security		
4.1	IoT Security - Distributed Security Platform	1	CO4
4.2	Mobile Cloud Robot Security Scenarios	1	CO4
4.3	User Privacy, Identity and Trust in 5G	1	CO5
4.4	5G Positioning - Outdoor versus Indoor Positioning Technologies	1	CO5
4.5	Passive versus Active Positioning - Security Threats and Privacy Issues in 5G Positioning	1	CO5

4.6	Physical-Layer Based Security Enhancements Mechanisms for Positioning in 5G - Cryptographic Techniques for Security and Privacy of Positioning	1	CO5
5	5G Cloud and Virtual Network Security		
5.1	Mobile Virtual Network Operators (MVNO) Security	1	CO6
5.2	Cloudification of the Network Operators - TaaS Deployment Security	1	CO6
5.3	5G, NFV, SDN and a Telco Cloud Security	1	CO6
5.4	NFV-based Security Services	1	CO6
5.5	Cloud and MEC Security	1	CO6
5.6	Security Architectures for 5G Cloud and MEC	1	CO6
5.7	Security and Privacy Issues in New 5G Technologies	1	CO6
	Total	36	

Course Designer(s):

1. Dr.G.S.R.Emil Selvan, Associate Professor, CSE

emil@tce.edu

22CSR01**EDGE AI**

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

This course on Edge AI is a first level course for aspiring engineers of Edge Computing/IoT/AI developers to acquire practice-based AI/Machine Learning skills and Edge Computing fundamentals. The course has been designed with a clear vision on future developments of Edge Computing in industrial application

Prerequisite

- Distributed Computing
- Artificial Intelligence
- Machine Learning
- Deep Learning

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Choose relational model of edge computing along with AI, Machine Learning and IoT	TPS3	70	80
CO2	Identify architecture and communication protocols for edge application constraints	TPS3	70	80
CO3	Develop Edge AI application with its core technological components	TPS3	70	80
CO4	Practice on tools supporting for ML,DL and Edge Computing solutions	TPS3	70	80
CO5	Use model-architecture co-design where real-time, hardware, energy, and privacy constraints of edge devices play	TPS3	70	80
CO6	Analyse Technical constraints required for designing edge application with proper societal requirements	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L	L	L	L		M	M	M		M	M	L	M
CO2.	S	M	L		L	L		M	M	M		M	M	L	M

CO3.	S	M	L		L	L		M	M	M		M	M	L	M
CO4.	S	M	L		L	L		M	M	M	L	M	M	L	M
CO5.	S	M	L		L	L		M	M	M	L	M	M	L	M
CO6.	S	S	M	L	S	S	M	M	M	M	M	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal						
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	5	5	20	-	-	-	30	-	-	-	-	-	-	-		1	4	10	-	-	-
CO2	5	5	25	-	-	-	35	-	-	-	-	-	-	-		1	4	10	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-		1	4	10	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20		1	4	10	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20		1	4	10	-	-	-
CO6	-	-	-	-	-	-	-	5	5	20	10	-	-	40	20	1	4	10	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Overview-Edge computing; Internet-of-Things (IoT); cyber-physical systems- Tiny ML and Neural Networks-Lightweight DL models

Architectures and Communication: hardware and application constraints-mobile Processors-Accelerators –Networking Protocols: Wi-Fi –Bluetooth Low Energy protocol- MQTT protocol-Binary data packing and unpacking – Real-Time database

Edge AI Development: Embedded Systems-MicroPython firmware and environments - Energy-aware machine learning (ML); Deep learning; model compression; Knowledge distillation;

Edge AI Applications: Federated learning; ML security; System optimization; Model-architecture co-design; object detection; social sensing;

Edge AI Deployment: On Device: Peripherals: ADC, UART, SPI, I2C, PWM, Timer: Temperature Sensor, Reading and decoding GPS Signals-Servomotor-Edge Server: Video Surveillance -Cloud Based: Autonomous Agents

Text Books

1. Perry lea, IoT and Edge Computing for Architects: Implementing Edge and IoT Systems from Sensors to Clouds with Communication Systems, Analytics, and Security, Packet Publishing, 2020, 2nd Edition
2. Fadi Al-Turjman, Edge Computing From Hype to Reality, EAI/Springer Innovations in Communication and Computing, 2019
3. Pete Warden and Daniel Situnayake, TinyML, O'Reilly, 2019

Reference Books & web resources

1. AWS IoT: Developing and Deploying an Internet of Things: https://courses.edx.org/courses/course-v1:AWS+OTP_AWSD5+1T2019/course/
1. <https://www.nvidia.com/en-us/on-demand/session/gtcfall21-a31535/>, Edge AI and Robotics, DLI

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Overview	
1.1	Edge computing;	1
1.2	Internet-of-Things (IoT);	1
1.3	Cyber-physical systems	1
1.4	Tiny ML and Neural Networks-Lightweight DL models	3
2	Architectures and Communication:	
2.1	Hardware and application constraints	1
2.2	Mobile processors-Accelerators	1
2.3	Networking Protocols: WiFi –Bluetooth Low Energy protocol- MQTT protocol	2
2.4	Binary data packing and unpacking	1
2.5	Real-Time database	1
3	Edge AI Development	
3.1	Embedded Systems	1
3.2	MicroPython firmware and environments	2

Module No.	Topic	No. of Periods
3.3	Energy-aware machine learning (ML)	1
3.4	Deep learning; model compression	2
3.5	Knowledge distillation	2
4	Edge AI Applications	
4.1	Federated learning	1
4.2	ML security	1
4.3	System optimization	1
4.4	Model-architecture co-design	2
4.5	object detection	1
4.6	social sensing	1
5	Edge AI Deployment	
5.1	On Device: Peripherals: ADC, UART, SPI, I2C, PWM, Timer: Temperature Sensor, Reading and decoding GPS Signals-Servomotor	3
5.2	Edge Server: Video Surveillance	3
5.3	Cloud Based: Autonomous Agents	3
	Total	36

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, rlsit@tce.edu
Computer Science and Engineering,

22CSRE1 **DECENTRALIZED
APPLICATIONS DEVELOPMENT**

Category L T P Credit
PEES 3 0 0 3

Preamble

This course explores the fundamentals and practical aspects of decentralized applications (DApps) development, covering blockchain networks, smart contracts, and the tools and techniques necessary for building secure and efficient DApps. Students will gain hands-on experience in designing DApp interfaces, managing decentralized storage, and implementing business logic within decentralized frameworks. Students will develop a comprehensive understanding of DApp ecosystems through case studies and projects.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the key concepts of blockchain, decentralized networks, and different consensus methods that power decentralized applications.	TPS3	70	80
CO2	Demonstrate proficiency in writing and deploying smart contracts using Solidity, enabling secure, automated transactions and interactions on the blockchain.	TPS3	70	80
CO3	Apply skills in front-end and back-end development, including the use of Web3.js and Ether.js libraries, to create functional and user-friendly decentralized applications.	TPS3	70	80
CO4	Identify various types of DApps, such as those for finance, identity, and marketplaces, and understand their real-world applications.	TPS3	70	80
CO5	Implement decentralized storage and databases, like IPFS and OrbitDB, to efficiently manage data in DApps.	TPS3	70	80
CO6	Develop end-to-end DApps that handle data, manage user interactions, and process transactions securely on a decentralized network.	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L		M	M	L	M	M	M		M	M	L	M

CO2.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO3.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO4.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO5.	S	M	L		M	M	L	M	M	M		M	M	L	M
CO6.	S	M	L	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	5	5	15	-	-	-	30	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	5	5	30	-	-	-	35	-	-	-	-	-	-	-	2	5	15	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-	2	-	15	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	5	15	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	5	15	-	-	-
CO6	-	-	-	-	-	-	-	5	5	10	20	-	-	60	-	-	-	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction : Introduction to Blockchain and Decentralized Network - Types of Blockchain- Consensus mechanisms- Blockchain and Decentralization - The History of Decentralized Applications-Working of Decentralized Apps -DApps Protocols

Smart contracts and Solidity Language : Solidity Data Types- Contract Structure -State Variables Functions- Decentralized Apps Using Ethereum DApps - Ether.js library -Front-end- User Interface (UI) design and development for Decentralized Applications -Web3.js- Smart Contracts (Backend of the App) –Storage-IPFS

Flourishing Dapp Ecosystem : Decentralized Wealth-Decentralized Identity-Decentralized Computing-Decentralized Bandwidth-Decentralized Markets for Decentralized Assets-Practical Decentralization

Decentralized Databases: OrbitDB - Peer to Peer Distributed Database- BigchainDB- Ties DB- Bluzelle- Amazon QLDB

Decentralized Apps and Frameworks: OpenBazaar- DTube- Ocean Protocol

Implementing Business Logic in DApps : Managing user data and state in DApps, Handling transactions and payments-Smart contract development techniques- End-to-end development of a DApp-Case study

Text Books

1. Shahid Shaikh ,”Building Decentralized Blockchain Applications: Learn How to Use Blockchain as the Foundation for Next-Gen Apps”, BPB Publications,2021
2. Siraj Raval,”Decentralized Applications”, O'Reilly Media,2016

Reference Books & web resources

1. Andreas M. Antonopoulos and Dr. Gavin Wood, "Mastering Ethereum Building Smart Contracts and DApps", O'Reilly, 2019
2. <https://www.udemy.com/course/decentralized-app-development-with-blockchain-novice-to-pro>
3. Decentralized Application (DApp) Development from coursera
4. <https://SoliditybyExample.org>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Introduction to Blockchain and Decentralized Network	1
1.2	Types of Blockchain	1
1.3	Consensus mechanisms	1
1.4	Blockchain and Decentralization	1
1.5	The History of Decentralized Applications	
1.6	Working of Decentralized Apps	1
1.7	DApps Protocols	1
2	Smart contracts and Solidity Language	
2.1	Solidity Data Types	1
2.2	Contract Structure- State Variables	1
2.3	Functions	1
2.4	Decentralized Apps Using Ethereum DApps	1
2.5	Ether.js library	1
2.6	Front-end- User Interface (UI) design and development for Decentralized Applications	1
2.7	Web3.js	1
2.8	Smart Contracts (Backend of the App)	1
2.9	Storage	1
2.10	IPFS	1
3	Flourishing Dapp Ecosystem	
3.1	Decentralized Wealth	1

Module No.	Topic	No. of Periods
3.2	Decentralized Identity	1
3.3	Decentralized Computing	1
3.4	Decentralized Bandwidth	1
3.5	Decentralized Markets for Decentralized Assets	1
3.6	Practical Decentralization	1
4	Decentralized Databases	
4.1	OrbitDB - Peer to Peer Distributed Database-	1
4.2	BigchainDB	2
4.3	Ties DB	2
4.4	Bluzelle	1
4.5	Amazon QLDB	1
5	Decentralized Apps and Frameworks	
5.1	OpenBazaar	1
5.2	DTube	1
5.3	Ocean Protocol	1
6.	Implementing Business Logic in DApps	
6.1	Managing user data and state in DApps	1
6.2	Handling transactions and payments	1
6.3	Smart contract development techniques- End-to-end development of a DApp	1
6.4	Case study	1
	Total	36

Course Designer(s):

- | | |
|---|---------------|
| 1. Dr.N.Shivakumar, Assistant Professor | shiva@tce.edu |
| 2. Dr.Raja Lavanya, Assistant Professor | rlit@tce.edu |

22CSRF1 SCALABLE MICROSERVICES

Category L T P Credit

PEES 3 0 0 3

Preamble

The objective of this course is to enhance students with principles of scalable microservice design using the event driven architecture. The students also gain knowledge about the practical implementation details of the design principles using the Spring Boot framework.

Prerequisite

Basic Knowledge of Cloud Computing and Software Engineering

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the strategies and best practices of designing scalable microservices	TPS3	70	80
CO2	Apply the event driven messaging patterns and protocols for efficient exchange of service messages across microservices.	TPS3	70	80
CO3	Develop microservice applications using the event driven and domain driven bounded context principles.	TPS3	70	80
CO4	Apply the event driven microservice data patterns and distributed database concepts to represent the data of the given application.	TPS3	70	80
CO5	Illustrate the microservice design framework for the given scenario using Spring Boot.	TPS3	70	80
CO6	Design and Deploy the given application as a microservice with the illustration of the multi-tier architecture.	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		L	L		M	M	M		M	M	L	M
CO2	S	M	L		L	L		M	M	M		M	M	L	M
CO3	S	M	L	L	M	L		M	M	M		M	M	L	M
CO4	S	M	L	L	M	L		M	M	M		M	M	L	M
CO5	S	M	L	L	M	L		M	M	M		M	M	L	M
CO6	S	M	L	L	M	M	L	M	M	M		M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	5	5	15	-	-	-	30	-	-	-	-	-	-	-	2	3	5	-	-	-
CO2	5	5	30	-	-	-	35	-	-	-	-	-	-	-	2	3	10	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-	2	3	15	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	15	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	15	-	-	-
CO6	-	-	-	-	-	-	-	5	5	30	-	-	-	60	2	3	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Scalable Microservices: Microservice, Event Driven Microservice: Architecture, Deployment, Technology, Autonomy, Resilience, Availability and Scalability. Basics of Containerization and Clustering.

Event-Driven Microservices: The Message Broker, Events and Messages, Asynchronous Messaging, Reactive Systems, Messaging Patterns-Publish-Subscribe, Work Queues, Filtering and Data Durability, Message Broker Protocols, Standards, and Tools, AMQP and RabbitMQ.

Building Event-Driven Microservices: N-Tier Architectures, Durable vs. Ephemeral Message Brokers and GDPR, Event-Driven Message Types, Common Event-Driven Messaging Patterns, Event-Driven Service Topologies, Organizing Event-Driven Microservice Boundaries: Organizational Composition, Domain-Driven Design and Bounded Contexts, Request-Driven vs. Event-Driven Services

Structural Patterns and Chaining process: Transactional consistency in Distributed Systems, Event-driven Orchestration Pattern, Event-driven Choreography pattern, Data Retrieval in Event-driven Architectures and Associated Patterns – Event Sourcing, Command sourcing, CQS and CQRS.

Microservice design using Spring Boot Framework: Setting Up the Development Environment with the Spring Boot Skeleton Web App, Three-Tier, Three-Layer Architecture, Modeling Domain-Driven Design, Business Logic, Presentation Layer, REST APIs with Spring Boot, Automatic Serialization and Testing.

Development and Deployment of Applications: Setting Up the Development Environment with the React Skeleton, A JavaScript Client, Rendering, Integration with the App, Adding Configuration to the Spring Boot App and Deploying the React, The Data Model - Choosing a Database, Spring Boot Data JPA, Entities and Repositories design, Service Layer, Controller Layer and User Interface deployment.

Learning Resources

1. Hugo Filipe Oliveira Rocha, "Practical Event-Driven Microservices Architecture Building Sustainable and Highly Scalable Event-Driven Microservices," Apress, 2022, ISBN-13 (pbk): 978-1-4842-7467-5, ISBN-13 (electronic): 978-1-4842-7468-2, <https://doi.org/10.1007/978-1-4842-7468-2>.
2. Sandeep Jagtap "Domain-Driven Design and Microservices Explained with Examples", Leanpub, 2023.

3. Rajiv Srivastava, "Cloud Native Microservices with Spring and Kubernetes: Design and Build Modern Cloud Native Applications using Spring and Kubernetes", BPB Publications, 2021, ISBN: 9390684315, 9789390684311.
4. Moises Macero Garcia, "Learn Microservices with Spring Boot a Practical Approach to RESTful Services Using an Event-Driven Architecture, Cloud-Native Patterns, and Containerization", Apress, Second Edition, 2020, ISBN-13 (pbk): 978-1-4842-6130-9, ISBN-13 (electronic): 978-1-4842-6131-6, <https://doi.org/10.1007/978-1-4842-6131-6>.
5. Harry Percival and Bob Gregory, "Architecture Patterns with Python Enabling Test-Driven Development, Domain-Driven Design, and Event-Driven Microservices" O'Reilly Media, First Edition, 2020, ISBN: 978-1-492-05220-3.
6. Thomas A. Limoncelli, Strata R. Chalup and Christina J. Hogan, "The Practice of Cloud System Administration DevOps and SRE Practices for Web Services", Volume 2, , Addison Wesley, Second Edition, 2015, ISBN-13: 978-0-321-94318-7, ISBN-10: 0-321-94318-X.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Scalable Microservices	
1.1	Microservice, Event Driven Microservice: Architecture, Deployment, Technology	1
1.2	Autonomy, Resilience, Availability and Scalability	1
1.3	Basics of Containerization	1
1.4	Basics of Clustering	1
2	Event-Driven Microservices	
2.1	The Message Broker, Events and Messages, Asynchronous Messaging, Reactive Systems.	2
2.2	Messaging Patterns-Publish-Subscribe, Work Queues, Filtering and Data Durability, Message Broker Protocols, Standards, and Tools.	2
2.3	AMQP and RabbitMQ.	2
3	Building Event-Driven Microservices	
3.1	N-Tier Architectures, Durable vs. Ephemeral Message Brokers and GDPR, Event-Driven Message Types, Common Event-Driven Messaging Patterns.	2
3.2	Event-Driven Service Topologies, Organizing Event-Driven Microservice Boundaries: Organizational Composition.	1
3.3	Domain-Driven Design and Bounded Contexts	2
3.4	Request-Driven vs. Event-Driven Services	1
4	Structural Patterns and Chaining process	
4.1	Transactional consistency in Distributed Systems	2
4.2	Event-driven Orchestration Pattern, Event-driven Choreography pattern.	2

Module No.	Topic	No. of Periods
4.3	Data Retrieval in Event-driven Architectures and Associated Patterns – Event Sourcing, Command sourcing	2
4.4	CQS and CQRS	2
5	Microservice design using Spring Boot Framework	
5.1	Setting Up the Development Environment with the Spring Boot Skeleton Web App.	1
5.2	Three-Tier and Three-Layer Architecture, Modeling Domain-Driven Design	2
5.3	Business Logic, Presentation Layer	1
5.4	REST APIs with Spring Boot, Automatic Serialization and Testing	2
6	Development and Deployment of Applications	
6.1	Setting Up the Development Environment with the React Skeleton, A JavaScript Client, Rendering, Integration with the App.	2
6.2	Adding Configuration to the Spring Boot App and Deploying the React.	1
6.3	The Data Model - Choosing a Database, Spring Boot Data JPA, Entities and Repositories design	2
6.4	Service Layer, Controller Layer and User Interface deployment	1
	Total	36

Course Designer(s):

1. Dr.G.Madhupriya
Associate Professor
gmadhupriya@tce.edu
2. Dr.K.Indira
Assistant Professor
kiit@tce.edu

22CSRG1**DEVOPS FOR CLOUD**

Category L T P Credit

PEES 3 0 0 3

Preamble

The objective of this course is to enhance students with a deep understanding of efficient service delivery practices during the build and deployment phases of Cloud services. Students will be able to design and deploy highly automated, reliable, and scalable service delivery pipelines that align with modern DevOps and SRE principles. It also covers the principles of DevOps culture and bridge the gap between development and operations for faster, more reliable software releases incorporated with CI/CD pipelines.

Prerequisite

Basic Knowledge of Cloud Computing

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the different service models in cloud computing, the types of machines used in cloud environments and the level of resource sharing in the cloud.	TPS3	70	80
CO2	Adapt the strategic shift from traditional software development approaches into key DevOps methodologies, values, and principles by exploring the practices of agile and Continuous Delivery.	TPS3	70	80
CO3	Illustrate the core values and principles of DevOps focusing on how they foster collaboration, integration, automation, and continuous improvement in software delivery.	TPS3	70	80
CO4	Experiment the modern DevOps practices in the build phase of service delivery and ensure fast and reliable deployment.	TPS3	70	80
CO5	Illustrate the deployment phase of service delivery in DevOps, focusing on efficient and automated strategies for promoting, installing, testing, and managing applications in production.	TPS3	70	80
CO6	Compute the Site Reliability Engineering (SRE) principles in a DevOps framework, using Google Cloud Platform (GCP) for cloud-native development	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		L			M	M	M		M	M		M
CO2	S	M	L		L			M	M	M		M	M		M
CO3	S	M	L		L			M	M	M		M	M		M
CO4	S	M	L	L	L	L		M	M	M		M	M	L	M
CO5	S	M	L	L	L	L		M	M	M		M	M	L	M
CO6	S	M	L	L	M	L	L	M	M	M	L	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	5	5	15	-	-	-	30	-	-	-	-	-	-	-	2	3	5	-	-	-
CO2	5	5	30	-	-	-	35	-	-	-	-	-	-	-	2	3	10	-	-	-
CO3	5	5	25	-	-	-	35	-	-	-	-	-	-	-	2	3	15	-	-	-
CO4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	15	-	-	-
CO5	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	15	-	-	-
CO6	-	-	-	-	-	-	-	5	5	30	-	-	-	60	2	3	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Cloud Service Platform - Level of Service Abstraction: Infrastructure as a Service, Platform as a Service, Software as a Service; Type of Machine: Physical Machines, Virtual Machines, Containers; Level of Resource Sharing: Compliance, Privacy, Cost and Control.

DevOps Strategy - The Traditional Approach vs The DevOps Approach, The Three Ways of DevOps: Workflow, Improve Feedback and Continual Experimentation and Learning; DevOps Evolution and Site Reliability Engineering.

DevOps Values and Principles: Relationships, Integration, Automation, Continuous Improvement, Common Nontechnical DevOps Practices, Common Technical DevOps Practices and Release Engineering DevOps Practices; Agile and Continuous Delivery.

Service Delivery: The Build Phase - Service Delivery Strategies: Pattern: Modern DevOps Methodology and Anti-pattern: Waterfall Methodology, Build-Phase Steps: Develop, Commit, Build, Package and Register, Build Console and Continuous Integration.

Service Delivery: The Deployment Phase - Deployment-Phase Steps: Promotion, Installation and Configuration; Testing and Approval; Operations Console; Infrastructure Automation Strategies: Preparing Physical Machines, Preparing Virtual Machines and Installing OS and Services; Continuous Delivery and Infrastructure as Code.

Site Reliability Engineering (SRE) Practices DevOps using Google Cloud: Basics of SRE, SRE's approach toward DevOps, SRE's: Technical and Cultural practices,

Implementing DevOps using Google Cloud: Cloud native development, CI in GCP, CD in GCP, Continuous monitoring on GCP, CI/CD pipeline in GCP.

Text Book

1. Thomas A. Limoncelli, Strata R. Chalup and Christina J. Hogan, "The Practice of Cloud System Administration DevOps and SRE Practices for WebServices", Volume 2, Addison Wesley, 2015.
2. Mohammed Ilyas Ahmed, "Cloud-Native DevOps: Building Scalable and Reliable Applications", Apress, 2024. ISBN: 9798868804069.
3. Justin Domingus, John Arundel, "Cloud Native DevOps with Kubernetes" O'Reilly Media, Second Edition, 2022, ISBN-13 : 978-1098116828
4. Craig Walls, "Spring Boot in Action", Manning Publications, 2016, ISBN: 1617292540, 9781617292545.

Reference Books & web resources

1. Pro PowerShell for Amazon Web Services: DevOps for the AWS Cloud, Brian Beach, Apress.
2. Hands-On Guide to AgileOps A Guide to Implementing Agile, DevOps, and SRE for Cloud Operations, Navin Sabharwal, Raminder Rathore, Udit Agrawal, Apress, 2022.
3. Google Cloud for DevOps Engineers: A practical guide to SRE and achieving Google's Professional Cloud DevOps Engineer certification, Sandeep Madamanchi, Packt Publishing, 2021.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Cloud Service Platform	
1.1	Level of Service Abstraction: Infrastructure as a Service, Platform as a Service, Software as a Service	2
1.2	Type of Machine: Physical Machines, Virtual Machines	
1.3	Containers-Level of Resource Sharing	2
1.4	Compliance-Privacy, Cost and Control.	1
2	DevOps Strategy	
2.1	The Traditional Approach vs The DevOps Approach	2
2.2	The Three Ways of DevOps	1
2.3	Workflow, Improve Feedback and Continual Experimentation and Learning	1
2.4	DevOps Evolution and Site Reliability Engineering.	1
3	DevOps Values and Principles	
3.1	Relationships, Integration, Automation, Continuous Improvement	1
3.2	Common Nontechnical DevOps Practices, Common Technical DevOps Practices	1

Module No.	Topic	No. of Periods
3.3	Release Engineering DevOps Practices	1
3.4	Continuous Delivery.	1
4	Service Delivery: The Build Phase	
4.1	Service Delivery Strategies: Pattern: Modern DevOps Methodology and Anti-pattern	2
4.2	Waterfall Methodology,	1
4.3	Build-Phase Steps: Develop, Commit, Build, Package	1
4.4	Register, Build Console and Continuous Integration.	1
5	Service Delivery: The Deployment Phase	
5.1	The Deployment Phase - Deployment-Phase Steps: Promotion, Installation and Configuration;	2
5.2	Testing and Approval; Operations Console	2
5.3	Infrastructure Automation Strategies: Preparing Physical Machines	2
5.4	Preparing Virtual Machines and Installing OS and Services	1
5.5	Continuous Delivery and Infrastructure as Code.	1
6	Site Reliability Engineering (SRE) Practices for DevOps using Google Cloud	
6.1	Basics of SRE, SRE's approach toward DevOps	2
6.2	SRE's : Technical and Cultural practices	1
6.3	Implementing DevOps using Google Cloud:	1
6.4	Cloud native development, CI in GCP, CD in GCP,	2
6.5	Continuous monitoring on GCP,	
6.6	CI/CD pipeline in GCP.	2
	Total	36

Course Designer(s):

1. Dr.G.Madhupriya, gmadhupriya@tce.edu
Associate Professor
2. Dr.K.Indira kiit@tce.edu
Assistant Professor

22CSRH1**BIG DATA ENGINEERING**

Category L T P Credit

PEES 3 0 0 3

Preamble

The course aims at facilitating the students to understand the concepts of big data and the need for big data analytics. It focuses on different technologies that are used to store, process and analyse large volume of data. Students will also learn about various analytical methods that can be applied on different types of data to produce meaningful conclusions. The course also introduces the concepts of streaming for building real time data analysis applications.

Prerequisite

Basics of Database Management Systems

Course Outcomes

On the successful completion of the course, students will be able to

CO#	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Recognize the need of a framework to process big data and identify the suitable analytical methods to analyse big data	TPS2	B	75
CO2	Use the Hadoop technologies for storing, processing and analysing huge amount of data.	TPS3	B	70
CO3	Exhibit the use of NoSQL databases for organizing and handling large scale data.	TPS3	B	70
CO4	Perform exploratory data analysis and evaluate Machine Learning models for big data	TPS3	B	70
CO5	Demonstrate the data analysis using spark and develop spark applications.	TPS3	B	70
CO6	Practice appropriate methods and techniques for mining data streams.	TPS3	B	70
CO7	Develop a complete data analytics solution for the given problem scenario.	TPS4	B	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L								L		L	L		L

CO2	S	M	L		L	L		L	M	M		M	M	L	M
CO3	S	M	L		L	L		L	M	M		M	M	L	M
CO4	S	M	L		L	L		L	M	M		M	M	L	M
CO5	S	M	L		L	L		L	M	M		M	M	L	M
CO6	S	M	L		L	L		L	M	M		M	M	L	M
CO7	S	S	M	L	M	M	L	L	M	M	L	M	S	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	15	-																						-	5	-			
CO2	5	20	20												20										2	5	12			
CO3	5	15	15												20										2	5	12			
CO4							5	10	15												20				2	5	12			
CO5							5	10	15												20				2	5	12			
CO6							5	20	15												20				2	5	12			
CO7															30	30					20	20								
Total	15	50	35				15	40	45						70	30					80	20			10	30	60			

Syllabus

Understanding Big Data: Evolution and Definition, Big Data Characteristics, Different types of Data, Challenges with Big Data - Big Data Architecture - big data applications, Traditional Business Intelligence versus Big Data – Data Lake, Architecture, Data lake vs Data Warehouse.

Analytics on Big Data: Defining big data analytics, Classification of Analytics – Challenges in Big data analytics.

Hadoop & Hadoop EcoSystem: Hadoop Architecture, HDFS - NameNode, Secondary NameNode, and DataNode, Hadoop MapReduce paradigm, Map and Reduce tasks, Job, Task trackers, Algorithms Using MapReduce: Matrix-vector multiplication, Matrix multiplication, Relational Algebra operations - Join operations: optimization for join operations, theta join, skew aware join, YARN & Zookeeper, Hadoop Cluster Setup & Hadoop Configuration.

Hadoop Technologies: Pig – Introduction, Parallel processing using Pig, Pig Architecture, Pig data model, Pig Latin- Input and output, Relational operators, User defined functions Hive –

Introduction, Hive Architecture, Hive QL-Data Definition and Data Manipulation, Hive QL queries-Map Reduce Scripts

NoSQL Data Management - Introduction to NoSQL – Sharding and Replication, ACID and BASE Properties, CAP Theorem, Data models – Column Family, Key-Value and Document data models, graph databases, Case studies: HBase – data model and implementations, Redis – Environment and Configuration, Commands, Neo4j – data model, Cassandra - Features, CQL, Key spaces.

Exploratory Data Analysis: Visualizations - Visual data analysis techniques- interaction techniques. **Machine Learning for Big Data:** Feature Engineering, Supervised and Unsupervised model selection and tuning

In memory Processing: Overview of Spark, Hadoop vs. Spark, Resilient Distributed Datasets(RDDs) in Spark, Creating RDD, RDD Operations, and Saving RDD, Data Frames & Spark SQL, Spark Streaming, Spark ML, Writing Spark Application - Spark Programming in Python, Application Execution.

Mining Data Streams: Introduction to Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments, Counting Oneness in a Window, Decaying Window, Real time Analytics Platform(RTAP) Applications - Case Studies - Real Time Sentiment Analysis- Stock Market Predictions

Text Book

1. Jure Leskovec, Anand Rajaraman, Jeffrey D. Ullman, “Mining of Massive Datasets”, Cambridge Universities Press, 3rd edition, 2020.
2. David Loshin, “Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph”, Elsevier, August 23, 2013.

Reference Books

1. Tom White, “Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale”, Orielly, 4th Edition, 2015.
2. Jeffrey Aven, “Data Analytics with Spark Using Python”, Addison-Wesley; 1st edition, 2018.
3. Brij B Gupta, Mamta, “Big Data Management and Analytics”, World Scientific Publishing Company, 2023.
4. Dan Sullivan, “NoSQL for Mere Mortals”, Addison-Wesley, 2015
5. <https://spark.apache.org/docs/latest/api/python/pyspark.html>
6. <https://github.com/adamjshook/mapreducepatterns>
7. <https://spark.apache.org/mllib/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Understanding Big Data (4)	

Module No.	Topic	No. of Periods
1.1	Evolution and Definition, Big Data Characteristics, Different types of Data	1
1.2	Challenges with Big Data, Big Data Architecture - Big Data applications	1
1.3	Traditional Business Intelligence versus Big Data, Data Lake, Architecture, Data lake vs Data Warehouse	1
1.4	Analytics on Big Data: Defining big data analytics, Classification of Analytics – Challenges in Big data analytics.	1
2	Hadoop & Hadoop Ecosystem (5)	
2.1	Hadoop Architecture, HDFS – NameNode, Secondary NameNode, and DataNode	1
2.2	Hadoop MapReduce paradigm, Map and Reduce tasks, Job, Task trackers, Algorithms Using MapReduce: Matrix-vector multiplication, Matrix multiplication	2
2.3	Relational Algebra operations - Join operations: optimization for join operations, theta join, skew aware join	1
2.4	YARN & Zookeeper, Hadoop Cluster Setup & Hadoop Configuration	1
3	Hadoop Technologies(5)	
3.1	Pig – Introduction, Parallel processing using Pig, Pig Architecture, Pig data model	1
3.2	Pig Latin- Input and output, Relational operators, User defined function	2
3.3	Hive – Introduction, Hive Architecture, Hive QL-Data Definition and Data Manipulation, Hive QL queries - Map Reduce Scripts	2
4	NoSQL Data Management (6)	
4.1	Introduction to NoSQL – Sharding and Replication, ACID and BASE Properties, CAP Theorem	1

Module No.	Topic	No. of Periods
4.2	Data models – Column Family, Key-Value and Document data models, graph databases	1
4.3	Case Studies: HBase – data model and implementations	1
4.4	Redis – Environment and Configuration, Commands	1
4.5	Neo4j – data model, Cassandra - Features, CQL, Key spaces	2
5	Exploratory Data Analysis(4)	
5.1	Visualizations - Visual data analysis techniques- interaction techniques	2
5.2	Machine Learning for Big Data: Feature Engineering, Supervised and Unsupervised model selection and tuning	2
6	In memory Processing (6)	
6.1	Overview of Spark, Hadoop vs. Spark	1
6.2	Resilient Distributed Datasets(RDDs) in Spark	1
6.3	Creating RDD, RDD Operations, and Saving RDD	1
6.4	Data Frames & Spark SQL, Spark Streaming, Spark ML	2
6.5	Writing Spark Application - Spark Programming in Python, Application Execution.	1
7	Mining Data Streams (6)	
7.1	Introduction to Streams Concepts – Stream Data Model and Architecture	1
7.2	Stream Computing - Sampling Data in a Stream	1
7.3	Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments	2
7.4	Counting Oneness in a Window, Decaying Window	1
7.5	Real time Analytics Platform(RTAP) Applications - Case Studies - Sentiment Analysis, Stock Market Predictions	1

Module No.	Topic	No. of Periods
	Total	36

Course Designer(s):

1. Dr B.Subbulakshmi, Assistant Professor, Dept. of Computer Science and Engineering bscse@tce.edu
2. Dr. M. Nirmala Devi, Assistant Professor (Selection Grade), Dept. of Computer Science and Engineering mnit@tce.edu

22CSRJ1**FEDERATED LEARNING**

Category L T P Credit

PEES 3 0 0 3

Preamble

This course provides the basic technologies and needs for Federated Learning (FL). The learners will understand different architectures and algorithms in FL and explore privacy preserving techniques in collaborative learning. At the end of the course the learners will be provided practical exposure to real-world federated learning applications.

Prerequisite

A basic knowledge is required in the following areas;

- Distributed Systems
- Deep Learning
- Python Programming

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Infer the fundamental concepts and types of federated learning	TPS2	70	80
CO2	Implement federated learning models using open source tools	TPS3	70	75
CO3	Identify the impact of data heterogeneity on model convergence and accuracy	TPS3	70	75
CO4	Determine security and privacy challenges in FL and its mitigation methods	TPS3	70	75
CO5	Illustrate scalability and resource management challenges of deploying FL on distributed machine learning	TPS3	70	75
CO6	Appraise FL to real-world case studies like personalized FL, cross silo vs. cross device FL	TPS4	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	M	L			L	L		M	M	M		M	M	L	M

CO2.	S	M	L		L	L		M	M	M		M	M	L	M
CO3.	S	M	L		L	L		M	M	M		M	M	L	M
CO4.	S	M	L		L	L		M	M	M		M	M	L	M
CO5.	S	M	L		L	L		M	M	M		M	M	L	M
CO6.	S	S	M	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2		Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	15	15		-	-	-		-	-	-	-	-	-	-		8		-	-	-	
CO2	5	10	20	-	-	-	50	-	-	-	-	-	-	-		4	15	-	-	-	
CO3	5	10	20		-	-	50	-	-	-	-	-	-	-		4	15		-	-	
CO4	-	-	-	-	-	-	-	5	5	20		-	-	30		4	15		-	-	
CO5	-	-	-	-	-	-	-	5	5	20		-	-	30		4	15		-	-	
CO6	-	-	-	-	-	-	-	5	5	10	20	-	-		40-	4	-	12	-	-	

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Federated Learning: Motivation and Need-Centralized vs. Decentralized Learning Paradigm-Taxonomy of FL: Horizontal, Vertical, and Transfer FL-Federated Learning workflow and Terminology-Cases Study: Google Gboard

Federated Learning Architectures and Algorithms: Client-Server Orchestration Model-Aggregation Techniques.-Knowledge Distillation-Handling Non-IID / skewed Data- Case Study: Credit Scoring

Security and Privacy in FL: Threat Models-Differential Privacy-Secure- CCPI, Multi-Party Computation-Homomorphic Encryption-Attack Vectors: Data Poisoning, Model Inversion, -Secure aggregation & federated analytics

System Design and Implementation—Platforms and Frameworks: TensorFlow Federated, PySyft, Flower-Federated Optimization Techniques--Case Study: Apple Siri, Google Gboard

Regulatory & Ethical Dimension-GDPR, HIPAA, RBI guidelines (India) relevance to FL- Fairness and bias in distributed data-Responsible AI frameworks

Applications and Case Studies: Cross silo-Federated Learning in Healthcare, Smart Devices, Finance, IoT-Personalized FL-FL in Edge and Mobile Devices-Open Research Problems

Reference Books & web resources

1. Jeyakrusna Sahoo, Mariya Ouaisa, Akarsh K. Nair "Federated Learning- Principles, Paradigms, and Applications", Taylor & Francis, 2024
2. Kairouz, P., McMahan, H. B., et al. "Advances and Open Problems in Federated Learning", *Foundations and Trends in Machine Learning*, 2021.
3. Li, T., Sahu, A. K., et al. "Federated Learning: Challenges, Methods, and Future Directions", *IEEE Signal Processing Magazine*, 2020.
4. Bonawitz, K. et al. "Towards Federated Learning at Scale: System Design", *SysML 2019*.
5. TensorFlow Federated Documentation – <https://www.tensorflow.org/federated>
6. PySyft and Flower Documentation (OpenMined.org)

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Federated Learning	
1.1	Motivation and Need	1
1.2	Centralized vs. Decentralized Learning paradigm	1
1.3	Taxonomy of FL: Horizontal, Vertical, and Transfer FL	1
1.4	Federated Learning workflow and terminology	1
1.5	Cases Study: Google Gboard	2
2	Federated Learning Architectures and Algorithms	
2.1	.Client Server Orchestration	1
2.2	Aggregation Techniques	1
2.3	Knowledge Distillation	2
2.4	Handling Non-IID / skewed Data	2
2.5	Case Study: Credit Scoring	1
3	Security and Privacy in FL	

Module No.	Topic	No. of Periods
3.1	Threat Models	1
3.2	Differential Privacy-Secure – CCPI, Multi-party Computation-Homomorphic Encryption	3
3.3	Attack Vectors: Data Poisoning, Model Inversion	3
3.4	Secure aggregation & federated analytics	2
4	System Design and Implementation	
4.2	Platforms and Frameworks: TensorFlow Federated, PySyft, Flower	2
4.3	Federated Optimization Techniques	2
4.5	Case Study: Apple Siri, Google Gboard	1
5	Regulatory & Ethical Dimension	
5.1	GDPR, HIPAA, RBI guidelines (India) relevance to FL	2
5.2	Fairness and bias in distributed data	2
5.3	Responsible AI frameworks	2
6	Applications and Case Studies	
6.1	Federated Learning in Healthcare, Smart Devices, Finance, IoT	1
6.2	Personalized FL-Cross Silo	1
6.3	FL in Edge and Mobile Devices- Open Research Problems	1
	Total	36

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, rlsit@tce.edu
Computer Science and Engineering,

22CSRK1**EDGE COMPUTING**

Category L T P Credit

PEES 3 0 0 3

Preamble

This course provides knowledge and abilities on how IoT and Edge Computing are used to meet application demands in Industrial intelligence. The objective of this course is to make students to understand computing and communication technologies used for IoT, Edge and Cloud Computing. This course also includes middleware architecture between Edge & Cloud and adding intelligence at the edge. At the end of the course students should be able to implement Edge-Cloud Systems for an application opportunity.

Prerequisite

- Algorithms
- Data Structure
- Network Programming
- Cloud Computing

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Interpret architectures, entities used in Edge Computing along with its benefits and challenges	TPS2	75	80
CO2	Implement the networking and routing protocols for communication and processing occurs in the Edge to cloud scenario	TPS3	70	75
CO3	Demonstrate the use of smart gateways and collaboration protocol using appropriate middleware infrastructure and lightweight OS	TPS3	70	75
CO4	Apply containerization and orchestration tools to manage edge data system	TPS3	70	75
CO5	Implement the tool for analyse the data at the edge to know the real-time insights and provide resolution to the problems in various domains	TPS3	70	75
CO6	Distinguish physical and logical workspace for the real time edge computing applications Edge-Fog- Cloud environment	TPS4	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L			L			L				L	L		

CO2.	S	M	L		M	M	L	M	L	L		L	M	L	L
CO3.	S	M	L		M	M	L	M	L	L		L	M	L	L
CO4.	S	M	L		M	M	L	M	L	L		L	M	L	L
CO5.	S	M	L		L	L		M	M	M		M	M	L	M
CO6.	S	S	M	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2		Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	10	20		-	-	-		-	-	-	-	-	-	-			8		-	-	-
CO2	5	10	20	-	-	-	50	-	-	-	-	-	-	-			4	15	-	-	-
CO3	5	10	20		-	-	50	-	-	-	-	-	-	-			4	15		-	-
CO4	-	-	-	-	-	-	-	5	5	20		-	-	20			4	15		-	-
CO5	-	-	-	-	-	-	-	5	5	20		-	-	20			4	15		-	-
CO6	-	-	-	-	-	-	-	5	5	15	15	-	-	20	40		4	-	12	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Foundation & Architecture of Edge Computing: Concepts, Characteristics and Attributes-Reference Architecture-Multi View, Concept View, Function View, Deployment View-Multi-access Edge Computing(MEC) reference model-Content Delivery Networks vs. MEC-Benefits-Drawbacks

Edge Networking and Protocols: Networking-5G/LTE-Network Protocols-MQTT, CoAP, WebSocket-5G MEC Architecture-Slicing in 5G. software-defined Clouds-Slicing management in Edge and Fog

Middleware Infrastructures: Cloud-fog-mist-edge Consortium-Mobile Cloud Computing vs Edge Computing, -Smart Gateway-IoT Hub, Event Hub-Collaboration Protocols-Lightweight OS ad frameworks-Case Study: Cisco Packet Tracer-Smart City, factories, traffic

Data management & orchestration: Time Series Database-Streamlining Analytics-Cloud Database-Data Scrapping-Docker container and Cloud native Kubernetes-Digital Twin-Case Study: Node-red-Temperature Monitoring, Prometheus tool

Edge Analytics: Edge Analytics Architecture-Challenges in Centralized IoT-OTA-Remote Diagnostics-Security, Privacy and Trust-Case Study: AWS Greengrass, Mender, Kafka

Edge Computing in Industry Application: Autonomous Vehicles & V2X (Vehicle –to Everything)-Healthcare-Green Edge Computing- Industry 4.0

Reference Books & web resources

1. Rajkumar Buyya, Satish Narayana Srirama, Fog and Edge Computing: Principles and Paradigms, Wiley 2019
2. Perry lea, IoT and Edge Computing for Architects: Implementing Edge and IoT Systems from Sensors to Clouds with Communication Systems, Analytics, and Security, Packet Publishing, 2020, 2nd Edition
3. Fadi Al-Turjman, Edge Computing From Hype to Reality, EAI/Springer Innovations in Communication and Computing, 2019
4. Jie Cao, Quan Zhang, Weisong Shi, Edge Computing: A Primer, SpringerBriefs in Computer Science, 2018
5. Edge Computing Consortium : <https://ecconsortium.eu/>
6. AWS IoT: Developing and Deploying an Internet of Things: <https://courses.edx.org/courses/course-v1:AWS+OTP+AWSD5+1T2019/course/>
7. IoT Networks and Protocols : <https://courses.edx.org/courses/course-v1:CurtinX+IOT3x+3T2020/course/>
8. Packet Tracer: <http://static-pt-assets.s3.amazonaws.com/tutorials71.htm#stub>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
1	Foundation & Architecture of Edge Computing		
1.1	Concepts, Characteristics and Attributes	1	CO1
1.2	Reference Architecture-Multi View, Concept View, Function View, Deployment View	3	CO1
1.3	Multi-access Edge Computing(MEC) reference model	1	CO1
1.4	Content Delivery Networks vs. MEC	1	CO1
2	Edge Networking and Protocols		
2.1	Networking-5G/LTE-	1	CO2
2.2	Network Protocols-MQTT, CoAP, WebSocket	2	CO2
2.3	5G MEC Architecture	1	CO2

2.4	Slicing in 5G. software-defined clouds	2	CO2
2.5	Slicing management in Edge and Fog	1	CO2
3	Middleware Infrastructures		
3.1	Cloud-fog-mist-edge consortium	1	CO3
3.2	Mobile Cloud Computing vs Edge Computing	1	CO3
3.3	Smart Gateway-IoT Hub, Event Hub	1	CO3
3.4	Collaboration Protocols	1	CO3
3.5	Lightweight OS and frameworks	1	CO3
3.6	Case Study: Cisco Packet Tracer-Smart City, factories, traffic	1	CO3
4	Data management & orchestration		
4.1	Time Series database	1	CO4
4.2	Streamlining Analytics	1	CO4
4.3	Cloud Database	1	CO4
4.4	Data Scrapping	1	
4.5	Docker container and Cloud native Kubernetes	2	CO4
4.6	Digital Twin	1	CO4
	Case Study: Node-red – Temperature Monitoring, Prometheus tool	1	CO4
5	Edge Analytics		
5.1	Edge Analytics Architecture	1	CO5
5.2	Challenges in Centralized IoT	1	CO5
5.3	OTA-Remote Diagnostics	1	CO5
5.4	Security, Privacy and Trust	1	CO5
5.5	Case Study: AWS Greengrass, Mender, Kafka	1	CO5
6	Edge Computing in Industry Application		

6.1	Autonomous Vehicles & V2X (Vehicle –to Everything), Digital Twin	1	CO6
6.2	Healthcare	1	CO6
6.3	Green Edge Computing	1	CO6
6.4	Industry 4.0	1	CO6
Total		36	

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, rlsit@tce.edu
Computer Science and Engineering,

22CSRL1

**BUSINESS PROCESS
AUTOMATION**

Category L T P Credit
PEES 2 0 2 3

(Theory cum Practical)

Preamble

Business process automation design and development course offers comprehensive knowledge and professional level skills focused on developing and deploying software robots to achieve automation. The course introduces RPA platform and enables a student to use RPA software to automate business processes.

Prerequisite

- Knowledge of Webapp development

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the operation of basic computing and programming concepts.	TPS2	70	85
CO2	Apply the business and management aspect of process automation.	TPS3	70	80
CO3	Automate business processes using various functionalities and features of RPA tool.	TPS3	70	80
CO4	Identify the process for image, audio, text and data tables automation.	TPS3	70	80
CO5	Deploy software bots independently for process automation.	TPS3	70	80
CO6	Develop software bots for email automation and exception handling.	TPS3	70	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L									L		L		L
CO2	S	M	L	M	S	L	L	L	L	L	M	L	M	M	L
CO3	S	M	L	M	S	L	L	L	L	L	M	L	M	M	L
CO4	S	M	L	M	S	L	L	L	L	L	M	L	M	M	L

CO5	S	M	L	M	S	L	L	L	L	L	M	L	M	M	L
CO6	S	M	L	M	S	L	L	L	L	L	M	L	M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						OCR						Model Lab						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10	-	-	-	-	-	-	-	-	-	-	10	-	-	-	-	-	-	-	-	-	-	2	15	-	-	-	-	-
CO2	10	10	20	-	-	-	-	-	-	-	-	-	-	15	-	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO3	10	10	20	-	-	-	-	-	-	-	-	-	-	20	-	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO4	-	-	-	-	-	-	10	10	10	-	-	-	-	20	-	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO5	-	-	-	-	-	-	5	10	20	-	-	-	-	20	-	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO6	-	-	-	-	-	-	5	10	20	-	-	-	-	15	-	-	-	-	-	-	20	-	-	-	5	10	-	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**ESSENTIALS OF PROGRAMMING FOR AUTOMATION**

Programming Concepts Basics -Python- Basic Web Concepts - Data Structures - Data Tables - Algorithms - Software Processes - .Net Framework - .Net Fundamentals - XML - HTML - CSS - Variable & Arguments.

RPA FOUNDATIONS AND BUSINESS USE CASES

RPA Basics - History of Automation - Programming Constructs in RPA - Types of Bots - Standardization of processes - RPA Development methodologies - RPA business case- Risks & Challenges with RPA - RPA emerging ecosystem, Agentic AI ,Blue Prism, Descript

RPA TOOL BASICS

Introduction to RPA Tool -Number Variables - Array Variables - Date and Time Variables - Data Table Variables-Importing New Namespaces - Control Flow Introduction - Loops - Advanced Control Flow - Sequences - Flowcharts - Data Manipulation Introduction - Scalar Variables, Collections and Tables - mapping of BPA to LLM

ADVANCED AUTOMATION CONCEPT AND TECHNIQUES

Recording and Advanced UI Interaction- Data Scraping –Audio Automation-Video Automation - Scraping Advanced Techniques - Selectors-Advanced Citrix Automation - Introduction Retrieval - Data Tables in BPA-Extracting Data from PDF - Extracting a single piece of data - Anchors - Automation of L1 and L2 jobs

EMAIL AUTOMATION & EXCEPTIONAL HANDLING Email Automation - Incoming Email Automation – Sending Email Automation-Debugging and Exception Handling - Debugging Tools – Strategies for Solving issues – Catching errors.

Text Book

- 1) Tripathi, Alok Mani. "Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool–UiPath". Packt Publishing Ltd, 2018.

Reference Books & web resources

1. Frank Casale, Rebecca Dilla, Heidi Jaynes, Lauren Livingston, "Introduction to robotic process automation: A primer", Institute of robotic process automation, 2015
2. Murdoch, Richard. "Robotic Process Automation: Guide to Building Software Robots, Automate Repetitive Tasks & Become An RPA Consultant." Middletown, DE. Omakustanne (2018).
3. <https://www.uipath.com/rpa/robotic-process-automation>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1.	Essentials of Programming for Automation (4)	
1.1	Programming Concepts Basics, Understanding the application	1
1.2	Basic Web Concepts ,Protocols, Email Clients, Data Structures	1
1.3	Data Tables, Algorithms , Software Processes Software Design, Scripting, Net Framework,.Net Fundamentals	1
1.4	XML - Control Structures and functions HTML - CSS - Variable & Arguments.	1
2	RPA Foundations and Business Use Cases (4)	
2.1	RPA Basics - History of Automation, RPA Vs Automation - Processes & Flowcharts	1
2.3	Programming Constructs in RPA Types of Bots - RPA Advanced Concepts	1
2.4	RPA Development methodologies, RPA business case	1

2.5	RPA Team - Process Design, RPA and emerging ecosystem. Popular tools ,Blue prism ,Descript	1
3.	RPA TOOL INTRODUCTION & BASICS (5)	
3.1	Introduction to RPA Tool - The User Interface - Variables -	1
3.2	Managing Variables - Naming Best Practices - The Variable Panel Managing Arguments - Naming Best Practices	1
3.4	Using Arguments - About Imported Namespaces - Importing New Namespaces -	1
3.5	Control Flow Introduction , Flowcharts - The Assign Activity - The Delay Activity -	1
3.7	Data Manipulation Introduction - Scalar Variables, Collections and Tables - Text Manipulation – Data Manipulation -Gathering and Assembling Data	1
4.	AUTOMATION TECHNIQUES (4)	
4.1	Recording and Advanced UI Interaction - Recording	1
4.2	Introduction - Basic and Desktop Recording - Web Recording - Input/ Output Methods	1
4.3	Screen Scraping - Data Scraping - Scraping Advanced Techniques	1
4.4	Customization - Debugging - Dynamic Selectors - Partial Selectors	1
5	ADVANCED AUTOMATION CONCEPT (4)	
5.1	RPA Challenge - Image, Audio, Video, Text & Advanced Citrix Automation	1
5.2	Introduction Retrieval - Advanced Citrix Automation Challenges -	1
5.3	Best Practices - Using tab for Images - Starting Apps - Excel Data Tablets & PDF - Data Tables in RPA - Excel and Data Tables basics	1
5.4	Data Manipulation in excel - Extracting Data from PDF- Extracting a single piece of data	1

6.	EMAIL AUTOMATION & EXCEPTIONAL HANDLING (3)	
6.1	Email Automation - Email Automation - Incoming Email Automation –	1
6.2	Sending Email automation - Debugging and Exception Handling -	1
6.3	Debugging Tools – Strategies for Solving issues – Catching errors	1
	Total	24

List of Experiments for practical hands on:

Experiment No.	Experiment Title (24 hours)	Course Outcomes
1	Automating File Management (Python)	CO2
2	Web Scraping and CSV Export (Python)	CO2
3	Automated Email Sending (Python)	CO2
4	Reading PDF and Extracting Data (Python)	CO3
5	Desktop Automation Using pyautogui (Python)	CO3
6	Web Form Automation with Selenium (Python)	CO3
7	Hello World Bot & Variable Handling	CO4
8	Web Data Scraping to Excel	CO4
9	Excel-Based Automation	CO5
10	PDF Invoice Extraction	CO5
11	Email Automation and Attachment Download	CO6
12	Exception Handling and Logging	CO6

Course Designer(s):

Dr.K.NarasimhaMallikarjunan, Associate Professor , CSE

arjunkambaraj@tce.edu

22CSR1**SOFTWARE TESTING AND
QUALITY CONTROL**

Category L T P Credit

PEES 3 0 0 3

Preamble

This course encourages students to learn and practice a disciplined approach to software testing and software quality assurance. Students will understand key testing strategies such as black-box and white-box testing, and explore various testing techniques. The course also teaches the basics of Software Quality Assurance, including ethics, quality checks, and standards, to help build reliable and high-quality software.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the key techniques and processes involved in software testing.	TPS2	70	85
CO2	Identify suitable tests to be carried out for given specification	TPS3	70	80
CO3	Design test cases for given software using appropriate test generation methods.	TPS3	70	80
CO4	Construct a test suite to meet the given adequacy criteria involving coverage and mutation score.	TPS3	70	80
CO5	Prepare a test plan for given software specification and develop scripts to automate the testing of a given software using appropriate testing tools	TPS3	70	80
CO6	Apply software quality assurance methods and measurements to check and improve software processes while following ethical rules and industry standards.	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L						L				L	L	-	-
CO2	S	M	L		L	L		M	M	M		M	M	M	M
CO3	S	M	L		L	L		M	M	M		M	M	M	M
CO4	S	M	L		L	L		M	M	M		M	M	M	M
CO5	S	M	L		L	L		M	M	M		M	M	M	M

CO6.	S	M	L		M	M	L	M	M	M	L	M	M	L	M
------	---	---	---	--	---	---	---	---	---	---	---	---	---	---	---

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	20		-	-	-	-	-	-	-	-	-		30											2	3	-	-	-	-
CO2	5	10	25	-	-	-	-	-	-	-	-	-			35										2	3	15	-	-	-
CO3	5	5	25	-	-	-	-	-	-	-	-	-			35										2	3	15	-	-	-
CO4	-	-	-	-	-	-	5	5	20	-	-	-									20				2	3	15	-	-	-
CO5	-	-	-	-	-	-	5	5	25	-	-	-									20				2	3	15	-	-	-
CO6	-	-	-	-	-	-	5	5	25	-	-	-									60				2	3	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Testing as an Engineering Activity, Principles of Testing, V-model concepts, Tester's Role in a Software Development Organization, Origins of Defects, Cost of defects, Defect Classes, The Defect Repository and Test Design, Developer/Tester Support of Developing a Defect Repository, Defect Prevention strategies.

Testing strategies and techniques: Testing strategies, Unit Testing, integration testing, system and acceptance testing, performance testing, regression testing, internationalization testing, ad hoc testing, object oriented testing, Usability and Accessibility Testing.

Black Box Approach to Test Case Design: Random Testing, Requirements based testing, Equivalence partitioning, boundary value analysis, cause effect graphing, state based testing, domain testing

White Box Approach to Test design: Test Adequacy Criteria, static testing vs. structural testing, adequacy criteria based on control flow, principles of mutation testing, equivalent mutants, fault detection using mutation, Test assessment using mutation

Test Management and Automation: People and organizational issues in testing, Organization structures for testing teams, testing services, Test Planning, Test Plan Components, Test Plan Attachments, Locating Test Items, test management, test process, Reporting Test Results-Test Automation: Software test automation, skill needed for automation, scope of automation, design and architecture for automation, requirements for a test tool, challenges in automation

Software Quality Assurance: Ethical Basis for Software Quality — Software Processes and Methodologies- Software Quality Assurance Metrics- Software Configuration Management- CMM (Capability Maturity Model)- Software Engineering Standards.

Text Book

1. Aditya P. Mathur "Foundations of Software Testing", Second Edition ,Pearson Education, 2014.
2. Claude Y. Laporte, Alain April , "Software Quality Assurance" Wiley, January 2018.

Reference Books & web resources

1. Srinivasan Desikan, Gopalaswamy Ramesh, "Software testing – principles and practices", First Edition ,Pearson Education, 2009
2. Roger S. Pressman, Software Engineering A Practitioner's Approach, Seventh Edition, Mcgraw Hill International Edition.,2010

Course Contents and Lecture Schedule

odule No.	Topic	No. of Periods
1	Introduction	
1.1	Testing as an Engineering Activity	1
1.2	Principles of Testing, V-model concepts	1
1.3	Tester's Role in a Software Development Organization	1
1.4	Origins of Defects, Cost of defects, Defect Classes	1
1.5	The Defect Repository and Test Design, Developer/Tester Support of Developing a Defect Repository,	1
1.6	Defect Prevention strategies	1
2	Testing strategies and techniques	
2.1	Testing strategies, Unit Testing, integration testing	1
2.2	System and acceptance testing	1
2.3	Performance testing, regression testing	1
2.4	Internationalization testing, ad hoc testing	1
2.5	Object oriented testing	1
2.6	Usability and Accessibility Testing	1
3	Black Box Approach to Test Case Design	
3.1	Random Testing, Requirements based testing	1
3.2	Equivalence partitioning	1
3.3	Boundary value analysis, cause effect graphing	1
3.4	State based testing	1
3.5	Domain testing	1
4	White Box Approach to Test design	

odule No.	Topic	No. of Periods
4.1	Test Adequacy Criteria, static testing vs. structural testing	1
4.2	Adequacy criteria based on control flow, principles of mutation testing	1
4.3	Equivalent mutants, fault detection using mutation	1
4.4	Test assessment using mutation	1
5	Test Management and Automation	
5.1	People and organizational issues in testing	1
5.2	Organization structures for testing teams, testing services	1
5.3	Test Planning, Test Plan Components	1
5.4	Test Plan Attachments, Locating Test Items	1
5.5	Test management, test process	1
5.6	Reporting Test Results-Test	1
5.7	Automation: Software test automation, skill needed for automation	1
5.8	Scope of automation, design and architecture for automation	1
5.9	Requirements for a test tool, challenges in automation	1
6	Software Quality Assurance	
6.1	Ethical Basis for Software Quality	1
6.2	Software Processes and Methodologies	1
6.3	Software Quality Assurance Metrics	1
6.4	Software Configuration Management	1
6.5	CMM (Capability Maturity Model)	1
6.6	Software Engineering Standards	1
	Total	36

Course Designer(s):

1. Dr.Raja Lavanya, Assistant Professor, rlit@tce.edu
Computer Science and Engineering,
2. Dr.N.Anita, Assistant Professor, naacse@tce.edu
Computer Science and Engineering,

22CSRN1	BEHAVIOURAL AND EMOTIONAL INTELLIGENCE	Category	L	T	P	C
		PEES	3	0	0	3

Preamble

Emotional Artificial Intelligence (EAI) is an emerging field where technology meets human emotions, enabling machines to understand and respond to how people feel. It is used in areas like education, mental health, marketing, and human-computer interaction. This course introduces Emotional AI and basic psychological concepts of emotions, along with hands-on techniques for recognizing emotions from text, voice, and facial expressions. Ethical considerations are also discussed, ensuring students gain the knowledge and practical skills to build responsible, emotion-aware technologies.

Prerequisite

- Basic knowledge of Machine Learning (ML) & Deep Learning (DL), Python programming and Natural Language Processing (NLP) concepts

Course Outcomes

On the successful completion of the course, students will be able to

CO. No	Course Outcomes (COs)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Relate the principles of Emotional AI, affective computing, and emotional theories to real world challenges in everyday environments, including concepts such as personality, perception, and stress.	TPS 3	70	75
CO2	Apply Natural Language Processing and machine learning techniques to perform emotion detection from textual data.	TPS 3	70	75
CO3	Employ speech processing and facial recognition techniques to determine emotional states from audio and visual inputs.	TPS 3	70	75
CO4	Interpret ethical, social, and design principles to produce transparent and responsible emotion-aware AI systems.	TPS 3	70	75

CO5	Analyze the effectiveness of different machine learning and deep learning models for emotion detection across text, speech, and facial modalities.	TPS 4	70	75
CO6	Infer design trade-offs and ethical implications in emotion-aware systems by examining real-world Emotional AI applications and case studies.	TPS 4	70	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L			L	L		L				M	L	L	L
CO2	S	M	L		M	L		M	L	L		M	M	L	L
CO3	S	M	L		M	L		M	L	L		M	M	L	L
CO4	S	M	L		L			M	L	L		M	M		L
CO5	S	M	L		M			M	L	L		M	M	L	L
CO6	S	M	L		M	L		M	L	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						Assignment 1			CAT 2						Assignment 2			Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6
CO1	10	10	10	-	-	-	-	-	20	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO2	10	10	10	-	-	-	-	-	20	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO3	-	20	20	-	-	-	-	-	60	-	-	-	-	-	-	-	-	-	2	5	10	-	-	-
CO4	-	-	-	-	-	-	-	-	-	10	10	10	-	-	-	-	-	10	2	5	10	-	-	-
CO5	-	-	-	-	-	-	-	-	-	10	10	-	15	-	-	-	-	30	2	5	-	10	-	-

CO6	-	-	-	-	-	-	-	-	-	10	10	-	15	-	-	-	-	60	-	5	-	10	-	-
-----	---	---	---	---	---	---	---	---	---	----	----	---	----	---	---	---	---	----	---	---	---	----	---	---

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Emotional AI & Affective Computing - Definition and scope of Emotional AI – Applications - education, mental health, Human Computer Interaction (HCI), marketing, robotics - Introduction to Affective Computing (Picard's work) - Basic concepts from psychology & neuroscience - Emotions vs Feelings vs Moods - Ekman's 6 basic emotions – James Lange theory – Human - centered AI and emotion – Personality - Meaning and development - Major determinants of Personality theories of Personality - Stress - causes and effect of job Stress

Emotion Detection from Text - Natural Language Processing techniques for text preprocessing - Sentiment analysis vs Emotion recognition - Emotion classification using: lexicon-based methods - BERT and GloVe embeddings for emotion detection - Deep Learning models (BiLSTM, Transformers) - Datasets: EmoLex, SemEval, GoEmotions

Emotion Detection from Speech and Facial Expressions - Speech features (pitch, energy, MFCCs) - DL models for Speech Emotion Recognition (SER) – Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) - Facial Emotion Recognition (FER2013 dataset) - Multimodal emotion recognition and fusion techniques - Case study - Fusion of speech and visual cues for emotion detection

Ethical, Social & Design Considerations - Privacy, bias and surveillance concerns - Emotion manipulation and user consent - Fairness and representation in emotion datasets - Ethical design of affective systems - Introduction to Explainable AI (XAI) - XAI for EEG emotion classification using DL

Applications & Real-World Systems - Emotion-aware chatbot and virtual assistant - Case Studies - Emotional AI in education - Replika - An emotionally aware chatbot for mental health support – Human - Robot interaction - Pepper Robot

Text Books

1. Affective Computing (The MIT Press) - Rosalind W. Picard, 1997
2. Interaction Design: Beyond Human-Computer Interaction (Wiley) - Helen Sharp, Jenny Preece, Yvonne Rogers, 2019

Reference Books

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O'Reilly Media, Géron, A, 2022.
2. Encyclopedia Of Human Computer Interaction, Claude Ghaoui, 2006
3. Applied machine learning and AI for engineers: Solve business problems that can't be solved algorithmically. O'Reilly Media, Jeff Prosise, 2022

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	Introduction to Emotional AI & Affective Computing	(7)
1.1	Definition and scope of Emotional AI	1
1.2	Applications - education, mental health, HCI, marketing, robotics	1
1.3	Introduction to Affective Computing (Picard's work)	1
1.4	Basic concepts from psychology & neuroscience Emotions vs Feelings vs Moods, Ekman's 6 basic emotions, James Lange theory	1
1.5	Human - Centered AI and emotion	1
1.6	Personality - Meaning and development - Major determinants of Personality theories of Personality	1
1.7	Stress - causes and effect of job Stress	1
2	Emotion Detection from Text	(6)
2.1	Natural Language Processing techniques for text preprocessing	2
2.2	Sentiment analysis vs. Emotion recognition	2
2.3	Emotion classification using - lexicon-based methods (NRC Emotion Lexicon)	2
3	Machine Learning & Deep Learning methodology	(6)
3.1	BERT and GloVe embeddings for emotion detection	2
3.2	Deep Learning models (BiLSTM, Transformers)	2
3.3	Datasets: EmoLex, SemEval, GoEmotions	2
4	Emotion Detection from Speech and Facial Expressions	(7)
4.1	Speech features: pitch, energy, MFCCs	1
4.2	DL models for speech emotion recognition (SER) – CNNs, RNNs	2
4.3	Facial Emotion Recognition (FER2013 dataset)	1
4.4	Multimodal emotion recognition and fusion techniques	2
4.5	Case study - Fusion of speech and visual cues for emotion detection	1
5	Ethical, Social & Design Considerations	(6)
5.1	Privacy, bias, and surveillance concerns	1
5.2	Emotion manipulation and user consent	1
5.3	Fairness and representation in emotion datasets	1
5.4	Ethical design of affective systems	1
5.5	Introduction to Explainable AI (XAI) - XAI for EEG emotion classification using DL	2
6	Applications & Real-World Systems	(4)
6.1	Emotion-aware chatbots and virtual assistants	1
6.2	Case Studies: Emotional AI in education	1
6.3	Case Studies: Replika – An emotionally aware chatbot for mental health support	1

6.4	Case Studies: Human - Robot interaction - Pepper Robot	1
Total Hours		36

Course Designer(s):

1. Mr. S. Santhana Hari, AP, CSE
2. Mrs. N. Sheerin Sitara, AP,CSE

sshcse@tce.edu
nsscse@tce.edu

22CSR1**SOFTWARE SECURITY**

Category L T P Credit

ENGINEERING

PEES 3 0 0 3

Preamble

This course aims at providing an understanding about the fundamental concepts and properties in software security, security process models and the software security practices. It also focuses on Software Security Knowledge for Architecture and design concepts of Software Security Testing.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the security concepts	TPS2	70	80
CO2	Develop a Security Assurance Case by applying the suitable security properties	TPS3	70	75
CO3	Apply the SQUARE process model to develop and document security requirements using systematic tools and methods.	TPS3	70	75
CO4	Apply elicitation, prioritization, and risk analysis to secure software design.	TPS3	70	75
CO5	Illustrate security principles, guidelines, attack patterns, and testing methods to enhance software architecture and design.	TPS3	70	75
CO6	Examine security failures, attacker perspectives, and system complexity challenges in secure software development.	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L		-								L	L		
CO2	S	M	L	-	-	L	L	L	L	L		L	M	L	L
CO3	S	M	L	-	M	L	L	L	L	L		L	M	L	L
CO4	S	M	L	-	-	L	L	L	L	L		L	M	L	L
CO5	S	M	L	-	-	L	L	M	L	L		L	M	L	L
CO6	S	M	L	-	-	L	L	L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10																							4	5				
CO2	5	5	20										50												4	5	10			
CO3	10	20	20										50													5	10			
CO4							5	10	10											30					4	5	10			
CO5							5	10	20											30					4	5	10			
CO6							10	10	20											40					4	5	10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: System Complexity- Software Assurance and Software Security-Processes and Practices in Software Security-Threats to Software Security -Sources of Software Insecurity-Software requirements in SDLC

Properties of Secure Software: Core Properties of Secure Software-Influential Properties of Secure Software-Influences on properties- Building Security Assurance.

Requirements Engineering: Quality Requirements-Security Requirements Engineering-SQUARE Process Model-Requirements Elicitation-Elicitation Evaluation Criteria-Requirements Prioritization-Identify Candidate Prioritization Methods- Prioritization Technique Comparison-Recommendations for Requirements Prioritization.

Security Practices: Architectural Risk Analysis: Characterization, Threat. Assessment, Determination, Risk. Risk Mitigation Planning. Recapping Architectural Risk Analysis.

Security principles: Security Principles-Security Guidelines, Attack Patterns-Software Security-DevOps and Micro services security- Testing: Contrasting Software Testing and Software Security Testing, Functional Testing, Risk-Based Testing.

Security and Complexity: System Assembly Challenges - Security Failures - Functional and Attacker Perspectives for Security Analysis- System Complexity Drivers and Security Deep Technical Problem Complexity

Text Book

1. Software Security Engineering A Guide for Project Managers by Julia H.Allen, ean J. Barnum, Robert J. Ellison and Gary McGraw, May 11, 2008
2. Mead, Nancy R., and Carol Woody. Cyber security engineering: A practical approach for systems and software assurance. Addison-Wesley Professional, 2016.

Reference Books

1. John Musa D, "Software Reliability Engineering", 2nd Edition, Tata McGraw-Hill, 2005
2. Du, Wenliang. Computer & internet security: a hands-on approach. Independently published, 2022.
3. Anderson, Ross J. Security engineering: a guide to building dependable distributed systems. John Wiley & Sons, 2010.
4. Suhel Ahmad Khan, Rajeev Kumar, Raees Ahmad Khan- Software Security Concepts & Practices – CRC Press, 2023

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction.	
1.1	System Complexity -Software Assurance and Software Security.	1
1.2	Processes and Practices in Software Security. Threats to Software Security	1
1.3	Sources of Software Insecurity-Software Requirements in SDLC	2
2	Properties of Secure Software	

Module No.	Topic	No. of Periods
2.1	Introduction, Defining Properties of Secure Software: Core Properties of Secure Software Software.	2
2.2	Influences on properties: The Defensive Perspective. The Attacker's Perspective.	2
2.3	Building a Security Assurance Case.	2
3	Requirements Engineering:	
3.1	Quality Requirements, Security Requirements Engineering.	2
3.2	SQUARE Process Model: A Brief Description of SQUARE, Tools, Expected results.	2
3.3	Requirements Elicitation	2
3.4	Requirements Prioritization: Identify Candidate Prioritization Methods, Prioritization Technique Comparison, and Recommendations for Requirements Prioritization.	2
4	Security Practices	
4.1	Architectural Risk Analysis: Characterization, Threat. Assessment, Determination, Risk	2
4.2	Risk Mitigation Planning. Recapping Architectural Risk Analysis.	2
5	Security principles:	
5.1	Software Security Knowledge for Architecture and Design Security Principles	2
5.2	Security Guidelines, and Attack Patterns: Security Principles, Security Guidelines, Attack Patterns..	2
5.3	Software Security Testing: Contrasting Software Testing and Software Security Testing, Functional Testing, Risk-Based Testing	2

Module No.	Topic	No. of Periods
6	Security and Complexity	
6.1	System Assembly Challenges - Security Failures	2
6.2	Functional and Attacker Perspectives for Security Analysis: Identity Management: Functional Perspective - Identity Management: Attacker's Perspective - Identity Management and Software Development	2
6.3	System Complexity Drivers and Security	2
6.4	Deep Technical Problem Complexity	2
	Total	36

Course Designers:

1. Dr. C. Senthil Kumar cskcse@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

PROGRAMME ELECTIVES

(TCE MOOC COURSES)

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSRQ1 GRAPH ANALYTICS

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

Graph analytics is a field of data analysis that focuses on studying the relationships between entities as vertices in a graph. This course will enable students to develop Graph analytics algorithms which are used in a wide range of applications, including recommendation systems, fraud detection, network optimization, and social media analysis. Students can also learn and practice the popular graph analytics algorithms and powerful graph databases to work with large-scale graphs and perform graph analytics.

Prerequisite

- knowledge of Data structures and Algorithms, Database systems, linear algebra, probability theory, statistics and Python programming

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Represent graph data using appropriate data structures and visualization techniques	TPS3	70	80
CO2	Implement fundamental graph traversal algorithms to find Path and distance	TPS3	70	80
CO3	Use Graph Technology, embedding and Machine learning to model the Graph	TPS3	70	80
CO4	Transform data in rectangular tables and documents in to graphs database	TPS3	70	80
CO5	Develop clusters to detect communities and Cliques for social and biological network analysis	TPS3	70	80
CO6	Use graph analytics and distributed algorithms for real life problem solving.	TPS3	70	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		M	M	M		M	M	L	M
CO2	S	M	L		M	L		M	M	M		M	M	L	M
CO3	S	M	L		M	L		M	M	M		M	M	L	M
CO4	S	M	L		M	L		M	M	M		M	M	L	M
CO5	S	M	L		M	L		M	M	M		M	M	L	M
CO6	S	M	L		M	L		M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment I						Assignment II						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	10	20												40										2	5	10			
CO2	5	10	10												30											5	10			
CO3	10	10	20												30										2	5	10			
CO4							10	10	20											30					2	5	10			
CO5							5	10	20											30					2	5	10			
CO6							5	10	10											40					2	10	5			
	20	30	50				20	30	50						100					100										

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Graph analysis and Working with Graphs: Graphs a mathematical models, Graph Theory in the analysis: Connection, Information Flow, Community, diversity, familiarity, influence, Graph as a data source, Types of Graphs, Representation of Graphs, Characterizing graphs, random graph models, Visualizing Graphs.

Standard graph algorithms: Graph Traversal, path and Distance, Shortest path algorithm-Dijkstra, Calculating Graph Diameter and density, dynamic graphs, graphs and linear algebra, Graph isomorphism problem

Graph Technology, embedding and Machine learning: Structured Graph, Knowledge Graph and maturity levels, Elastic Graphs, Classification Graph, **Graph embedding:** force-directed layout, deepwalk, node2vec **Machine learning on graph:** Graph neural networks and auto encoder

Graph as databases: Transforming data in rectangular tables for use in Graphs-Creating simple graph, Transforming data from document for use in graphs-Scraping data from semi structured documents and creating edge list.

Social and Biological networks analysis- Components, partitions and Cluster- Connected components of disconnected graphs, Vertex partitioning and clustering, Community detection, Cliques , Assortativity in networks: Categorical or nominal, Degree Assortativity, Vertex and Graph similarity

Graph Analytics algorithms: matching, ordering, and coloring, Pagerank, triangle counting, genome assembly, **Big graph processing:** distributed algorithms, GraphX, Neo4J, Distributed graph algorithm --**Recommendation systems:** Collaborative filtering

Text Book

1. Keith McNulty, "Handbook of Graphs and Networks in People Analytics: With Examples in R and Python", Chapman and Hall/CRC, First Edition, 2022.
2. Estelle Scifo, "Hands-On Graph Analytics with Neo4j: Perform graph processing and visualization techniques using connected data across your enterprise", Packt Publishing; First Edition, 2020.

Reference Books & web resources

1. David A. Bader, "Massive Graph Analytics" (Chapman & Hall/CRC Data Science Series), Taylor and Francis CRC Press; First Edition, 2022
2. T. Ananth Kumar , E. Golden Julie , Y. Harold Robinson, S. M. Jaisakthi, "Simulation and Analysis of Mathematical Methods in Real-Time Engineering Applications", Wiley, First Edition, 2021.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lecture Hours
1	Graph analysis and Working with Graphs:	
1.1	Graphs a mathematical models, Graph Theory in the analysis: Connection, Information Flow, Community, diversity, familiarity, influence	3
1.2	Graph as a data source, Types of Graphs, Representation of Graphs,	1
1.3	Characterizing graphs, random graph models, Visualizing Graphs.	2
2	Standard graph algorithms:	
2.1	Graph Traversal, path and Distance	1
2.2	Shortest path algorithm-Dijkstra	1
2.3	Calculating Graph Diameter and density	1
2.4	Dynamic graphs, graphs and linear algebra and Graph isomorphism problem	1
3	Graph Technology, embedding and Machine learning:	
3.1	Structured Graph, Knowledge Graph and maturity levels, Elastic Graphs, Classification Graph,	3
3.2	Graph embedding: force-directed layout, deepwalk, node2vec	2
3.3	Machine learning on graph: Graph neural networks and autoencoder	2
4	Graph as databases	
4.1	Transforming data in rectangular tables for use in Graphs-Creating simple graph	2
4.2	Transforming data from document for use in graphs-Scraping data from semi structured documents and creating edge list.	3
5	Social and Biological networks analysis-	
5.1	Components, partitions and Cluster- Connected components of disconnected graphs	2
5.2	Vertex partitioning and clustering, Community detection, Cliques ,	2
5.3	Assortativity in networks: Categorical or nominal, Degree Assortativity	2
5.4	Vertex and Graph similarity	
6	Graph Analytics algorithms:	
6.1	matching, ordering, and coloring, Pagerank, triangle counting, genome assembly,	3

6.2	Big graph processing: distributed algorithms, GraphX, Neo4J, Distributed graph algorithm	2
6.3	Recommendation systems: Collaborative filtering	1
	Total Hours	36

Course Designer(s):

1. Dr. M.Nirmala Devi, AP,CSE
2. Dr.B.Subbulakshmi, AP,CSE
3. Dr.M.K.Kavitha Devi, Prof.,CSE

mnit@tce.edu
bscse@tce.edu
mkkdit@tce.edu

22CSRR1**AGILE PROJECT DEVELOPMENT****Category L T P Credit****PEES 3 0 0 3****Preamble**

To impart a thorough understanding of the principles and practices used in agile software development. This course focuses on providing hands-on experience in designing and developing software systems using agile methodologies. Consequently students undertake a project, working through a number of stages of the development of medium size software.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the differences between traditional, Agile and hybrid project management and key principles of agile software development	TPS2	70	80
CO2	Identify overwhelming variety of frameworks associated with Agile, like XP, Scrum, Kanban, Lean.	TPS3	70	75
CO3	Apply Visual Management, information radiators and team management techniques	TPS3	70	75
CO4	Develop user stories from specifications and create product roadmap	TPS3	70	75
CO5	Utilize the estimates and requirements and propose a plan for Release planning	TPS3	70	75
CO6	Create backlogs and burn-down charts to monitor progress of the project	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	M	L				M	M	M	M	M	M	M	M	M	M
CO2	S	M	L			M	M	M	M	M	M	M	M	M	M
CO3	S	M	L		L	M	M	M	M	M	M	M	M	M	M
CO4	S	M	L		L	M	M	M	M	M	M	M	M	M	M
CO5	S	M	L		L	M	M	M	M	M	M	M	M	M	M
CO6	S	M	L		L	M	M	M	M	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1	Assig	CAT2	Assig	Terminal
----	------	-------	------	-------	----------

							nment-1							nment-2						
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	5	5													2	5				
CO2	10	10	25				50								2	5	12			
CO3	10	10	25				50								2	5	12			
CO4								-	10	20					2	5	12			
CO5								5	10	20				50	2	5	12			
CO6								5	10	20				50	-	5	12			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

UNDERSTANDING AGILITY

Introducing Agile Project Management-How agile projects work-Agile Project Management Is Becoming Agile- Product Management, Differences between managing a project versus developing a product , Why agile product development works better-Appling the Agile Manifesto and Principles, Understanding the Agile Manifesto , Outlining the Four Values of the Agile Manifesto- Agile Principles.

AGILE WORKS BETTER AND AGILE BEHAVIOURS

How Agile Approaches Beat Historical Approaches- Agility Is About Being Customer Focused-Agile Approaches, Understanding Kanban, An overview of lean-An overview of scrum. An overview of extreme programming - Agile Behaviors in Action- Establishing Product owner, team member. Scrum master- Stakeholders- Agile mentor-Establishing New Values, Changing Team Philosophy, Dedicated team. Cross-functionality, Self-organization, Self-management, Size-limited teams.

AGILE PLANNING AND EXECUTION.

Defining the Product Vision and Product Roadmap. Agile Planning. Progressive elaboration. Inspect and adapt. Defining the Product Vision. Creating a Product Roadmap. Completing the Product Backlog.

PLANNING RELEASES AND SPRINTS

Refining Requirements and Estimates. What is a user story? Steps to create a user story. Breaking down requirements. Estimation poker. Affinity estimating. Release Planning. Preparing for Release. Preparing the product for deployment. Prepare for operational support. Preparing the marketplace .Sprint Planning. The sprint backlog. The sprint planning meeting.

SCRUM LIFECYCLE

Sprint Planning, Daily Scrum Sprint Review, Sprint Retrospective Working throughout the Day. Planning Your Day: The Daily Scrum. Tracking Progress. The sprint backlog. The task board. Agile Roles in the Sprint. The Sprint Review. Preparing to demonstrate. The sprint review meeting. Collecting feedback in the sprint review meeting. The Sprint Retrospective . Planning for retrospectives. The retrospective meeting. Inspecting and adapting.

Text Book

1. Mark C. Layton , Steven J. Ostermiller , Dean J. Kynaston, "Agile Project Management For Dummies", 3rd Edition 2020, John Wiley & Sons,
2. Greg Caldwell, "Agile Project Management The Complete Guide for Beginners to Scrum, Agile Project Management, and Software Development" 2021 Alakai Publishing LLC

Reference Books & web resources

1. Mark C. Layton, Steven J Ostermiller, and Dean J. Kynaston "Agile Project Management" 3rd Edition 2017, [Wiley](#) publishers.
2. Eric Lean, "Agile Project Management The Complete Beginner's Guide to Learn Project Management Step by Step" 202, 1Kdx Publishing Limited

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	UNDERSTANDING AGILITY	
1.1	Introducing Agile Project Management	1
1.2	How agile projects work-Agile Project Management Is Becoming Agile	1
1.3	Product Management,	1
1.4	Differences between managing a project versus developing a product	1
1.5	Why agile product development works better	1
1.6	Applying the Agile Manifesto and Principles,	1
1.7	Understanding the Agile Manifesto , Outlining the Four Values of the Agile Manifesto-	1
1.8	Agile Principles	1
2	AGILE WORKS BETTER AND AGILE BEHAVIOURS	
2.1	How Agile Approaches Beat Historical Approaches- Agility Is about Being Customer Focused-Agile Approaches	1
2.2	Understanding Kanban, An overview of lean-An overview of scrum. An overview of extreme programming	2
2.3	Agile Behaviors in Action- Establishing Product owner, team member.	1
2.4	Scrum master- Stakeholders- Agile mentor- Establishing New Values,	2
2.5	Changing Team Philosophy, Dedicated team. Cross-functionality	1
2.6	Self-organization, Self-management ,Size-limited teams	1

Module No.	Topic	No. of Periods
3	AGILE PLANNING AND EXECUTION	
3.1	Defining the Product Vision and Product Roadmap	1
3.2	Agile Planning- Progressive elaboration. Inspect and adapt	2
3.3	Defining the Product Vision-Creating a Product Roadmap	2
3.4	Completing the Product Backlog	1
4	PLANNING RELEASES AND SPRINTS	
4.1	Refining Requirements and Estimates. What is a user story- Steps to create a user story.	1
4.2	Breaking down requirements- Estimation poker. Affinity estimating	1
4.3	Release Planning-Preparing for Release. Preparing the product for deployment .	2
4.4	Prepare for operational support. Preparing the marketplace .	1
4.5	Sprint Planning-The sprint backlog,The sprint planning meeting	2
5	SCRUM LIFECYCLE	
5.1	Sprint Planning, Daily Scrum Sprint Review	1
5.2	Sprint Retrospective Working throughout the Day-Planning Your Day: The Daily Scrum . Tracking Progress	1
5.3	The sprint backlog. The task board. Agile Roles in the Sprint	1
5.4	The Sprint Review. Preparing to demonstrate. The sprint review meeting. Collecting feedback in the sprint review meeting.	2
5.5	The Sprint Retrospective. Planning for retrospectives. The retrospective meeting. Inspecting and adapting	2
	Total hours	36

Course Designer(s):

- | | |
|---|----------------|
| 1. Dr.N.Shivakumar
Assistant Professor,CSE | shiva@tce.edu |
| 2. Mrs.Raja Lavanya,
Assistant Professor,CSE | rlit@tce.edu |
| 3. Mrs.G.Bhavani,
Assistant Professor,CSE | gbicse@tce.edu |

22CSRS1	CYBER SECURITY AND CYBER LAW	Category	L	T	P	Credit
		PEES	3	0	0	3

Preamble

This course aims at providing an understanding of the cyberspace, cybercrime, cyber laws and intellectual property issues in cyberspace.

Prerequisite

Data Communication and Networks

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the cybercrime scenario and challenges.	TPS2	70	80
CO2	Illustrate various cybercrimes, the motivation behind them and the overall global perspective of cybercrime.	TPS3	70	75
CO3	Demonstrate the types of cybercrime with respect to the motivations and methods of operation of offenders.	TPS3	70	75
CO4	Identify appropriate law enforcement strategies.	TPS3	70	75
CO5	Apply diverse viewpoints to ethical issues in the cyberspace.	TPS3	70	75
CO6	Interpret the Intellectual property issues in the cyberspace	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L		-								L	L		
CO2	S	M	L	-	-	L	L	L	L	L		L	M	L	L
CO3	S	M	L	-	M	L	L	L	L	L		L	M	L	L
CO4	S	M	L	-	-	L	L	L	L	L		L	M	L	L
CO5	S	M	L	-	-	L	L	M	L	L		L	M	L	L
CO6	S	M	L	-	-	L	L	L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
TP Scale																														
CO1	10	10																							2	10				
CO2	10	10	10												50										2	5	10			
CO3	10	20	20												50										4	5	10			
CO4							10		10												30				4	5	10			
CO5							10	10	20												30				4	5	10			
CO6							10	20	10												40				4		10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Cyberspace, Cybersecurity, Nature of threats, Enabling peoples, Cybercriminals, Cybersecurity Policy, Current cybercrime scenario and challenges. Classification of Cybercrimes.

Cyber Crime: How Criminals plan them, Categories of Cyber Crimes, How Criminal Plans the Attack: Active Attacks, Passive Attacks, Social Engineering, Classification of Social Engineering, Cyber Stalking: types of Stalkers, Cyber Cafe and Cyber Crimes, Botnets, Attack Vectors, Cyber Crime and Cloud Computing, E-mail, Spoofing, Spamming, Cyber Defamation, Internet Time Theft, Salami Attack, Salami technique Data Diddling, Forgery, Web Jacking, Newsgroup Spam, Industrial Spying, Hacking, Online Frauds, Software Piracy, Computer Sabotage Email Bombing, Computer Network Intrusion, Password Sniffing, Credit Card Frauds, Identity Theft.

Tools and Methods used in Cyber Crime: Proxy server and Anonymizers, Phishing: How Phishing works? How password cracking works? Keynoters and Spywares, Virus and Worms, Trojan Horses and Backdoors, Dos and DDoS Attacks, SQL Injection, Buffer Overflow.

Legal Perspectives: The Legal Perspectives, Cyber Crime Indian Perspectives, The Cyber Crime and Indian ITA 2000/2001, Hacking and Indian Laws, Global Perspective on Cyber Crime, Cyber Crime and extended Enterprise, Cyber Crime and Punishment.

Computer Ethics, Privacy & Legislation

Computer ethics, moral and legal issues, descriptive and normative claims, Professional Ethics, code of ethics and professional conduct. Privacy, Computers and privacy issue, Digital Evidence Controls, Evidence Handling Procedures, Basics of Indian Evidence ACT, Negotiable Instruments Act 1881(Amended), IPC and CrPC, Electronic Communication Privacy ACT, Legal Policies, legislative background, Computer contracts, liability for

defective hardware and software, software contracts, web and hardware contracts, electronic contracts and torts, liabilities.

Intellectual property issues in cyberspace

Introduction to intellectual property, WIPO, computer software copyrights, copyright in databases and electronic publishing, the law of confidence, product designs, international law, Copyright, Trade Secrets, Trademarks, Patents, Design, protection of intellectual property, Protection options – Encryption, copyright on web-content, copyright on software. Ethical Decision Making: Types of ethical choices, making defensible decisions

Text Book

1. Abraham R. Wagner, Nicholas Rostow, "Cybersecurity and Cyberlaw", 2021
2. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole, Sunit Belapur, 2012, Wiley
3. James X. Dempsey, Cybersecurity Law Fundamentals, 2021
4. Carrie E. Cope, Dirk E. Ehlers, Keith W. Mandell, Cyber Risks, Social Media and Insurance: A Guide to Risk Assessment and Management, 2022
5. Robert P. Merges, Peter S. Menell, Mark A. Lemley, "Intellectual Property in New Technological Age", 2016.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction.	
1.1	Cyberspace, Cyber Security, Nature of threats	1
1.2	Enabling peoples, Cybercriminals, Cybersecurity Policy, Current cybercrime scenario	1
1.3	Challenges, Classification of Cybercrimes	2
2	Cyber Crime	
2.1	How Criminals plan them, Categories of Cyber Crimes	1
2.2	How Criminal Plans the Attack: Active Attacks, Passive Attacks	1
2.3	Social Engineering, Classification of Social Engineering	1
2.4	Cyber Stalking: types of Stalkers, Cyber Cafe and Cyber Crimes	1
2.5	Botnets, Attack Vectors, Cyber Crime and Cloud Computing, E-mail, Spoofing, Spamming	1
2.6	Cyber Defamation, Internet Time Theft, Salami Attack, Salami technique Data Diddling	1
2.7	Forgery, Web Jacking, Newsgroup Spam, Industrial Spying, Hacking, Online Frauds, Software Piracy	1

Module No.	Topic	No. of Periods
2.8	Computer Sabotage Email Bombing, Computer Network Intrusion, Password Sniffing, Credit Card Frauds, Identity Theft.	1
3	Tools and Methods used in Cyber Crime	
3.1	Proxy server and Anonymizers, Phishing: How Phishing works?	1
3.2	How password cracking works?	1
3.3	Keynoters and Spywares, Virus and Worms	1
3.4	Trojan Horses and Backdoors.	1
3.5	DoS and DDoS Attacks	1
3.6	SQL Injection, Buffer Overflow	1
4	Legal Perspectives	
4.1	The Legal Perspectives, Cyber Crime Indian Perspectives	1
4.2	The Cyber Crime and Indian ITA 2000/2001	1
4.3	Hacking and Indian Laws, Global Perspective on Cyber Crime.	1
4.4	Cyber Crime and extended Enterprise	1
4.5	Cyber Crime and Punishment	1
5	Computer Ethics, Privacy & Legislation	
5.1	Computer ethics, moral and legal issues, descriptive and normative claims	1
5.2	Professional Ethics, code of ethics and professional conduct. Privacy, Computers and privacy issue	1
5.3	Digital Evidence Controls, Evidence Handling Procedures, Basics of Indian Evidence ACT, Negotiable Instruments Act 1881(Amended)	1
5.4	IPC and CrPC, Electronic Communication Privacy ACT, Legal Policies, legislative background,	1
5.5	Computer contracts, liability for defective hardware and software, software contracts	1
5.6	web and hardware contracts, electronic contracts and torts, liabilities.	1
6	Intellectual property issues in cyberspace	
6.1	Introduction to intellectual property	1

Module No.	Topic	No. of Periods
6.2	WIPO, computer software copyrights, copyright in databases and electronic publishing	1
6.3	the law of confidence, product designs,	1
6.4	international law, Copyright, Trade Secrets, Trademarks,	1
6.5	Patents, Design, protection of intellectual property, Protection options	1
6.6	Encryption, copyright on web-content, copyright on software.	1
6.7	Ethical Decision Making: Types of ethical choices, making defensible decisions	1
	Total	36

Course Designers:

- | | | |
|----|----------------------|--|
| 1. | Dr. C. Senthil Kumar | cskcse@tce.edu |
| 2. | Dr. M. Vijayalakshmi | mviji@tce.edu |
| 3. | Dr.J.Dharani | jdice@tce.edu |

22CSRT1**CYBER PHYSICAL SYSTEM**

Category

L

T

P

Credit

PEES

3

0 0

3

Preamble

This course gives students the basic knowledge to the design and analysis of cyber physical systems. It also explores the cyber physical system components and their behaviour in the design of autonomous systems. The case study application shows the significance of the cyber physical system in all disciplines.

Prerequisite

- Basic knowledge of physical systems, behaviour modelling, state transition, computation methods and network communication.

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the basic process of cyber physical systems.	TPS 2	70	80
CO2	Identify the physical and cyber components for the given problem scenario.	TPS 3	70	75
CO3	Model the physical system for the given components based on their behaviour and interaction.	TPS 3	70	75
CO4	Identify the semantics of concurrent composition of components.	TPS 3	70	75
CO5	Simulate the design of the system with concurrency in hardware and software components.	TPS 3	70	75
CO6	Estimate the desired and undesired behaviour of the designed system specification.	TPS 3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L						M	M	M		L	L		L
CO2	S	M	L					M	M	M		L	M		L
CO 3	S	M	L		M	L		M	M	M		L	M	L	L
CO 4	S	M	L					M	M	M		L	M		L
CO	S	M	L		M	L		M	M	M		L	M	L	L

5															
CO 6	S	M	L		L			M	M	M		L	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO 1	5	10	-	-	-	-	-	-	-	-	-	-	-	10	-	-	-	-	-	-	-	-	-	-	2	6	-	-	-	-
CO 2	5	10	20	-	-	-	-	-	-	-	-	-	-	15	30	-	-	-	-	-	-	-	-	-	2	6	10	-	-	-
CO 3	10	10	30	-	-	-	-	-	-	-	-	-	-	15	30	-	-	-	-	-	-	-	-	-	2	6	10	-	-	-
CO 4	-	-	-	-	-	-	10	10	15	-	-	-	-	-	-	-	-	-	-	5	5	-	-	-	2	6	10	-	-	-
CO 5	-	-	-	-	-	-	5	10	15	-	-	-	-	-	-	-	-	-	-	15	30	-	-	-	2	6	10	-	-	-
CO 6	-	-	-	-	-	-	10	10	15	-	-	-	-	-	-	-	-	-	-	15	30	-	-	-	2	6	12	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Basics of Cyber Physical Systems: Features of Cyber Physical Systems, Components of CPS - Computation, Communication and Control, The CPS Process steps -Modelling, Design and Analysis.

CPS Models: Synchronous models- Reactive Components, Properties of Components, Composing Components, Safety Requirements, Asynchronous models - Asynchronous Processes, Components and Coordination Protocols, Liveness Requirements.

Modelling Dynamic Behaviours: Continuous Dynamics – Newtonian Mechanics, Actor models, Properties and Feedback Control, Discrete Dynamics – Finite State Machine, Extended State Machine, Non determinism, Hybrid Systems – Modal Models, Classes of Hybrid Systems, Composition of State Machines – Concurrent Composition, Hierarchical State Machine.

Concurrent Models of Computation: Structure of Models, Synchronous-Reactive Models, Dataflow Models of Computation, Timed Models of Computation.

Design: Sensors and Actuators – Models and Common Sensors, Embedded Processors – Types of Processors and Parallelism, Memory Architectures – Memory Hierarchy and Models, Input and Output – I/O Hardware and Sequential Software, Scheduling.

Analysis: Reachability analysis and model checking- Open Closed systems, Abstraction, model checking liveness property, Quantitative analysis, Applications of CPS: Health and Smart Vehicles.

Text Books

1. Edward Ashford Lee and Sanjit Arunkumar Seshia, "Introduction to Embedded Systems - A Cyber-Physical Systems Approach", Second Edition, Massachusetts Institute of Technology (MIT) Press, 2017.
2. Rajeev Alur, "Principles of Cyber-Physical Systems", Massachusetts Institute of Technology (MIT) Press, 2015.

Reference Books & web resources

1. Raj Rajkumar Dionisio de Niz Mark Klein, "Cyber-Physical Systems", Addition Wesley, Pearson Education 2017.
2. Danda B. Rawat Joel J.P.C. Rodrigues Ivan Stojmenovic, "Cyber-Physical Systems From Theory to Practice", CRC Press, 2016 by Taylor & Francis.
3. <https://www.coursera.org/learn/cyber-physical-systems>
4. https://onlinecourses.nptel.ac.in/noc23_cs62/preview

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Basics of Cyber Physical Systems	
1.1	Features of Cyber Physical Systems	1
1.2	Components of CPS - Computation, Communication and Control	1
1.3	The CPS Process steps -Modelling, Design and Analysis	2
2	CPS Models	
2.1	Synchronous models - Reactive Components, Properties of Components	1
2.2	Composing Components, Safety Requirements	2
2.3	Asynchronous models - Asynchronous Processes	1
2.4	Components and Coordination Protocols	2
2.5	Liveness Requirements	1
3	Modelling Dynamic Behaviors	
3.1	Continuous Dynamics – Newtonian Mechanics, Actor models, Properties and Feedback Control	2
3.2	Discrete Dynamics – Finite State Machine, Extended State Machine, Non determinism	2
3.3	Hybrid Systems – Modal Models, Classes of Hybrid Systems	2
3.4	Composition of State Machines – Concurrent Composition, Hierarchical State Machine	2

Module No.	Topic	No. of Periods
4	Concurrent Models of Computation	
4.1	Structure of Models	1
4.2	Synchronous-Reactive Models	1
4.3	Dataflow Models of Computation	1
4.4	Timed Models of Computation	1
5	Design	
5.1	Sensors and Actuators – Models and Common Sensors	1
5.2	Embedded Processors – Types of Processors and Parallelism	2
5.3	Memory Architectures – Memory Hierarchy and Models	1
5.4	Input and Output – I/O Hardware and Sequential Software	1
5.5	Scheduling	2
6	Analysis	
6.1	Reachability analysis and model checking- Open Closed systems	1
6.2	Abstraction, model checking liveness property	1
6.3	Quantitative analysis	2
6.4	Applications of CPS: Health and Smart Vehicles	2
	Total	36

Course Designer(s):

1. Dr.G.Madhupriya, gmadhupriya@tce.edu
Associate Professor, CSE
2. Mr.D.Nagendrakumar, dnkcse@tce.edu
Assistant Professor, CSE

22CSRUI**QUANTUM COMPUTING**

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

This course introduces the fundamental concepts and principles of quantum computing, quantum algorithms, quantum information theory and applications of quantum computing to machine learning tasks.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the fundamental concepts and implications of quantum computing	TPS2	70	80
CO2	Develop quantum algorithms for a variety of problems	TPS3	70	75
CO3	Implement quantum machine learning algorithms with quantum simulators	TPS3	70	75
CO4	Illustrate how quantum information theory helps to understand the communication and processing of quantum information	TPS3	70	75
CO5	Examine the principles of quantum error correction and its applications in quantum computing.	TPS3	70	75
CO6	Use the potential applications of quantum computing in machine learning	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L			M	L		M	L	L		M	M	M	L

CO 2	S	S	M	L	M	L		M	L	L		M	S	M	L
CO 3	S	M	L		M	L		M	L	L		M	M	M	L
CO 4	S	M	L		M	L		M	L	L		M	M	M	L
CO 5	S	M	L		L			M	L	L		M	M	M	L
CO 6	S	M	L		L			M	L	L		M	M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass 2	Terminal					
TP S Sca le	1	2	3	4	5	6	3	1	2	3	4	5	6	3	1	2	3	4	5	6
CO 1	5	10	20	-	-	-	30	-	-	-	-	-	-	-	2	3	12	-	-	-
CO 2	5	10	20	-	-	-	35	-	-	-	-	-	-	-	2	3	12	-	-	-
CO 3	5	5	20	-	-	-	35	-	-	-	-	-	-	-	2	3	12	-	-	-
CO 4	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	12	-	-	-
CO 5	-	-	-	-	-	-	-	5	5	20	-	-	-	20	2	3	12	-	-	-
CO 6	-	-	-	-	-	-	-	5	5	30	-	-	-	60	2	3	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Quantum Computing: Motivation - Turning Machine - Overview of quantum computing- classical and quantum computing - Qubits and quantum states - Quantum gates and circuits – Representations of Qubits – Superpositions.

Quantum Algorithms: Introduction - Deutsch's algorithm - The Deutsch–Jozsa algorithm – Shor's Algorithm - Grover's Algorithm.

Quantum Machine Learning: Overview - Quantum computing and classical machine learning - Potential benefits and challenges of quantum machine learning.

Quantum Information Theory: Quantum states and measurements - Entanglement and quantum teleportation - Quantum communication- Quantum cryptography.

Quantum Error Correction: Quantum error correction codes - Fault-tolerant quantum computation.

Text Book

1. Riley T. Perry, "The Temple of Quantum Computing", Riley T. Perry, 2004.
2. Michael A. Nielsen, Issac L. Chuang, "Quantum Computation and Quantum Information", Tenth Edition, Cambridge University Press, 2010.
3. Chris Bernhardt, "Quantum Computing for Everyone", The MIT Press, Reprint edition, 8 September 2020.

Reference Books & web resources

1. Qiskit Textbook: Quantum Computing with Qiskit
2. Scott Aaronson, "Quantum Computing Since Democritus", Cambridge University Press, 2013.
3. N. David Mermin, "Quantum Computer Science: An Introduction", Cambridge University Press, 2007.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Quantum Computing	
1.1	Motivation	1
1.2	Turning Machine	1
1.3	Overview of quantum computing	1
1.4	Classical and Quantum computing	1
1.5	Qubits and quantum states	2
1.6	Quantum gates and circuits	2
1.7	Representations of Qubits	1

Module No.	Topic	No. of Periods
1.8	Superpositions	1
2	Quantum Algorithms	
2.1	Introduction	1
2.2	Deutsch's algorithm	2
2.3	The Deutsch–Jozsa algorithm	2
2.4	Shor's Algorithm	2
2.5	Grover's Algorithm	2
3	Quantum Machine Learning	
3.1	Overview	1
3.2	Quantum computing and classical machine learning	2
3.3	Potential benefits and challenges of quantum machine learning	2
4	Quantum Information Theory	
4.1	Quantum states and measurements	2
4.2	Entanglement and quantum teleportation	2
4.3	Quantum communication	2
4.4	Quantum cryptography	2
5	Quantum Error Correction	
5.1	Quantum error correction codes	2
5.2	Fault-tolerant quantum computation	2
	Total	36

Course Designer(s):

1. Dr.M.Vijiayalakshmi, Professor ,CSE
2. Dr.C.Santhiya, Assistant Professor, CSE
3. Dr.K.Indira,Assistant Professor,CSE

mviji@tce.edu
csit@tce.edu
kiit@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E./B.Tech DEGREE PROGRAMME

ELECTIVE COURSE FOR MINOR

**STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSQA1	AI FOR ENGINEERS	Category	L	T	P	C
		PEES	3	0	0	3

Preamble

This syllabus on AI for Engineers is intended for students without a background in computer science and provides an introduction to the concepts and applications of Artificial Intelligence (AI). The syllabus emphasizes on the basics of AI, including Machine Learning, Neural Networks and provides an overview of the technology's capabilities and limitations. The modules in the syllabus reflect an approach to problem-solving via AI technologies, and the syllabus focuses on the strategies, and techniques in developing AI applications. The goal is to equip students with the knowledge and skills

Prerequisite

- Fundamentals of python programming language

Course Outcomes

On the successful completion of the course, students will be able to

CO. No	Course Outcomes (COs)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the basic concepts of Artificial Intelligence and show how they transform engineering practices using Machine Learning and Deep Learning	TPS 2	70	80
CO2	Develop Machine Learning solutions by determining appropriate projects, computing visual insights, and preparing CRISP-DM-based workflows using Python.	TPS 3	70	75
CO3	Illustrate fundamental AI algorithms such as Neural Networks and Convolutional Neural Networks, including tuning parameters, applying filters, and performing object detection.	TPS 3	70	75
CO4	Apply AI techniques to address problems in sectors such as healthcare, agriculture, transportation, smart manufacturing and demonstrate solutions to common deployment challenges like overfitting and generalization	TPS 3	70	75
CO5	Interpret AI concepts to examine ethical issues, social impact, adversarial risks, job displacement, and bias in models	TPS 3	70	75
CO6	Construct AI-driven approaches to solve engineering case studies focused on manufacturing, traffic management, and sentiment analysis.	TPS 3	70	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L			L	L		L				M
CO2	S	M	L		M	L		M	L	L		M
CO3	S	M	L		M	L		M	L	L		M
CO4	S	M	L		L			M	L	L		M
CO5	S	M	L		L			M	L	L		M
CO6	S	M	L		M	L		M	L	L		M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						Assignment 1			CAT 2						Assignment 2			Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6
CO1	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	10	15	20	-	-	-	-	-	40	-	-	-	-	-	-	-	-	-	2	5	12	-	-	-
CO3	10	15	20	-	-	-	-	-	60	-	-	-	-	-	-	-	-	-	2	5	12	-	-	-
CO4	-	-	-	-	-	-	-	-	-	5	10	15	-	-	-	-	-	30	2	5	12	-	-	-
CO5	-	-	-	-	-	-	-	-	-	5	10	15	-	-	-	-	-	40	2	5	12	-	-	-
CO6	-	-	-	-	-	-	-	-	-	5	10	25	-	-	-	-	-	30	-	5	12	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction - Basic Artificial Intelligence terminologies - Scope and limitations of AI - Explanation of Machine learning and Deep learning - Examples of AI.

Understanding AI Workflows - Machine Learning project lifecycle - Approaches for processing and visualizing data - Choosing AI project - Technical tools for AI - Use of Python in AI – CRISP DM Framework and Methodology.

Fundamental Algorithms of AI - Introduction to Neural Networks (NN) - Convolutional Neural Network (CNN) - Parameters tuning - Convolution function - Different types of filters and activation function - Object detection frameworks

Building AI Applications - Survey of major AI application areas such as healthcare, agriculture, transportation, and smart manufacturing – understanding real-world use cases (Spam Email Detection) and the impact of AI across industries - Common pitfalls in AI deployment, including challenges related to data quality, overfitting, generalization issues.

AI and Society - Societal impact of AI - Discrimination and Bias in AI models - Adversarial Attacks - Role of AI in developing economies - Ethical considerations - Impact of AI on jobs and the future of work.

Case Studies - Predictive Maintenance in Manufacturing - Smart Traffic Management System - AI for Power Demand Prediction - Sentiment Analyzer for Product Reviews - Self-driving cars

Text Books

1. AI for Everyone? Critical Perspectives, University of Westminster Press, Pieter Verdegem, 2021 – 1st Edition
2. The Hundred-Page Machine Learning Book, Andriy Burkov, 2019 – 1st Edition

Reference Books

1. Artificial intelligence: A modern approach, Russell, S. J., & Norvig, P, Pearson, 2020 – 4th Edition
2. Artificial Intelligence: The Basics, Kevin Warwick, Routledge, 2012 – 1st Edition

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	Introduction	(6)
1.1	Basic AI terminology	2
1.2	Scope and limitations of AI – Explanation of Machine Learning and Deep Learning.	2
1.3	Machine Learning and Deep Learning examples	2
2	Understanding AI Workflows	(6)
2.1	Machine Learning project lifecycle	1
2.2	Approaches for processing and visualize data	1
2.3	Choose an AI project	1
2.4	Technical tools for AI	1
2.5	Use of python in AI related projects	1
2.6	CRISP DM Framework and Methodology	1
3	Fundamental Algorithms of AI	(6)
3.1	Introduction to Neural Networks (NN)	1
3.2	Introduction to Convolutional Neural Network (CNN)	1
3.3	NN vs CNN - Various Components of CNN	1
3.4	Parameters tuning, Convolution function	1
3.5	Different types of filters and activation function	1
3.6	Object detection frameworks.	1
4	Building AI Applications	(6)
4.1	Major AI application areas such as healthcare, agriculture, transportation, and smart manufacturing	2
4.2	Real world use cases (Spam Email Detection) and the impact of AI	2
4.3	Common pitfalls in AI deployment, including challenges related to data quality, overfitting, generalization issues	2
5	AI and Society	(6)
5.1	The societal impact of AI	1
5.2	Discrimination and Bias in AI models	1
5.3	Adversarial attacks on AI	1
5.4	AI in developing economies - Ethical considerations	1
5.5	Impact of AI on jobs and the future of work	2
6	Case Studies	(6)
6.1	Predictive Maintenance in Manufacturing	1

6.2	Smart Traffic Management System	1
6.3	AI for Power Demand Prediction	1
6.4	Sentiment Analyzer for product reviews	1
6.5	Self-driving cars	2
Total Hours		36

Course Designer(s):

1. Dr. K. Sundarakantham, Professor, CSE
2. Mr. S.Santhana Hari, Assistant Professor, CSE

kskcse@tce.edu
sshcse@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

PROGRAMME ELECTIVES

(TCE MOOC COURSES)

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2025 - 2026 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSR1**STATISTICAL LEARNING**

Category L T P Credit

PEES 3 0 0 3

Preamble

This course focuses on developing a theoretical understanding of the statistical properties of learning algorithms. Learners will explore model estimation, evaluation, and selection methods, along with advanced approaches such as regularization, ensemble learning, and Bayesian inference. Students will be equipped to design, analyse, and evaluate predictive models across diverse data-driven applications.

Prerequisite

Knowledge on Machine Learning, Linear Algebra, Calculus, Probability and Statistics

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the fundamental concepts of statistical learning, including learning paradigms, model evaluation, optimization basics, model complexity, and generalization control techniques.	TPS2	70	75
CO2	Use core regression and classification models, together with resampling and model-selection methods, to build and evaluate reliable predictive solutions.	TPS3	70	70
CO3	Utilize model-selection, regularization, and sparsity-based techniques to effectively manage high-dimensional data and produce interpretable predictive models.	TPS3	70	70
CO4	Employ flexible and non-linear modelling approaches to represent complex patterns using splines, additive models, local methods, and multi-instance learning techniques	TPS3	70	70
CO5	Implement ensemble methods, kernel techniques, and margin-based classifiers—such as decision trees, random forests, boosting, and SVMs to develop basic predictive models	TPS3	70	70
CO6	Practice Bayesian inference, sampling techniques, and unsupervised learning methods to model uncertainty, uncover structure in data, and interpret probabilistic patterns.	TPS3	70	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PS O1	PS O2	PSO 3
CO1	M	L					L	L	L		M	L		L
CO2	S	M	L	L	L	L	L	L	L	M	M	M	L	L
CO3	S	M	L		L	L	L	L	L	L	M	M	L	L
CO4	S	M	L		L	L	L	L	L	L	M	M	L	L
CO5	S	M	L	L	L	L	L	L	L	M	M	M	L	L
CO6	S	M	L		L	L	L	L	L	L	M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	15											2	5				
CO2	5	15	20										2	5	12			
CO3	5	15	20										2	5	12			
CO4							5	10	15				2	5	12			
CO5							5	10	15				-	5	12			
CO6							5	15	20				2	5	12			

CO	Assignment 1						Assignment 2					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1												
CO2			60									
CO3			40									
CO4									30			
CO5									30			
CO6									40			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**Foundations of Statistical Learning**

Introduction to Statistical Learning - Learning with complete data – Estimation, Prediction, Generalization, Supervised vs Unsupervised, Regression vs Classification – Model and Task-specific metrics – Bias–Variance Trade-off – Underfitting vs Overfitting, Learning parameters and optimization methods – objective functions, loss functions, Gradient Descent, Stochastic Gradient Descent, convergence criteria, Metrics for Optimal Parameter selection – validation curves, learning curves, regularization tuning, early stopping. Basics of Statistical Learning Theory -

Overview of PAC Learning and VC dimension, Model Complexity and Generalization Control, Role of Regularization and Sparsity for simple models.

Core Statistical Models

Simple and Multiple Linear Regression – Logistic Regression – Generative Models for Classification - Linear and Quadratic Discriminant Analysis (LDA, QDA) – Naïve Bayes - Generalized Linear Model – Linear and Poisson Regression - Resampling Methods - Cross-Validation, Bootstrap and Model selection criteria - AIC, BIC, DIC, Clustering Methods - k-means clustering, hierarchical clustering.

Regularization, Sparsity & High Dimensional Learning

Subset Selection Methods - Best Subset and Stepwise Selection, Sparse Models & Sparsity-Inducing Methods – Ridge, LASSO, Elastic Net, Feature Selection via Sparsity, High Dimensional Learning – sparsity vs overfitting, Dimension Reduction Methods – Principal Components Analysis, Factor models, Multi-Task Learning – shared representations, Joint sparsity across tasks, Advanced Topics in Regularization – Sparse inverse covariance estimation

Flexible and Non-linear Modelling

Polynomial Regression – Basis Functions – Regression Splines – Smoothing Splines – Local Regression (LOESS) – Generalized Additive Models (GAMs) for Regression and Classification, Multi-instance Learning – Bag-level vs instance level labels, Sparse Additive Models, Optimization considerations in Nonlinear models.

Ensemble Models and Margin-Based Learning

Decision Trees – Bagging, Random Forests, Boosting and Bayesian Additive Regression Trees - Support Vector Machines – Maximum Margin Classifier – Kernel Trick and dual representation, Margin Theory – Generalization via margins, Relationship to VC dimension, Kernel Methods, Adaptive Game-Heuristic Modelling, Multi-agent learning and adversarial optimization

Bayesian Learning, Sampling & Probabilistic Models

Bayesian Inference: Prior, Likelihood, Posterior, MAP Estimation – Overview of Markov Chain Monte Carlo (MCMC), Monte Carlo estimation and uncertainty quantification, Hidden Markov Models (HMMs), Gaussian Mixture Models

Text Book

1. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Jonathan Taylor, "An Introduction to Statistical Learning with Applications in Python", Springer Cham, First Edition, 2023.

Reference Books & web resources

1. Johannes Lederer, "A First Course in Statistical Learning With Data Examples and Python Code", Springer Nature Switzerland, First Edition, 2025.
2. Tushar Gulati, "Core Concepts in Statistical Learning", Educohack Press, 2025.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Foundations of Statistical Learning	

Module No.	Topic	No. of Periods
1.1	Introduction to Statistical Learning - Estimation, Prediction, Generalization	1
1.2	Supervised vs Unsupervised- Regression vs Classification	1
1.3	Model and Task-specific metrics-Bias–Variance Trade-off, Underfitting vs Overfitting	1
1.4	Learning parameters and optimization methods-objective functions, loss functions, Gradient Descent, Stochastic Gradient Descent, Convergence criteria,	1
1.5	Metrics for Optimal Parameter selection, validation curves, learning curves, regularization tuning, early stopping	1
1.6	Basics of Statistical Learning Theory - Overview of PAC Learning and VC dimension,	2
1.7	Model Complexity and Generalization Control, Role of Regularization and Sparsity for simple models.	1
2	Core Statistical Models	
2.1	Simple and Multiple Linear Regression, Logistic Regression	1
2.2	Generative Models for Classification - Linear and Quadratic Discriminant Analysis (LDA, QDA)	1
2.3	Naïve Bayes, Generalized Linear Model - Linear and Poisson Regression	1
2.4	Resampling Methods, Cross-Validation and Bootstrap	1
2.5	Model selection criteria - AIC, BIC, DIC	1
2.6	Clustering Methods - k-means clustering, hierarchical clustering.	1
3	Regularization, Sparsity & High Dimensional Learning	
3.1	Subset Selection Methods: Best Subset, and Stepwise Selection	1
3.2	Sparse Models & Sparsity-Inducing Methods – Ridge, LASSO, Elastic Net, Feature Selection via Sparsity	1
3.3	High Dimensional Learning – sparsity vs overfitting, Dimension Reduction Methods - Principal Components Analysis, Factor models	2
3.4	Multi-Task Learning – shared representations, Joint sparsity across tasks	1
3.5	Advanced Topics in Regularization – Sparse inverse covariance estimation	1
4	Flexible and Non-linear Modelling	

Module No.	Topic	No. of Periods
4.1	Polynomial Regression – Basis Functions – Regression Splines, Smoothing Splines – Local Regression(LOESS)	2
4.2	Generalized Additive Models (GAMs) for Regression and Classification	1
4.3	Multi-instance Learning – Bag-level vs instance level labels	1
4.4	Sparse Additive Models, Optimization considerations in Nonlinear models	1
5	Ensemble Models and Margin-Based Learning	
5.1	Decision Trees – Bagging, Random Forests, Boosting and Bayesian Additive Regression Trees	2
5.2	Support Vector Machines – Maximum Margin Classifier - kernel trick and dual representation	1
5.3	Margin Theory – Generalization via margins, Relationship to VC dimension, Kernel Methods	1
5.4	Adaptive Game-Heuristic Modelling, Multi-agent learning and adversarial optimization	2
6	Bayesian Learning, Sampling & Probabilistic Models	
6.1	Bayesian Inference: Prior, Likelihood, Posterior, MAP Estimation	1
6.2	Overview of Markov Chain Monte Carlo (MCMC)	1
6.3	Monte Carlo estimation and uncertainty quantification	1
6.4	Hidden Markov Models (HMMs), Gaussian Mixture Models	2
	Total	36

Course Designer(s):

1. Dr.B.Subbhulakshmi, bscse@tce.edu
2. Mrs.S.Jeniba, sjacse@tce.edu

22CSRW1**GENERATIVE AI FOR
SOFTWARE TESTING**

Category L T P Credit

PEES 3 0 0 3

Preamble

This course provides an introduction to the methodologies, and practical applications of Generative AI for enhancing software testing processes. It focuses on how AI models can automate test case generation, bug detection, and quality improvement. Students will learn to apply AI tools to enhance efficiency, accuracy, and reliability in software development.

Prerequisite

Basics of Software engineering

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe test driven development principles and role of generative AI in software testing and development.	TPS2	70	80
CO2	Use prompt engineering principles and utilize different LLMs for code review, debugging, and documentation to enhance testing workflows.	TPS3	70	75
CO3	Adapt AI-assisted tools and LLMs to validate and improve code functionality through automated and intelligent testing techniques.	TPS3	70	75
CO4	Demonstrate effective test plans, generate test cases and automate documentation processes using AI models and prompt-driven test strategies.	TPS3	70	75
CO5	Use AI techniques to manage synthetic test data and perform exploratory, analytical, and transformational testing processes.	TPS3	70	75
CO6	Illustrate Code LLMs for software quality assurance through automated code analysis, testing, and security validation to improve overall software reliability.	TPS3	70	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	M	L												
CO2	S	M	L		M	M	M	M	M		M	M	L	M

CO3	S	M	L		M	M	M	M	M		M	M	L	M
CO4	S	M	L		M	M	M	M	M		M	M	L	M
CO5	S	M	L	L	M	M	M	M	M		M	M	L	M
CO6	S	M	L	L	M	M	M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	20		-	-	-	-	-	-	-	-	-													2	5				
CO2	5	10	20	-	-	-	-	-	-	-	-	-			50										2	5	15			
CO3	5	10	25	-	-	-	-	-	-	-	-	-			50										2		15			
CO4	-	-	-	-	-	-	5	5	20	-	-	-									20				2	5	15			
CO5	-	-	-	-	-	-	5	5	25	-	-	-									20				2	5	10			
CO6	-	-	-	-	-	-	5	5	25	-	-	-									60						15			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Test Driven Development- Principles of Test-Driven Development (TDD) , Role of Generative AI in enhancing the TDD process, Testing types: functional, non-functional, test plans, test cases, and reports, Generative AI in Code and Software Development - Code generation- AI tools like GitHub Copilot, OpenAI Codex, AI-assisted debugging, Ethical and responsible AI use in testing environments

Large Language Models and Prompt Engineering: Overview of LLM basics and architectures- LLMs in software engineering (ChatGPT, Claude, Gemini) - Risks and limitations of LLMs in testing and development- Role of LLMs in improving test accuracy and coverage -Principles of prompt engineering- Working with various LLMS for code review, debugging, and documentation

AI-assisted testing: Testing Tools (CodeceptJS, Selenium) - Usage of LLMs in testing-pairing with LLMs: Analyzing ideas, Analyzing code-Using LLMs to generate and refine test cases - Automating unit and integration test creation-Enhancing test coverage using AI-generated edge cases- Self-Script Generation Using Large Language Models

Test Planning with AI support: Overview of test planning in modern testing -Focused prompts with use of Models-Combining models and LLMs to assist test planning-LLMs and Test Cases- Unit testing - Automating test documentation and traceability matrices with AI

Data creation using AI: Generating and transforming data with LLMs- Prompting LLMs to generate simple data sets- Transforming test data with LLMs-Processing complex data with LLM- Exploratory testing with AI prompts

Software Quality Assurance with Code LLMs: Introduction to Software quality assurance - Role of Code LLMs in SQA -Code quality evaluation using AI -Static and dynamic analysis using LLMs- Code Correctness-Code security

Text Book

1. Mark Winteringham, "Software Testing with Generative AI", Manning, 2024.
2. Aditya P. Mathur "Foundations of Software Testing", Second Edition, Pearson Education, 2014.
3. Srinivasan Desikan, Gopalaswamy Ramesh, "Software testing – principles and practices", First Edition, Pearson Education, 2009.

Reference Books & web resources

1. <https://www.coursera.org/learn/generative-ai-course-in-software-testing-and-documentation>
2. <https://www.udemy.com/course/generative-ai-in-software-automation-testing/?couponCode=MT251103G2>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Test Driven Development- Principles of Test-Driven Development (TDD)	1
1.2	Role of Generative AI in enhancing the TDD process	1
1.3	Testing types: functional, non-functional, test plans, test cases, and reports	1
1.4	Generative AI in Code and Software Development	1
1.5	Code generation- AI tools like GitHub Copilot, OpenAI Codex, AI-assisted debugging	1
1.6	Ethical and responsible AI use in testing environments	1
2	Large Language Models and Prompt Engineering	
2.1	Overview of LLM basics and architectures	1
2.2	LLMs in software engineering (ChatGPT, Claude, Gemini, etc.)	2
2.3	Risks and limitations of LLMs in testing and development	1
2.4	Role of LLMs in improving test accuracy and coverage	1
2.5	Principles of prompt engineering	1
2.6	Working with various LLMS for code review	1
2.7	Debugging and documentation	1
3	AI-assisted testing	
3.1	Testing Tools (CodeceptJS, Selenium)	1
3.2	Usage of LLMs in testing - pairing with LLMs	1

3.3	Analyzing ideas, Analyzing code Using LLMs to generate and refine test cases	1
3.4	Automating unit and integration test creation	1
3.5	Enhancing test coverage using AI-generated edge cases- Self-Script Generation Using Large Language Models	1
4	Test Planning with AI support	
4.1	Overview of test planning in modern testing	1
4.2	Focused prompts with use of models	1
4.3	Combining models and LLMs to assist test planning	2
4.4	LLMs and Test cases	1
4.5	Unit testing	1
4.6	Automating test documentation and traceability matrices with AI	1
5	Data creation using AI	
5.1	Generating and transforming data with LLMs	1
5.2	Prompting LLMs to generate simple data sets	1
5.3	Transforming test data with LLMs	1
5.4	Processing complex data with LLM	1
5.5	Exploratory testing with AI prompts	1
6	Software Quality Assurance with Code LLMs	
6.1	Introduction to Software quality assurance	1
6.2	Role of Code LLMs in SQA	1
6.3	Code quality evaluation using AI	1
6.4	Static and dynamic analysis using LLMs	1
6.5	Code correctness , Code security	1
	Total	36

Course Designer(s):

- 1.Dr.S.Prasanna, Associate Professor, Department of CSE, sprcse@tce.edu
- 2.Dr. Raja Lavanya, Assistant Professor, Department of CSE, rlit@tce.edu
- 3.Dr.N.Anita, Assistant Professor, Department of CSE, naacse@tce.edu

22CSRY1 PRIVACY- PRESERVING MACHINE LEARNING

Category L T P Credit
PEES 3 0 0 3

Preamble

Traditional machine learning models often require access to large datasets for training, and these datasets may contain sensitive or personally identifiable information. Privacy-preserving machine learning is motivated by the desire to harness the power of machine learning while safeguarding individuals' sensitive information.

Prerequisite

A Basic course on Machine Learning

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the fundamental concepts of Differential Privacy	TPS3	70	80
CO2	Apply the differential privacy techniques in various machine learning algorithms	TPS3	70	80
CO3	Illustrate various secure computation protocols for machine learning.	TPS3	70	80
CO4	Apply the secure computation protocols in machine learning algorithms.	TPS3	70	80
CO5	Illustrate the protocols for computing on encrypted data	TPS3	70	80
CO6	Analyze the homomorphic encryption protocols for machine learning classification algorithms	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1.	S	M	L		L	M	M	L	M	M		M	L	L
CO2.	S	M	L		L	L	L	M	M	M		M	L	M
CO3	S	M	L		L	L	L	M	M	M		M	L	M
CO4	S	M	L		L	L	L	M	M	M	L	M	L	M
CO5	S	M	L		L	L	L	M	M	M		M	L	M
CO6	S	S	M	L	M	S	M	L	M	M	L	M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass2		Terminal					
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	5	10	20	-	-	-	30	-	-	-	-	-	-	-		-	5	10	-	-	-
CO2	5	10	20	-	-	-	30	-	-	-	-	-	-	-		2	5	10	-	-	-
CO3	-	10	20	-	-	-	40	-	-	-	-	-	-	-		2	5	10	-	-	-
CO4	-	-	-	-	-	-	-	5	10	15	-	-	-	20		2	5	10	-	-	-
CO5	-	-	-	-	-	-	-	5	10	15	-	-	-	20		2	5	10	-	-	-
CO6	-	-	-	-	-	-	-	-	10	10	20	-	-	40	20	2	-	-	15	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Privacy - Definition - Security Threats against Individual Privacy - Differential Privacy for machine learning – Privacy-preserving data analysis – The model of computation – Formalizing Differential Privacy – Basic Techniques and Composition Theorems – Laplace mechanism.

Applying differential privacy in machine learning - Output perturbation – Differentially private supervised learning – Differentially private linear regression – Differentially private logistic regression – Differentially private unsupervised learning – Differentially private K-means clustering.

Secure Computation – Secret sharing – Additive secret sharing – Threshold secret sharing - Polynomials over fields – Shamir secret sharing.

Secure Federated Learning – Federated Learning Fundamentals – Secure aggregation motivation – Secure aggregation preliminaries – Masking – Handling dropped users - Practical secure aggregation protocol.

Computation on Encrypted Data - Homomorphic Encryption schemes - Quadratic Residuosity Cryptosystem – Fully homomorphic encryption scheme.

Machine learning classification over encrypted data – Private hyperplane decision – Polynomial form of a decision tree – Private evaluation of a polynomial - Private decision tree protocol - Privacy Preserving Deep Learning - Application of Privacy in Social Networks.

Text Books

1. J. Morris Chang, Di Zhuang, G. Dumindu Samaraweera, Wilko Henecka, "Privacy-Preserving Machine Learning", Manning publishers, March 2023.
2. Kwangjo Kim and Harry Chandra Tanuwidjaja, "Privacy-Preserving Deep Learning A Comprehensive Survey", Springer 2021.

Reference Books & web resources

1. Dwork and Roth, "The Algorithmic Foundations of Differential Privacy", Now publishers, Foundations and Trends in Theoretical Computer Science, 2014.
2. Ronald Cramer, Ivan Damgaard and Jesper Buus Nielsen, "Secure Multiparty Computation and Secret Sharing - An Information Theoretic Approach", Cambridge Press, 2015.
3. Bonawitz, Keith, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. "Practical secure aggregation for privacy-preserving machine learning." In proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1175-1191. 2017.
4. Bost, Raphael, Raluca Ada Popa, Stephen Tu, and Shafi Goldwasser. "Machine learning classification over encrypted data." Cryptology ePrint Archive, 2014.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Differential Privacy for machine learning	
1.1	Privacy - Definition - Security Threats against Individual Privacy	1
1.2	Differential Privacy - Privacy-preserving data analysis - The model of computation	1
1.3	Formalizing Differential Privacy	1
1.4	Basic Techniques and Composition Theorems	1
1.5	Laplace mechanism	1
2	Applying differential privacy in machine learning	
2.1	Output perturbation	1
2.2	Differentially private linear regression	2
2.3	Differentially private logistic regression	2
2.4	Differentially private K-means clustering	2
3	Secure Computation	
3.1	Secret sharing	1
3.2	Additive secret sharing	1
3.3	Threshold secret sharing	1
3.4	Polynomials over fields	1

Module No.	Topic	No. of Periods
3.5	Shamir secret sharing	2
4	Secure Federated Learning	
4.1	Federated Learning Fundamentals	1
4.2	Secure aggregation motivation	1
4.3	Secure aggregation preliminaries	1
4.4	Masking and Handling dropped users	2
4.5	Practical secure aggregation protocol	2
5	Computation on Encrypted Data	
5.1	Homomorphic Encryption schemes	1
5.2	Quadratic Residuosity Cryptosystem	2
5.3	Fully homomorphic encryption scheme	2
6	Machine learning classification over encrypted data	
6.1	Private hyperplane decision	1
6.2	Polynomial form of a decision tree	1
6.3	Private evaluation of a polynomial	1
6.4	Private decision tree protocol	1
6.5	Privacy Preserving Deep Learning	1
6.5	Application of Privacy in Social Networks	1
	Total	36

Course Designer(s):

1. Dharani J, Assistant Professor, CSE jdicse@tce.edu
2. Ramkumar M.P, Associate Professor, CSE ramkumar@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E./B.Tech DEGREE PROGRAMME

ELECTIVE COURSE FOR MINOR

**STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

**22CSQB1 EXPLORATORY DATA
ANALYSIS AND VISUALIZATION**

Category	L	T	P	Credit
ME	3	0	0	3

Preamble

Data Analysis refers to all the tools and activities involved in processing data to develop valuable insights and interpretations. Data Visualization deals with picturing the interpretations to develop trends and conclusions. In Data Visualization, information is organized into charts, plots, infographics, and animations. This course will enable students to develop visual displays of information, communicate complex data relationships and data-driven insights in a way which is easy to understand.

Prerequisite

- Basic Knowledge of linear algebra, probability theory and statistics

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Practice Python libraries for Data Science using Numpy, Pandas, matplotlib , Seaborn	TPS3	B	70
CO2	Perform quantitative data analysis using descriptive and inferential statistics	TPS3	B	70
CO3	Use Regression Analysis, Classification, Clustering, Association Analysis, and Natural Language Processing for qualitative Data Analysis	TPS3	B	70
CO4	Derive conclusions from data exploration using appropriate charts and plots	TPS3	B	70
CO5	Create interactive dashboards to gain meaningful insights from data visualization	TPS3	B	70
CO6	Implement effective data storytelling to make better business decisions using Data Visualization	TPS3	B	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L		M	L		M	M	M		M	M	L	M
CO 2	S	M	L		M	L		M	M	M		M	M	L	M
CO 3	S	M	L		M	L		M	M	M		M	M	L	M

CO 4	S	M	L		M	L		M	M	M		M	M	L	M
CO 5	S	M	L		M	L		M	M	M		M	M	L	M
CO 6	S	M	L		M	L		M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment I						Assignment II						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	10	20												40										2	5	10			
CO2	5	10	10												30											5	10			
CO3	10	10	20						10						30										2	5	10			
CO4							10	10	20												30				2	5	10			
CO5							5	10	10												30				2	5	10			
CO6							5	10	10												40				2	5	10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Fundamentals of Data Analysis and Visualization-Python for Data Science: Data preparation and cleaning: handling missing data, outliers, and noisy data using Numpy, Pandas, Sklearn, understanding data types and data representation: categorical, hierarchical, relational, temporal & spatial, Dimensional modelling, data Transformation and Integration, Visualising data using matplotlib, seaborn, plotly, GGPlot2- Julia - high-performance scientific, mathematical, AI and data applications.

Quantitative Data Analysis-Basics of Linear Algebra, Probability and Statistics, Descriptive Analysis: Handling Numeric data, Patterns and Trends, Frequencies, Percentages, and Measures of central tendency, Mean, Mode, and Median, Inferential Analysis: Differences and correlations between data sets, ANOVA, t-Tests, and Chi-Square, Hypothesis testing.

Qualitative Data Analysis-Handling Non-numeric data such as audio, video recordings, images, texts, and transcripts. Regression: Ridge regression, LASSO. Classification: K-NN, Bayes, Logistic Regression, Artificial Neural networks (ANNs), Deep NNs, CNN, overfitting, regularization, Clustering: k-means, expectation-maximization **Case studies:** Text Analytics, Pattern Recognition, Business Intelligence: Product placement and pricing recommendation, time-series data visualization and forecasting.

Data Visualization- Object Oriented Interface in Matplotlib 2-D: bar charts, bubble charts, polar charts, Range chart, Box-and-whisker plots, univariate scatter plots, histograms, word cloud, pie chart, waffle chart, stacked bar chart, tree map. 3-D: surfaces, contours, hidden

surfaces, pm3d colouring, 3D mapping: multi-dimensional data visualization: multimodality of data-OLAP Operations and interactive visualization techniques, manifold visualization, graph data visualization, Annotation

Visualization & Tableau - Introduction to BI, connecting Data Sources-Tableau Generated Fields-Data Manipulation & Customization-Using Parameters & adding Statistics, Pareto Chart, Bump Chart, Maps, Gantt Chart, Butterfly Chart, Donut Chart

Creating Dashboards & Stories - Dashboard Design Approach: Healthcare Quality Dashboard, Airline Quality Dashboard, Manufacturing Quality Dashboard. Types of Visualization: Multivariate visualization, Geospatial visualization, Interactive visualization, Critiques in data visualization, Data visualization with ethics.

Text Books

1. Bharti Motwani, "Data Analytics using Python", 1st Edition, Wiley Publications, 2020
2. Wes McKinney, "Python for Data Analysis", 3rd Edition, O'Reilly Publications, 2022.
3. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow", 3rd Edition, O'Reilly Publications, 2023.

Reference Books & web resources

1. Andy Kirk, Data Visualization A Handbook for Data Driven Design, First Edition. Sage Publications, 2016.
2. Stefanie Molin, Ken Jee, "Hands-On Data Analysis with Pandas: A Python data science handbook for data collection, wrangling, analysis, and visualization, 2nd Edition, Packt Publishing Limited, 2021
3. Cole Nussbaumer Knaflic, "Storytelling with Data: A Data Visualization Guide for Business Professionals", 1st Edition, Wiley Publishers, 2015.
4. James R Evans, "Business Analytics – Methods, Models and Decisions", Pearson Publications, 2013.
5. R. N. Prasad, Seema Acharya, "Fundamentals of Business Analytics", Wiley Publications, 2015.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lecture Hours
1	Fundamentals of Data Analysis and Visualization (5)	
1.1	Python for Data Science: Data preparation and cleaning using Numpy, Pandas, Sklearn	2
1.2	Understanding data types and Data representation: Chart types: categorical, hierarchical, relational, temporal & spatial	2
1.3	Visualising data using Seaborn and Matplotlib	1
2	Quantitative Data Analysis (6)	
2.1	Basics of Linear Algebra, Probability and Statistics	2
2.2	Descriptive Analysis: Handling Numeric data, Patterns and Trends, Frequencies, Percentages	1
2.3	Measures of central tendency: Mean, Mode, and Median	1
2.4	Inferential Analysis: Differences and correlations between data sets-ANOVA, t-Tests, and Chi-Square, Hypothesis testing	2
3	Qualitative Data Analysis (9)	

3.1	Handling non-numeric data such as audio, video recordings, images, texts, and transcripts.	1
3.2	Regression: Logistic Regression, Ridge regression, LASSO	2
3.3	Classification- K-NN, naïve Bayes	1
3.4	Artificial Neural networks (ANNs), Deep NNs, CNN, overfitting, regularization	2
3.5	Clustering: k-means, expectation-maximization	1
3.6	Case studies: Text Analytics, Pattern Recognition	1
3.7	Business Intelligence: Product placement and pricing recommendation	1
4	Data Visualization (6)	
4.1	Object Oriented Interface in Matplotlib-2-D: bar charts, bubble charts, polar charts, Range chart, Box-and-whisker plots	2
4.2	Univariate scatter plots, histograms, word cloud, pie chart, waffle chart, stacked bar chart, tree map	2
4.3	3-D: surfaces, contours, hidden surfaces, pm3d colouring, 3D mapping	1
4.4	multi-dimensional data visualization; understanding data types and data representation: categorical, hierarchical, relational, temporal & spatial,; graph data visualization; Annotation	1
5	Visualization & Tableau (5)	
5.1	Introduction to BI, connecting Data Sources: Tableau Generated Fields	1
5.2	Data Manipulation & Customization, Using Parameters & adding Statistics	2
5.3	Pareto Chart, Bump Chart, Maps, Gantt Chart, Butterfly Chart, Donut Chart	2
6	Creating Dashboards & Stories (5)	
6.1	Dashboard Design Approach: Healthcare Quality Dashboard, Airline Quality Dashboard, Manufacturing Quality Dashboard.	2
6.2	Types of Visualization: Multivariate visualization, Geospatial visualization, Interactive visualization	2
6.3	Critiques in data visualization, Data visualization with ethics	1
	Total Hours	36

Course Designer(s):

1. Dr.B.Subbulakshmi, AP,CSE
2. Dr. M.Nirmala Devi, AP,CSE

bscse@tce.edu
mnit@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

PROGRAMME ELECTIVES

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSPW1 NETWORK MANAGEMENT SYSTEM
[Common for CSE / IT/ CSBS]

Category	L	T	P	Credit
PSE	3	0	0	3

Preamble

This course aims to provide knowledge on managing complex communication networks through NMS frameworks, protocols, and automation tools. It equips students with practical skills in monitoring, configuring, and troubleshooting network infrastructure through both traditional and modern network management protocols.

Prerequisite

Basic Computer Networks, Java Fundamentals, Python and Basics of Telecommunication

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the architecture, functions, and models of network management systems including FCAPS, TMN, and eTOM frameworks.	TPS2	70	75
CO2	Apply standard network management protocols such as SNMP, NETCONF, RESTCONF, and YANG models to configure, monitor, and manage network devices effectively.	TPS3	70	75
CO3	Apply SNMP, NETCONF, and RESTCONF protocols for managing and configuring network elements.	TPS3	70	75
CO4	Apply network fault management principles including suppression, correlation, and root cause analysis (RCA) using NMS tools.	TPS3	70	75
CO5	Implement NBI interfaces and APIs to enable network observability and service orchestration in simulated environments	TPS3	70	75
CO6	Apply the concepts of NFVI, VNF on boarding, SDN, and network slicing to build and test virtualized and software-defined network environments in lab setups	TPS3	70	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1.	M	L									L	L		L
CO2.	S	M	L	M	S	L	L	L	L	L	M	M	M	L
CO3	S	M	L	M	S	L	L	L	L	L	M	M	M	L
CO4	S	M	L	M	S	L	L	L	L	L	M	M	M	L

CO5	S	M	L	M	S	L	L	L	L	L	M	M	M	L
CO6	S	M	L	M	S	L	L	L	L	L	M	M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10	-	-	-	-	-	-	-	-	-	-	-	10	-	-	-	-	-	-	-	-	-	-	2	15	-	-	-	-
CO2	10	10	20	-	-	-	-	-	-	-	-	-	-	-	15	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO3	10	10	20	-	-	-	-	-	-	-	-	-	-	-	20	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO4	-	-	-	-	-	-	10	10	10	-	-	-	-	-	20	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO5	-	-	-	-	-	-	5	10	20	-	-	-	-	-	20	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO6	-	-	-	-	-	-	5	10	20	-	-	-	-	-	15	-	-	-	-	-	20	-	-	-	-	5	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**Mobile Network Management Frameworks and Architectures**

Overview of Mobile and Telecom Networks (GSM, LTE) and Core Components (BTS, BSC, MSC)– TMN and eTOM Frameworks – Layers, Logical Architecture, and Functional Areas.– EMS and NMS Architecture – Hierarchical Structure, Interfaces, and Roles.– FCAPS Model – Fault, Configuration, Accounting, Performance, and Security.– Integration of TMN, eTOM, and FCAPS for End-to-End Network Management.

SNMP-Based Network Management and Protocol Fundamentals

SNMP Architecture – Manager, Agent, MIB, and OID Structure.– SNMP Operations – GET, SET, TRAP, and INFORM; Message Workflow.– Security and Access Control in SNMP (v1, v2c, v3).– Role of SNMP in Fault and Performance Management within FCAPS.– Evolution Toward Model-Driven Management – Limitations and Transition Path.

Model-Driven Network Management with YANG and NETCONF

Model-Driven Concepts – Benefits over Traditional Protocols.– YANG Data Model – Structure, Syntax, and Standard Model Definitions.– NETCONF Protocol – Architecture, Capabilities, and RPC Operations.– Configuration Operations – get-config, edit-config, commit, and validation.– Integration of YANG with NETCONF for Device and Service Management.

RESTCONF, APIs, and SDN-Based Fault Management

RESTCONF Protocol – Operations, Resource Hierarchy, and Data Representation. – REST API Flow – Request/Response Cycle, Commands, and Payload Formats. – Alarm Lifecycle Management – Generation, Correlation, Suppression, and RCA. – SDN Controllers – Functional Components (Routing, Switching, Rollback). – Fault and Performance Monitoring through RESTCONF and SDN Integration.

Network Virtualization, NFV Architecture, and Orchestration

Network Function Virtualization (NFV) – Concept, Need, and Benefits. – NFV Infrastructure (NFVI) and MANO Framework – VIM, VNFM, and NFVO. – Integration of NFV with SDN for Dynamic Resource Allocation. – API-Driven Orchestration and Zero-Touch Provisioning (ZTP).– Service Lifecycle Management and Policy-Based Network Automation.

AI-Driven Network Orchestration and Emerging Technologies

AI-Driven Network Orchestration – Concepts, Architecture, and Applications. – ONAP Framework – Service Orchestration, Policy Management, and Automation. – AI/ML for Predictive Analytics, Fault Detection, and Network Optimization.– Telemetry with NETCONF/RESTCONF for Observability and Insights.– Emerging Technologies – PON, GPON, XPON, DOCSIS, 6G, and Quantum-Aware Networking.

Text Book

1. Mani Subramanian. Network Management: Principles and Practice. Pearson Education, 2nd Edition, 2010. ISBN: 978-8131708983.
2. Yu-Chu Tian and Jing Gao. Network Analysis and Architecture. Springer Nature Singapore Pte Ltd., 2024. ISBN: 978-981-99-5647-0.
3. Siamak Azodolmolky. Software Defined Networking with OpenFlow. Packt Publishing, 2014. ISBN: 978-1782166334.
4. Sami Aidarous and Thomas Plevyak. Telecommunication Network Management: Technologies and Implementations. IEEE Communications Society Press, 2000. ISBN: 978-0780334540.

Reference Books & web resources

1. Jason Edelman, Scott S. Lowe, and Matt Oswalt. Network Programmability and Automation: Skills for the Next-Generation Network Engineer. O'Reilly Media, 2018. ISBN: 978-1491931256.
2. Martin Björklund. YANG – A Data Modeling Language for the Network Configuration Protocol (NETCONF) (RFC 7950). Internet Engineering Task Force (IETF), 2016. Available at: <https://datatracker.ietf.org/doc/html/rfc7950>
3. Alexander Clemm. Network Management Fundamentals. Cisco Press, 2007. ISBN: 978-1587201370.
4. Michael W. Lucas. Networking for Systems Administrators. Tilted Windmill Press, 2019. ISBN: 978-1642350333.
5. Carla Schroder. Linux Networking Cookbook. O'Reilly Media, 2007. ISBN: 978-0596102487.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Mobile Network Management Frameworks and Architectures(6)	
1.1	Overview of Mobile and Telecom Networks – GSM, LTE, Core Components (BTS, BSC, MSC)	1
1.2	Introduction to Network Management – Role and Importance of NMS/EMS	1
1.3	TMN Framework – Layers, Logical Architecture, and Functional Areas	1
1.4	eTOM Framework – Strategy, Infrastructure, and Operations	1
1.5	FCAPS Model – Fault, Configuration, Accounting, Performance, Security	1
1.6	Integration of TMN, eTOM, and FCAPS for End-to-End Network Management	1
2	SNMP-Based Network Management and Protocol Fundamentals(6)	
2.1	SNMP Architecture – Manager, Agent, MIB, and OID	1

2.2	SNMP Operations – GET, SET, TRAP, INFORM	1
2.3	SNMP Versions – v1, v2c, v3 and Security	1
2.4	Fault & Performance Mgmt using SNMP within FCAPS	1
2.5	SNMP Integration with NMS Tools (OpenNMS, Zabbix)	1
2.6	Limitations of SNMP and Transition to Model-Driven Management	1
3	Model-Driven Network Management with YANG and NETCONF(6)	
3.1	Introduction to Model-Driven Management	1
3.2	YANG Data Model – Structure and Syntax	1
3.3	Standard YANG Models – IETF, OpenConfig	1
3.4	NETCONF Protocol – Architecture, Capabilities, RPC Operations	1
3.5	Configuration Operations – get-config, edit-config, commit	1
3.6	Integration of YANG and NETCONF for Device Management	1
4	RESTCONF, APIs, and SDN-Based Fault Management(6)	
4.1	RESTCONF Protocol – Operations, Resource Hierarchy	1
4.2	REST API Flow – Request/Response Cycle, Payloads	1
4.3	Alarm Lifecycle – Generation, Correlation, Suppression	1
4.4	Root-Cause Analysis (RCA) and Fault Suppression	1
4.5	SDN Controllers – Control Plane, Functions	1
4.6	RESTCONF & SDN Integration for Fault/Performance Management	1
5	Network Virtualization, NFV Architecture, and Orchestration(6)	
5.1	Network Virtualization – Concept and Need	1
5.2	NFV Infrastructure (NFVI) and MANO Framework	1
5.3	Roles of VIM, VNFM, NFVO in NFV Ecosystem	1
5.4	Integration of NFV with SDN Controllers	1
5.5	API-Driven Orchestration (RESTCONF/NETCONF)	1
5.6	Zero-Touch Provisioning (ZTP) & Policy-Based Automation	1
6	AI-Driven Network Orchestration and Emerging Technologies(6)	
6.1	AI-Driven Network Orchestration – Concepts and Architecture	1
6.2	ONAP Framework – Service Orchestration & Policy Mgmt	1

6.3	AI/ML for Predictive Analytics & Fault Detection	1
6.4	Network Slicing for 5G/IoT Environments	1
6.5	Telemetry with NETCONF/RESTCONF – Observability	1
6.6	Future Technologies – PON, GPON, XPON, DOCSIS, 6G, Quantum Networking	1
	Total	36

Assignments

Experiment No.	Experiment Title	Course Outcomes
1	SNMP Setup and Configuration (using OpenNMS, Zabbix, or similar tool)	CO2
2	NETCONF and YANG Modelling Hands-on with Cisco/YangSuite	CO2
3	RESTCONF API Usage for Configuration Management	CO2
4	Fault Management Simulation and RCA Lab	CO3
5	Virtualized Network Lab: NFVI,	CO3
6	VNF Lifecycle Management	CO3
7	SDN Controller (OpenDaylight) Setup and Flow Management	CO4
8	Service Orchestration Simulation (Ansible, etc.)	CO4
9	Network Slicing Setup in Simulated Environments	CO5
10	Practical Observability Tools: Grafana, Prometheus Integration	CO5
11	Practical Observability Tools: Prometheus Integration	CO6
12	Integration of Monitoring Tools with REST APIs	CO6

Course Designer(s):

1. Dr.K.Narasimha Mallikarjunan, Associate Professor , arjunkambaraj@tce.edu
Dept of Computer Science and Engineering
2. Dr.S.Ilankumaran Assistant Professor Dept of siit@tce.edu
Information Technology
3. Dr.P.Sharmila, Assistant Professor ,Dept of Information psaca@tce.edu
Technology
4. Mr.T.Siva,Assistant Professor ,Dept of Computer tsca@tce.edu
Science and Business Systems

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

**PROGRAMME ELECTIVES
(Honors)**

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

22CSRZ1**GPU ARCHITECTURE AND
PROGRAMMING**

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

This course introduces Learners to the foundations, programming models, and optimization techniques for GPU computing. It emphasizes the Single Instruction Multiple Thread (SIMT) architecture, CUDA and OpenCL programming environments, and methods for achieving high performance and scalability in heterogeneous systems. Students will gain hands-on experience in developing, debugging, and tuning GPU-accelerated applications across various domains.

Prerequisite

- Computer Architecture
- Data Structures
- Operating System
- Programming: C/C++

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate GPU architecture, SIMT execution model, and differences from CPU-based systems.	TPS2	70	85
CO2	Develop CUDA programs to optimize GPU kernels for memory and thread utilization.	TPS3	70	80
CO3	Construct OpenCL programs to solve compute-intensive problems	TPS3	70	80
CO4	Apply Optimization methods in GPU kernels for efficient memory and thread utilization.	TPS3	70	80
CO5	Use GPU integration for acceleration in AI/ML and data science workflows	TPS3	70	80
CO6	Discover emerging GPU Computing trends in heterogeneous, cloud, and edge GPU computing.	TPS4	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1.	M	L			L	L	M	M	M		M	M	L	M
CO2.	S	M	L	L	M	L	M	M	M		M	M	L	M
CO3.	S	M	L	L	M	L	M	M	M		M	M	L	M
CO4.	S	M	L	L	M	L	M	M	M		M	M	L	M
CO5.	S	M	L	L	M	L	M	M	M		M	M	L	M
CO6.	S	S	M	M	M	L	M	M	M	L	M	S	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass2		Terminal					
	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
TPS Scale	1	2	3	4	5	6	3	1	2	3	4	5	6	3	4	1	2	3	4	5	6
CO1	15	15		-	-	-		-	-	-	-	-	-	-		2	6		-	-	-
CO2	5	10	20	-	-	-	50	-	-	-	-	-	-	-		1	3	15	-	-	-
CO3	5	10	20		-	-	50	-	-	-	-	-	-	-		1	3	15		-	-
CO4	-	-	-	-	-	-	-	5	10	20		-	-	30		1	3	15		-	-
CO5	-	-	-	-	-	-	-	5	10	20		-	-	30		1	3	15		-	-
CO6	-	-	-	-	-	-	-	5	10		15	-	-		40	1	3		12	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Fundamentals of GPU Computing: Evolution of GPU Computing – From Graphics to GPGPU-GPU Architecture – SIMT Model, Thread Hierarchy-Parallel Programming Models – CUDA, OpenCL, Open ACC, CUDA Tuned Libraries-Programming Environment Setup, Device Query, and Memory Management-Server level Configuration-**Case Study:** Vector Addition using CUDA **Open Source Tools:** NVIDIA CUDA Toolkit, Nsight Systems, OpenCL SDK

CUDA Programming Model: CUDA Thread Hierarchy: Grids, Blocks, Threads- Memory Model: Global, Shared, Constant, and Texture Memory-Kernel Launch and Synchronization-Error Handling, Streams, and Events-**Case Study:** Image Filtering using CUDA, Multi GPU with CUDA **Tools:** NVIDIA Nsight Compute, cuBLAS, cuFFT

OpenCL Programming and Portability: OpenCL Platform Model, Execution Model, Data Transfer Overhead-OpenCL Kernels and Buffers-Performance Comparison: CUDA vs OpenCL-Portability Across GPUs and CPUs-**Case Study:** Matrix Multiplication using OpenCL(Vanilla) **Tools:** AMD ROCm, Intel OpenCL SDK

Optimization and Performance Tuning: Profiling GPU Applications-Memory Coalescing, Occupancy, and Latency Hiding -Loop Unrolling, Tiling, and Kernel Fusion- Power and Thermal Considerations-Quantization Aware Training, Performance Metrics: Compute, Memory, Execution **Case Study:** Convolution Optimization for Deep Learning **Tools:** NVIDIA Nsight Systems, GPUPerfStudio, nvprof

Framework Integration: GPU Acceleration in Machine Learning (TensorFlow, PyTorch)-RAPIDS for GPU Data Science-Multi-GPU Programming and Peer-to-Peer Communication-GPU Virtualization and Cloud GPU Platforms (AWS, Azure, GCP)-**Case Study:** Training CNN Models using GPU **Tools:** TensorFlow-GPU, PyTorch, RAPIDS cuDF

Recent Trends and Research Directions: Heterogeneous Computing: GPU + CPU + FPGA-GPU Clusters and Distributed Deep Learning (Horovod, NCCL)-Edge GPU Computing (Jetson Nano, Coral TPU)- Energy-efficient GPUs-Future Architectures: NVIDIA Hopper, AMD CDNA, Intel ARC-**Case Study:** Edge AI Inference on Jetson Nano **Tools:** NVIDIA JetPack SDK, Horovod, NCCL

Text Book

1. David Kirk, Wen-mei Hwu, "Programming Massively Parallel Processors: A Hands-on Approach"; Morgan Kaufman; 2022

Reference Books & web resources

1. John L. Hennessy and David A. Patterson, "Computer Architecture -- A Quantitative Approach", Morgan Kaufman; 2017
2. Shane Cook, "CUDA Programming: A Developer's Guide to Parallel Computing with GPUs"; Morgan Kaufman; 2012
3. Benedict Gaster, Lee Howes, David R. Kaeli, "Heterogeneous Computing with OpenCL", Morgan Kaufman; 2016
4. NPTEL-GPU Architecture and Programming :
https://onlinecourses.nptel.ac.in/noc20_cs41/preview

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Fundamentals of GPU Computing (5)	
1.1	Evolution of GPU Computing – From Graphics to GPGPU	1
1.2	GPU Architecture – SIMT Model, Thread Hierarchy	1
1.3	Parallel Programming Models – CUDA, OpenCL, Open ACC, CUDA Tuned Libraries	1
1.4	Programming Environment Setup, Device Query, and Memory Management – Server Level Configuration	1
1.5	Case Study: Vector Addition using CUDA Open Source Tools: NVIDIA CUDA Toolkit, Nsight Systems, OpenCL SDK	1
2	CUDA Programming Model (7)	
2.1	CUDA Thread Hierarchy: Grids, Blocks, Threads	1
2.2	Memory Model: Global, Shared, Constant, and Texture Memory	1
2.3	Kernel Launch and Synchronization	2
2.4	Error Handling, Streams, and Events	2
2.5	Case Study: Image Filtering using CUDA, Multi GPU with CUDA Tools: NVIDIA Nsight Compute, cuBLAS, cuFFT	1
3	OpenCL Programming and Portability (6)	
3.1	OpenCL Platform Model, Execution Model, Data Transfer Overhead	2
3.2	OpenCL Kernels and Buffers	1
3.3	Performance Comparison: CUDA vs OpenCL	1
3.4	Portability Across GPUs and CPUs	1

Module No.	Topic	No. of Periods
3.5	Case Study: Matrix Multiplication using OpenCL Tools: AMD ROCm, Intel OpenCL SDK	1
4	Optimization and Performance Tuning (7)	
4.1	Profiling GPU Applications	1
4.2	Memory Coalescing, Occupancy, and Latency Hiding	1
4.3	Loop Unrolling, Tiling, and Kernel Fusion	1
4.4	Power and Thermal Considerations, Quantization Aware Training	2
4.5	Performance Metrics: Compute, Memory, Execution	1
4.6	Case Study: Convolution Optimization for Deep Learning , Tools: NVIDIA Nsight Systems, GPUPerfStudio, nvprof	1
5	Framework Integration (5)	
5.1	GPU Acceleration in Machine Learning (TensorFlow, PyTorch)	1
5.2	RAPIDS for GPU Data Science	1
5.3	Multi-GPU Programming and Peer-to-Peer Communication	1
5.4	GPU Virtualization and Cloud GPU Platforms (AWS, Azure, GCP)	1
5.5	Case Study: Training CNN Models using GPU Tools: TensorFlow-GPU, PyTorch, RAPIDS cuDF	1
6	Recent Trends and Research Directions (6)	
6.1	Heterogeneous Computing: GPU + CPU + FPGA	1
6.2	GPU Clusters and Distributed Deep Learning (Horovod, NCCL)	1
6.3	Edge GPU Computing (Jetson Nano, Coral TPU)	1
6.4	Energy-efficient GPUs	1
6.5	Future Architectures: NVIDIA Hopper, AMD CDNA, Intel ARC	1
6.6	Case Study: Edge AI Inference on Jetson Nano Tools: NVIDIA JetPack SDK, Horovod, NCCL	1
	Total	36

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, rlsit@tce.edu
Computer Science and Engineering,

22CSTA1 COMPUTATIONAL GEOMETRY

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

This course introduces the geometric foundations of algorithm design, emphasizing data structures and computational methods for solving geometric problems. Students will learn to model, analyse, and apply geometric algorithms in various engineering domains.

Prerequisite

- Data Structures and Algorithms
- Discrete Mathematics and Linear Algebra

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the fundamental geometric concepts, transformations, and classification of problems in Computational Geometry	TPS2	70	80
CO2	Demonstrate the concept of convex hulls to solve real-world geometric applications.	TPS3	70	75
CO3	Use algorithms to detect and handle intersections among geometric objects in various applications.	TPS3	70	75
CO4	Use triangulation techniques to decompose geometric shapes and solve problems in modelling, rendering, and robotic navigation.	TPS3	70	75
CO5	Solve proximity and spatial-partitioning problems using Voronoi and Delaunay-based solutions	TPS3	70	75
CO6	Implement geometric data structures for spatial search and navigation systems.	TPS3	70	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1.	M	L										L		
CO2.	S	M	L	L		S		M			S	M	M	M
CO3	S	M	L	M		S	L	M	M		S	M	M	M
CO4	S	M	L	M		S		M			S	M	M	M
CO5	S	M	L	M		S		M			S	M	M	M
CO6	S	M	L	M	L	S	L	M	M		S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4			1	2	3	4	5	6
TPS Scale																														
CO1	5	10																							2	5				
CO2	5	15	20												50										2	5	12			
CO3	5	15	25												50										2	5	12			
CO4							5	10	15												20				2	5	12			
CO5							5	10	20												30				2	5	12			
CO6							5	10	20												50				5	12				

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**Introduction to Computational Geometry**

Overview and Scope of Computational Geometry, Geometric Objects and Representations, Geometric transformations, Classification of Geometric Problems: Static, Query and Dynamic, Geometric Techniques: Line-intersection tests, Point-in-polygon, Sweep line concepts.

Convex Hulls

Concept of Convex Hulls - Convex Hull Algorithms in the Plane: Graham's Scan Algorithm, Jarvi's March, Divide and Conquer Algorithm, Case Study: Shape analysis - Boundary estimation.

Linear Programming and Intersection

Linear Programming in Geometry: Half-Plane Intersection, Incremental and Randomized Linear Programming, Duality, Geometric Intersection Algorithms: Line segment intersection, Plane Sweep Algorithm. Case Study: Intersection handling in VLSI layout

Triangulation

Triangulations and types: Monotone and Simple Polygon Triangulations, Point-Set Triangulations, Optimization and Advanced Triangulation: Guarding and Triangulations - Steiner triangulation – Delaunay refinement, Case Study: Finite Element Analysis: Biomedical models.

Voronoi Diagrams and Delaunay Triangulation

Properties of Voronoi Diagram - Computing the Voronoi Diagram: Fortune Sweep Algorithm, Divide and Conquer Algorithm, Closest pair Problems, Relationship between Voronoi and Delaunay Triangulation, Computing the Delaunay Triangulation, Case Study: Mesh Generation, Spatial analysis.

Geometric Data Structures and Searching

Geometric Data Structures: Doubly Connected Edge List (DCEL) representation, Computing Overlays of Planar subdivisions, Spatial Data Structures (Quadrees, k-d Trees and R-Trees), Locality Sensitive Hashing (LSH): Point location – Range and Nearest - Neighbour searching. GPU architectures for computational geometry: Parallel geometric processing. Applications:

GIS map overlays, ISRO Bhoomi dataset for spatial queries.

Text Book

1. Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars, "Computational Geometry Algorithms and Applications", Third Edition, Springer, 2013.

Reference Books & web resources

1. Joseph O'Rourke "Computational Geometry in C" Cambridge University Press.1998
2. Preparata, Franco P., and Shamos, Michael I.. Computational Geometry: An Introduction. Germany, Springer New York, 2012.
3. K. Mulmuley, Computational Geometry: An Introduction through Randomized Algorithms, PrenticeHall, 1994
4. Patil, Devang. Principles of Computational Geometry. India, Educohack Press, 2025.
5. Computational Geometry: Exploring Geometric Insights for Computer Vision. N.p., One Billion Knowledgeable, 2024.
6. NPTEL-IIT Delhi https://onlinecourses.nptel.ac.in/noc22_cs41/preview

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Computational Geometry	
1.1	Overview and Scope of Computational Geometry	1
1.2	Geometric Objects and Representations, Geometric transformations	1
1.3	Classification of Geometric Problems: Static, Query and Dynamic	1
1.4	Geometric Techniques: Line-intersection tests	1
1.5	Point-in-polygon, Sweep line concepts.	1
2	Convex Hulls	
2.1	Concept of Convex Hulls	1
2.2	Convex Hull Algorithms in the Plane: Graham's Scan Algorithm	2
2.3	Jarvi's March, Divide and Conquer Algorithm	1
2.4	Case Study: Shape analysis - Boundary estimation	2
3	Linear Programming and Intersection	
3.1	Linear Programming in Geometry: Half-Plane Intersection	1
3.2	Incremental and Randomized Linear Programming	1
3.3	Duality	1
3.4	Geometric Intersection Algorithms: Line segment intersection	1
3.5	Plane Sweep Algorithm	1
3.6	Case Study: Intersection handling in VLSI layout	1
4	Triangulation	

4.1	Triangulations and types: Monotone and simple polygon triangulations, point-Set triangulations	1
4.2	Optimization and Advanced Triangulation: Guarding and Triangulations	1
4.3	Steiner triangulation	1
4.4	Delaunay refinement	1
4.5	Case Study: Finite Element Analysis: Biomedical models.	2
5	Voronoi Diagrams and Delaunay Triangulation	
5.1	Properties of Voronoi Diagram - Computing the Voronoi Diagram: Fortune Sweep Algorithm	1
5.2	Divide and Conquer Algorithm, Closest pair Problems	1
5.3	Relationship between Voronoi and Delaunay Triangulation	1
5.4	Computing the Delaunay Triangulation	1
5.5	Case Study: Mesh generation, Spatial analysis	2
6	Geometric Data Structures and Searching	
6.1	Geometric Data Structures: Doubly Connected Edge List (DCEL) representation	1
6.2	Computing overlays of planar subdivisions, Spatial data structures (Quadrees, k-d Trees and R-Trees)	1
6.3	Locality Sensitive Hashing (LSH): Point location – Range and Nearest - Neighbour searching.	1
6.4	GPU architectures for computational geometry, Parallel geometric processing	2
6.5	Applications: GIS map overlays, ISRO Bhoomi dataset for spatial queries.	2
	Total	36

Course Designer(s):

1. Mrs.G.Bhavani, AP/CSE,
2. Mrs.J.Vijayalakshmi, AP/CSE

gbicse@tce.edu
jvicse@tce.edu

22CSTB1**QUANTUM ALGORITHMS AND
CRYPTOGRAPHY**

Category	L	T	P	Credit
PEES	3	0	0	3

Preamble

The course will cover the impact on cryptography created by the advent of quantum computers. Significant progress has been made in recent times to develop quantum computers, so it is necessary to base cryptography on problems that remain hard against quantum attackers. In this course, we will study the foundations of quantum computing and the important role of quantum computers in cryptography.

Prerequisite

- Linear Algebra

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the quantum mechanics primitives that aid in design of quantum algorithms	TPS2	70	80
CO2	Model the power of various quantum algorithms	TPS3	70	75
CO3	Model computationally-bounded cryptographic protocols	TPS3	70	75
CO4	Apply quantum mechanics to construct cryptographic protocols	TPS3	70	75
CO5	Demonstrate the notion of lattices and bases	TPS3	70	75
CO6	Model lattice based cryptosystem that can resist quantum attacks	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1	M	L		L	L	L	L	M	M	M		L	L	M
CO2	S	M	L		L	L	L	M	M	M		M	L	M
CO3	S	M	L		L	L	L	M	M	M		M	L	M
CO4	S	M	L		L	L	L	M	M	M	L	M	L	M
CO5	S	M	L		L	L	L	M	M	M		M	L	M
CO6	S	M	L	L	M	M	L	M	M	M	L	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1	CAT2	Assignment 1	Assignment 2	Terminal
----	------	------	--------------	--------------	----------

TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20																							2	10				
CO2	10	10	20												50										2	10	10			
CO3		10	20												50											5	10			
CO4							5	10	20																2	5	10			
CO5							5	10	20																2	5	10			
CO6								10	20																2	5	10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Quantum Mechanics: Qubits - Dirac Notation – Superposition - Single Qubit System - Reversible Operations - Unitary Transformations - Single Qubit Gates, Two Qubit Systems, Tensor Products and Two Qubit Gates - Quantum Circuits - Quantum Entanglement and Interference – No-Cloning - Superdense Coding.

Quantum Algorithms - Quantum Fourier Transform - Inverse Quantum Fourier Transform - Phase Kickback – Deutsch and Deutsch-Jozsa Algorithm – Simon's Algorithm over $\mathbb{Z}_n$ – Bernstein-Vazirani Algorithm - Shor's Algorithm.

Quantum Cryptography – Principles of Cryptographic Protocols: Security Model - Hardness Assumptions – Security Proof – Symmetric Key Encryption - Public Key Encryption – Encrypting Quantum States - Mixed States - Quantum Key Distribution – BB84 Protocol - Quantum One-Time Pad - Quantum Public Key Encryption.

Lattices and Bases – Hidden Subgroup problem - Introduction to Lattices – Lattice Based One Way Functions.

Post-Quantum Cryptography - Learning with Errors – Public Key Encryption from Learning with Errors – Lattice-based Digital Signature scheme - Quantum Oblivious Transfer.

Text Book

1. Introduction to Quantum Cryptography by Thomas Vidick, California Institute of Technology, USA, Stephanie Wehner, Delft University of Technology, The Netherlands. Cambridge University Press, (2023).
2. Post-Quantum Cryptography by Daniel J. Bernstein, Johannes Buchmann, Erik Dahmen, Springer (2009).
3. Quantum Computation and Quantum Information, M. A. Nielsen and I. Chuang, Cambridge University Press (2010).
4. Introduction to Modern Cryptography by Jonathan Katz and Yehuda Lindell, Chapman & Hall CRC, (2008).

Reference Books & web resources

5. Quantum computing by Oded Regev
https://cims.nyu.edu/~regev/teaching/quantum_fall_2019/

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Quantum Mechanics	
1.1	Qubits, Dirac Notation	1
1.2	Superposition, Reversible Operations, Unitary Transformations, Tensor Products	1

Module No.	Topic	No. of Periods
1.3	Qubit Systems and Gates, Quantum Circuits	2
1.4	Quantum Entanglement and Interference	2
1.5	No-Cloning and Superdense Coding	2
2	Quantum Algorithms	
2.1	Quantum Fourier Transform, Inverse Quantum Fourier Transform	2
2.2	Phase Kickback	1
2.3	Deutsch and Deutsch-Jozsa Algorithm	2
2.4	Simons Algorithm over Z_n	2
2.5	Bernstein-Vazirani Algorithm	1
2.6	Shor's Algorithm	2
3	Quantum Cryptography	
3.1	Principles of Cryptographic Protocols: Security Model, Hardness Assumptions, Security Proof	1
3.2	Symmetric Key Encryption	1
3.3	Public Key Encryption	1
3.4	Mixed States - Quantum Key Distribution - BB84 Protocol	2
3.5	Quantum One-Time Pad	1
3.6	Quantum Public Key Encryption	2
4	Lattices and Bases	
4.1	Hidden Subgroup problem	1
4.2	Introduction to Lattices	1
4.3	Lattice Based One Way Functions	2
5	Post-Quantum Cryptography	
5.1	Learning with Errors	2
5.2	Public Key Encryption from Learning with Errors	2
5.3	Lattice-based Digital Signature scheme	1
5.4	Quantum Oblivious Transfer	1
	Total	36

Course Designer(s):

1. Dharani J, Assistant Professor, CSE

jdicse@tce.edu

22CSTC1	PROBABILISTIC MODELLING	GRAPHICAL	Category	L	T	P	Credit
			PEES	2	1	0	3

Preamble

This course provides a comprehensive introduction to probabilistic graphical models, with an emphasis on their applications in machine learning. It integrates concepts from probability theory, graph theory, and optimization to develop a unified framework for reasoning under uncertainty. This course bridges theory and practice through modern probabilistic programming tools for real-world applications. At the end of the course, students will be able to model real-world problems using graphical models, design and implement inference algorithms, and learn both the parameters and structure of graphical models from data.

Prerequisite

- Basic Knowledge of probability, calculus, linear algebra, and discrete mathematics

Course Outcomes

On the successful completion of the course, students will be able to:

CO1	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct different types of graphical models, such as Bayesian Networks and Markov Random Fields, using concepts from probability and graph theory.	TPS3	70	80
CO2	Demonstrate the use of conditional independence and factorization to interpret the structure and properties of graphical models.	TPS3	70	75
CO3	Implement exact inference methods like variable elimination and belief propagation to calculate probabilities in graphical models.	TPS3	70	75
CO4	Utilize approximate inference and sampling methods, such as variational techniques and Markov Chain Monte Carlo, to solve problems in probabilistic models.	TPS3	70	75
CO5	Employ parameter estimation and structure learning algorithms for graphical models using data.	TPS3	70	75
CO6	Investigate the solutions for complex, data-intensive problems in domains such as NLP, bioinformatics, and vision, using HMMs, DBNs, VAEs, and Python libraries.	TPS4	70	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	S	M	L		M		L	L	L	L	M	M	L	L
CO2	S	S	M	L	M	L	L	L	L	L	M	M	L	L
CO3	S	M	L		M		L	L	L	L	M	M	L	L
CO4	S	M	L		M		L	L	L	L	M	M	L	L

CO5	S	M	L		M		L	L	L	L	M	M	L	L
CO6	S	S	M	L	M	L	L	L	L	L	M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	10	15												30										2	5	10			
CO2	5	10	15												30										2	5	10			
CO3	5	10	25												40										2	5	10			
CO4							5	10	15											30					2	5	10			
CO5							5	10	15											30					2	5	10			
CO6							5	10	15	10										20	20				5		10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**Review of Probability Theory and Graph Theory**

Graphical Models: Introduction - Directed Models (Bayesian Network), Undirected Models (Markov Random Fields), Dynamic Models (Hidden Markov Model & Kalman Filters) and Factor Graph; Conditional Independence (Bayes Ball Theorem and D-separation), Markov Blanket, Factorization (Hammersley-Clifford Theorem), Equivalence (I-Maps & Perfect Maps); Factor Graphs - Representation, Relation to Bayesian Network and Markov Random Field.

Inference in graphical models: Exact Inference - Variable Elimination, Elimination Orderings, Relation to Dynamic Programming, Dealing with Evidence, Forward-Backward Algorithm, Viterbi Algorithm; Junction Tree Algorithm; Belief Propagation (Sum Product); Approximate Inference – Reparameterization and Inverse Parameterization-Variational Methods (Mean Field, Kikuchi & Bethe Approximation), Expectation Propagation, Gaussian Belief Propagation; MAP Inference - Max-Product, Graph Cuts, Linear Programming Relaxations to MAP (Tree-Reweighted Belief Propagation, MPLP); Sampling - Markov Chain Monte Carlo- Handling rejection sampling-Metropolis Hastings, Gibbs (Collapsing & Blocking), Particle filtering.

Learning in Graphical Models: Parameter Estimation - Expectation Maximization, Maximum Likelihood Estimation, Maximum Entropy, Pseudolikelihood, Bayesian Estimation, Conditional Likelihood, Structured Prediction; Learning with Approximate Inference; Learning with Latent Variables; Structure Learning, Structure Search, L1 priors.

Advanced Topics and Applications

Temporal models: Hidden Markov Models (HMMs) and Dynamic Bayesian Networks (DBNs) – Probabilistic relational models – Deep probabilistic models and Variational Autoencoders – Applications in natural language processing, bioinformatics, vision, and recommendation systems – Implementation using Python libraries (pgmpy, PyMC3 etc.)-Mini Project.

Text Book

1. Koller, D. and Friedman, N., "Probabilistic Graphical Models: Principles and Techniques", MIT Press, 1st Edition, 2009.
1. Sucar, L. E., "Probabilistic Graphical Models: Principles and Applications", Springer Cham, 2nd ed., 2021.

Reference Books & web resources

1. Kevin P. Murphy, "Probabilistic Machine Learning: An Introduction", Adaptive Computation and Machine Learning series, MIT Press, 1st Edition, 2022.
2. David Barber, "Bayesian Reasoning and Machine Learning", Cambridge University Press, 1st Edition, 2012.

3. Martin J. Wainwright and Michael I. Jordan, "Graphical Models, Exponential Families, and Variational Inference", Foundations and Trends in Machine Learning, 1st Edition, 2008.
4. https://github.com/dherault/coursera_probabilistic_graphical_models
5. <https://www.cse.iitd.ac.in/~parags/teaching/2015/au15/col776/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Review of Probability Theory and Graph Theory	2
1.1	Basics of Probability, Random Variables, Joint Distributions, Conditional Probability, Conditional Independence	1
1.2	Graph Theory Basics – Paths, Cliques, Subgraphs, Cycles	1
2	Graphical Models	8
2.1	Introduction – Directed (Bayesian Networks), Undirected (MRFs), Dynamic Models (HMMs, Kalman Filters), Factor Graphs	3
2.2	Conditional Independence – Bayes Ball Theorem, D-Separation, Markov Blanket	2
2.3	Factorization – Hammersley-Clifford Theorem, Equivalence (I-Maps & Perfect Maps)	1
2.4	Factor Graph Representation; Relation to Bayesian Network & Markov Random Field	2
3	Inference in Graphical Models (Exact Methods)	8
3.1	Variable Elimination, Elimination Orderings, Relation to Dynamic Programming	2
3.2	Dealing with Evidence	1
3.3	Forward-Backward Algorithm	1
3.4	Viterbi Algorithm	1
3.5	Junction Tree Algorithm – Construction & Message Passing	2
3.6	Belief Propagation – Sum-Product Algorithm	1
4	Approximate & MAP Inference	7
4.1	Variational Methods – Mean Field, Kikuchi & Bethe Approximation	2
4.2	Expectation Propagation & Gaussian Belief Propagation	1
4.3	MAP Inference – Max-Product, Graph Cuts	1
4.4	Linear Programming Relaxations – TRW-BP, MPLP	1
4.5	Sampling – MCMC, Rejection Handling, Metropolis-Hastings	1
4.6	Gibbs Sampling (Collapsing & Blocking), Particle Filtering	1
5	Learning in Graphical Models	5
5.1	Parameter Estimation – MLE, MAP, EM, Maximum Entropy	2
5.2	Bayesian Estimation, Pseudolikelihood, Conditional Likelihood	1
5.3	Structured Prediction & Learning with Latent Variables	1
5.4	Structure Learning – Structure Search, L1 priors	1
6	Advanced Topics and Applications	6
6.1	Temporal Models – HMMs & Dynamic Bayesian Networks (DBNs)	2
6.2	Probabilistic Relational Models	1
6.3	Deep Probabilistic Models & Variational Autoencoders	1
6.4	Applications – NLP, Bioinformatics, Vision, Recommendation Systems	1
6.5	Implementation using Python Libraries – pgmpy, PyMC3, TFP	1
	Mini Project	
Total	36 Periods	

Course Designer(s):

1. Dr M Nirmala Devi, AP(Sel.Gr),CSE, TCE

mnit@tce.edu

22CSTD1 COMPUTATIONAL COMPLEXITY

Category	L	T	P	Credit
PEES	2	1	0	3

Preamble

Computational Complexity explores the fundamental limits of computation through the study of time, space, and randomness in algorithms. It classifies problems into complexity classes such as P, NP and PSPACE, highlighting the boundaries between tractable and intractable computation.

Prerequisite

- Discrete Mathematics
- Theory of Computation
- Linear Algebra
- Data Structures and Algorithms

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the fundamental concepts of computational models and complexity classes based on time.	TPS 2	70	80
CO2	Apply the concepts of polynomial-time reductions and NP-completeness to classify computational problems.	TPS 3	70	75
CO3	Illustrate the power of parallel computation models by studying Boolean circuits and their implications.	TPS 3	70	75
CO4	Apply lower bound techniques and communication complexity concepts to evaluate the computational limitations.	TPS 3	70	75
CO5	Analyse the relationships among randomized complexity classes	TPS 4	70	75
CO6	Solve advanced complexity concepts to understand their role in characterizing computational hardness.	TPS 4	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PS O1	PS O2	PSO 3
CO1	M	L						L	L	L	L	L		L

CO2	S	M	L		L	L	L	L	M	M	L	M	L	L
CO3	S	M	L	L	L	L	M	L	M	M	L	M	L	L
CO4	S	M	L	L	L	L	M	L	M	M	L	M	L	M
CO5	S	S	M	L	L	L	S	L	M	M	L	S	L	L
CO6	S	S	M	L	L	L	S	L	M	M	L	S	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10											20												2	5				
CO2	10	10	20											20											2	5	10			
CO3		20	20											60											2	5	10			
CO4							10	10	10											10					2	5	10			
CO5							10	10		15										30					2	5	10	10		
CO6							10	10		15										60						5		10		

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Foundations of Computational Complexity - Turing Machines - Time Complexity Classes (P, NP, CoNP, NP hard, NP complete) -Polynomial-time Reductions – Greedy Approach – Hamiltonian Cycle Reduction - Graph Colouring - Cook-Levin Theorem - Satisfiability and Circuit Value Problems - Alternating Turing Machines and their relation to PH - Ladner's Theorem – Equivalence and Non-Equivalence of P and NP – Quantum NP

Space Complexity and Structural Results - Space Complexity Classes - Configuration Graphs and Reachability Problems - Space Hierarchy Theorem - Savitch's Theorem Immerman-Szelepcsényi Theorem - PSPACE-Completeness and NL-Completeness - Time–Space Trade-offs for SAT

Circuit Complexity - Circuit Model of Computation, Boolean Circuits, Uniformity - Non-uniform Classes - Parallel Computation Classes - Karp-Lipton Theorem - Circuit Hierarchy Theorem and Constant-depth Circuits

Non-uniform Computation - Shannon's Lower Bound, Lupanov's Upper Bound - Circuit Lower Bounds - Razborov-Smolensky Method, Monotone Circuits - Branching Programs and Barrington's Theorem - Communication Complexity - Connections between Circuit and Communication Complexity

Randomized and Probabilistic Complexity - Probabilistic Turing Machines - Randomized complexity classes - Randomized Space Complexity – Reduction in Randomized Complexity Theory (Polynomial time Reduction, Hamiltonian Reduction, Randomized Reduction) - Pseudorandom Generators and the Nisan-Wigderson Construction - Derandomization - Polynomial Identity Testing and Randomized Algorithms - Perfect Matching and RNC Algorithms

Interactive Proofs, Counting, and Advanced Topics - Interactive Proof Systems - Sum-Check Protocol, Random Self-Reducibility - Valiant's Theorem - Toda's Theorem - Probabilistically Checkable Proofs - PCP Theorem – Approximation algorithm – Greedy Set Cover – Vertex Cover 2-Approximation – Christofides for TSP.

Text Book

1. Jayant Ramaswamy, "Introduction to Mathematical Logic", Educohack Press, 2025.
2. Avi Wigderson, "Mathematics and Computation: Ideas Revolutionizing Technology and Science", Princeton University Press, 2019.
3. Oded Goldreich, "Computational Complexity: A Conceptual Perspective", Cambridge University Press, 2008.
4. Heribert Vollmer, "Introduction to Circuit Complexity: A Uniform Approach", Springer, 1999.

Reference Books & web resources

1. Sanjeev Arora and Boaz Barak, "Computational Complexity: A Modern Approach", Cambridge University Press, 2009.
2. C. H. Papadimitriou, "Computational Complexity", Pearson, 1994.
3. Michael Sipser, "Introduction to the Theory of Computation", Cengage, 2014.
4. Lecture notes on "A Taste of Circuit Complexity Pivoted at NEXP not in ACC (and more)" by Gil Cohen
5. https://onlinecourses.nptel.ac.in/noc21_cs90/preview

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Foundations of Computational Complexity	(6)
1.1	Turing Machines	1
1.2	Time Complexity Classes, Polynomial-time Reductions	1
1.3	Greedy Approach, Hamiltonian Cycle Reduction, Graph Colouring, Cook-Levin Theorem	1
1.4	Satisfiability and Circuit Value Problems	1
1.5	Alternating Turing Machines and their relation to PH, Ladner's Theorem	1
1.6	Equivalence and Non-Equivalence of P and NP – Quantum NP	1
2	Space Complexity and Structural Results	(6)
2.1	Space Complexity Classes, Configuration Graphs and Reachability Problems	2
2.2	Space Hierarchy Theorem, Savitch's Theorem, Immerman-Szelepcsényi Theorem	2
2.3	PSPACE-Completeness and NL-Completeness, Time-Space Tradeoffs for SAT	2
3	Circuit Complexity	(6)
3.1	Circuit Model of Computation, Boolean Circuits, Uniformity	2

Module No.	Topic	No. of Periods
3.2	Non-uniform Classes, Parallel Computation Classes, Karp-Lipton Theorem	2
3.3	Circuit Hierarchy Theorem and Constant-depth Circuits	2
4	Non-Uniform Computation	(6)
4.1	Shannon's Lower Bound, Lupanov's Upper Bound - Circuit Lower Bounds	2
4.2	Razborov-Smolensky Method, Monotone Circuits, Branching Programs and Barrington's Theorem	2
4.3	Communication Complexity	1
4.4	Connections between Circuit and Communication Complexity	1
5	Randomized and Probabilistic Complexity	(6)
5.1	Probabilistic Turing Machines, Randomized Complexity Classes	1
5.2	Randomized Space Complexity, Reduction in Randomized Complexity Theory	2
5.3	Pseudorandom Generators and the Nisan-Wigderson Construction	1
5.4	Derandomization, Polynomial Identity Testing and Randomized Algorithms	1
5.5	Perfect Matching and RNC Algorithms	1
6	Interactive Proofs, Counting, and Advanced Topics	(6)
6.1	Interactive Proof Systems, Sum-Check Protocol, Random Self-Reducibility	1
6.2	Valiant's Theorem, Unique SAT, Toda's Theorem	1
6.3	Probabilistically Checkable Proofs (PCP), PCP Theorem	1
6.4	Hardness of Approximation, Quantum Computation, Grover's Algorithm	1
6.5	Approximation algorithm – Greedy Set Cover – Vertex Cover 2-Approximation – Christofides for TSP	2
	Total	36

Course Designer(s):

1. Dr. N. Sheerin Sitara, nsscse@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E./B.TECH DEGREE PROGRAMME

ELECTIVE COURSE FOR MINOR

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSQC1**MACHINE LEARNING FOR ENGINEERS**

Category	L	T	P	Credit
ME	2	1	0	3

Preamble

The objective of this course is to introduce the fundamental principles of Machine Learning (ML) applicable across engineering domains. The students to develop and evaluate ML models for multidisciplinary engineering problems. They have an ability to demonstrate the integration of ML tools with engineering workflows for data-driven decision-making. They will understand the concept of interpretability, ethics, and sustainability in ML-based engineering solutions.

Prerequisite

- Probability and Statistics
- Foundations of Computer Programming

Course Outcomes

On the successful completion of the course, students will be able to

Cos	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the foundational concepts of Machine Learning, mathematical tools and computational packages	TPS 2	70	80
CO2	Apply classical ML algorithms (Supervised and Unsupervised) for modelling and training engineering data.	TPS 3	70	75
CO3	Use back-propagation, regularization, and cross-validation techniques to design a neural networks and deep learning models.	TPS 3	70	75
CO4	Implement deep learning architectures (RNN, CNN, and Transfer Learning) to solve real-world engineering problems such as forecasting, defect detection, and imaging.	TPS 3	70	75
CO5	Use a method of advanced generative models such as GANs and DCGANs for engineering applications like synthetic data or image generation to give solutions for real-time problems.	TPS 3	70	75
CO6	Design a system using ML approaches to solve real-world problems in multidisciplinary engineering domains through case studies.	TPS 3	70	75

Mapping with Programme Outcomes

CO →	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO1	PSO2	PSO3
CO1	M	L	–	–	–	–	–	–	–	–		L	–	–
CO2	S	M	L	L	–	S	–	M	–	–	S	M	M	M
CO3	S	M	L	M	–	S	–	M	–	–	S	M	M	M

CO4	S	M	L	M	–	S	–	M	–	–	S	M	M	M
CO5	S	M	L	M	–	S	–	M	M	L	S	M	M	M
CO6	S	M	L	M	L	S	M	M	M	M	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	5	15	25	-	-	-	-	-	-	-	-	-	-	-	50	-	-	-	-	-	-	-	-	-	2	7	10	-	-	-
CO3	5	15	25	-	-	-	-	-	-	-	-	-	-	-	50	-	-	-	-	-	-	-	-	-	2	7	10	-	-	-
CO4	-	-	-	-	-	-	5	10	15	-	-	-	-	-	-	-	-	-	-	-	20	-	-	-	2	7	10	-	-	-
CO5	-	-	-	-	-	-	5	10	20	-	-	-	-	-	-	-	-	-	-	-	30	-	-	-	2	7	10	-	-	-
CO6	-	-	-	-	-	-	5	10	20	-	-	-	-	-	-	-	-	-	-	-	50	-	-	-	-	7	10	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Foundations of Machine Learning

Overview of Machine Learning in Science & Engineering: Definition, ML Types, Need, Scientific data vs Engineering data, Scientific ML techniques and Engineering ML techniques - Data Handling and Preprocessing - Data Augmentation - Data Visualization Tools: Pandas, scikit - learn, Seaborn / Matplotlib and Low-Code Tools (Google AutoML).

Classical ML Techniques

Supervised Learning: Linear & Logistic Regression - Support Vector Machines (SVM) - Decision Trees and Random Forests - Naïve Bayes Classifier. Unsupervised Learning: k-Means Clustering - k-Nearest Neighbors (kNN) - Dimensionality Reduction: PCA.

Neural Networks and Deep Learning

Fundamentals of Neural Networks: Perceptron, MLPs, Backpropagation – Regularization, activation functions - Cross-Validation - Regularization – CNN – Operations and Architectures, RNN – LSTM, Introduction to Deep Learning Environments: TensorFlow / PyTorch.

Applied Models in Engineering Backgrounds:

RNN - Time-series data for energy load forecasting and sensor monitoring, CNN – image or signal data for defect detection and structural imaging, Drug Discovery: QSAR modeling, Predictive Maintenance: Anomaly Detection in Sensor Data, Navigation Systems: Simultaneous Localization and Mapping.

Advanced Models

Introduction to GANs - Architecture of GANs – Generator, Discriminator, Ethics: Bias and Fairness, Transparency, Data Privacy - Case Study: Tumor Segmentation Improvement and Medical Image Synthesis

Applications in Science and Engineering

Case Study: Civil Engineering: Fault and Crack Detection in Structures - Mechanical Engineering: Predictive Maintenance and Vibration Analysis - Electrical & Electronics: Power load Forecasting and Fault Diagnosis - Chemical & Environmental: Process Optimization, Pollution Monitoring - Computer & IT: Cybersecurity Anomaly Detection, Optimization Problems - Biomedical Engineering: Disease Classification

Text Books

1. Tom M. Mitchell, Machine Learning, India Edition 2013, Mcgraw Hill International Edition.
2. Aurelien Geron., "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow" (3rd Edition), O'Reilly Media, 2023.

Reference Books & Web Resources

1. Pankaj Mehta et al., "A High-Bias, Low-Variance Introduction to Machine Learning for Physicists", Physics Reports, Vol. 810, 2019.
2. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning", MIT Press, 1st Edition, 2016.
3. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 1st Edition, 2012.
4. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer Science & Business Media, 1st Edition, 2006.
5. Balaji Srinivasan and Ganapathy Krishnamurthi, "Machine Learning for Engineering and Science Applications", NPTEL, IIT Madras, 2023.
nptel.ac.in/courses/106106198.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Foundations of Machine Learning	
1.1	Overview of Machine Learning in Science & Engineering: Definition, , ML Types, Need, Scientific data vs Engineering data, Scientific ML techniques and Engineering ML techniques	1
1.2	Data Handling and Preprocessing - Data Augmentation	1
1.3	Data Visualization Tools: Pandas, scikit - learn, Seaborn / Matplotlib	2
1.4	low-code tools (Google AutoML).	1
2	Classical ML Techniques	
2.1	Supervised Learning: Linear & Logistic Regression	1
2.2	Support Vector Machines (SVM)	1
2.3	Decision Trees and Random Forests	1
2.4	Naïve Bayes Classifier	1
2.5	Unsupervised Learning: k-Means Clustering - k-Nearest Neighbors (kNN)	1
2.6	Dimensionality Reduction: PCA.	1
3	Neural Networks and Deep Learning	
3.1	Fundamentals of Neural Networks: Perceptron, MLPs	1
3.2	Backpropagation - Regularization and activation functions	1
3.3	Cross-Validation - Regularization	1
3.4	CNN – Operations and Architectures	1

Module No.	Topic	No. of Periods
3.5	RNN – LSTM	1
3.6	Introduction to Deep Learning Environments: TensorFlow / PyTorch	1
4	Applied Models in Engineering Backgrounds	
4.1	RNN - Time-series data for energy load forecasting and sensor monitoring -	2
4.2	CNN - image or signal data for defect detection and structural imaging,	2
4.3	Drug Discovery: QSAR modelling, Predictive Maintenance: anomaly detection in sensor data,.	1
4.4	Navigation Systems: SLAM	1
5	Advanced Models	
5.1	Introduction to GANs - Architecture of GANs - generator, discriminator	2
5.2	Ethics: Bias and fairness, transparency, Data privacy	1
5.3	Case study: Tumor segmentation improvement	2
5.4	Medical image synthesis	2
6	Applications in Science and Engineering	
6.1	Case Study: Civil Engineering: Fault and crack detection in structures	1
6.2	Mechanical Engineering: Predictive maintenance and vibration analysis	1
6.3	Electrical & Electronics: Power load forecasting and fault diagnosis	1
6.4	Chemical & Environmental: Process optimization, pollution monitoring	1
6.5	Computer & IT: Cybersecurity anomaly detection, optimization problems - Biomedical Engineering: Disease classification	2
	Total	36

Course Designer(s):

1. Mrs.J.Vijayalakshmi, Assistant Professor, jvicse@tce.edu CSE
2. Mrs.D.Priyadharshini, Assistant Professor, dpicse@tce.edu, CSE

CURRICULUM AND SYLLABI

FOR

INDUSTRY SUPPORTED COURSES

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2025 - 2026



THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CS1A1	GENERATIVE AI TOOLS
----------------	----------------------------

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course explores Generative Artificial Intelligence tools, focusing on algorithms that create new content such as images and text. It provides a historical overview of generative models, and dives into foundational concepts like neural networks and backpropagation. The course also examines advanced techniques like Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), discussing their applications in data generation, image synthesis, and more. The ethical considerations, including bias and privacy, are addressed to provide a comprehensive view of the challenges in Generative AI.

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the key concepts and various types of generative models, including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and autoregressive models for data generation and vector quantized diffusion models	TPS3	B	75
CO2	Apply generative techniques to create synthetic data, generate images, and perform text-to-image synthesis	TPS3	B	75
CO3	Demonstrate the ability to use Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) to solve real-world problems, such as data augmentation and image restoration	TPS3	B	75
CO4	Apply best practices to fine-tune generative models, including hyperparameter tuning, transfer learning, and regularization to avoid overfitting	TPS3	B	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS 01	PS 02	PS 03
CO1	S	M	L	L	L	L		L	M			M	M	L	L
CO2	S	M	L		L	L		L	M			M	M	L	L
CO3	S	M	L		M	L		L	M	M		M	M	L	L
CO4	S	M	L		L	L		L	M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Terminal					
	1	2	3	4	5	6
TPS Scale						
CO1	5	5	15	-	-	-
CO2	5	5	15	-	-	-

CO3	5	5	15	-	-	-
CO4	5	5	15	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Generative AI – Historical overview of generative modeling – Importance of generative models in AI and machine learning – Types of generative models: GANs, VAEs, autoregressive models and vector quantized diffusion models

Understanding Generative Models – Probabilistic modeling and generative processes - Generative Adversarial Networks (GANs) –Generator and discriminator architectures in GANs – Training process: adversarial training, Nash equilibrium – Variants of GANs: DCGANs, WGANs, CGANs - Variational Autoencoders (VAEs) –Encoder and decoder architectures in VAEs – Latent space representation and sampling techniques–Loss functions for training VAEs: reconstruction loss, KL divergence

Applications of GANs and VAEs –Image generation and synthesis using GANs –Data generation and augmentation with VAEs - Training and Fine-Tuning Generative Models – Hyper parameter tuning for generative models – Transfer learning and pretraining techniques - Ethics and Challenges in Generative AI – Bias and fairness issues in generative models – Privacy concerns in generating synthetic data

Hands-on Projects and Demonstrations - Building a simple GAN, VAE model in TensorFlow or PyTorch - Generating images or text using pre-trained generative models - Demonstrating the impact of hyperparameters on model performance

Text Book

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow Concepts, Tools, and Techniques to Build Intelligent Systems - Aurelien Geron, Third Edition O'Reilly 2022.

Reference Books & web resources

1. <https://www.oreilly.com/library/view/generative-deeplearning/9781492041931/ch01.html>
2. <https://www.mygreatlearning.com/blog/generative-ai-models/>
3. <https://www.geeksforgeeks.org/what-is-generative-ai/>
4. <https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/what-is-generative-ai>
5. <https://www.baeldung.com/cs/vae-vs-gan-image-generation>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Generative AI (2)	
1.1	Evolution of Generative AI – Benefits - Historical overview of generative modeling - Importance of generative models in AI and machine learning	1
1.2	Types of generative models: GANs, VAEs, and autoregressive models and vector quantized diffusion models	1
2	Understanding Generative Models (5)	
2.1	Probabilistic modeling and generative processes	1

Module No.	Topic	No. of Periods
2.2	Generative Adversarial Networks (GANs) - Generator and discriminator architectures in GANs	
2.3	Training process: adversarial training, Nash equilibrium	1
2.4	Variants of GANs: DCGANs, WGANs, and CGANs	1
2.5	Variational Autoencoders (VAEs) - Encoder and decoder architectures in VAEs	1
2.6	Latent space representation and sampling techniques	1
2.7	Loss functions for training VAEs: reconstruction loss, KL divergence	
3	Applications of GANs and VAEs (3)	
3.1	Image generation and synthesis using GANs - Data generation and augmentation with VAEs	1
3.2	Training and Fine-Tuning Generative Models - Hyperparameter tuning for generative models - Transfer learning and pretraining techniques	1
3.3	Ethics and Challenges in Generative AI - Bias and fairness issues in generative models - Privacy concerns in generating synthetic data	1
4	Hands-on Projects and Demonstrations (4)	
4.1	Building a simple GAN, VAE model in TensorFlow or PyTorch	2
4.2	Generating images or text using pre-trained generative models - Demonstrating the impact of hyperparameters on model performance	2
	Total	14

Course Designers

1. Mr. Vijay Shankar, Software Engineer, Ericsson
2. Dr.K.Sundarakantham, Professor, CSE
3. Mr. S. Santhana Hari, AP, CSE
4. Ms. R. Harine Rajashree, AP, CSE

vvs221199@gmail.com
 kskcse@tce.edu
sshcse@tce.edu
rhcse@tce.edu

22CS1B1**CONTAINERIZED CI AND CD**

Category L T P Credit

PEES 1 0 0 1

Preamble

This course helps the students to understand the concept and significance of containerization in deploying microservices with CI/CD. It also teaches the students about the benefits of using containerized orchestration and pipelining in the CI and CD process. The course gives practical hands-on with the tools for CI/CD, containerization and orchestration using Jenkins, Docker and Kubernetes.

Prerequisite

- Nil

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify the significance of CI/CD and design software applications with CI/CD principles	TPS3	B	80
CO2	Use the CI/CD tools to develop software using CI/CD methodology	TPS3	B	80
CO3	Develop and deploy microservice applications using containers	TPS3	B	80
CO4	Apply container orchestration and containerized pipeline	TPS3	B	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO1.	S	M	L		L				L	L		L	M		
CO2.	S	M	L		M	L			M	M		M	M	L	L
CO3.	S	M	L		M	L			M	M		M	M	L	L
CO4.	S	M	L		M	L			M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Level	Terminal Examination
Remember	10
Understand	30
Apply	60
Analyse	-
Evaluate	-
Create	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Basics of CI and CD: Overview of software development process using monolithic and microservices architecture principles and challenges, Need for CI/CD, Basics of DevOps, Example of microservice software development with and without CI/CD, an overview using git and an overview using a simple script.

Case study for CI/CD: Jenkins overview and setup, My first Jenkins job, Jenkins pipelines, Declarative vs scripted pipelines, Jenkins plugins and integration with git.

Containerization: Introduction to containerization - Need of containers, Docker basics, A sample software project with and without docker.

Container Orchestration and Pipelining: Scaling up docker – Kubernetes, Understanding the basics of infrastructure as code using minikube, Basics of containerized pipeline.

Text Book

1. Marko Anastasov, Jerome Petazzoni, Pablo Tomas Fernandez Zavalia, "CI/CD with Docker and Kubernetes-How to Deliver Cloud Native Applications at High Velocity", Second Edition, 2022.
2. John Edward Cooper Berg, "DevOps. Building CI/CD Pipelines with Jenkins, Docker Container, AWS ECS, JDK 11, Git and Maven 3", Amazon, 2019.

Reference Books & web resources

1. <https://docs.docker.com/get-started/>
2. <https://kubernetes.io/docs/tutorials/kubernetes-basics/create-cluster/cluster-intro/>
3. <https://minikube.sigs.k8s.io/docs/start/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Basics of CI and CD	
1.1	Overview of software development process using monolithic and microservices architecture principles and challenges, Need for CI/CD	1
1.2	Basics of DevOps	1
1.3	Example of microservice software development with and without CI/CD	1
1.4	An overview using git and an overview using a simple script.	1
2	Case study for CI/CD	
2.1	Jenkins overview and setup	1
2.2	My first Jenkins job, Jenkins pipelines, Declarative vs scripted pipelines	2
2.3	Jenkins plugins and integration with git	1
3	Containerization	
3.1	Introduction to containerization - Need of containers	1
3.2	Docker basics and A sample software project with and without docker	2
4	Container Orchestration and Pipelining	

Module No.	Topic	No. of Periods
4.1	Scaling up docker using Kubernetes	1
4.2	Understanding the basics of infrastructure as code using minikube	1
4.3	Basics of containerized pipeline	1
	Total	14

Course Designer(s):

1. Mr.S.Raghav, Software Development Engineer II, Amazon
sraghavmdu@gmail.com
2. Dr.G.Madhupriya,gmadhupriya@tce.edu
3. Dr.K.Indra, kiit@tce.edu



22CS1C1**AR FOR INDUSTRY 4.0**

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course provides a comprehensive understanding of AR technologies and their applications within the context of Industry 4.0. Throughout this course, students will delve into the theoretical foundations and practical implementations of AR, exploring its role in reshaping manufacturing, maintenance, training, and other critical domains

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Understand the basics of AR for user interaction and advanced interaction used in game development.	TPS2	B	80
CO2	Classify Rendering Optimizations in project and Hand-Gesture-Recognition for wearable devices .	TPS3	B	75
CO3	Apply the knowledge of designing Industrial 4.0 Systems for various applications.	TPS3	B	75
CO4	Experiment the Industrial Metaverse and digital representations of physical industrial assets and spaces that people can control, communicate, and interact .	TPS3	B	75

Mapping with Programme Outcomes

Co s	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
C O1	M	L						L	L	L		L	L		L
C O2	S	M	L		M			L	L	L		L	M	L	L
C O3	S	M	L		M	M		L	L	L		L	M	L	L
C O4	S	M	L		M	M		L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Level	Terminal Examination
Remember	10
Understand	30
Apply	60
Analyse	-
Evaluate	-
Create	-

Syllabus

Overview of AR concepts- Understanding of User Experience Design & Development, Key Factors about Game Development.

Optimization Techniques for AR -Maximizing performance(LOD), HLSL Shader programming Lighting Techniques, Testing Process (Responsive test VR) , Gesture Input(AR/MR).

Applications of AR in Industry operations - Applications of AR and VR for supply chain: a macro perspective approach, Convergence of AR/VR with IoT in manufacturing and their novel usage in IoT

Industrial Metaverse -Proficiency of Metaverse using virtual reality for industry and users perspective, Exploring practical use-cases of augmented reality using photogrammetry - 3D reconstruction tools in the Metaverse

Reference Books & web resources

1. Richa Goel, Sukanta Kumar Baral, Tapas Mishra, and Vishal Jain, "Augmented and Virtual Reality in Industry 5.0", De Gruyter 2023
2. Carsten Röcker, Sebastian Büttner, "Human-Technology Interaction Shaping the Future of Industrial User Interfaces", Springer International Publishing, 2023

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Overview of AR concepts	
1.1	Understanding of User Experience Design & Development	1
1.2	Custom inputs and movements to control key factors in game development.	1
2	Optimization Techniques for AR	
2.1	Maximizing performance(LOD)	1
2.2	HLSL Shader programming Lighting Techniques	1
2.3	Testing Process (Responsive test VR)	1
2.4	Gesture Input(AR/MR).	1
3	Applications of AR in Industry operations	
3.1	Applications of AR and VR for supply chain: a macro perspective approach	1
3.2	Convergence of AR/VR with IoT in manufacturing and their novel usage in IoT	2
4	Industrial Metaverse	
4.1	Proficiency of Metaverse using virtual reality for industry and users perspective	1
4.2	Exploring practical use-cases of augmented reality using photogrammetry	2

Module No.	Topic	No. of Periods
4.3	3D reconstruction tools in the Metaverse	2
	Total	14

Course Designers:

1. Mr.S.Gokulakannan seenivasangokulakannan@gmail.com
2. Dr.N.Shivakumar shiva@tce.edu
3. Dr.RajaLavanya rlit@tce.edu



22CS1D1	DATA VISUALIZATION USING POWER BI
----------------	--

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

In today's data-driven world, organizations need to quickly and accurately make informed decisions based on vast amounts of data. Power BI is a powerful data visualization tool that allows users to create interactive dashboards and reports that help them analyze and communicate insights from data. Data visualization is an essential skill for anyone who wants to understand and communicate insights from data. This course is designed to inculcate the basics of data visualization with Power BI.

Prerequisite

Basic understanding of data analysis concepts

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the concepts of Business Intelligence and Power BI installation	TPS2	B	80
CO2	Build the data transformation using the query editor and Managing Data Relationships	TPS3	B	75
CO3	Create calculated measures and columns visuals in Power BI	TPS3	B	75
CO4	Make use of dashboards that supports on- premise data gateway and connecting with databases.	TPS3	B	75

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L			L			L		L		L	L		L
CO2	S	M	L		M			L	L	M	L	L	M	L	M
CO3	S	M	L		M			L	L	M	L	L	M	L	M
CO4	S	M	L		M			L	L	M	L	L	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern:

Cognitive Levels	Terminal Examination
Remember	20
Understand	40
Apply	40
Analyse	0
Evaluate	0
Create	0

Syllabus

Introduction to Power BI-Introduction to Business Intelligence: Self-Service Business Intelligence (SSBI)-Introduction to Power BI: Traditional BI vs. Power BI-Uses of Power BI-The Flow of Work in Power BI-Basic Components of Power BI-Introduction to Building Blocks of Power BI-Data model and importance of Data Modelling

Power BI Desktop and Data Transformation-Data Sources in Power BI Desktop-Loading Data in Power BI Desktop-Views in Power BI Desktop-Query Editor in Power BI-Transform, Clean, Shape, and Model Data-Manage Data Relationship- Editing a Relationship-Cross Filter Direction-Saving Work file

Data Analysis Expression (DAX) -Introduction to DAX-Importance of DAX- Data Types in DAX-DAX Calculation Types-Steps to Create Calculated Columns-Measures in DAX- DAX Syntax, Functions and Operators-DAX Tables and Filtering

Data Visualization-Introduction to Visuals in Power BI: Visualization Charts in Power BI- Matrixes and Tables- Slicers and Map Visualizations- Modifying Colors in Charts and Visuals: Shapes, Text Boxes, and Images- Custom Visuals-Page Layout and Formatting- Bookmarks and Selection Pane-KPI Visuals

Power BI Service-Introduction to Power BI Service-Creating a Dashboard- Configuring a Dashboard- Power BI Embedded- Bookmarks and buttons **Connectivity Modes**-Data Sources Supported in Power BI- Exploring Live Connections to Data Sources-Connecting Directly to SQL Azure- Connecting Directly to SQL Server Analysis Services/My SQL- Connecting Power BI in Excel

References

1. Data Visualization with Microsoft Power BI, Alex Kolokolov, Maxim Zelensky, O'Reilly Media Inc, 2024
2. Data Visualization : Using Power BI, Orange and Excel, Dr. Shirshendu Roy, Notion Press, 2021
3. Power BI Data Analysis and Visualization, Suren Machiraju, Suraj Gaurav, De Gruyter, 2018

Course Contents and Lecture Schedule

Data Visualization Using Power BI		
Module Number	Topic	No. of Lectures
1	Introduction to Power BI	
1.1	Introduction to Business Intelligence: Self-Service Business Intelligence (SSBI)-Introduction to Power BI: Traditional BI vs. Power BI	1
1.2	Uses of Power BI-The Flow of Work in Power BI-Basic Components of Power BI	1
1.3	Introduction to Building Blocks of Power BI-Data model and importance of Data Modelling	1
2	Power BI Desktop and Data Transformation	

2.1	Data Sources in Power BI Desktop-Loading Data in Power BI Desktop	1
2.2	Views in Power BI Desktop-Query Editor in Power BI-Transform, Clean, Shape, and Model Data	1
2.3	Manage Data Relationship- Editing a Relationship-Cross Filter Direction-Saving Work file	1
3	Data Analysis Expression (DAX)	
3.1	Introduction to DAX-Importance of DAX- Data Types in DAX- DAX Calculation Types	1
3.2	Steps to Create Calculated Columns-Measures in DAX- DAX Syntax, Functions and Operators-DAX Tables and Filtering	1
4	Data Visualization	
4.1	Introduction to Visuals In Power BI: Visualization Charts in Power BI-Matrixes and Tables	1
4.2	Slicers and Map Visualizations- Modifying Colors in Charts and Visuals: Shapes, Text Boxes, and Images	1
4.3	Custom Visuals-Page Layout and Formatting- Bookmarks and Selection Pane-KPI Visuals	1
5	Power BI Service	
5.1	Introduction to Power BI Service-Creating a Dashboard-Configuring a Dashboard- Power BI Embedded- Bookmarks and buttons	1
5.2	Connectivity Modes -Data Sources Supported in Power BI-Exploring Live Connections to Data Sources	1
5.3	Connecting Directly to SQL Azure- Connecting Directly to SQL Server Analysis Services/My SQL- Connecting Power BI in Excel	1
	Total Lectures	14

Course Designers:

1. Dr. G.S.R.EmilSelvan - email@tce.edu
2. Dr. M.P. Ramkumar - ramkumar@tce.edu
3. Mr. M. Midunprasath - midunprasath.10697163@ltimindtree.com

22CS1E1**DESIGN PATTERN**

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course introduces the fundamental principles of software design patterns in software development by providing reusable solutions to common design problems, promoting modular and maintainable code, and facilitating collaboration and communication among team members. Students will learn various design patterns categorized into creational, structural, behavioral, and application pattern. This course provides a comprehensive understanding of software design patterns and their applications in building robust, flexible and maintainable software systems.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify and illustrate the common design problems in software development and justify the use of appropriate design patterns to address the requirements.	TPS3	B	75
CO2	Explore with the features of structural design pattern to promote the modular and reusable design for easier maintenance of software systems.	TPS3	B	75
CO3	Adapt the significance of behavioural design pattern to enable code reusability across different parts of the system.	TPS3	B	75
CO4	Apply Application design pattern in the chosen application to meet the code level requirements of Industry Standardization.	TPS3	B	75

Mapping with Programme Outcomes

Co s	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
C O1	S	M	L	L	L	L		L	M			M	M	L	L
C O2	S	M	L		L	L		L	M			M	M	L	L
C O3	S	M	L		M	L		L	M	M		M	M	L	L
C O4	S	M	L		L	L		L	M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Level	Terminal Examination
Remember	10
Understand	30
Apply	60
Analyse	-
Evaluate	-
Create	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Design Pattern-History of Patterns-Advantage and Metrics - Classification of Patterns

Catalog: Creation Patterns-Structural Patterns-Behavioural Patterns

Creation Pattern- Factory Method-Abstract Factory-Builder-Prototype-Singleton

Structural Pattern- Adapter-Bridge-Composite-Decorator-Façade-Flyweight-Proxy

Behavioural Pattern- Chain of Responsibility-Command-Iterator-Mediator-Memento-Observer-State-Strategy-Template-Methods-Visitor

Application Pattern: MVC, MVP, MVI, MVVM and VIPER –Micro service Design Pattern

Text Book

1. Head First Design Patterns: A Brain-Friendly Guide" by Eric Freeman, Elisabeth Robson, Bert Bates, Kathy Sierra (2021).
2. "Hands-On Design Patterns with Kotlin" by Alexey Soshin (2019)
3. Learning JavaScript Design Patterns: A JavaScript and Node.js Developer's Guide" by Eduardo Diaz (2020).
4. "JavaScript Design Patterns: 10 Coding Patterns You Should Learn to Master Efficient JavaScript Programs" by Christopher Hayes (2021).

Reference Books & web resources

1. "Learning JavaScript Design Patterns: A JavaScript and Node.js Developer's Guide" by Eduardo Diaz (2020).
2. Microservices Patterns: With examples in Java 1st Edition, Kindle Edition by Chris Richardson (2018)
3. <https://microservices.io/patterns/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Introduction	
1.2	Design Pattern	1
1.3	History of Patterns	
1.4	Advantage and Metrics	1
1.5	Classification of Patterns	1
2	Creation Pattern	

Module No.	Topic	No. of Periods
2.1	Factory Method	1
2.2	Abstract Factory-Builder	1
2.3	Prototype-Singleton	1
3	Structural Pattern	
3.1	Bridge-Composite	1
3.2	Adapter- -Decorator-Façade	1
3.3	Flyweight-Proxy	
4	Behavioural Pattern	
4.1	Chain of Responsibility	1
4.2	Command-Iterator-Mediator	1
4.3	Memento-Observer-State	
4.4	Strategy-Template	1
4.5	Methods-Visitor	
5	Application Pattern	
5.1	MVC, MVP	1
5.2	MVI, MVVM	
5.3	VIPER	1
5.4	Micro service Design Pattern	1
	Total	14

Course Designers:

- | | | | |
|----|----------------|---------------------------|----------------|
| 1. | Mr.C.Manimaran | c.manimrn12@gmail.com | |
| | | Senior Software Engineer, | Walmart Global |
| | | Tech,Bangalore | |
| 2. | Dr.K.Indira | kiit@tce.edu | |
| 3. | Dr.RajaLavanya | rlit@tce.edu | |

22CS1F1 VIRTUALIZATION ESSENTIALS

Category L T P Credit

PEES 1 0 0 1

Preamble

This course empowers the students with the knowledge and skills necessary to build the potential of virtualization concepts and resource provision management with VM Management in open source virtualization environment, KVM. It also provides students with an understanding of cloud computing concepts, technologies and practices with an open source cloud platform.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify and illustrate the virtualization principles and its type that can be used for the given application.	TPS3	B	75
CO2	Explore the features of virtualization management in Enterprise Virtualized Environments.	TPS3	B	75
CO3	Utilize the open source virtualization tools to simulate the virtual environment for the given application.	TPS3	B	75
CO4	Use open source cloud platform to deploy and manage applications.	TPS3	B	75

Mapping with Programme Outcomes

Co s	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L		L	L		L	M	L		M	M	L	L
CO 2	S	M	L		L	L		L	M	L		M	M	L	L
CO 3	S	M	L	L	M	L		L	M	M		M	M	L	L
CO 4	S	M	L	L	M	L		L	M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Level	Continuous Assessment Test	Assignment	Terminal Examination
Remember	-	-	10
Understand	20	20	30
Apply	80	80	60
Analyse	-	-	-

Evaluate	-	-	-
Create	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Virtualization - Overview of virtualization, hypervisors, types of hypervisors, types of virtualization, virtualization in cloud, Virtualization Advantages and disadvantages, VM Basics - VM, Resource Provisioning, Virtual CPU, Virtual Memory, Virtual networking and virtual graphics processing.

Virtual Machine Management: Working with Virtual Machines-Creating templates and Clones, Modifying Virtual Machines, Migrating Virtual Machines, Creating Virtual Machine Snapshots.

Enterprise Solutions: VMWare ESXi, Microsoft Hyper-V, Virtual Box and VMWare Workstation-Image building - Image conversion - Image resizing.

Open source virtualization KVM - Introduction, Installation, Configuring, creating guest VMs, managing and monitoring guest VMs.

Cloud Computing: Virtualization Vs Cloud Computing - Cloud - Service and Deployment models -. Open Source Cloud platform: Introduction to OpenStack Cloud Component-Installation-Identity Service-Image Service-Networking Service-Compute Service.

Text Book

1. M. Portnoy, "Virtualization Essentials", United Kingdom: John Wiley and Sons, 2023.
2. G. Blokdyk, "Virtualization Management: A Complete Guide"StarBooks Publisher, 2020.
3. Kusnetzky, Dan. Virtualization: A Manager's Guide. United States, O'Reilly Media, Incorporated, 2011.

Reference Books & web resources

1. P. von Oven, Mastering VMware Horizon 8: An Advanced Guide to Delivering Virtual Desktops and Virtual Apps, 1st ed., Apress, 2021.
2. K. Jackson, C. Bunch, E. Sigler, and J. Denton, OpenStack Cloud Computing Cookbook, 4th ed., Packt Publishing, 2018.
3. G. Blokdyk, Virtualization Management: A Complete Guide. 5StarBooks Publisher, 2020.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Overview of virtualization	1
1.2	Hypervisors	
1.3	Types of hypervisors	
1.4	Types of virtualization	1
1.5	Virtualization in cloud	1
1.6	Virtualization Advantages and disadvantages	
1.7	VM Basics - VM, Resource Provisioning, Virtual CPU,.	1
1.8	Virtual Memory, Virtual networking and virtual graphics processing	1
2	Virtual Machine Management	

Module No.	Topic	No. of Periods
2.1	Working with Virtual Machines	1
2.2	Creating templates and Clones	
2.3	Modifying Virtual Machines	
2.4	Migrating Virtual Machines	1
2.5	Migrating Virtual Machines	
2.6	Creating Virtual Machine Snapshots	1
3	Enterprise Solutions	
3.1	VMWare ESXi	1
3.2	Microsoft Hyper-V	
3.3	Virtual Box and VMWare Workstation	1
3.4	Image building - Image conversion - Image resizing	
4	Open source virtualization KVM	
4.1	Installation and Configuring host	1
4.2	Creating guest VMs	
4.3	Managing and monitoring guest VM	
5	Cloud Computing	
5.1	Virtualization Vs Cloud Computing	1
5.2	Cloud - Service models	
5.3	Deployment models	
5.4	Open Source Cloud platform: Introduction to OpenStack Cloud Component	2
5.5	Installation-Identity Service-Image Service	
5.6	Networking Service-Compute Service	
	Total	14

Course Designers:

1. Mr.E.Mahendran, Joint Director
2. Dr.G.Madhupriya Associate Profesor
3. Dr.K Indira, Assistant Professor

mahendran.e@gmail.com
 CDAC, Chennai
 gmadhupriya@tce.edu
 kiit@tce.edu

22CS1G1	SOFTWARE DEFINED NETWORKS ESSENTIALS
----------------	---

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course helps students understand the significance of SDN, which enables scalable, flexible, and secure network infrastructures. By leveraging these technologies, students will be able to build and manage cloud networks with efficient resource utilization, robust security, and dynamic scalability, aligning with modern demands in cloud computing and network management.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify the significance of Software Define Network.(CO1)	TPS3	70	80
CO2	Adapt the concepts of service provider network virtualization to optimize the backbone networks to accommodate growing user demands and new services.	TPS3	70	80
CO3	Demonstrate the SDN stems from its centralized architecture, which enhances QoS, availability, and consistency while simplifying configuration management.	TPS3	70	80
CO4	Adapt the features of Open Daylight, ONOS, and Mininet SDN Controllers to provide a comprehensive ecosystem for developing, testing, and deploying SDN solutions across varied network environments	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		L				L	L		L	M		
CO2	S	M	L		M	L			M	M		M	M	L	L
CO3	S	M	L		M	L			M	M		M	M	L	L
CO4	S	M	L		M	L			M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Levels	Continuous Assessment Tests	Assignment	Terminal Examination
Remember	-	-	-
Understand	20	20	20
Apply	80	80	80
Analyse	-	-	-
Evaluate	-	-	-
Create	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: History and evolution of SDN-Architecture of SDN-Components

Scalability: Data Centres-Service provider networks-ISP Automation

Reliability: QoS-Service Availability-Consistency-Configuration Management-Access Control Violations-Network Operating System (NOS).

SDN Architecture - Planes – data and control management - Interfaces – northbound and south bound-SDN Controllers: OpenDaylight - ONOS – Mininet- Open vSwitch.

Network Virtualization-Network Function Virtualization and Network Slicing.

Text Book

1. Kshira Sagar Sahoo, Bibhudatta Sahoo and Brojo Kishore Mishra, "Software-Defined Networking for Future Internet Technology Concepts and Applications" CRC Press, 2021.
2. R. Sadique, S. Rajeev, "SDN and NFV Security", Springer, ISBN: 9783031205456, 2023.

Reference Books & web resources

1. Larry Peterson, Carmelo Cascone, Bruce Davie, "Software-Defined Networks: A Systems Approach", Systems Approach LLC, 2021
2. [Anand Nayyar](#), [Bhawna Singla](#), [Preeti Nagrath](#), "Software Defined Networks: Architecture and Applications", John Wiley & Sons, 2022
3. [Mangesh M. Ghonge](#), [Sabyasachi Pramanik](#), [Amol D. Potgantwar](#), "Software Defined Networking for Ad Hoc Networks", Springer Nature, 2022.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	History and evolution of SDN	1
1.2	Architecture of SDN	1
1.3	Components	1
2	Scalability	
2.1	Data Centres	1

Module No.	Topic	No. of Periods
2.2	Service provider networks	2
2.3	ISP Automation	1
3	Reliability	
3.1	QoS	1
3.2	Service Availability	
3.3	Consistency	1
3.4	Configuration Management	
3.5	Access Control Violations	1
3.6	Network Operating System	1
4	SDN Architecture	
4.1	Planes – data and control management	1
4.2	- Interfaces – northbound and south bound	
4.3	SDN Controllers: OpenDaylight	1
4.4	ONOS – Mininet- Open vSwitch.	
5	Network Virtualization	
5.1	Network Function Virtualization	1
5.2	Network Slicing	
	Total	14

Course Designer(s):

Mahendran E Industry Area of expertise	mahendran.e@gmail.com CDAC, Chennai Grid and Cloud computing (15 years of experience)
Dr.K.NarasimhaMallikarjunan	arjunkambaraj@tce.edu
Dr.K.Indra	kiit@tce.edu

22CS1H1**PROMPT ENGINEERING FOR AI APPLICATIONS**

Category L T P Credit

PEES 1 0 0 1

Terminal: Practical Exam

Preamble

Prompt engineering involves the design and refinement of inputs (prompts) for AI language models such as GPT to generate the intended outputs. It's an essential competency in the creation of AI-driven systems, particularly in Natural Language Processing (NLP), and is increasingly significant across diverse domains such as software development, data analysis, and automation. This course aims to equip students with foundational skills in prompt engineering, enabling them to effectively interact with AI models for solving real-world and technical problems.

Prerequisite

Foundations of Artificial Intelligence

Course Outcomes

On the successful completion of the course, students will be able to

CO#	Course Outcome 1 (CO1)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Understand foundational principles of prompt engineering and differentiate between black-box and white-box LLMs..	TPS2	B	75
CO2	Structure effective prompts for diverse tasks, adapting approaches for different LLM architectures.	TPS3	B	70
CO3	Adapt white-box models like LLaMA for specific engineering and technical applications.	TPS3	B	70
CO4	Evaluate ethical implications related to LLM transparency and prompt engineering.	TPS4	B	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L								L		L	L		L
CO2	S	M	L		L	L		L	M	M		M	M	L	M
CO3	S	M	L		L	L		L	M	M		M	M	L	M
CO4	S	S	M	L	M	M	L	L	M	M		M	S	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

Cognitive Level	Internal Exam	External /Terminal Exam
	Mini Project	Practical Exam
Remember	-	-
Understand	30	30
Apply	40	40
Analyse	30	30
Evaluate	-	-
Create	-	-

Syllabus

Introduction to AI Language Models and Prompt Engineering: Overview of black-box LLMs like ChatGPT, GPT-4, and basic white-box models like LLaMA, LLaMA-2, **Prompt Engineering Basics:** Importance of prompt engineering in AI interactions, key applications across various domains, **Types of Prompts:** zero-shot, few-shot, and multi-turn prompting.

Prompting Techniques and White Box Models: Core Prompting Techniques: Structuring questions, setting context, and specifying response format. **Advanced Prompt Structuring:** Adjusting tone, style, and length to refine responses. **Prompt Engineering in White-Box Models:** LLaMA and similar models, Architecture and customization for specific tasks.

Transition from Black-Box to White-Box LLMs: Overview of transparent models like LLaMA, their architecture, and practical applications in engineering. **Optimization and Adaptation Techniques for White-Box Models:** fine-tuning, parameter adjustments, and customizations that enhance model performance. **Applications in Engineering Tasks:** Case studies focused on technical writing, problem-solving, and data analysis using white-box LLMs.

Ethical Considerations and Future Trends in AI and LLMs: Addressing bias, ambiguity, and sensitivity in prompt engineering, especially when transitioning between black-box and white-box LLMs, Ethical challenges and emerging trends in white-box LLMs and AI.

Mini Project:

- **Prompt Engineering Project Using White-Box and Black-Box LLMs:** Students will develop a structured prompt for an engineering or business task using both types of LLMs to compare and contrast performance and ethical implications.
- **Evaluation Focus:** The assignment will include prompt design, analysis of response differences, and ethical evaluation based on model transparency.

Learning Resources

1. James Phoenix, Mike Taylor, "Prompt Engineering for Generative AI: Future-Proof Inputs for Reliable AI Outputs", O'Reilly Media; 1st edition, 2024
2. Da Sachin Sharma, "Prompt DOT AI (Artificial Intelligence): Art of writing Generative AI Prompts", Notion Press, 1st edition, 2023.
3. <https://developers.google.com/machine-learning/resources/prompt-eng>
4. <https://www.promptingguide.ai/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to AI Language Models and Prompt Engineering (3)	
1.1	Overview of black-box LLMs like ChatGPT, GPT-4, and basic white-box models like LLaMA, LLaMA-2	1
1.2	Prompt Engineering Basics: Importance of prompt engineering in AI interactions, key applications across various domains,	1
1.3	Types of Prompts: zero-shot, few-shot, and multi-turn prompting.	1
2	Prompting Techniques and White Box Models(3)	
2.1	Core Prompting Techniques: Structuring questions, setting context, and specifying response format.	1
2.2	Advanced Prompt Structuring: Adjusting tone, style, and length to refine responses.	1
2.3	Prompt Engineering in White-Box Models: LLaMA and similar models, Architecture and customization for specific tasks	1
3	Transition from Black-Box to White-Box LLMs (3)	
3.1	Overview of transparent models like LLaMA, their architecture, and practical applications in engineering	1
3.2	Optimization and Adaptation Techniques for White-Box Models: fine-tuning, parameter adjustments, and customizations that enhance model performance	1
3.3	Applications in Engineering Tasks: Case studies focused on technical writing, problem-solving, and data analysis using white-box LLMs.	1
4	Ethical Considerations and Future Trends in AI and LLMs (5)	

Module No.	Topic	No. of Periods
4.1	Addressing bias, ambiguity, and sensitivity in prompt engineering - Transitioning between black-box and white-box LLMs	1
4.2	Ethical challenges and emerging trends in white-box LLMs and AI.	1
4.3	Practical Lab - structured prompt for an engineering or business task using both types of LLMs	2
4.4	Iterative Prompt Refinement- Iterating Prompts for Accuracy and Relevance	1
	Total	14

Course Designer(s):

1. Mr. S. Maheswaran Senior Manager R&D, Aptean India Pvt Ltd
Maheswaran.Samuthiram@aptean.com bscse@tce.edu
2. Dr B.Subbulakshmi, Assistant Professor, Dept. of Computer Science and Engineering
3. Dr. M. Nirmala Devi, Assistant Professor (Selection Grade), Dept. of mnit@tce.edu
Computer Science and Engineering

22CS1J1	LOWCODE - NOCODE FOR APPLICATION MODERNIZATION	Category L T P Credit PEES 1 0 0 1
---------	---	---------------------------------------

Preamble

This course examines the growing adoption of Low-Code/No-Code (LCNC) platforms and agile approaches by IT and Business teams. Participants will learn how to leverage the power of Generative AI to significantly accelerate their digital transformation journey.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the principles of Low-Code/No-Code (LCNC), including democratization and citizen programming, to evaluate its benefits for application development.	TPS3	70	70
CO2	Illustrate how industry reports evaluate the offerings, value proposition, challenges, and future trends of Low-Code/No-Code (LCNC) technology.	TPS3	70	70
CO3	Develop cloud-native, AI/ML and Computer Vision apps using LCNC platforms and GenAI	TPS3	70	70
CO4	Use LCNC platforms to build and modernize applications.	TPS3	70	70

Mapping with Programme Outcomes

Co s	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	S	M	L		L	L		L	M	L		M	M	L	L
CO 2	S	M	L		L	L		L	M	L		M	M	L	L
CO 3	S	M	L	L	M	L		L	M	M		M	M	L	L
CO 4	S	M	L	L	M	L		L	M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Internal						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6
TPS Scale												
CO1	5	5	15	-	-	-	5	5	15	-	-	-
CO2	5	5	15	-	-	-	5	5	15	-	-	-
CO3	5	5	15	-	-	-	5	5	15	-	-	-
CO4	5	5	15	-	-	-	5	5	15	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: LCNC definition -Principles of LCNC -Democratization and Citizen Programming-Benefits-Reduced development time, Enhanced collaboration and other.

Industry Reports Review: Understand the offerings, Value, Challenges, Limitations and future trends.

Building Applications with LCNC Tools: Overview -Cloud Native Applications-Model driven AI/ML-Computer Vision applications using visual programming approach and generative AI capabilities like Co-Pilots.

Platforms: Mendix-Rapid Miner-H2O -Open Source -AutoML-DataRobot-RobFlow-GitHub Copilot.

UseCases: Modernization of applications (LCNC AI/ML)

Hands-on Lab Sessions – LCNC platforms

Reference Books & web resources

1. Björkholm, Mika; Petranen, Aleksi (January 2025). "Technology Report: Automated Code Generation & AI Tools on Low-Code/No-Code". *ResearchGate*.
2. <https://www.g2.com/categories/no-code-development-platforms#learn-more>
3. No-Code Low-Code Citizen Development eBooks Citizen Development - T <https://quixy.com/blog/top-no-code-low-code-citizen-development-ebooks/op> Essential Game Changing No-Code Low-Code Citizen Development eBooks in 2025 | Quixy

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	LCNC definition	1

Module No.	Topic	No. of Periods
1.2	Principles of LCNC	
1.3	Democratization and Citizen Programming	
1.4	Benefits	
2	Industry Reports Review	
2.1	Understand the offerings	1
2.2	Value, Challenges, Limitations and future trends.	1
3	Building Applications with LCNC Tools	
3.1	Overview	1
3.2	Cloud Native Applications	
3.3	Model driven AI/ML	1
3.4	Computer Vision applications using visual programming approach and generative AI capabilities like Co-Pilots	
4	Platforms	
4.1	Mendix	1
4.2	Rapid Miner	
4.3	H2O	1
4.4	Open Source	
4.5	AutoML	4
4.6	DataRobot	
4.7	RobFlow	
4.8	GitHub , Copilot.	
5	UseCases	
5.1	Modernization of applications (LCNC AI/ML)	
5.2	Hands-on Lab Sessions – LCNC platforms	2

Module No.	Topic	No. of Periods
	Total	14

Course Designers:

1. Dr.Sambath Narayanan Sambath.narayanan@gmail.com
Parthasarathy Dataever Consulting,Bangalore.
Director
2. Dr.C.Santhiya csit@tce.edu
Assistant Professor

22CS1K1**ENTERPRISE APPLICATION
DEVELOPMENT**

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course objective is to strengthen the individual's knowledge and practical skills in full stack web development, encompassing of creating dynamic, responsive, and secure web applications. This course provides the hands-on training on how to use React components to create interactive interfaces, connecting with ORM Data model, Creating and Consuming APIs and finally host the application in the CI/CD pipeline.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Illustrate the basic principles of Front-End, Back-End and their working framework for an application.	TPS3	70	80
CO2	Experiment with the features of React Components and State management for an interactive application.	TPS3	70	80
CO3	Apply the basic CRUD operation and consuming the API with secured authentication mechanism for an application.	TPS3	70	80
CO4	Build an application and consuming RESTful APIs service and integrate with user authentication methods.	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1.	S	M	L		L	L		M	M	M		M	M	L	M
CO2.	S	M	L		L	L		M	M	M		M	M	L	M
CO3.	S	M	L		L	L		M	M	M		M	M	L	M
CO4.	S	M	L	L	L	L		M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	INTERNAL						TERMINAL					
TPS SCALE	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	5	-	-	-	-	5	5	-	-	-	-
CO2	5	5	25	-	-	-	5	5	25	-	-	-
CO3	5	5	20	-	-	-	5	5	20	-	-	-
CO4	5	5	15	-	-	-	5	5	15	-	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Front-end and back-end Technologies-Development tools and Environment-Version Control

React: Typescript: components and Props-Tree Shaking –Bundling-State management-interactive web applications

Data Connectivity: Django and Python-Setup-Views-Routing-URL.

API Connectivity: RESTful Endpoints-Serializing Data-CRUD-Authentication and Authorization.

Integration: Consuming RESTful APIs with React-User authentication and JWT in React-Django apps.

Data and Deployment Management: ORM- Data Modeling- Migrating database models Continuous integration and Continuous deployment.

Text Book

1. Mangabo, Kolawole. Full Stack Django and React: Get Hands-on Experience in Full-stack Web Development with Python, React, and AWS. United Kingdom, Packt Publishing, 2023.
2. Bhowmick, Sneha Deep. The Full Stack Development Book. India, WHITE FALCON PUB, 2019.
3. Andrawos, Mina. Hands-On Full Stack Development with Go: Build Full Stack Web Applications with Go, React, Gin, and GopherJS. United Kingdom, Packt Publishing, 2019.

Reference Books & web resources

1. Hinkula, Juha. Full Stack Development with Spring Boot and React: Build Modern and Scalable Full Stack Applications Using the Power of Spring Boot and React, 3rd Edition. United Kingdom, Packt Publishing, 2022.
2. Zammetti, Frank. Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, and Docker. United States, Apress, 2020.
3. <https://www.coursera.org/learn/the-full-stack#modules>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction	
1.1	Front-end and back-end Technologies-	1
1.2	Development tools and environment	
1.3	Version Control	
2	React:	
2.1	Typescript: components and Props-Tree Shaking – Bundling	1
2.2	State management	
2.3	Interactive web applications	1
3	Data Connectivity	
3.1	Django and Python	1
3.2	Setup-Views	
3.3	Routing-URL.	1
4	API Connectivity	
3.1	RESTful Endpoints	2
3.2	Serializing Data	1
3.3	CRUD-Authentication and Authorization.	
3.4	Integration	
3.5	Consuming RESTful APIs with React	1
3.6	User authentication and JWT in React	
3.7	Django apps.	1

Module No.	Topic	No. of Periods
4	Data and Deployment Management:	
4.1	ORM	2
4.2	Data Modeling	
4.3	Migrating database models	2
4.4	Continuous integration and Continuous deployment.	
	Total	14

Course Designer(s):

1. Mr.Vijay Shankar,Software Developer,Ericson,Bangalore.
2. Dr.RajaLavanya, Assistant Professor, Computer Science and Engineering, rlit@tce.edu
3. Ms.G.Bhavani, Assistant Professor, Computer Science and Engineering, gbicse@tce.edu

22CS1L1	CLOUD COMPUTING WITH AWS
----------------	---------------------------------

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course aims to provide basic understanding of deploying cloud applications using Amazon Web Service.

Prerequisite

- Computer Networking fundamentals

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome Statement	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Comprehend the basic terminologies in cloud computing and AWS	TPS2	70	80
CO2	Create and deploy server less functions.	TPS3	70	75
CO3	Examine the working of AWS storage and network services.	TPS3	70	75
CO4	Comprehend the basic terminologies in cloud computing and AWS	TPS3	70	75

Mapping with Programme Outcomes

Co s	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO 1	M	L							M	L	L	M	M		L
CO 2	S	M	L		L	L			M	L	L	M	M	L	L
CO 3	S	M	L		M	L			M	M	L	M	M	L	L
CO 4	S	M	L		M	L			M	M	L	M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	INTERNAL						TERMINAL					
	1	2	3	4	5	6	1	2	3	4	5	6
TPS SCALE												
CO1	5	10	-	-	-	-	5	10	-	-	-	-
CO2	5	10	10	-	-	-	5	10	10	-	-	-
CO3	5	10	15	-	-	-	5	10	15	-	-	-

CO4	5	10	15	-	-	-	5	10	15	-	-	-
-----	---	----	----	---	---	---	---	----	----	---	---	---

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to AWS & Cloud Computing -Overview of cloud computing. Benefits of using cloud- Introduction to AWS.

AWS compute services - EC2: creating, configuring, and launching Instances-Elastic Beanstalk: deploying and managing web Applications-Lambda: creating and deploying server less functions-Auto Scaling: automatic scaling of EC2 instances

AWS storage services -S3: creating and managing buckets, objects, and lifecycles-EBS: creating and attaching volumes to Instances-Glacier: creating and managing archives

AWS networking services - VPC: creating and configuring virtual private clouds-Route 53: creating and managing DNS records-CloudFront: creating and managing content delivery networks

AWS security & Identity services -IAM: creating and managing users, groups, and policies-Security Groups: managing inbound and outbound traffic rules-CloudTrail: logging AWS account activity- KMS: managing encryption keys

Reference Books & web resources

1. Mark Wilkins, "Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud" First Edition, Pearson, 2019.
2. <https://www.coursera.org/specializations/aws-fundamentals>
3. <https://www.coursera.org/professional-certificates/aws-cloud-solutions-architect>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours
1	Introduction to AWS & Cloud Computing	
1.1	Overview of cloud computing, Benefits of using cloud.	1
1.2	Introduction to AWS.	1
2	AWS compute services	
2.1	EC2: creating, configuring, and launching Instances.	1
2.2	Elastic Beanstalk: deploying and managing web Applications	1
2.3	Lambda: creating and deploying server less functions	1
2.4	Auto Scaling: automatic scaling of EC2 instances	1
3	AWS storage services	
3.1	S3: creating and managing buckets, objects and lifecycles	1
3.2	EBS: creating and attaching volumes to Instances	1
3.3	Glacier: creating and managing archives	1
4	AWS networking services	
4.1	VPC: creating and configuring virtual private clouds, Route 53: creating and managing DNS records.	1
4.2	CloudFront: creating and managing content delivery networks.	1
5	AWS security & Identity services	
5.1	IAM: creating and managing users, groups, and policies.	1
5.2	Security Groups: managing inbound and outbound traffic rules.	1
5.3	CloudTrail: logging AWS account activity, KMS: managing encryption keys.	1

	TOTAL	14
--	-------	----

Course Designers

1. Mr. T. Sasi Kumar, tmsasikumar@gmail.com
Lead Consultant,
ThoughtWorks
2. Dr. C. Senthil Kumar cskcse@tce.edu
3. Dr. M. Vijayalakshmi mviji@tce.edu

:

22CS1M1	RETRIEVAL-AUGMENTED GENERATION
----------------	---------------------------------------

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

This course covers Retrieval-Augmented Generation (RAG), enhancing AI with real-world info. It explores how RAG works through retrievers and generators, using tools like LangChain, FAISS, and ChromaDB. Students will learn about LLMs, vector databases, and real-world applications in chatbots, education, and support. The course also touches on ethical issues and future trends in AI.

Prerequisite

- Basic understanding of large language models

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the core concepts of Retrieval-Augmented Generation (RAG) by integrating retrieval mechanisms with language models to improve response accuracy.	TPS3	70	75
CO2	Apply vector-based search techniques and embedding models to retrieve relevant documents from large corpora using tools like FAISS and ChromaDB.	TPS3	70	75
CO3	Determine the ability to build and implement RAG pipelines using retrievers and generators for real-world applications such as chatbots and educational assistants.	TPS3	70	75
CO4	Apply best practices in grounding language models with external data while addressing ethical concerns related to data privacy, hallucination, and fairness.	TPS3	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS 01	PS 02	PS 03
CO1	S	M	L		M	M		M	M			M	M	L	L
CO2	S	M	L		L	M		M	M			M	M	L	L
CO3	S	M	L		M	M		M	M	M		M	M	L	L
CO4	S	M	L		L	M		M	M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Internal						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	5	15	-	-	-	5	5	15	-	-	-
CO2	5	5	15	-	-	-	5	5	15	-	-	-
CO3	5	5	15	-	-	-	5	5	15	-	-	-
CO4	5	5	15	-	-	-	5	5	15	-	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to RAG - Retrieval-Augmented Generation- RAG importance in modern AI - Real-world examples (ChatGPT w/ browsing, Perplexity.ai)-Case study: Use RAG to fetch and explain current medical literature to doctors.

Basics of Language Models - Large Language Models (LLMs) - Limitations of LLMs (hallucination, outdated data) - Difference between Retrieval-based and Generation-based systems-Case study:Use RAG to pull recent papers or citations into research writing assistants.

Information Retrieval Concepts - Introduction to document search and retrieval - Vector Database- How embeddings help in finding relevant info-Case study: Match user queries with product catalogues using dense vector search.

Components of a RAG System - Retriever: Fetching relevant documents - Generator: Creating answers based on documents - Overview of the RAG architecture.

Applications of RAG - Use cases: Chatbots, Customer Support, Education - Industry examples using RAG - Benefits of grounding LLMs in real data.

Tools & Ecosystem (Brief Overview) - Popular tools: La.ngChain, Haystack, FAISS, ChromaDB - Language model providers: OpenAI, HuggingFace, Meta AI (RAG model).

Future Trends & Ethical Considerations - Future of grounded generation - Ethical concerns: accuracy, privacy, and data handling.

Text Book

1. Christopher D. Manning, PrabhakarRaghavan, and HinrichSchütze, "Introduction to Information Retrieval," Cambridge University Press, 2008.

Reference Books& web resources

1. <https://mallahyari.github.io/rag-ebook/intro.html>
2. https://huggingface.co/learn/cookbook/en/advanced_rag
3. <https://arxiv.org/pdf/2501.09223v1>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to RAG	
1.1	Retrieval Augmented Generation- RAG importance in modern AI	1
1.2	Real-world examples. Case study: Use RAG to fetch and explain current medical literature to doctors.	1
2	Basics of Language Models	
2.1	Large Language Models (LLMs) - Limitations of LLMs (hallucination, outdated data) -	1
2.2	Difference between Retrieval-based and Generation-based systems- Case study: Use RAG to pull recent papers or citations into research writing assistants.	1
3	Information Retrieval Concepts	
3.1	Introduction to document search and retrieval - Vector Database	1
3.2	How embeddings help in finding relevant info- Case study: Match user queries with product catalogues using dense vector search.	1
4	Components of a RAG System	
4.1	Retriever: Fetching relevant documents	1
4.2	Generator: Creating answers based on documents - Overview of the RAG architecture.	1
5	Applications of RAG	
5.1	Use cases: Chatbots, Customer Support, Education	1
5.2	Industry examples using RAG - Benefits of grounding LLMs in real data.	1
6	Tools & Ecosystem (Brief Overview)	
6.1	Popular tools: LangChain, Haystack, FAISS	1
6.2	ChromaDB - Language model providers: OpenAI, HuggingFace, Meta AI (RAG model)	1
7	Future Trends & Ethical Considerations	

Module No.	Topic	No. of Periods
7.1	Future of grounded generation	1
7.2	Ethical concerns: accuracy, privacy, and data handling.	1
	Total	14

Course Designer(s):

1. Ashokkumar Pudhuraja
2. Dr.K.Sundarakantham, Professor, CSE
3. Mrs.S.Saradha,AP,CSE

ZOHO
 kskcse@tce.edu
 ssacse@tce.edu

CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

INDUSTRY SUPPORTED COURSES

**FOR THE STUDENTS ADMITTED FROM THE
ACADEMIC YEAR 2025 - 2026 ONWARDS**

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CS1N1	IDENTITY AND ACCESS MANAGEMENT	Category	L	T	P	Credit
		PEES	1	0	0	1

Preamble

Cybersecurity is the practice of protecting people, systems and data from cyberattacks by using various technologies, processes and policies. Identity and Access Management (IAM) is one of the cybersecurity disciplines that deals with provisioning and protecting digital identities for people and access permissions in IT systems. IAM tools help ensure that the right people can access the right resources for the right reasons at the right time.

This course aims to bring greater awareness and knowledge around Cybersecurity – IAM domain to students and train them with basic hands-on skills in an IAM tool effectively.

Prerequisite

IBM Cyber Security Foundational Course (online – 1 to 2 hrs max)

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
1	Describe the fundamental concepts of Identity and Access Management (IAM) and its key domains, and summarize the essential entities and processes involved in Identity Security.	TPS2	70	75
2	Discuss the key protocols associated with Identity Security, including SAML, OpenID Connect (OIDC), and related authentication and authorization frameworks.	TPS2	70	75
3	Develop identity creation and application security using risk-based access policies and approval workflows in an IAM security tool, and apply authentication and authorization APIs to protect custom web applications.	TPS3	70	70
4	Explore emerging trends and advanced topics in Security, including Blockchain and Agentic AI, and identify practical approaches to safeguard personal data and privacy in everyday digital activities.	TPS2	70	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1.	M	L			L	L	L	L	L		M	L	L	L
CO2.	M	L			L	L	L	L	L		M	L	L	L

CO3.	S	M	L	L	M	M	M	M	M	L	M	M	M	M
CO4.	M	L			L	L	L	L	L		M	L	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Internal						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6
CO1		20					5	15				
CO2		25					5	20				
CO3		20	20				5	15	20			
CO4		15					5	10				

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to IAM domain and concepts:

Basic Introduction to Cyber Security - CIA Triad and Basics, Confidentiality, Integrity, Availability.

Introduction to IAM and Four A's that matter most - Administration, Authentication, Authorization, Audit. **IAM sub-domains and concepts** - Identity Management (IM), Access Management (AM), Identity Governance and Administration (IGA), Privilege Access Management(PAM), Customer IAM (CIAM), Cloud IAM and Cloud IEM (CIEM). **IAM Entities and Key Processes - IAM key entities** - Directory, Identity, Resources, Entitlement, Account, Risk, **IAM Key Processes** - IM - Directory, Lifecycle Management, Identity Issuance, Identity Attestation, AM - RBAC, ABAC, IGA - Entitlements, Account Provisioning, Approvals/Certifications, SOD Risk Detection and Mitigation, PAM - Secrets Mgmt., Secured Admin access to systems, Multi-level Approvals, Zero Standing Privileges, Audit/Compliance, CIAM - Silent registration, Progressive Profiling, Data Privacy & Consents, Self-service, Fraud detection/prevention.

Practical: Identity creation and Access provisioning - Practical Sessions on IBM Verify SaaS, create user and groups, create roles and permissions, Create Application, Enable Access and configure Single-Sign-On, configure access policy with Multi-Factor Authentication, Configure Entitlements, Configure Access Requests and Approval flows

Introduction to Identity Protocols/Standards - SAML, OIDC, FIDO2, SCIM.

Practical: Develop a Web App & protect using branded login page and identity APIs - CIAM Demos, develop a custom Web App, Protect the App using IBM Verify SaaS - using branded Login page and Verify APIs of Authentication and Authorization.

Identity Advanced, Trends and Digital Hygiene - Advanced Topics - Blockchain and Identity, Quantum and Cyber Security, AI and Cyber Security, Recent trends in IAM - Identity-centric Cyber Security, Zero Trust, Identity Fabric, NHI & Agentic NHI, Zero-Knowledge Proofs, Attacks – case studies, security events, anomaly detection. Practical Guide to Digital Life Hygiene - Identity Theft Protection, Data Privacy controls, Web/Mobile Hygiene practices

Text Books

1. Yvonne Wilson, Abhishek Hingnikar, Solving Identity Management in Modern Applications: Demystifying OAuth 2.0, OpenID Connect, and SAML 2.0, APress, 2019.

2. Morey J. Haber, Darran Rolls, Identity Attack Vectors: Implementing an Effective Identity and Access Management Solution, APress, 2019

Web resources

1. <https://medium.com/@tahirbalarabe2/identity-and-access-management-the-four-as-that-matter-468a581ee1e2>
2. <https://www.okta.com/sites/default/files/2023-05/identity-as-a-service-dummies-EN.pdf>
3. <https://auth0.com/resources/ebooks/ciam-for-dummies>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours
1	Introduction to IAM domain and concepts	
1.1	Basic Introduction to Cyber Security - CIA Triad and Basics (Pre-requisites) Confidentiality, Integrity, Availability	1
1.2	Introduction to IAM and Four A's that matter most Administration, Authentication, Authorization, Audit	
1.3	IAM sub-domains and concepts - Identity Management (IM) - Access Management (AM) - Identity Governance and Administration (IGA) - Privilege Access Management(PAM) - Customer IAM (CIAM) - Cloud IAM and Cloud IEM (CIEM)	
2	IAM Entities and Key Processes	
2.1	IAM key entities - Directory, Identity, Resources, Entitlement, Account, Risk	1
2.2	IAM Key Processes - IM - Directory, Lifecycle Management., Identity Issuance, Identity Attestation - AM - RBAC, ABAC - IGA - Entitlements, Account Provisioning, Approvals/Certifications, SOD Risk Detection and Mitigation - PAM - Secrets Mgmt., Secured Admin access to systems, Multi-level Approvals, Zero Standing Privileges, Audit/Compliance - CIAM - Silent registration, Progressive Profiling, Data Privacy & Consents, Self-service, Fraud detection/prevention - Cloud IAM & CIEM - Cloud services protection through native IAM, Cloud Infrastructure protection through permission creep	
3	Practical : Identity creation and Access provisioning	
3.1	Practical Sessions on IBM Verify SaaS - Create user and groups - Create roles and permissions - Create Application - Enable Access and configure Single-Sign-On - Configure access policy with Multi-Factor Authentication	5

	● Configure Entitlements ● Configure Access Requests and Approval flows	
4	Introduction to Identity Protocols/Standards	
4.1	● SAML ● OIDC ● FIDO2 ● SCIM	2
5	Practical : Develop a Web App & protect using branded login page and identity APIs	
5.1	● CIAM Demos	4
5.2	● Develop a custom Web App ● Protect the App using IBM Verify SaaS - using branded Login page and Verify APIs of Authentication and Authorization	
6	Identity Advanced, Trends and Digital Hygiene	
6.1	● Advanced Topics - Blockchain and Identity, Quantum and Cyber Security, AI and Cyber Security	1
6.2	● Recent trends in IAM - Identity-centric Cyber Security, Zero Trust, Identity Fabric, NHI & Agentic NHI, Zero-Knowledge Proofs, Attacks – case studies, security events, anomaly detection	
6.3	● Practical Guide to Digital life Hygiene - Identity Theft Protection, Data Privacy controls, Web/Mobile Hygiene practices	
Total		14

Course Designer(s):

1. Mr. Sivapatham, M. | Senior Product Manager – IBM | sivapatham@in.ibm.com
2. Dr. Subbulakshmi, B., Ph.D. | Assistant Professor – TCE |bscse@tce.edu
3. Mr.Nagendrakumar, D |Assistant Professor – TCE| dnkcse@tce.edu

22CS1P1 MOBILE BACKEND SERVICE

Category L T P Credit

PEES 0 0 2 1

Preamble

The course introduces learners to Mobile Backend as a Service (MBaaS), a modern cloud-based approach for building scalable and efficient mobile applications. It focuses on applying MBaaS platforms like Firebase, AWS Amplify, and Azure Mobile Apps to handle authentication, data storage, push notifications, and server less functions. Students gain hands-on experience in integrating APIs, managing real-time databases, and deploying cloud-connected mobile back ends using CI/CD and monitoring tools. The course emphasizes practical implementation to prepare students for developing scalable, secure, and cloud-enabled mobile solutions.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Identify the concepts of mobile application architecture and MBaaS fundamentals to design scalable and cloud-connected mobile systems.	TPS3	70	75
CO2	Explore MBaaS architecture components such as authentication, data modelling, and real-time synchronization to develop secure and responsive mobile back ends.	TPS3	70	75
CO3	Adapt the significance of MBaaS services and integration techniques to connect APIs, manage cloud storage, deploy server less functions, and enable push notifications in mobile applications.	TPS3	70	75
CO4	Apply deployment and monitoring practices to build, test, and release MBaaS-enabled mobile applications with CI/CD pipelines, analytics, and performance tracking.	TPS3	70	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	S	M	L	M	M	M	M	M	M	M	M	M	M	M
CO2	S	M	L	M	M	M	M	M	M	M	M	M	M	M
CO3	S	M	L	M	M	M	M	M	M	M	M	M	M	M
CO4	S	M	L	M	M	M	M	M	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Assignment 1						Terminal					
	(Mini Project)						(Mini Project)						(Practical Exam)					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1		20	20						40					20	20			
CO2			20						20						20			
CO3			20						20						20			
CO4			20						20						20			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to MBaaS: Overview of mobile application Architectures-Traditional backend vs. MBaaS-Limitations-Core MBaaS features: authentication, storage, push notifications, cloud functions-MBaaS providers: Firebase, AWS Amplify, Azure Mobile Apps, Backendless

MBaaS Architecture and Core Components: MBaaS architecture: client, SDK, API, Database-Data modeling and cloud Databases-Authentication and authorization (OAuth, JWT)-Real-time synchronization and offline caching(web/mobile)- data breaches, data security, and capability management for mobile platforms.

MBaaS Services and Integration: API gateway and service Integration-Cloud storage and media Management-Push notifications and messaging services-Serverless functions for business Logic-Integrating third-party APIs-SDKs for Android, iOS, and web applications

Development, Deployment, and Monitoring: Building and connecting a mobile app with MBaaS-Authentication with social providers (Google, Facebook)-Deploying cloud functions and backend Logic-Configuring analytics and crash reporting-CI/CD for MBaaS apps

Text Book

1. Kameron Hussain, Frahaan Hussain, "Mastering Firebase: The Complete Guide to Building and Scaling Apps", 2024.
2. Laurence Moroney, "The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform", 2022.

Reference Books & web resources

1. Julian Braun," Programming Backend with Go: Build robust and scalable back ends for your applications using the efficient and powerful tools of the Go ecosystem", Asian Publishing House,2025
2. Miriam Le," Mobile System Design: Real-World Architecture for Scalable Apps: A Practical Guide to Designing APIs, Sync, and Offline-Ready Mobile Experiences",2025

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to MBaaS:	
1.1	Overview of mobile application Architectures	1
1.2	Traditional backend vs. MBaaS-Limitations	1
1.3	Core MBaaS features: authentication, storage, push notifications, cloud functions	1

Module No.	Topic	No. of Periods
1.4	MBaaS providers: Firebase, AWS Amplify, Azure Mobile Apps, Backend less	1
2	MBaaS Architecture and Core Components:	
2.1	MBaaS architecture: client, SDK, API, database	1
2.2	Data modelling and cloud databases (Firestore, DynamoDB)	1
2.3	Authentication and authorization (OAuth, JWT)-	1
2.4	Real-time synchronization and offline caching(web/mobile)	1
2.5	Data breaches, data security, and capability management for mobile platforms.	
3	MBaaS Services and Integration:	
3.1	API gateway and service integration	1
3.2	Cloud storage and media management	1
3.3	Push notifications and messaging services	
3.4	Server less functions for business logic	1
3.5	Integrating third-party APIs	
3.6	SDKs for Android, iOS, and web applications	
4	Development, Deployment, and Monitoring:	
4.1	Building and connecting a mobile app with MBaaS	1
4.2	Authentication with social providers (Google, Facebook)	1
4.3	Deploying cloud functions and backend logic	1
4.4	Configuring analytics and crash reporting	
4.5	CI/CD for MBaaS apps	
	Total	14

Course Designer(s):

- 1.Mr.C. Manimaran, Architect, Persistent systems,Bangalore,c.manimrn12@gmail.com
- 2.Dr.K. Indira, Assistant Professor, Department of CSE,TCE,kiit@tce.edu
- 3.Dr. RajaLavanya, Assistant Professor, Department of CSE,TCE,rlit@tce.edu

**PROGRAMMING AGENTIC AI:
TOOLS, FRAMEWORKS, AND
ARCHITECTURES**

22CS1Q1

Category	L	T	P	Credit
PEES	1	0	0	1

Preamble

Agentic Artificial Intelligence (Agentic AI) represents a transformative evolution in the field of intelligent systems. The emergence of Agentic AI signifies a major step toward **AI systems that act rather than merely react**. Through continuous reasoning, self-correction, and contextual understanding, Agentic AI can manage complex workflows, pursue long-term objectives, and execute tasks with minimal human oversight. This course aims to equip students with foundational skills to build and test a working AI Agent,

Prerequisite

Fundamental understanding of programming in Python. Familiarity with Linux, shell commands. Preferably, exposure to MS VS Code.

Course Outcomes

On the successful completion of the course, students will be able to

CO#	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explore the foundational principles of Agentic AI and differentiate between various AI models	TPS3	70	70
CO2	Structure the AI Agents using AI Agent Tool kit	TPS3	70	70
CO3	Adapt AI Agents for different technical applications	TPS3	70	70
CO4	Explore your own working AI Agent that is build/run/test with Google Agent Development Kit (ADK).	TPS3	70	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1.	S	M	L	L	L	L		L	M			M	M	L
CO2.	S	M	L		L	L		L	M			M	M	L

CO3.	S	M	L		M	L		L	M	M		M	M	L
CO4.	S	M	L		L	L		L	M	M		M	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assignment 1						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6
TPS Scale												
CO1			40					5	20			
CO2			20					5	20			
CO3			20					5	20			
CO4			20					5	20			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Foundations: Foundations of AI - Machine Learning, Neural Networks. **Evolution:** Machine Learning to Neural Networks to Language Models - Scaling Laws. **LLMs:** Transformers & LLM - Parameters - LLM Training Process - RLHF - **Applications & Challenges** - Applications, Safety, Security & Governance.

From LLMs to Agents: Constraints of LLMs. LLMs to Agentic Workflows to Agents. Chain of Thought - ReAct - **Tools:** Calculators, Web Search, Real-world connection **Context & Memory:** Context, short & long-term memory - **Collaboration:** Discovery, Communication, Security

AI Agents Architecture: core components—planner, memory, executor— communication protocols and modular design. Model Context Protocol (MCP) - structured interoperability between models and tools. LangGraph, and ADKs - agent workflows **Popular AI Agents:** Gemini CLI, Microsoft Copilot, Google Code Assist, Perplexity Comet - **Gemini CLI:** Demonstration

Building AI Agents: Langchain & Google ADK - **Design:** Sequential, Parallel, Loop Agents - Toolset - Agent2Tool **Run & Test** AI Studio - adkweb - MCP Inspector **Deployment** Agent Runtime - Detailed walkthrough & explanation of an AI agent

Mini Project:

Building your own AI Agent: Students will be able to build a working AI Agent with Google ADK, and get introduced to Advanced AI Agent Architectures

Text Book

1. Micheal Lanham, "AI Agents in Action", Manning Publications, 1st edition, 2025.

Reference Books & web resources

1. Poole, David L., and Alan K. Mackworth, "Artificial Intelligence: Foundations of Computational Agents", Cambridge University Press, 3rd edition, 2023.

2. Sumit Ranjan, Divya Chembachere, Lanwin Lobo, Agentic AI in Enterprise: Harnessing Agentic AI for Business Transformation, Apress, 2025
3. Ken Huang, Chris Hughes, Securing AI Agents: Foundations, Frameworks, and Real-World Deployment, Springer Cham, 2025
4. <https://cloud.google.com/discover/what-are-ai-agents>
5. <https://modelcontextprotocol.io/docs/getting-started/intro>
6. <https://google.github.io/adk-docs/>
7. <https://copilot.microsoft.com/>
8. <https://docs.cloud.google.com/gemini/docs/codeassist/gemini-cli>
9. <https://aistudio.google.com/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Foundations of AI & LLMs (2)	
1.1	Foundations: Foundations of AI - Machine Learning, Neural Networks. Evolution: Machine Learning to Neural Networks to Language Models - Scaling Laws. LLMs: Transformers & LLM - Parameters - LLM Training Process - RLHF	1
1.2	Applications & Challenges - Applications, Safety, Security & Governance.	1
2	From LLM to Workflows / Agents (2)	
2.1	From LLMs to Agents: Constraints of LLMs. LLMs to Agentic Workflows to Agents. Chain of Thought - ReAct - Tools: Calculators, Web Search, Real-world connection Context & Memory: Context, short & long-term memory	1
2.2	Collaboration: Discovery, Communication, Security	1
3	AI Agents Architecture (3)	
3.1	AI Agents Architecture: core components—planner, memory, executor— communication protocols and modular design. Model Context Protocol (MCP) - structured interoperability between models and tools. LangGraph, and ADKs - agent workflows	1
3.2	Popular AI Agents: Gemini CLI, Microsoft Copilot, Google Code Assist, Perplexity Comet	1
3.3	Gemini CLI: Demonstration	1
4	Building your own AI Agent (7)	
4.1	Building AI Agents: Langchain & Google ADK	1

Module No.	Topic	No. of Periods
4.2	Design: Sequential, Parallel, Loop Agents - Toolset - Agent2Tool	1
4.3	Run & Test AI Studio - adkweb	2
4.4	MCP Inspector Deployment Agent Runtime - Detailed walkthrough & explanation of an AI agent	3
	Total	14

Course Designer(s):

1. Mr. A. Gerald Naveen, Staff Engineer & Manager, ageraldnaveen@gmail.com
Google Cloud Security, Google
2. Dr.K.Sundarakantham, Professor(CAS), Department kskcse@tce.edu
of Computer Science and Engineering
3. Mr.S.Santhana Hari, Assistant Professor, sshcse@tce.edu
Department of Computer Science and Engineering
4. Ms.C.Karthika, Assistant Professor, Department of ckcse@tce.edu
Computer Science and Engineering

CURRICULUM AND SYLLABI

FOR

INTERDISCIPLINARY ELECTIVES

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2025 - 2026



THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

22CSGA1	APPLICATIONS OF VIRTUAL REALITY
----------------	--

Category	L	T	P	Credit
IE	3	0	0	3

Preamble

Students will gain a foundational knowledge of virtual reality environment, the ways to interact, and their various areas of applications. Students will gain a foundational knowledge of building VR projects

Prerequisite

Nil.

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the components, benefits of Virtual Reality.	TPS2	B	75
CO2	Demonstrate the working of the input and output devices that are interfaced with virtual reality Environments	TPS2	B	75
CO3	Design and implement the suitable virtual reality modelling techniques for the given problem	TPS2	B	75
CO4	Identify the applications of VR in Business, Manufacturing, Architecture and Construction and Entertainment and other applications	TPS3	B	70
CO5	Develop an application in AR/VR using Unity game engine for a suitable problem given.	TPS3	B	70
CO6	Design and implement a team-project for developing AR/VR project using any Game engine.	TPS3	B	70

Mapping with Programme Outcomes and Programme Specific Outcomes

CoS	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L								L		L	M		L
CO2	M	L			L			L	L	L		L	M	L	L
CO3	M	L			L			L	L	L		L	M		L
CO4	S	M	L		M	L	L	L	L	L		M	M	L	L
CO5	S	M	L		M	L	L	L	L	L		M	M	L	L
CO6	S	M	L	L	M	L	L	L	L	L	M	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

CO	CAT										Assignment										Terminal					
	1					2					1					2										
TPS Scale	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	6
CO1	7	20		-	-	-	-	-	-	-		10	-	-	-	-	-	-	-	-	4	10		-	-	-
CO2	7	20		-	-	-	-	-	-	-		10	-	-	-	-	-	-	-	-	4	10		-	-	-
CO3	6	20	20	-	-	7	20	20	-	-	-	10	-	-	-	-	-	-	-	-	4	10		-	-	-
CO4	-	-	-	-	-	7	20	20	-	-	-	-	10	-	-			-	-	-	4	10	15	-	-	-
CO5	-	-	-	-	-	6	20		-	-	-	-	10	-	-			30	-	-	4	10	15	-	-	-
CO6	-	-	-	-	-				-	-	-	-	50	-	-		20	70	-	-	-	-			-	-

Syllabus

VIRTUAL REALITY ENVIRONMENTS: The development of VR: Computer Graphics, Real-time computer graphics, Virtual environments, Requirements for VR, benefits of Virtual reality. Augmented Reality – Computer Generated Worlds, Understanding Virtual Space. Defining Visual Space and Content. Defining Position and Orientation.

3D USER INTERFACES

Head-Mounted Displays, Augmenting Displays, Binocular Displays, Smartphone-Based displays, Sensors for Tracking Position, Orientation, and Motion, Trackers, 2D Versus 3D Interaction and Navigation, Hand and Gesture Tracking, Whole Body Tracking.

AR/VR Programming – Creation of VR environment, Digital twin and VR/AR, Unity 3D, setting up the Unity project, Unity Editor and its interface, Understanding Game Objects and Components, creating a basic scene in Unity and adding objects, setting up an AR project in Unity, Using AR Foundation, Deploying AR experiences onto mobile devices, Visual scripting in Unity.

AR VR Applications- Business and Manufacturing, Areas of application Product development and Marketing, Real Estate Sales Applications, Entertainment Applications and production. Education Applications, Automotive Engineering, Aerospace Engineering, Nuclear Engineering and Manufacturing. Public Safety and Military Applications flight simulation and training.

AR VR Project Development- Exploring a VR Model in a HMD and navigation. A game scene to demonstrate a player movement to demonstrate RigidBody, collider and C# scripts. Development of Mobile AR applications using AR Core template.

Text books

1. Anna Braun, Raffael Rizzo, Birmingham, "XR Development with Unity A beginner's guide to creating virtual, augmented, and mixed reality experiences using Unity", 2023.
2. William R.Sherman, Alan B.Craig, "Understanding Virtual Reality – Interface, Application, Design", The Morgan Kaufmann Series, 2018.

REFERENCES:

1. Steven M. LaValle, "Virtual Reality", Cambridge University Press, 2023.
2. Steve Aukstakalnis, "Practical Augmented Reality", Addison Wesley 2017.

3. Alan B. Craig, William R. Sherman, Jeffrey D. Will, "Developing Virtual Reality Applications", 2017.
4. Matjaz Mihelj, Domen Novak, Samo Begus, "Virtual Reality Technology and Applications", 1st Edition, Springer Netherlands, 2014.
5. Grigore C. Burdea, Philip Coiffet, "Virtual Reality Technology", 2nd Edition, Wiley India, 2017.
6. Gerard Jounghyun Kim, "Designing Virtual Reality Systems, the Structured Approach" · Springer London, 2005.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
1	VIRTUAL REALITY ENVIRONMENTS		CO1
1.1	The development of VR: Computer Graphics, Real-time computer graphics,	1	CO1
1.2	Virtual environments, Requirements for VR, benefits of Virtual reality.	1	CO1
1.3	Augmented Reality – Computer Generated Worlds, Understanding Virtual Space.	2	CO1
1.4	Defining Visual Space and Content. Defining Position and Orientation.	2	CO1
2	Interactive devices		CO2
2.1	Head-Mounted Displays, Augmenting Displays, Binocular Displays.	2	CO2
2.2	Smartphone-Based displays, Sensors for	1	CO2
2.3	Tracking Position, Orientation, and Motion, Trackers, 2D Versus 3D Interaction and Navigation,	1	CO2
2.4	Hand and Gesture Tracking, Whole Body Tracking.	2	CO2
3	VR Programming		CO3
3.1	Creation of VR environment, Digital twin and VR/AR,	1	CO3, CO6
3.2	Unity 3D, setting up the Unity project, Unity Editor and its interface,	2	CO3, CO6
3.3	Understanding Game Objects and Components,	2	CO3, CO6
3.4	Creating a basic scene in Unity and adding objects, Setting up an AR project in Unity,	2	CO3, CO6
3.5	Using AR Foundation, Deploying AR experiences onto mobile devices,	2	CO3, CO6
3.6	Visual scripting in Unity.	1	CO3, CO6
4.1	Business and Manufacturing, Areas of application Product development and Marketing,	1	CO 4
4.2	Real Estate Sales Applications, Entertainment Applications and production.	1	CO4
4.3	Education Applications, Automotive Engineering, Aerospace Engineering, Nuclear Engineering and Manufacturing.	1	CO4

4.4	Public Safety and Military Applications flight simulation and training.	1	CO4,
5.1	Exploring a VR Model in a HMD and navigation. template.	3	CO5, CO6
5.2	A game scene to demonstrate a player movement to demonstrate RigidBody, collider C# scripts and	3	CO5, CO6
5.3	visual scripting.	2	
5.4	Development of Mobile AR applications using AR Core	2	CO5, CO6
	Total	36	

Course Designers:

1. Dr.N.Shivakumar shiva@tce.edu
2. Ms. G. Bhavani gbicse@tce.edu



22CSGB1	CLOUD ESSENTIALS
----------------	-------------------------

Category	L	T	P	Credit
IE	3	0	0	3

Preamble

This course covers the history and background of cloud computing the essential technical knowledge needed to build applications in the cloud and how to utilize the cloud services for architecting a cloud application environment. Students will able to learn how to build, deploy and manage applications on cloud.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO. No	Course Outcomes (COs)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Articulate the main concepts, key technologies, strengths, limitations of cloud computing and the possible applications for state-of-the-art cloud computing	TPS2	B	75
CO2	Identify the architecture and infrastructure of cloud computing, including cloud delivery and deployment models	TPS3	B	70
CO3	Examine the virtualization technology that enhance cloud computing	TPS3	B	70
CO4	Analyze the various cloud resource management techniques to identify the optimal solution for allocation and scheduling	TPS3	B	70
CO5	Identify threats and assess risks associated with cloud services	TPS3	B	70
CO6	Simulate and create cloud environments	TPS3	B	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L											L		
CO2	S	M	L					L	L	L		L	M		L
CO3	S	M	L					L	L	L		L	M		L
CO4	S	M	L					L	L	L		L	M		L
CO5	S	M	L					L	L	L		L	M		L
CO6	S	M	L		L	L		L	L	L	L	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1 (Theory Component)						CAT2 (Theory Component)						Terminal (Theory Component)					
	100						100						100					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10											10					
CO2	10	10	20											10				
CO3	10		30											10	20			
CO4							20	10					10	10				
CO5							10	10	15						20			
CO6							10	10	15						10			

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

CO	Assignment 1						Assignment 2					
	1	2	3	4	5	6	1	2	3	4	5	6
TPS Scale												
CO1												
CO2												
CO3			30									
CO4			40						30			
CO5			30						30			
CO6									40			

Syllabus

Introduction - Cloud components - Essential characteristics - Architectural and Technological Influences – Operational Influences.

Cloud Infrastructure – Perspective of Data Centers – NIST Cloud Computing Reference Architecture and Types - Services: Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS) - Challenges and risks in cloud adoption. Cloud deployment model: Public clouds, Private clouds, Community clouds, Hybrid clouds.

Virtualization – Introduction – Implementation Levels of Virtualization-Virtualization Structures/Tools and Mechanism - Virtualization of CPU, Memory, I/O Devices – Porting Applications and VM Migration – Container based virtualization - Microservice deployment in Docker

Cloud Resource Management – Resource Management types and strategies - Allocation Methods – Policies – Tools for Resource Management (IBM Cloud Orchestrator, AWS Cloud Formation)

Cloud Security - Infrastructure Security – Data Security and Storage – Identity and Access Management -Trust, Reputation, Risk – Authentication in Cloud

Cloud Middleware - Simulating a cloud environment using cloudsim - Create a cloud environment using openstack.

Reference Books & web resources

1. Thomas Erl, Zaigham Mahmood, Ricardo Puttini, "Cloud Computing: Concepts, Technology & Architecture", Prentice Hall Service Technology Series, 2013.
2. John Rittinghouse, James Ransome, "Cloud Computing: Implementation, Management and Security", CRC Press 2010.
3. Greg Schulz, "Cloud and Virtual Data Storage Networking", CRC Press, 2012.
4. Nelson Ruest, Danielle Ruest, "Virtualization, A Beginner's Guide", McGraw-Hill Companies, 2009.
5. <https://www.edx.org/learn/cloud-computing>.
6. <https://www.coursera.org/browse/information-technology/cloud-computing>
7. <https://docs.docker.com/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	CO
1	Introduction (4)		
1.1	Cloud components	1	CO1
1.2	Essential characteristics	1	
1.3	Architectural and Technological Influences	2	
1.4	Operational Influences		
2	Cloud Infrastructure (7)		
2.1	Perspective of Data Centers	1	CO2
2.2	NIST Cloud Computing Reference Architecture and types	1	
2.3	Services – SaaS, PaaS, IaaS	2	
2.4	Challenges and risks in cloud adoption	1	
2.5	Cloud deployment model - Public clouds, Private clouds Community clouds, Hybrid clouds	2	
3	Virtualization (8)		
3.1	Introduction	1	CO3
	Implementation Levels of Virtualization		
3.2	Virtualization Structures/Tools and Mechanism	2	CO3
3.3	Virtualization of CPU, Memory, I/O Devices	1	
3.4	Porting Applications and VM Migration	2	
3.5	Container based virtualization	1	
3.6	Microservice deployment in Docker	1	
4	Cloud Resource Management (6)		
4.1	Resource Management types and strategies	1	CO4
4.2	Allocation Methods	2	
4.3	Policies	1	
4.4	Tools for Resource Management	2	
5	Cloud security and storage (5)		
5.1	Infrastructure Security	1	CO5
5.2	Data Security and Storage	1	
5.3	Identity and Access Management	1	
5.4	Trust, Reputation, Risk	1	

5.5	Authentication in Cloud	1	
6	Cloud Middleware (6)		
6.1	Simulating a cloud environment using cloudsim	3	CO6
6.2	Create a cloud environment using openstack	3	
	Total	36	

Course Designers:

1. Ms. C.Santhiya, Assistant Professor, CSE

csit@tce.edu



22CSGC1**ESSENTIALS OF PYTHON FOR
ARTIFICIAL INTELLIGENCE**

Category L T P Credit

IE 3 0 0 3

Preamble

Python is a preferred language for developing Artificial Intelligence(AI) applications because of its ease of use and robust library set of pre-built functions, for deep learning, machine learning, and data analysis. The main objective of this course is to train the students on solving AI challenges. The students will explore the use of Python's advanced module features and apply them in probability, statistical testing, signal processing, financial forecasting, and various other applications and they can quickly build AI models.

Prerequisite

- Basic knowledge of Programming

Course Outcomes

On the successful completion of the course, students will be able to

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Interpret the key concepts of AI and AI Project Lifecycle to solve real time challenges	TPS2	B	75
CO2	Implement Python Variables, Expressions, Control Structures and Data Structures	TPS3	B	70
CO3	Demonstrate Python Modules, Functions, Packages, Object Oriented Concepts and Exception Handling	TPS3	B	70
CO4	Prepare AI datasets using Numpy and Pandas libraries	TPS3	B	70
CO5	Develop AI Applications using Visualization and AI Modelling libraries	TPS3	B	70
CO6	Choose suitable AI Frameworks to solve real time projects	TPS4	B	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	M	L						L	L	L	L	L	L	L	L
CO2.	S	M	L		L			M	M	M	M	M	M	L	M
CO1	S	M	L		L			M	M	M	M	M	M	L	M
CO2	S	M	L		L			M	M	M	M	M	M	L	M
CO3	S	M	L		L			M	M	M	M	M	M	L	M
CO4	S	M	L	L	M	L	L	M	M	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	15													30										2	5				
CO2	5	10	25												30										2	5	10			
CO3	5	10	25												40										2	5	15			
CO4							5	10	20												30				2	5	15			
CO5							5	10	20												30				2	5	10			
CO6							5	10	15												10	30				5	10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus**Introduction to Artificial Intelligence**

History and Transition to AI Era- AI in Modern World - Key Concepts in AI- Understanding Machine Learning: Supervised- Classification, Regression -Unsupervised – Clustering, Dimension Reduction, Anomaly Detection- Reinforcement learning- Deep Learning

Introduction to Python: Python Variables, Expressions, Control Structure, Data Structures

Jupyter Notebooks-Data Types- Expressions- Operators - Control Structure and statements: Conditionals- Iteration - Data Structures: Lists, Tuple, Set, Dictionary-operations- methods-parameters- Advanced list processing – Strings Operations.

Python Modules, Functions, Packages, Object Oriented Concepts and Exception Handling:

Function: parameters and arguments- Fruitful functions-return values- local and global scope- recursion- lambda- filter- reduce- map- zip- Concepts of Packages- class- objects- Exception Handling: Assertion, except- try-finally- user defined exception.

AI Project Lifecycle

Idea and Conceptualization-Defining Objectives-Research and Feasibility Analysis- Data Collection and Preparation- Choosing Right Framework and Libraries - Designing the AI Model- Training the AI Model- Model Optimization and Tuning-Integration and Deployment - Testing and Quality Assurance-Ethical Considerations and Compliance

Python for AI

Data Handling- **Numpy**: NDARRAYS- Reshape-Flatten- Expand-Stack-Concatenate -Linear Algebra- **Pandas**: Access data from Series- object attributes –Data Frame-Attributes and Operations-Data Manipulation: Feature Aggregation-Pivot Tables-Combine Dataset-Data Cleaning

Visualization, AI Modelling and AI Applications

Matplotlib and Seaborn for Data Visualization- Scikit-Learn for Machine Learning- TensorFlow and PyTorch for Deep Learning- Keras for Neural Networks-Image Classification- Classification with Pre-Trained Models -Classification with Custom Trained Models

Case study: Coffee Preference Prediction App- Analysis based on various factors like taste preferences, consumption habits, and contextual data

Mini project Topics [Not limited to]:

Healthcare: Predictive Diagnostics, Personalized Medicine, Robotic Surgeries.

Finance: Credit Scoring, Fraud Detection.

Retail: Customer Segmentation, Inventory Management, Personalized Recommendation Systems.

Manufacturing: Production Processes, Predictive Maintenance, Supply Chain Management.

Text Book

1. Pramod Gupta, Anupam Bagchi, "Essentials of Python for Artificial Intelligence and Machine Learning", Synthesis Lectures on Engineering, Science, and Technology Series, Springer, First Edition, 2023.
2. Patrick J, "Python AI Programming: Navigating fundamentals of ML, deep learning, NLP, and reinforcement learning in practice", GitforGits Publishers, First Edition, 2024

Reference Books & web resources

1. Allen Downey, Jeffrey Elkner, Chris Meyers, "Learning with PYTHON: How to Think Like a Computer Scientist", O'Reilly Publishers, 3rd Edition(GNU), 2020.
2. G Venkatesh and Madhavan Mukund, "Computational Thinking: A Primer for Programmers and Data Scientists", 1st Edition, Notion Press, 2021.
3. John V Guttag, "Introduction to Computation and Programming Using Python: With Applications to Computational Modelling and Understanding Data", Third Edition, MIT Press 2021
4. Eric Matthes, "Python Crash Course, A Hands-on Project Based Introduction to Programming", 3rd Edition, No Starch Press, US, 2023.
5. Paul Deitel, Dr. Harvey Deitel "Python for Programmers: with Big Data and Artificial Intelligence Case Studies", Pearson Higher Ed, First Edition, 2019
6. <https://www.python.org/>

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Artificial Intelligence	
1.1	History and Transition to AI Era, AI in Modern World, Key Concepts in AI	1
1.2	Understanding Machine Learning: Supervised- Classification, Regression	2
1.3	Unsupervised – Clustering, Dimension Reduction, Anomaly Detection- Reinforcement learning- Deep Learning	3

Module No.	Topic	No. of Periods
2	Introduction to Python: Python Variables, Expressions, Control Structure, Data Structures	
2.1	Jupyter Notebooks, Data Types, Expressions, Operators	1
2.2	Control Structure and statements: Conditionals, Iteration	1
2.3	Data Structures: Lists, Tuple, Set, Dictionary-operations- methods- parameters	3
2.4	Advanced list processing , Strings Operations	1
3	Python Modules, Functions, Packages, Object Oriented Concepts and Exception Handling	
3.1	Function: parameters and arguments, Fruitful functions: return values- local and global scope	2
3.2	Recursion, lambda, filter, reduce, map, zip, Concepts of Packages	2
3.3	Class- objects- Exception Handling: Assertion, except- try-finally- user defined exception	2
4	Understanding AI Project Lifecycle	
4.1	Idea and Conceptualization, Defining Objectives, Research and Feasibility Analysis, Data Collection and Preparation	2
4.2	Choosing Right Framework and Libraries , Designing the AI Model , Training the AI Model , Model Optimization and Tuning	2
4.3	Integration and Deployment , Testing and Quality Assurance ,Ethical Considerations and Compliance	2
5	Python for AI	
5.1	Data Handling - Numpy : NDARRAYS, Reshape, Flatten, Expand, Stack, Concatenate, Linear Algebra	3
5.2	Pandas : Access data from Series, object attributes, Data Frame ,Attributes and Operations ,Data Manipulation: Feature Aggregation, Pivot Tables, Combine Dataset, Data Cleaning	3
6	Data Visualization, AI Modelling and AI Applications	
6.1	Matplotlib and Seaborn for Data Visualization	1
6.2	Scikit-Learn for Machine Learning, TensorFlow and PyTorch for Deep Learning, Keras for Neural Networks	2
6.3	Image Classification, Classification with Pre-Trained Models , Classification with Custom Trained Models	2
6.4	Case study : Coffee Preference Prediction App-Analysis based on various factors like taste preferences, consumption habits, and contextual data	1
	Mini Project	

Module No.	Topic	No. of Periods
	Total	36

Course Designer(s):

1. Dr.M.Nirmala Devi

mnit@tce.edu



22CSGD1**DRONE SYSTEMS**

Category	L	T	P	Credit
IE	3	0	0	3

Preamble

This course provides a foundational understanding of drone systems, including their design, control, and navigation. It covers key topics like UAV communication, regulations, and emerging technologies. Students will gain insights into real-world applications and the future scope of drone innovations.

Prerequisite

- Basic Knowledge of Electronics, Automation and Communication systems.

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome 1	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the fundamental principles of UAV architecture, types, and power systems to identify appropriate drone configurations for various applications.	TPS3	75	80
CO2	Illustrate the principles of drone aerodynamics and control strategies including PID tuning and sensor integration for stable flight.	TPS3	75	80
CO3	Construct suitable navigation approaches using SLAM, Kalman filters, and obstacle avoidance for autonomous drone operations.	TPS3	75	80
CO4	Select appropriate communication protocols and interpret UAV regulatory frameworks, including DGCA and international norms	TPS3	75	80
CO5	Classify different modes of drone coordination and analyze safety measures, fault tolerance, and reliability in swarm-based operations.	TPS4	75	80
CO6	Examine drone applications across sectors and evaluate the use of AI, IoT, and Blockchain technology.	TPS4	75	80

Mapping with Programme Outcomes

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L									L	L		
CO2	S	M	L		L					L		M	M		M
CO3	S	M	L		L					L		M	M		M
CO4	S	M	L		M		M	M	M			M	M		M

CO5	S	S	M	L			M	M	M	L		M	M		M
CO6	S	S	M	L	M	L		M	M	L	L	M	S	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Ass1		Ass2		Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	3	4	3	4	1	2	3	4	5	6
CO1	5	10	10	-	-	-	-	-	-	-	-	-	20	-	-	-	2	5	10	-	-	-
CO2	5	10	25	-	-	-	-	-	-	-	-	-	40	-	-	-	2	5	10	-	-	-
CO3	-	10	25	-	-	-	-	-	-	-	-	-	40	-	-	-	-	5	10	-	-	-
CO4	-	-	-	-	-	-	5	10	20	-	-	-	-	-	20	-	2	5	10	-	-	-
CO5	-	-	-	-	-	-	-	10	-	20	-	-	-	-	-	40	2	5	-	10	-	-
CO6	-	-	-	-	-	-	5	10	-	20	-	-	-	-	-	40	2	5	-	10	-	-

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Drone Systems – Overview of UAVs – Types and classifications – Major components and functions – Power Systems – Applications and Characteristics

Aerodynamics and Control Systems – Multirotor Aerodynamics – Forces and moments – Flight modes - Basic Motion Control – Basics of PID control – Sensor integration (IMU, GPS, GNSS) – Stability and tuning of controllers.

Autonomous Navigation and Path Planning – SLAM and localization – Path planning – Obstacle detection and avoidance using LiDAR/vision sensors – Kalman filter

Communication and Regulations – Drone communication protocols (Wi-Fi, LoRa, 5G) – D2D and D2X – DGCA and global UAV regulations – Regulatory Aspects and Ethical Considerations.

Swarm Robotics, Safety, and Reliability – Introduction to Swarm Robotics - Principles of Multi-Agent Coordination. Safety Considerations: Potential Hazards, Basic Safety Procedures. Reliability - Fault Tolerance.

Emerging Applications and Technologies – Use cases in agriculture, surveillance, logistics – AI in drones – IoT integration – Blockchain for drone identity and data security.

Text Book

1. Paul Gerin Fahlstrom, Thomas James Gleason, Mohammad H. Sadraey, "Introduction to UAV Systems", Wiley Fifth Edition, 2022
2. Randal W. Beard and Timothy W. McLain, "Small Unmanned Aircraft: Theory and Practice", Princeton and Oxford University Press, 2012
3. Roland Siegwart, Illah R. Nourbakhsh, and Davide Scaramuzza, "Introduction to Autonomous Mobile Robots", MIT Press 2011.

Reference Books

1. Nikolaus Correll, Bradley Hayes, Christoffer Heckman, Alessandro Roncone, "Introduction to Autonomous Robots: Mechanisms, Sensors, Actuators, and Algorithms" MIT Press, 2022.
2. A. R. Jha, "Theory, Design and Applications of Unmanned Aerial Vehicles", CRC Press, Taylor & Francis Group 2017
3. Kimon P. Valavanis, George J. Vachtsevanos, "Handbook of Unmanned Aerial Vehicles", Springer 2015
4. Richard K. Barnhart, Stephen B. Hottman, Douglas M. Marshall, Eric Shappee, "Introduction to Unmanned Aircraft Systems", CRC Press, Taylor & Francis Group 2012.
5. Reg Austin, "Unmanned Aircraft Systems Uavs Design, Development and Deployment", Wiley 2010.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Drone Systems	
1.1	Overview of UAVs	1
1.2	Types and classifications	1
1.3	Major components and functions	1
1.4	Power Systems	1
1.5	Applications and Characteristics	1
2	Aerodynamics and Control Systems	
2.1	Multirotor Aerodynamics, Forces and moments.	2
2.2	Flight modes, Basic Motion Control - PID control	2
2.3	Sensor integration (IMU, GPS)	2
2.4	Stability and tuning of controllers.	1
3	Autonomous Navigation and Path Planning	
3.1	SLAM and localization	1
3.2	Path planning	1
3.3	Obstacle detection and avoidance using LiDAR/ vision sensors	2
3.4	Kalman filter	2
4	Communication and Regulations	
4.1	Drone communication protocols (Wi-Fi, LoRa, 5G)	2
4.2	D2D and D2X	2

Module No.	Topic	No. of Periods
4.3	DGCA and global UAV regulations	2
4.4	Regulatory Aspects and Ethical Considerations.	1
5	Swarm Robotics, Safety, and Reliability	
5.1	Introduction to Swarm Robotics, Principles of Multi-Agent Coordination.	2
5.2	Safety Considerations: Potential Hazards, Basic Safety Procedures.	1
5.3	Reliability, Fault Tolerance	2
6	Emerging Applications and Technologies	
6.1	Use cases in agriculture, surveillance, logistics	2
6.2	AI in drones – IoT integration	2
6.3	Blockchain for drone identity and data security.	2
	Total	36

Course Designer(s):

1. Dr.K.Narasimhamallikarjunan (arjunkambaraj@tce.edu)
2. Mr.D.Nagendra Kumar (dnkcse@tce.edu)

22CSGE1 UI AND UX DESIGN

Category L T P Credit

IE 3 0 0 3

Preamble

This course equips students with essential skills in user-centered design thinking, wireframing and prototyping techniques, usability testing, and visual design principles. Through hands-on projects and real-world case studies, students will apply these skills to solve real user problems and develop innovative solutions, preparing them for careers in design and technology.

Prerequisite

- NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Show the differences between UI and UX, the history of UX, the key stages of design thinking and identify the user requirements.	TPS3	70	80
CO2	Demonstrate the user needs using research methods and develop personas, user stories, and problem statements.	TPS3	70	80
CO3	Apply UI design principles to design visually clear and consistent interfaces using patterns and style guides.	TPS3	70	80
CO4	Build wireframes, prototypes, and mockups using digital tools, and assess them through usability testing.	TPS3	70	80
CO5	Organize user flows and structure content using flow diagrams and information architecture methods.	TPS3	70	80
CO6	Construct UX solutions for AI, IoT, and voice interfaces, and recognize the role of DesignOps.	TPS3	70	80

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	S	M	L					M	M			M	M		L
CO2	S	M	L			L	L	M	M	M		M	M	L	M
CO3	S	M	L		L	L	L	M	M	M		M	M	L	M
CO 4	S	M	L		L	L	L	M	M	M		M	M	L	M

CO 5	S	M	L		L	L	L	M	M	M		M	M	L	M
CO 6	S	M	L	L	M	M	L	M	M	M	L	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	5	20	-	-	-	-	-	-	-	-	-			30										2	3	10			
CO2	5	5	25	-	-	-	-	-	-	-	-	-			35										2	3	15			
CO3	5	5	25	-	-	-	-	-	-	-	-	-			35										2	3	10			
CO4	-	-	-	-	-	-	5	5	20	-	-	-									20				2	3	10			
CO5	-	-	-	-	-	-	5	5	25	-	-	-									20				2	3	15			
CO6	-	-	-	-	-	-	5	5	25	-	-	-									60				2	3	10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Foundations of design and User Requirements : UI vs. UX Design - History of UX. Need of UI and UX - Core Stages of Design Thinking - Divergent and Convergent Thinking, User requirements – Definition, Types of user research - Qualitative research, Quantitative research. Tools to collect user requirements – personal observation, interviews, questionnaire, User/ Expert reviews, User requirement analysis - Understanding target audience and client requirements, Competitive analysis, Affinity mapping

Case study: Use Design tool for user requirement collection and analysis of various interfaces

UI Design: Visual and UI Principles - UI Elements and Patterns - Interaction Behaviors and Principles - Branding - Style Guides- Visual Communication Design - effective visual communication for graphical user interface

Case study: Design a Visually Consistent Mobile App Interface using tools like Figma, Adobe XD

UX Design: Introduction to User Experience – Importance of User Experience - Understanding User Experience - Defining the UX Design Process and its Methodology - Research in User Experience Design - Tools and Method used for Research - User Needs and its Goals - Know about Business Goals

Case study: Conduct UX Research and Create Personas & Journey Map

Wireframing, Prototyping And Testing : Sketching Principles - Sketching Red Routes - Responsive Design - Wireframing - Creating Wireflows - Building a Prototype - Building High-Fidelity Mockups - Designing Efficiently with Tools - Interaction Patterns - Conducting Usability Tests - Other Evaluative User Research Methods - Synthesizing Test Findings - Prototype Iteration.

Case study: Translate wireframes into a working prototype and test it for usability using open source tools

Conceptual Design and UX Structuring: Identifying and Writing Problem Statements - Identifying Appropriate Research Methods - Creating Personas - Solution Ideation - Creating User Stories - Creating Scenarios - Flow Diagrams - Flow Mapping - Information Architecture.
Case study: Use conceptual design techniques to map out UX structure before UI begins.

UX for Emerging Technologies : AI-driven UX Design (Designing for Chatbots, Smart Assistants) - Designing for IoT Interfaces - Voice, Gesture, and Biometric Interfaces- (DesignOps) Overview

Text Book

1. Joel Marsh, "UX for Beginners", O'Reilly, 2022
2. Jon Yablonski, "Laws of UX using Psychology to Design Better Product & Services" O'Reilly 2021

Reference Books & web resources

1. Jenifer Tidwell, Charles Brewer, Aynne Valencia, "Designing Interface" 3rd Edition, O'Reilly 2020
2. Steve Schoger, Adam Wathan "Refactoring UI", 2018

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Foundations of Design and User Requirements	
1.1	UI vs. UX Design - History of UX - Need of UI and UX	1
1.2	Core Stages of Design Thinking	1
1.3	Divergent and Convergent Thinking	1
1.4	User requirements – Definition	1
1.5	Types of user research - Qualitative research, Quantitative research	1
1.6	Tools to collect user requirements – personal observation, interviews, questionnaire, User/ Expert reviews	1
1.7	User requirement analysis - Understanding target audience and client requirements, Competitive analysis, Affinity mapping	2
2	UI Design	
2.1	Visual and UI Principles	1
2.2	UI Elements and Patterns	1
2.3	Interaction Behaviors and Principles	1
2.4	Branding - Style Guides	1

Module No.	Topic	No. of Periods
2.5	Visual Communication Design	1
2.6	Effective visual communication for graphical user interface	1
3	UX Design	
3.1	Introduction to User Experience – Importance of User Experience - Understanding User Experience	1
3.2	Defining the UX Design Process and its Methodology	1
3.3	Research in User Experience Design	1
3.4	Tools and Method used for Research	1
3.5	User Needs and its Goals- Know about Business Goals	1
4	Wireframing, Prototyping And Testing	
4.1	Sketching Principles- Sketching Red Routes	1
4.2	Responsive Design	1
4.3	Wireframing - Creating Wireflows	1
4.4	Building a Prototype	1
4.5	Building High-Fidelity Mockups- Designing Efficiently with Tools	1
4.6	Interaction Patterns	1
4.7	Conducting Usability Tests - Other Evaluative User Research Methods	1
4.8	Synthesizing Test Findings - Prototype Iteration.	1
5	Conceptual Design and UX Structuring	
5.1	Identifying and Writing Problem Statements	1
5.2	Identifying Appropriate Research Methods	1
5.3	Creating Personas - Solution Ideation	1
5.4	Creating User Stories - Creating Scenarios	1
5.5	Flow Diagrams - Flow Mapping	1
5.6	Information Architecture	1
6	UX for Emerging Technologies	

Module No.	Topic	No. of Periods
6.1	AI-driven UX Design (Designing for Chatbots, Smart Assistants)	1
6.2	Designing for IoT Interfaces - Voice, Gesture, and Biometric Interfaces	1
6.3	DesignOps Overview	1
	Total	36

Course Designer(s):

1. Dr.Raja Lavanya,Assistant Professor,CSE rlit@tce.edu

22CSGF1	COMPUTER SECURITY ESSENTIALS	Category	L	T	P	Credit
		IE	3	0	0	3

Preamble

This course aims at providing an understanding of the fundamentals of the cyber security.

Prerequisite

NIL

Course Outcomes

On the successful completion of the course, students will be able to

CO	Course Outcome (CO)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Interpret the fundamental elements of cyber security.	TPS2	70	80
CO2	Administer the functioning of networks, internet and the concepts underlying network security..	TPS3	70	75
CO3	Dramatize the different variants of cyber threats.	TPS3	70	75
CO4	Illustrate various security primitives of information security.	TPS3	70	75
CO5	Examine the concepts underlying system security.	TPS3	70	75
CO6	Discover the tools and techniques for digital forensics and incident response.	TPS3	70	75

Mapping with Programme Outcomes

Cos	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO 3
CO1	M	L		-								L	L		
CO2	S	M	L	-	-	L	L	L	L	L		L	M	L	L
CO3	S	M	L	-	M	L	L	L	L	L		L	M	L	L
CO4	S	M	L	-	-	L	L	L	L	L		L	M	L	L
CO5	S	M	L	-	-	L	L	M	L	L		L	M	L	L
CO6	S	M	L	-	-	L	L	L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment1						Assignment2						Terminal					
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10																							2	10				
CO2	10	10	10												50										2	5	10			
CO3	10	20	20												50										4	5	10			
CO4							10													30					4	5	10			
CO5							10	10	20											30					4	5	10			
CO6							10	20	10											40					4		10			

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Cybersecurity: Introduction – Types of Cyber Threats – Basic Security Terminology: CIA Triad – Confidentiality – Integrity – Availability, Authentication – Access Policy.

Network Security: Overview – Overview - Network Basics - How the Internet Works - Basic Network Utilities - Transport layer security - Internet protocol security - Wireless security - Email security - Network monitoring - Intrusion detection system - Virtual private network and firewall.

Cyber Threats: DoS Attacks - Malware: Viruses, Trojan Horses, Buffer-Overflow Attack - Spyware – Other Forms of Malware - Detecting and Eliminating Viruses and Spyware - The Dark Web - Recent Cyber Threats - Twitter attack - GPT attack - Data leaks.

Information Security: Overview of Cryptography – Encryption: Caesar Cipher, Hill Cipher – Hash Function – Digital Signatures – Digital Certificates - Data Privacy - Blockchain.

Systems Security - Malware - Program analysis - Penetration testing - Embedded system and hardware security - Mobile security - Secure storage management - Cyber-physical system security - IoT security.

Digital Forensics and Incident Response: Incident Response – Preparation – Detection and Analysis – Recovery – Post-incident Activities; Digital Forensics – Data Collection and Examination – Analysis and Reporting - Cyber resilience - Assured Delays - Security Assessment.

Text Books

1. Chuck Easttom, "Computer Security Fundamentals", 2023.
2. William Stallings, "Cryptography and Network Security Principles and Practice", 8th Edition, Pearson, 2023.

3. Behrouz A. Forouzan, “Data Communications and Networking with TCP/IP Protocol Suite” 6th Edition, 2022.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to Cybersecurity	
1.1	Introduction – CIA Triad	1
1.2	Confidentiality – Integrity – Availability	2
1.3	Authentication – Access Policy	2
2	Network Security	
2.1	Overview - Network Basics	1
2.2	How the Internet Works	1
2.3	Basic Network Utilities	1
2.4	Transport layer security - Internet protocol security	1
2.5	Wireless security - Email security	1
2.6	Network monitoring - Intrusion detection system - Virtual private network and firewall	2
3	Types of Cyber Threats	
3.1	DoS Attacks	1
3.2	Malware: Viruses, Trojan Horses	1
3.3	Buffer-Overflow Attack	1
3.4	Spyware – Other Forms of Malware - Detecting and Eliminating Viruses and Spyware	1
3.5	The Dark Web	1
3.6	Recent Cyber Threats - Twitter attack - GPT attack - Data leaks	1
4	Information Security	
4.1	Overview	1
4.2	Encryption: Caesar Cipher, Hill Cipher	1
4.3	Hash Function	1
4.4	Digital Signatures	1
4.5	Digital Certificates - Data Privacy	1
4.6	Blockchain	1

Module No.	Topic	No. of Periods
5	Systems Security	
5.1	Malware - Program analysis	1
5.2	Penetration testing	1
5.3	Embedded system and hardware security	1
5.4	Mobile security - Secure storage management	1
5.5	Cyber-physical system security	1
5.6	IoT security	1
6	Digital Forensics and Incident Response	
6.1	Incident Response – Preparation – Detection and Analysis	1
6.2	Recovery – Post-incident Activities	1
6.3	Digital Forensics – Data Collection and Examination	1
6.4	Digital Forensics – Analysis and Reporting	1
6.5	Cyber resilience	1
6.6	Assured Delays - Security Assessment	1
	Total	36

Course Designers:

1. Dr. M. Vijayalakshmi mviji@tce.edu
2. Dr. J. Dharani jdicse@tce.edu

22CSGG1	DEEP LEARNING FOR LANGUAGE PROCESSING	Category IE	L 3	T 0	P 0	Credit 3
----------------	--	----------------	--------	--------	--------	-------------

Preamble

This course focuses on enabling machines to understand and interact using human language. This course introduces the fundamental concepts of NLP, including text and speech processing, language modelling, and syntactic and semantic analysis. Students will explore both the theoretical foundations and practical tools used to build intelligent language applications. Special emphasis is placed on handling ambiguity, processing Indian languages, and applying NLP techniques to real-world domains such as education, law, and conversational AI.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Infer the fundamentals, origins, and challenges of Natural Language Processing, including various levels of language processing and grammar in both global and Indian language contexts.	TPS2	70	85
CO2	Apply basic text preprocessing techniques such as tokenization, stemming, lemmatization, and vector space modeling to prepare textual data for NLP tasks.	TPS3	70	80
CO3	Evaluate probabilistic language models including unigram, bigram, and trigram models for word prediction and text analysis.	TPS3	70	80
CO4	Apply foundational concepts of articulatory and acoustic phonetics along with digital signal processing techniques to segment and classify speech data.	TPS3	70	80
CO5	Implement popular NLP libraries and frameworks to develop and experiment with language processing tools and applications.	TPS3	70	80
CO6	Design NLP-based solutions for real-world applications such as sentiment analysis, text summarization, question answering, chatbots, and information extraction.	TPS3	70	80

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	M	L				L				L	L	L	L	L	L
CO2.	S	M	L		M				M	M	M	M	M	M	M
CO3.	S	M	L		M					M	M		M	M	M

CO4.	S	M	L		M				M	M	M		M	M	M
CO5.	S	M	L		M	M				M	M	M	M	M	M
CO6.	S	M	L		M				M	M	M	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass2	Terminal					
	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	10	10	-	-	-	-	30	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	10	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO3	10	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO4	-	-	-	-	-	-	-	5	10	20	-	-	-	20	2	5	12	-	-	-
CO5	-	-	-	-	-	-	-	5	10	15	-	-	-	20	2	5	12	-	-	-
CO6	-	-	-	-	-	-	-	10	10	15	-	-	-	60		5	12	-	-	-

*Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to NLP: What is NLP- Origins and challenges of NLP- Generic NLP system - Introduction to various levels of natural language processing- Knowledge in language Processing- Ambiguity in Natural language-Computational requirements for NLP- Language and Grammar- Processing Indian Languages- The role of language Models

Text Processing: Text Processing Basics - Probability and NLP - Lower case conversion- Character Encoding-Word Segmentation- Sentence Segmentation-Stop Words Removal- Punctuation and digits removal- Stemming and lemmatization- Vector Space Models-Statistical Properties of words.

Language Models: What is word prediction- The definition of probabilistic language models - Estimating parameters -Basic Smoothing-Evaluating language models -Simple N-gram models - bigram and trigram language - Neural Language Models-RNN-LSTM-Transformers-BERT-GPT- Llama-DeepSeek- Out of vocabulary words.

Speech Processing: Speech fundamentals: articulatory phonetics – production and classification of speech sounds- Word Boundary Detection- acoustic phonetics – acoustics of speech production- review of digital signal processing concepts

NLP Libraries: Regex (Regular Expressions)-NLTK (Natural Language Toolkit)-spaCy- TextBlob-Textacy-VADER -Gensim-AllenNLP-Stanza-Pattern-PyNLPI-Hugging Face Transformer-flair Library-FastText-Polyglot

Applications of NLP: Text Summarization-Sentiment Analysis-Question answering- Information Extraction-Basic Dialogue Systems- Spam Detection-Building a chatbot-NLP in Education and Legal Systems

Text Books

1. Daniel Jurafsky, James H. Martin—Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, Pearson Publication, 2014.
2. Steven Bird, Ewan Klein and Edward Loper, —Natural Language Processing with Python, First Edition, O'Reilly Media, 2009.
3. Lawrence Rabiner And Biing-Hwang Juang, “Fundamentals Of Speech Recognition”, Pearson Education, 2003.

Reference Books

1. Frederick Jelinek, “Statistical Methods Of Speech Recognition”, MIT Press, 1997.
2. Breck Baldwin, “Language Processing with Java and LingPipe Cookbook”, Atlantic Publisher, 2015
3. Richard M Reese, —Natural Language Processing with Java, O'Reilly Media, 2015

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	Introduction to NLP	
1.1	What is NLP- Origins and challenges of NLP- Generic NLP system	1
1.2	Introduction to various levels of natural language processing	1
1.3	Knowledge in language processing	1
1.4	Ambiguity in Natural language-Computational requirements for NLP	1
1.5	Language and Grammar-Processing Indian Languages- The role of language Models	1
2	Basic Text Processing:	
2.1	Text Processing Basics - Probability and NLP	1
2.2	Lower case conversion- Character Encoding	1
2.3	Word Segmentation- Sentence Segmentation-Stop words removal	2
2.4	Punctuation and digits removal	1
2.5	Stemming and lemmatization	1
3	Language Models:	
3.1	What is word prediction- The definition of probabilistic language models	1
3.2	Estimating parameters - Basic Smoothing- Evaluating language models	1
3.3	Simple N-gram models - bigram and trigram language	1

Module No.	Topic	No. of Periods
3.4	Neural Language Models-RNN-LSTM	1
3.5	Transformers-BERT-GPT- Llama-DeepSeek	1
3.6	Out of vocabulary words	1
4	Basic Speech Processing:	
4.1	Speech fundamentals: articulatory phonetics	1
4.2	production and classification of speech sounds	1
4.3	Word Boundary Detection	1
4.4	acoustic phonetics	1
4.5	acoustics of speech production	1
4.6	review of digital signal processing concepts	1
5	Overview of NLP Libraries	
5.1	Regex (Regular Expressions)-NLTK (Natural Language Toolkit)	2
5.2	spaCy-TextBlob-Textacy	1
5.3	VADER -Gensim-AllenNLP-Stanza-Pattern	2
5.4	PyNLPI-Hugging Face Transformer-flair Library-FastText-Polyglot	2
6	Applications of NLP	
6.1	Text Summarization-Sentiment Analysis-Question answering	2
6.2	Information Extraction-Basic Dialogue Systems-	2
6.3	Spam Detection-Building a chatbot	1
6.4	NLP in Education and Legal Systems	1
	Total	36

Course Designer(s):

1. Dr. K. Sundarakantham
2. Mrs.S.Jeniba

kskcse@tce.edu
sjacse@tce.edu

22CSGH1**EXTENDED REALITY TECHNOLOGIES**

Category	L	T	P	Credit
IDE	3	0	0	3

Preamble

Students will gain a foundational knowledge of virtual reality environment, the ways to interact, and their various areas of applications. Students will gain a foundational knowledge of building VR projects

Prerequisite

Nil

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the components, usage of Virtual, Augmented and Mixed Reality	TPS2	B	75
CO2	Understand Immersive Experience and Game Development with AR VR Environments	TPS2	B	75
CO3	Design and develop programs using suitable AR/VR Development tools	TPS2	B	75
CO4	Set up AR/VR project to use AI tools in prototyping the workflow and suitable applications areas	TPS3	B	70
CO5	Develop an application in AR/VR using Unity game engine for a suitable problem given.	TPS3	B	70
CO6	Design and implement a team-project for developing AR/VR project using any Game engine.	TPS3	B	70

Mapping with Programme Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L								L		L	M		L
CO2	M	L			L			L	L	L		L	M	L	L
CO3	M	L			L			L	L	L		L	M		L
CO4	S	M	L		M	L	L	L	L	L		M	M	L	L
CO5	S	M	L		M	L	L	L	L	L		M	M	L	L
CO6	S	M	L	L	M	L	L	L	L	L	M	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

CO	CAT										Assignment										Terminal					
	1					2					1					2										
TPS Scale	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	6
CO1	10	25		-	-	-	-	-	-	-		-	-	-	-	-	-	-	-	-	4	10		-	-	-
CO2	10	25		-	-	-	-	-	-	-		10	-	-	-	-	-	-	-	-	4	10		-	-	-
CO3	5	25		-	-	5	15	-	-	-	-	40	-	-	-	-	-	-	-	-	4	10		-	-	-
CO4	-	-	-	-	-	5	20	15	-	-	-	-	50	-	-			-	-	-	4	10	15	-	-	-
CO5	-	-	-	-	-	5	20	15	-	-	-	-	-	-	-			30	-	-	4	10	15	-	-	-
CO6	-	-	-	-	-				-	-	-	-	-	-	-			70	-	-	-	-			-	-

Syllabus

VIRTUAL ENVIRONMENTS: Choosing reality: VR, AR, or MR., Real-time computer graphics, Requirements for VR, AR, and MR, – **Environmental storytelling.** Computer Generated Worlds for VR , AR Foundation package, OpenXR, Universal Render Pipeline (URP), SDKs for XR development, XR Interaction Toolkit, Defining Visual Space for MR, Defining Position and Orientation.

3D USER INTERFACES

Head-Mounted Displays, Augmenting Displays, Binocular Displays, Smartphone-Based displays, Sensors for Tracking Position, Orientation, and Motion, Trackers, 2D Versus 3D Interaction and Navigation, Hand and Gesture Tracking, Whole Body Tracking. Web3, Metaverse and Virtual Assets, tokens and dApp.

AR/VR Programming – Creation of VR environment, Unity 3D, setting up the Unity project, Unity Editor and its interface, Understanding Game Objects and Components, creating a basic scene in Unity and adding objects, setting up an AR project in Unity, Using AR Foundation, Deploying AR experiences onto mobile devices, Visual scripting in Unity. Digital Twin models

Generative Artificial Intelligence (AI) in AR/VR, Using AI Assistant to set up a scene, Animation Generator to bring character to life, Sprite Generator to make an avatar icon, Use Sound Generator for background audio. Finalize and share your project, **AR VR case studies-** Business and Manufacturing, Product visualization and Marketing, Public Safety and Military flight simulation and training.

AR VR Project Development- Exploring a VR Model in a HMD and navigation. A game scene to demonstrate a player movement to demonstrate Rigibody, collider and C# scripts. Development of Mobile AR applications using AR Core template.

Text books

1. Anna Braun, Raffael Rizzo, Birmingham, “XR Development with Unity A beginner’s guide to creating virtual, augmented, and mixed reality experiences using Unity”, 2023.
2. Create virtual and mixed reality experiences in Unity, UNITY 2022 LTS edition.
3. William R.Sherman, Alan B.Craig, "Understanding Virtual Reality – Interface, Application, Design", The Morgan Kaufmann Series, 2019.

REFERENCES:

1. Steven M. LaValle, "Virtual Reality", Cambridge University Press, 2023.
2. Alan B. Craig, William R. Sherman, Jeffrey D. Will, "Developing Virtual Reality Applications", 2017.
3. Grigore C. Burdea, Philip Coiffet, "Virtual Reality Technology", 2nd Edition, Wiley India, 2017.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
1	Virtual Reality Environments		CO1
1.1	Choosing reality: VR, AR, or MR.	1	CO1
1.2	Real-time computer graphics, Requirements for VR, AR, and MR, – Environmental storytelling..	1	CO1
1.3	Computer Generated Worlds for VR , AR Foundation package, OpenXR,	2	CO1
1.4	Universal Render Pipeline (URP), SDKs for XR development, XR Interaction Toolkit,	1	CO1
1.5	Defining Visual Space for MR. Defining Position and Orientation	1	CO1
2	Interactive devices		CO2
2.1	Head-Mounted Displays, Augmenting Displays, Binocular Displays.	1	CO2
2.2	Smartphone-Based displays, Sensors for	1	CO2
2.3	Tracking Position, Orientation, and Motion, Trackers, 2D Versus 3D Interaction and Navigation,	1	CO2
2.4	Hand and Gesture Tracking, Whole Body Tracking.	1	CO2
2.5	Web3, Metaverse and Virtual Assets, tokens and dApp.	2	CO2
3	VR Programming		CO3
3.1	Creation of VR environment, Digital twin and VR/AR,	1	CO3
3.2	Unity 3D, setting up the Unity project, Unity Editor and its interface,	2	CO3, CO6
3.3	Understanding Game Objects and Components,	2	CO3, CO6
3.4	Creating a basic scene in Unity and adding objects, Setting up an AR project in Unity,	2	CO3, CO6
3.5	Using AR Foundation, Deploying AR experiences onto mobile devices,	1	CO3, CO6
3.6	Visual scripting in Unity. Digital Twin models	2	CO3
4	Generative Artificial Intelligence (AI) in AR/VR		
4.1	Using AI Assistant to set up a scene.	1	CO4, CO6
4.2	Animation Generator to bring character to life, Sprite Generator to make an avatar icon,	2	CO4, CO6
4.3	Use Sound Generator for background audio. Finalize and share your project.	1	CO4, CO6

4.4	AR VR case studies- Business and Manufacturing, Product visualization and Marketing, Public Safety and Military flight simulation and training.	2	CO4,
5	AR VR Project Development		
5.1	Exploring a VR Model in a HMD and navigation. template.	2	CO5, CO6
5.2	A game scene to demonstrate a player movement to demonstrate Rigidbody, collider C# scripts and	2	CO5, CO6
5.3	Visual scripting.	2	CO5, CO6
5.4	Development of Mobile AR applications using AR Core	2	CO5, CO6
	Total	36	

1. Dr. N. Shivakumar shiva@tce.edu

