CURRICULUM AND DETAILED SYLLABI

FOR

M.C.A DEGREE PROGRAMME

Department Vision and Mission

ACADEMIC YEAR 2026 – 2027 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

DEPARTMENT OF COMPUTER APPLICATIONS

Vision

Transforming individuals into globally competent computing professionals to address technological challenges and contribute to societal development

Mission

The Department of MCA is committed to:

- Empowering students with knowledge, skills, and attitude to develop competency in Computer applications through a well-defined curriculum
- Adopting innovative content delivery, assessment and continuous improvement methods to achieve desired outcomes
- Promoting active collaboration with Industry to address the technological needs of the society.
- Practicing ethical standards by the faculty and students.

PEOs

PEO1: Graduates will pursue a professional career in the domain of Computer Applications in industry or academia.

PEO2: Graduates will demonstrate technical, managerial, and leadership skills to develop and manage computing solutions that address societal and industrial needs.

PEO3: Graduates will demonstrate ethics, leadership, communication skills and lifelong learning for sustained career growth and societal contribution.

POs

PO1 (Foundation Knowledge):

Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving.

PO2 (Problem Analysis):

Identify, review, formulate and analyze problems for primarily focusing on customer requirements using critical thinking frameworks.

PO3 (Development of Solutions):

Design, develop and investigate problems with an innovative approach for solutions incorporating ESG/SDG goals.

PO4 (Modern Tool Usage):

Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.

PO5 (Individual and Teamwork):

Function and communicate effectively as an individual or a team leader in diverse and multidisciplinary groups. Use methodologies such as agile.

PO6 (Project Management and Finance):

Use the principles of project management such as scheduling, work breakdown structure and be conversant with the principles of Finance for profitable project management.

PO7 (Ethics):

Commit to professional ethics in managing software projects with financial aspects. Learn to use new technologies for cyber security and insulate customers from malware.

PO8 (Life-long Learning):

Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

CURRICULUM AND DETAILED SYLLABI

FOR

M.C.A DEGREE PROGRAMME

CREDIT DISTRIBUTION

&

SCHEDULING OF COURSES

ACADEMIC YEAR 2026 – 2027 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 - 625015

Credit Distribution for MCA Degree Programme – 2026 – 2027 Batch

S.No	Category of Courses	Credits Regular
A	Basic Science Courses (BSC)	04
B	Professional Core Courses (PCC)	**48**
C	Professional Elective Courses (PEC)	**14**
D	Employability Enhancement Courses (EEC)	**16**
E	Internship and Mandatory Audit Courses (AC) as per Regulatory authorities (Non-Credit and not included in CGPA)	-
	Minimum Credits to be earned for the award of the Degree	**82**

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI – 625 015

DEPARTMENT OF COMPUTER APPLICATIONS

SCHEDULING OF COURSES

M	Theory / Theory cum Practical / Practical								Audit Courses (Mandatory Non- credit)	Credit
	1	2	3	4	5	6	7	8		
I	26CASA0 Mathematical Foundations of Computer Science (BSC-4)	26CACA0 Data Structures (PCC-4)	26CACB0 Database Systems (PCC-4)	26CACC0 Computer Networks (PCC-4)	26CACD0 Python for Data Science (PCC-4) TCP	26CACE0 Database Systems Laboratory (PCC Lab 0:2)	26CACF0 Data Structures Laboratory (PCC Lab-0:2)	26CAHA0 Professional Communication (EEC 0:2)	26CCAA0 Vision and Speed Maths (AC-0)	26
II	26CACG0 Design and Analysis of Algorithms (PCC-4)	26CACH0 Object Oriented Programming (PCC-4)	26CACJ0 Operating Systems (PCC-4)	26CAEA0 Elective-I (PEC-4)	26CACK0 Full Stack Application Development (PCC-4) TCP	26CACL0 Object Oriented Programming Laboratory (PCC Lab 0:2)	26CACM0 Software development Laboratory (PCC Lab 0:2)	26CACN0 Design and Analysis of Algorithms Laboratory (PCC Lab 0:2)	26CCAB0 Foundations of Problem Solving -1 (AC-0)	26
III	26CACP0 Machine Learning (PCC-4)	26CAEB0 Elective-II (PEC-4)	**26CAEC0 Elective -III (PEC-3)**	**26CAED0 Elective-IV (PEC-3)**		26CACQ0 Mobile Application Development Laboratory (PCC Lab 0:2)		26CAHD0 Mini Project (EEC 0:2)		18
IV	26CAHE0 Project work (EEC 0:12)									12
	Total									**82**

Category-wise credits BSC: 4 | PCC: **48** | PEC: **14** | EEC: **16** | Total: 82

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI- 625 015
(A Govt. Aided, ISO 9001:2008 certified Autonomous Institution affiliated to Anna University)

MCA (Master of Computer Applications) Programme

Categorization of Courses

List of Basic Science Course (4)

26CASA0- Mathematical Foundations of Computer Science (4)

List of Professional Core Courses (46)

26CACA0 - Data Structures (4)
26CACB0 - Database Systems (4)
26CACC0 - Computer Networks (4)
26CACD0 – Python for Data Science (4)
26CACE0 - Database systems Laboratory (2)
26CACF0 - Data Structures Laboratory (2)
26CACG0 - Design and Analysis of Algorithms (4)
26CACH0- Object Oriented Programming (4)
26CACJ0- Operating Systems (4)
26CACK0- Full Stack Application Development (4)
26CACL0 - Object Oriented Programming Laboratory (2)
26CACM0- Software development Laboratory (2)
26CACN0-Design and Analysis of Algorithm Laboratory (2)
26CACP0 - Machine Learning (4)
26CACQ0 - Mobile Application Development Laboratory (2)

Programme Elective Courses (14)

26CAEA0 - Elective-I (4)
26CAEB0 - Elective-II (4)
26CAEC0 - Elective-III (3)
26CAED0 - Elective-Iv (3)

Employability Enhancement Courses (EEC) (16)

26CAHA0 - Professional Communication (2)
26CAHD0 – Mini Project (2)
26CAHE0 – Project work (12)

Audit Course (0)

26CCAA0 – Vision and Speed Maths
26CCAB0- Foundations of Problem Solving -1

CURRICULUM AND DETAILED SYLLABI

FOR

M.C.A DEGREE PROGRAMME

FIRST & SECOND SEMESTER

ACADEMIC YEAR 2026 – 2027 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

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

Master of Computer Applications (M.C.A) Degree Program (2 Years)
COURSES OF STUDY
(For the candidates admitted from 2026-2027 onwards)

FIRST SEMESTER

Course code	Name of the Course	Category	No. of Hours / Week			Credits
			L	T	P	
THEORY						
26CASA0	Mathematical Foundations of Computer Science	BSC	3	1	-	4
26CACA0	Data Structures	PCC	3	1	-	4
26CACB0	Database Systems	PCC	3	1	-	4
26CACC0	Computer Networks	PCC	3	1	-	4
THEORY CUM PRACTICAL						
26CACD0	Python for Data Science	PCC	1	-	6	4
PRACTICAL						
26CACE0	Database systems Laboratory	PCC	-	-	4	2
26CACF0	Data Structures Laboratory	PCC	-	-	4	2
26CAHA0	Professional Communication	EEC	-	1	2	2
26CCAA0	Vision and Speed Maths	AC	-		-	-
Total			13	5	16	26

SECOND SEMESTER

Course code	**Name of the Course**	**Category**	**No. of Hours / Week**			**Credits**
			L	**T**	**P**	
THEORY						
26CACG0	Design and Analysis of Algorithms	PCC	3	1	-	4
26CACH0	Object Oriented Programming	PCC	3	1	-	4
26CACJ0	Operating Systems	PCC	3	1	-	4
THEORY CUM PRACTICAL						
26CACK0	Full Stack Application Development	PCC	1	-	6	4
ELECTIVE						
26CAEA0	Elective – I	PEC	3	1	-	4
PRACTICAL						
26CACL0	Object Oriented Programming Laboratory	PCC	-	-	4	2
26CACM0	Software Development Laboratory	PCC	-	-	4	2
26CACN0	Design and Analysis of Algorithm Laboratory	PCC			4	2
26CCAB0	Foundations of Problem Solving -1	AC			-	-
	Total		**13**	**4**	**18**	**26**

Passed in BoS Meeting on 23.05.2026 Approved in 71st Academic Council Meeting on 27.06.2026

THIRD SEMESTER

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
26CACP0	Machine Learning	PCC	3	1	-	4
26CAEB0	Elective – II	PEC	3	1	-	4
26CAEC0	Elective – III	PEC	3	-	-	3
26CAED0	Elective – III	PEC	3	-	-	3
PRACTICAL						
26CACQ0	Mobile Application Development Laboratory	PCC	-	-	4	2
26CAHD0	Mini Project	EEC	-	-	4	2
Total			**12**	**2**	**8**	**18**

FOURTH SEMESTER

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
PRACTICAL						
26CAHE0	Project work	EEC	-	-	24	12
Total					**24**	**12**

BSC : Basic Science Courses (Foundation Courses)
PCC : Programme Core Courses
PEC : Programme Elective Courses
EEC : Employability Enhancement Courses

L : Lecture

T : Tutorial

P : Practical

Note:

1 Hour Lecture/week is equivalent to 1 credit

1 Hour Tutorial/week is equivalent to 1 credit

2 Hour Practical/week is equivalent to 1 credit

Total credits to be earned for the award of degree: 82

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI – 625 015

Master of Computer Applications (M.C.A) Degree Program

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2026-2027 onwards)

FIRST SEMESTER

S. No.	Sub. Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Pass Mark for 100	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	26CASA0	Mathematical Foundations of Computer Science	3	40	60	100	45	50
2	26CACA0	Data Structures	3	40	60	100	45	50
3	26CACB0	Database Systems	3	40	60	100	45	50
4	26CACC0	Computer Networks	3	40	60	100	45	50
THEORY CUM PRACTICAL								
5	26CACD0	Python for Data Science	3	50	50	100	45	50
PRACTICAL								
6	26CACE0	Database systems Laboratory	3	60	40	100	45	50
7	26CACF0	Data Structures Laboratory	3	60	40	100	45	50
8	26CAHA0	Professional Communication	3	60	40	100	45	50

SECOND SEMESTER

S.No.	Sub. Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Pass Mark for 100	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	26CACG0	Design and Analysis of Algorithms	3	40	60	100	45	50
2	26CACH0	Object Oriented Programming	3	40	60	100	45	50
3	26CACJ0	Operating Systems	3	40	60	100	45	50
THEORY COM PRACTICAL								
4	26CACK0	Full Stack Application Development	3	50	50	100	45	50
ELECTIVE								
5	26CAEA0	Elective - I	3	40	60	100	45	50
PRACTICAL								
6	26CACL0	Object Oriented Programming Laboratory	3	60	40	100	45	50
7	26CACM0	Software Development Laboratory	3	60	40	100	45	50
8	26CACN0	Design and Analysis of Algorithm Laboratory	3	60	40	100	45	50

THIRD SEMESTER

S.No.	Sub. Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Pass Mark for 100	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	26CACP0	Machine Learning	3	40	60	100	45	50
2	26CAEB0	Elective – II	3	40	60	100	45	50
3	26CAEC0	Elective – III	3	40	60	100	45	50
4	26CAED0	Elective – IV	3	40	60	100	45	50
PRACTICAL								
6	26CACP0	Mobile Application Development Laboratory	3	60	40	100	45	50
7	26CAHD0	Mini Project	3	60	40	100	45	50

FOURTH SEMESTER

S.No.	Sub. Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Pass Mark for 100	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
PRACTICAL								
1	26CAHE0	Project work	3	60	40	100	45	50

* Continuous Assessment (CA) evaluation pattern will differ from subject to subject 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.

CURRICULUM AND DETAILED SYLLABI

FOR

M.C.A DEGREE PROGRAMME

FIRST & SECOND SEMESTER

ACADEMIC YEAR 2026 – 2027 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

		Category	L	T	P	Credit
26CASA0	**Mathematical Foundations of Computer Science**	BSC	3	1	0	4

Preamble

Discrete mathematics forms the mathematical foundation for computer science and engineering applications by developing concepts of logic, proof techniques, and counting principles for systematic problem solving. Emphasis is given to recurrence relations and generating functions for solving recursive processes, along with graph and coloring techniques for modeling networks. Fundamental concepts of number theory and their applications are also explored.

Prerequisite

- Nil

Course Outcomes

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

No.	Course Outcome	TPS Level	Performance Indicator	CO Weightage	SDG addressed	SDG Level
CO1	Apply propositional logic, predicates, quantifiers, logical equivalences, and rules of inference to construct and validate mathematical arguments.	TPS3	PO1	17%	SDG4	2
CO2	Apply counting principles, permutations, combinations, and related techniques to solve discrete problems.	TPS3	PO1	16%	SDG4	2
CO3	Solve recurrence relations and apply generating functions to computational problems.	TPS3	PO1	17%	SDG4	2
CO4	Use graph concepts, representations, and connectivity properties to model and examine graph structures.	TPS3	PO1, PO3	17%	SDG4	2
CO5	Use concepts of Euler and Hamilton paths, shortest path algorithms, and graph coloring techniques to solve real-world problems.	TPS3	PO1, PO3	17%	SDG4	2
CO6	Apply number theoretic concepts, including divisibility, primes, and congruences, in solving computational and mathematical problems.	TPS3	PO1, PO4	16%	SDG4	2

Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	**34**	**34**			**17**
CO2	**32**	**32**			**16**
CO3	**34**	**34**			**17**
CO4			**34**	**34**	**17**
CO5			34	34	**17**
CO6			**32**	**32**	**16**
Total	**100**	**100**	**100**	**100**	**100**

Syllabus

The Foundations: Logic and Proofs: Propositional logic – Applications of Propositional Logic – Propositional Equivalences – Predicates and Quantifiers – Nested Quantifiers- Rules of inference.

Counting: The Basics of Counting –The Pigeonhole Principle – Inclusion – Exclusion- Permutations and Combinations.

Advanced Counting Techniques: Applications of Recurrence Relations – Solving Linear Homogeneous Recurrence Relations with Constant Coefficients – Linear Nonhomogeneous Recurrence Relations with Constant Coefficients- Generating functions.

Graphs: Graphs and Graph Models – Graph Terminology and Special Types of Graphs – Representing Graphs and Graph Isomorphism – Connectivity – Euler and Hamilton Paths – Shortest Path Problems--Graph Coloring.

Number Theory: Divisibility and Modular Arithmetic - Integer Representations and Algorithms - Primes and Greatest Common Divisors - Solving Congruences - Applications of Congruences.

Learning Resources

Text Book

1. Kenneth H. Rosen., "Discrete Mathematics and Its Applications", 8th Edition, McGraw-Hill Education, 2019.
The Foundations: Logic and Proofs: [1.1 - 1.6]
Counting: [6.1, 6.2, 8.5, 6.3]
Advanced Counting Techniques: [8.1, 8.2, 8.4]
Graphs: [10.1 – 10.6,10.8]
Number Theory: [4.1 – 4.5]
2. J.P. Tremblay and R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science",1st Edition, McGraw Hill Education,2017.

Reference Books & Web resources

1.S.B. Singh, Jai Kishore and Ekata Gupta, "Discrete Structures", 3rd Edition, Khanna Publishing House, 2017.
2. S.B. Singh, "Combinatorics and Graph Theory", 3rd Edition, Khanna Publishing House, 2018.
3. Ralph p. Grimaldi, "Discrete and Combinatorial Mathematics An Applied Introduction",5th Edition, Pearson Education,2019.

Course Contents and Lecture Schedule

Module No	Topics	No of Periods
1	**The Foundations: Logic and Proofs**	
1.1	Propositional logic	1
1.2	Applications of propositional logic	2
	Tutorial	1
1.3	Propositional Equivalences	1
1.4	Predicates and Quantifiers	2
	Tutorial	1
1.5	Nested Quantifiers	1
1.6	Rules of inference	1
2	**Counting**	
2.1	The Basics of counting	1
	Tutorial	1
2.2	The Pigeonhole Principle	1
2.3	Inclusion – Exclusion	1

2.4	Permutation and Combinations	1
	Tutorial	1
3	**Advanced Counting Techniques**	
3.1	Applications of Recurrence Relations	1
3.2	Solving Linear Homogeneous Recurrence Relations with Constant Coefficients	2
	Tutorial	1
3.3	Linear Nonhomogeneous Recurrence Relations with Constant Coefficients	1
3.4	Generating Functions	2
	Tutorial	1
4	**Graphs**	
4.1	Graphs and Graph Models	1
4.2	Graph Terminology and Special Types of Graphs	2
	Tutorial	1
4.3	Representing Graphs and Graph Isomorphism	2
4.4	Connectivity	1
	Tutorial	1
4.5	Euler and Hamilton Paths	2
4.6	Shortest – Path Problems	1
	Tutorial	1
4.7	Graph Coloring	2
5	**Number Theory**	
5.1	Divisibility and Modular Arithmetic	1
	Tutorial	1
5.2	Integer Representations and Algorithms	2
5.3	Primes and Greatest Common Divisors	1
	Tutorial	1
5.4	Solving Congruences	2
5.5	Applications of Congruences	1
	Tutorial	1
	Total	48

Course Designer(s):

1. Dr. A. Anitha, Assistant Professor, Department of Mathematics, anithavalli@tce.edu
2. Dr. P. Krishnapriya, Assistant Professor, Department of Mathematics, pkamat@tce.edu
3. Dr. P. Jeyadurga, Assistant Professor, Department of Mathematics, pjdmat@tce.edu

		Category	L	T	P	Credit
26CACA0	**DATA STRUCTURES**	**PCC**	**3**	**1**	**0**	**4**

1. Preamble

This course introduces various data structures and their operations for effective data manipulation. Students will learn how to organize, access, and update data efficiently through computer programs.

2. Prerequisite

- Nil

3. Course Outcomes

After the successful completion of the course, students will be able to:

No.	Course Outcome	TPS Level	Program outcomes	CO Weightage	SDG addressed	SDG Level
CO1	Use different types of linked lists, such as singly, doubly, and circular linked lists, to solve simple computing problems	TPS3	PO1, PO2	15%	SDG 4	2
CO2	Implement stacks, linear queues, circular queues, and deques to solve simple problems such as expression evaluation, symbol balancing, and task scheduling	TPS3	PO1, PO2	15%	SDG 9	2
CO3	Use binary trees and different types of heaps, such as binary heap, leftist heap, and binomial queue, for data organization, priority handling, and encoding applications	TPS3	PO1, PO2	20%	SDG 9	2
CO4	Apply binary search trees and balanced binary search trees, such as AVL trees, Splay trees, and Red-Black trees, to improve search efficiency and manage dynamic data.	TPS3	PO1, PO2	20%	SDG 9	2
CO5	Apply multiway search trees, different types of tries, and disjoint sets to solve problems related to information retrieval, pattern matching, and union-find operations.	TPS3	PO1, PO2	15%	SDG 9	2
CO6	Use internal and external sorting methods, searching techniques, and hashing techniques, including collision resolution and extendible hashing, to solve computational problems efficiently.	TPS3	PO1, PO2	15%	SDG 9	2

4. Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	30	25			15
CO2	30	25			15
CO3	40	50			20
CO4			40	50	20
CO5			30	25	15
CO6			30	25	15
Total	**100**	**100**	**100**	**100**	**100**

5. Syllabus

Module 01: 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.
Module 02: 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.
Module 03: STACKS AND QUEUES:
Stack – Operations – Applications – Balancing Symbols – Evaluating arithmetic expressions - Infix to Postfix conversion – Function Calls – Queue – Operations – Circular Queue – DeQue – Applications of Queues – Scheduling - Applications of DeQue – clock-wise and anti-clockwise rotational operation.
Module 04: 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.
Module 05: BINARY SEARCH TREES:
Binary Search Tree (BST) – Balanced BST - AVL Trees – Red Black Trees – Splay Trees – Operations
Module 06: 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
Module 07: DISJOINT SETS:
Disjoint-set operations - Representation – Smart union algorithms – Path Compression
Module 08: 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.

6. Learning Resources

6.1 Text Book

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

6.2 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.

7. Course Contents and Lecture Schedule

Module No	Topics	No of Periods
0	Syllabus Discussion (CO, CO vs PO, Assessment, Prerequisite, 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 - BigOh, Omega, Theta	2
2	**LINKED LISTS**	
2.1	Abstract Data Types (ADTs) – List ADT – Array-based implementation- Linked list implementation	2
2.2	Singly linked lists	2
2.3	Doubly-linked lists	2
2.4	Circularly linked lists	2
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**	
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
3.4	Circular Queue, DeQue	1
3.5	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	2
5.2	AVL Trees	2
5.3	Red Black Trees	2
5.4	Splay Trees	2
6	**MULTIWAY SEARCH TREE**	
6.1	Multiway Search Trees - B-Tree	1
6.2	Tries - Standard Tries - Compressed Tries	1
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	2
8	**SEARCHING, SORTING AND HASHING TECHNIQUES**	
8.1	Sorting –Insertion sort- Shell sort –. Merge Sort	2
8.2	External Sorting - The Simple Algorithm - Multiway Merge - Replacement Selection	1
8.3	Searching – Linear Search – Binary Search	2
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 Periods	48

8. Course Designer(s):

1. Dr.S.R. Mathu sudhanan, Assistant Professor, CA srmca@tce.edu

		Category	L	T	P	Credit
26CACB0	**DATABASE SYSTEMS**	**PCC**	**3**	**1**	**0**	**4**

Preamble

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

- NIL

Course Outcomes

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

No.	Course Outcome	TPS Level	Program Outcome	CO Weightage	SDG addressed	SDG Level
CO1	Make use of data models such as Entity Relationship (ER) model and Relational model for a given application requirement. (Apply)	TPS 3	PO1, PO2	10%	SDG 09	02
CO2	Demonstrate/Manipulate relational database using Relational languages and PL/SQL sub programs.	TPS 3	PO1, PO4	12%	SDG 09	02
CO3	Apply normalized databases for a given application by incorporating various constraints and normal forms.	TPS 3	PO2, PO3	15%	SDG 09	02
CO4	Apply data structures like indexes and hash tables to support fast retrieval of data and explain query processing and optimization methods.	TPS 3	PO2, PO3	17%	SDG 09	02
CO5	Make use of different forms of transactions, concurrency control and recovery mechanisms to maintain data consistency in a multi user environment.	TPS 3	PO1, PO2, PO3	20%	SDG 09	02
CO6	Demonstrate NoSQL models for building the databases.	TPS 3	PO3, PO4,	16%	SDG 09	02

Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	20	20			10
CO2	40	40			20
CO3	40	40			20
CO4			40	40	20
CO5			40	40	20
CO6			20	20	10
Total	**100**	**100**	**100**	**100**	**100**

Syllabus

Module1: 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

Module2: 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.

Module3: 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.

Module4: 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.

Module5: Data Storage and structures: 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.

Module6: No SQL databases: CAP Theorem – Document Based systems – Key value Stores – Column Based Systems – Graph Databases.

Text Book

1. Silberschatz, A, Henry F. Korth, and S. Sudharshan, “Database System concepts”, 7th Edition, Tata McGraw Hill, 2021.

Reference Books & Web resources

1. RamezElmasri and Shamkant B. Navathe, “Fundamentals of Database Systems”, 7thEdition, Pearson/Addisionwesley, 2021
2. C. J. Date, A. Kannan and S. Swamynathan, “An Introduction to Database Systems”, 8thed, Pearson Education, 2006
3. Raghu Ramakrishnan, “Database Management Systems”, Third Edition, McGraw Hill, 2022.
4. Pramod J. Sadalage, Martin Fowler, “NoSQL Distilled - A Brief Guide to the Emerging World of Polyglot Persistence”, Pearson Education, 2212.
5. Andreas Meier, Michael Kaufmann, “ SQL& NoSQL Databases - Models, Languages, Consistency Options and Architectures for Big Data Management” , Springer Fachmedien Wiesbaden, 2021.

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	2
1.3	Data Models – ER model	1
1.4	Relational model - Concept of relation, Constraints	2
1.5	Mapping ER model to Relational Model	2
2	**Relational Languages**	
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
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- Recursive Queries	2
3	**Database design**	
3.1	Decomposition – Lossless decomposition	1
3.2	Functional Dependencies - Closure set of FD's for key identification	2

3.3	Normal forms – 1NF, 2NF, 3NF	2
3.4	Dependency Preserving – BCNF, Multivalued Dependencies and 4NF	2
3.5	Join dependencies and 5NF	1
4	**Transaction and Concurrency control**	
4.1	Transaction concepts	1
4.2	Concurrent Execution, Serializability	2
4.3	Concurrency Control - Lock based protocol,	2
4.4	Deadlock handling- Deadlock detection and prevention	1
4.5	Recovery System - Log-based Recovery, Recovery with Concurrent Transaction	2
5	**Data Storage and structures**	
5.2	Database Compression, De-Duplication	1
5.3	File Organization - Indexing – Ordered Index	1
5.4	B+ tree – properties , insertion, deletion	2
5.5	Hashing - Static and dynamic hashing	2
5.6	**Query Processing and Optimization -** Basics of query processing and optimization	2
6	**Introduction to No SQL databases (4)**	
6.1	Introduction – CAP Theorem	2
6.2	Document Based systems – Key value Stores	2
6.3	Column Based Systems – Graph Databases	2
	Total	**48 Hours**

Course Designer(s):

Dr.S.VIJAYALAKSHMI, Professor , Department of Computer Applications , svlcse@tce.edu

		Category	L	T	P	Credit
26CACC0	**COMPUTER NETWORKS**	PCC	3	1	0	4

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

NIL

Course Outcomes

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

No.	Course Outcome	TPS Level	Program Outcome	CO Weightage	SDG addressed	SDG Levl
CO1	Apply the operation of network applications with an understanding of network models, switching techniques, and layered architecture.	TPS 3	PO1, PO2	12	SDG 9	2
CO2	Demonstrate the concepts of transmission and multiplexing techniques.	TPS 3	PO1, PO4	17	SDG 9	2
CO3	Solve flow and error control issues in the data link layer using appropriate techniques.	TPS 3	PO2, PO3, PO4	15	SDG 9	2
CO4	Compute the performance implications of multiple access protocols.	TPS 3	PO2, PO3	20	SDG 11	2
CO5	Implement routing and forwarding solutions for packet switching networks with an understanding of router architectures, algorithms, and protocols.	TPS 3	PO1, PO4	16	SDG 9	2
CO6	Operate transport layer , presentation layer, and Application layer protocols under different networking scenarios.	TPS 3	PO3, PO4	20	SDG 9	2

Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	25	10			10
CO2	40	40			20
CO3	35	50			20
CO4			40	50	20
CO5			30	25	10
CO6			40	25	20
Total	**100**	**100**	**100**	**100**	**100**

Syllabus

Module 1: Network Models, Types and Switching

Network Models: 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 - Infrastructure and Ad hoc networks.

Module 2: Physical Layer

Functions of Physical Layer, Data Transmission Concepts, Analog and Digital Transmission, Transmission Modes – Simplex, Half Duplex, Full Duplex, Transmission Media – Guided Media and Unguided Media, Multiplexing Techniques – Frequency Division Multiplexing (FDM), Time Division Multiplexing (TDM), Wavelength Division Multiplexing (WDM), Bandwidth Utilization, Advantages and Applications of Multiplexing.

Module 3: Data Link Layer

Functions of Data Link Layer, Framing Techniques – Character Count, Byte Stuffing, Bit Stuffing, Error Detection and Correction – Types of Errors, Single Bit Error, Burst Error, Error Detection Methods – Parity Check, Cyclic Redundancy Check (CRC), Checksum, Forward Error Correction Techniques, Data Link Control, Flow Control and Error Control, Media Access Control (MAC) – CSMA/CD, CSMA/CA, Channelization Methods – FDMA, TDMA, CDMA, Applications of MAC Protocols.

Module 4: Network Layer

Functions of Network Layer, Connecting Devices – Hubs, Bridges, Switches, Routers, Gateways, Routing and Forwarding Concepts, Routing Algorithms – Distance Vector Routing, Link State Routing, Routing Tables and Path Determination, Network Performance Metrics – Delay, Throughput, Packet Loss, Jitter, Congestion Control Techniques, Internet Protocol (IP), IPv4 Addressing – Classful Addressing and Classless Addressing, IPv6 Addressing, Internet Control Message Protocol (ICMP), Address Resolution Protocol (ARP), Reverse Address Resolution Protocol (RARP), IP Datagram Structure.

Module 5: Transport Layer and Session layer

Functions of Transport Layer, Process-to-Process Communication, Port Addressing, User Datagram Protocol (UDP) – Features of UDP, UDP Header Format, Applications of UDP, Transmission Control Protocol (TCP) – Features of TCP, TCP Header Format, Connection Establishment, Three-Way Handshaking, Flow Control, Error Control, Comparison between TCP and UDP, Reliable Data Transfer Mechanisms. Session Layer: Introduction to Session Layer, Functions of Session Layer, Session Establishment, Management, and Termination, Dialog Control, Synchronization and Checkpoint Mechanism.

Module 6: Presentation and Application Layer

Presentation Layer – Functions of Presentation Layer, Data Translation, Data Encryption and Decryption, Data Compression, Authentication Techniques, Security Services in Networks. Application Layer – Functions of Application Layer, File Transfer Protocol (FTP), TELNET, Hyper Text Transfer Protocol (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

2. Computer Networks: A Systems Approach, Larry L. Peterson, Bruce S. Davie, Elsevier, Mar 2011
3. Larry L.Peterson and Bruce S. Davie, "Computer Networks – A systems Approach" Fifth Edition, Elsevier, 2019
4. Computer Networking: A Top-Down Approach featuring the Internet, 6th edition, James F. Kurose, Pearson Education India, 2013.
5. https://nptel.ac.in/courses/106105183 – Computer Networks And Internet Protocol, IIT Kharagpur**,** Prof. Soumya Kanti Ghosh Prof. Sandip Chakraborty - NPTEL Source

Course Contents and Lecture Schedule

1	**Network Models, Types and Switching**	
1.1	The OSI Model – TCP/IP Protocol Suite – OSI versus TCP/IP	2
1.2	Circuit-Switched Networks and Packet Switching	2
1.3	Structure of a Switch and Switching in TCP/IP Layers	1
1.4	WAN, MAN and LAN	1
1.5	WLAN – Infrastructure and Ad hoc Networks	1
1.6	Applications of Different Network Types	1
2	**Physical Layer**	
2.1	Functions of Physical Layer and Data Transmission Concepts	1
2.2	Analog and Digital Transmission	1
2.3	Transmission Modes – Simplex, Half Duplex and Full Duplex	1
2.4	Guided Media and Unguided Media	1
2.5	Multiplexing Techniques – FDM, TDM and WDM	2
2.6	Bandwidth Utilization, Advantages and Applications of Multiplexing	2
3	**Data Link Layer**	
3.1	Functions of Data Link Layer and Framing Techniques	1
3.2	Character Count, Byte Stuffing and Bit Stuffing	1
3.3	Types of Errors – Single Bit Error and Burst Error	1
3.4	Error Detection Methods – Parity Check, CRC and Checksum	2
3.5	Forward Error Correction Techniques	1
3.6	Data Link Control, Flow Control and Error Control	1
3.7	Media Access Control – CSMA/CD and CSMA/CA	1

3.8	Channelization Methods – FDMA, TDMA and CDMA	1
3.9	Applications of MAC Protocols	1
4	**Network Layer**	
4.1	Functions of Network Layer and Connecting Devices	1
4.2	Routing and Forwarding Concepts	1
4.3	Distance Vector Routing and Link State Routing	2
4.4	Routing Tables and Path Determination	1
4.5	Network Performance Metrics – Delay, Throughput, Packet Loss and Jitter	1
4.6	Congestion Control Techniques	1
4.7	Internet Protocol and IP Datagram Structure	1
4.8	IPv4 Addressing – Classful and Classless Addressing	1
4.9	IPv6 Addressing	1
4.10	ICMP, ARP and RARP	1
5	**Transport Layer and Session Layer**	
5.1	Functions of Transport Layer, Process-to-Process Communication and Port Addressing	1
5.2	UDP – Features, Header Format and Applications	1
5.3	TCP – Features and Header Format	1
5.4	Connection Establishment and Three-Way Handshaking	1
5.5	Flow Control, Error Control and Reliable Data Transfer Mechanisms, Comparison between TCP and UDP	1
5.6	Functions of Session Layer, Session Establishment, Management, and Termination, Dialog Control, Synchronization and Checkpoint Mechanism	1
6	**Presentation and Application Layer**	
6.1	Functions of Presentation Layer and Data Translation	1
6.2	Data Encryption, Decryption and Authentication Techniques	1
6.3	Data Compression and Security Services in Networks	1
6.4	Functions of Application Layer	1
6.5	FTP, TELNET, HTTP and Secure Shell (SSH)	2
6.6	Domain Name System (DNS), Client-Server Communication and Web Services	1
	Total	**48**

Course Designer(s):

1. Dr.C.Mahadevi Assistant Professor Computer Applications cmids@tce.edu

26CACD0	PYTHON FOR DATA SCIENCE	Category TCP(P)	L 1	T 0	P 6	Credits 4

Preamble **Terminal Exam: Practical**

The course on Python for Data Science is intended to introduce students to the fundamental concepts of data science and data analysis using Python. The course aims to provide exposure to collecting, cleaning, manipulating, and analysing data through Python libraries and tools. It aims to train the students in performing data operations and deriving meaningful insights from real-world datasets. This course also includes a laboratory component designed to provide hands-on experience in implementing data science techniques and practical data analysis applications using Python.

Prerequisite

NIL

Course Outcomes

No.	Course Outcome	TCE Proficiency Scale	Program Outcome	CO Weigh tage	SDG Addressed	SDG Level
CO1	Apply Python language basics, control structures, data structures, functions, and file handling techniques to develop Python programs.	TPS – 3	PO1, PO2	20%	SDG 4	2
CO2	Use NumPy arrays, universal functions, array operations, linear algebra, and random number generation techniques for numerical computing.	TPS - 3	PO1, PO4	15%	SDG 9	2
CO3	Demonstrate Pandas data structures, data loading, descriptive statistics, data aggregation, and preprocessing techniques for data analysis.	TPS - 3	PO2, PO4	15%	SDG 4	2
CO4	Implement data loading, data transformation, string manipulation, database interaction, and data wrangling operations using Python libraries.	TPS - 3	PO2, PO3, PO4	15%	SDG 9	2
CO5	Compute data aggregation, group operations, pivot tables, cross tabulation, and date-time analysis using Pandas techniques.	TPS - 3	PO3, PO4	15%	SDG 4	2
CO6	Demonstrate data visualization techniques for categorical, multivariable, distribution, and time series data using Python visualization libraries.	TPS - 3	PO3, PO4, PO8	20%	SDG 11	2

Assessment Pattern

CO	Theory		Practical		Terminal Exam
	CAT1	CAT2	OCR/Continuous Assessment	Model Test CO-wise	
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	40	-	25	25	25
CO2	30	-	10	10	10
CO3	30	-	10	10	10
CO4		-	15	15	15
CO5		-	15	15	15
CO6		-	25	25	25
Total	100	100	100	100	100

Assessment Pattern: Practical Component

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, Post-lab	75	Conducted throughout the semester
Model Test	CO-wise (Cognitive, Psychomotor, Affective)	25	Rubrics covering all Domains of learning
Total Internal Marks		**100**	
End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	Rubrics covering all Domains of learning

Syllabus

Module1:

Python Language Basics and Data Structures: Python Language Basics - Scalar Types - Control Flow. Data Structures and Sequences: Tuple - List - Built-in Sequence Functions - dict - set- List, Set, and Dict Comprehensions. Functions: Namespaces, Scope, and Local Functions - Returning Multiple Values - Functions Are Objects - Files and the Operating System.

Module 2:
Numpy Basics: The NumPy ndarray: A Multidimensional Array Object - Universal Functions: Fast Element-Wise Array Functions - Array-Oriented Programming with Arrays - File Input and Output with Arrays - Linear Algebra - Pseudorandom Number Generation.

Module 3:
Pandas Basics: Introduction to pandas Data Structures –Loading and Understanding Data- Data aggregation for computing Descriptive Statistics- Data Cleaning and Preprocessing

Module 4:
Data Loading, Storage, and File Formats: Reading and Writing Data in Text Format - Binary Data Formats - Interacting with Web APIs - Interacting with Databases. Data Exploration: Data Transformation - String Manipulation. Data Wrangling: Hierarchical Indexing - Combining and Merging Datasets - Reshaping and Pivoting.

Module 5:
Data Wrangling: Data Aggregation and Group Operations: GroupBy Mechanics - Data Aggregation - Apply: General split-apply-combine - Pivot Tables and Cross Tabulation - Date and Time Data Types.

Module 6:
Data Visualization: Introduction to Data Visualization- Visualizing categorical data, visualizing time series data, Visualizing multiple variables -Visualizing Distribution & Relationships - Multivariate and Time Series Visualization exploration, Case Study – Regression and Classification.

Text Book

1. Jeronimo Reguera, “Python for Data Science”, Publiconsulting Media, 2020.
2. Wes McKinney, “Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Jupyter”, O’Reilly Media, 2022.
3. Dr. Mohd. Abdul Hameed, “Python for Data Science”, Wiley Emerging Technology Series, 2020.
4. Veit Schiele, “Python for Data Science”, Release 24.3.0, 2024.
5. Steve Blair, “Python Data Science: The Ultimate Handbook for Beginners on How to Explore NumPy for Numerical Data, Pandas for Data Analysis, IPython, Scikit-Learn and TensorFlow for Machine Learning and Business”, Independently Published, 2019.
6. Wes McKinney, “Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython”, O’Reilly Media, 2017.

Reference Books & web resources

1. S. K. Mukhiya and U. Ahmed, “Hands-on Exploratory Data Analysis with Python”, Packt Publishing, 2020.
2. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, O’Reilly Media, 2017.
3. D. Cielen, A. D. B. Meysman and M. Ali, “Introducing Data Science”, Manning Publications, 2016.
4. Matthew O. Ward, Georges Grinstein and Daniel Keim, “Interactive Data Visualization: Foundations, Techniques, and Applications”, A. K. Peters/CRC Press, 2015.
5. NPTEL course on "Python for Data Science" by Prof. Ragunathan Rengasamy, IIT Madras - https://nptel.ac.in/courses/106106212
6. Swayam MOOC course on "Data Science Using Python" by Prof. Faisal Anwer, Prof. Mohammad Nadeem, Aligarh Muslim University - https://www.classcentral.com/course/swayam-data-science-using-python-292953.
7. NPTEL course on "Data Analytics with Python", Prof. A. Ramesh, IIT Roorkee.- https://nptel.ac.in/courses/106107220

Course Contents and Lecture Schedule

Module No	Topics	No of Periods	Course Outcomes
1	**Python Language Basics and Data Structures**	(2)	
1.1	Introduction to Python Language Basics and Scalar Types	1	CO1
1.2	Control Flow, Data Structures, Functions, Files, and Operating System Handling	1	CO1
2	**NumPy Basics**	(2)	
2.1	Introduction to NumPy ndarray and Universal Functions	1	CO2
2.2	Array Programming, Linear Algebra, File I/O, and Random Number Generation	1	CO2
3	**Pandas Basics**	(2)	
3.1	Introduction to Pandas Data Structures and Data Loading	1	CO3
3.2	Descriptive Statistics, Data Cleaning, and Preprocessing	1	CO3
4	**Data Loading, Storage, and Data Wrangling**	(2)	
4.1	Data Loading, Web APIs, and Database Interaction	1	CO4
4.2	Data Transformation, String Manipulation, Merging, and Pivoting	1	CO4
5	**Data Aggregation and Group Operations**	(2)	
5.1	GroupBy Mechanics, Aggregation, and Split-Apply-Combine	1	CO5
5.2	Pivot Tables, Cross Tabulation, and Date-Time Data Types	1	CO5
6	**Data Visualization and Case Studies**	(2)	
6.1	Data Visualization Techniques and Relationship Analysis	1	CO6
6.2	Multivariate Visualization, Regression, and Classification Case Studies	1	CO6
	Total No. of Hours	**12**	

List of Experiments with CO Mapping

Sl. No.	Name of the Experiment	Level of Experiment	Objective(s)	COs Addressed
1	**Python Language Basics and Data Structures**			
1.1	Programs using Python language basics, scalar types, control flow statements, tuples, lists, dictionaries, sets, sequence functions, and comprehensions	02	To apply Python programming fundamentals and data structures for problem solving	CO1

1.2	Programs using functions, namespaces, scope, local functions, returning multiple values, and functions as objects	02	To demonstrate the use of functions and scope management in Python programs	CO1
1.3	Program to perform file handling operations including file creation, reading, writing, appending, and counting the number of words, lines, and characters in a file	02	To implement file handling operations and operating system functionalities using Python	CO1
2	**NumPy Basics**			
2.1	Program using NumPy arrays to perform array creation, reshaping, indexing, slicing, and mathematical operations	02	To implement multidimensional array operations using NumPy	CO2
2.2	Program using NumPy to generate random numbers and perform statistical operations on arrays	02	To apply NumPy functions for statistical analysis and random data generation	CO2
2.3	Program to convert NumPy arrays into Pandas DataFrames and perform descriptive statistical analysis	02	To demonstrate integration of NumPy and Pandas for data analysis	CO2, CO3
2.4	Program using NumPy linear algebra functions to compute determinant, inverse, eigenvalues, and eigenvectors of matrices	02	To solve linear algebra problems using NumPy functions	CO2
3	**Pandas Basics**			
3.1	Program to create and manipulate a Pandas DataFrame using different data sources such as lists, dictionaries, and CSV files	02	To create and manage structured datasets using Pandas DataFrames	CO3
3.2	Program to load and understand datasets using Pandas functions	02	To perform dataset loading and exploratory analysis using Pandas	CO3
3.3	Program to perform descriptive statistics and data aggregation operations using Pandas	02	To compute statistical measures and aggregate data for analysis	CO3

3.4	Program to perform data cleaning and preprocessing techniques using Pandas	02	To implement preprocessing techniques for improving data quality	CO3
4	**Data Loading, Storage, and Data Wrangling**			
4.1	Program to read and write text and binary data files using Python	02	To perform data loading and storage operations using file formats	CO4
4.2	Program to interact with Web APIs and retrieve JSON data for analysis	02	To access and process data from Web APIs	CO4
4.3	Program to connect and interact with databases using Python	02	To implement database operations using Python libraries	CO4
4.4	Program to perform string manipulation and data transformation operations	02	To apply string processing and transformation techniques for data exploration	CO4
4.5	Program to perform hierarchical indexing, merging, reshaping, and pivoting operations using Pandas	02	To implement data wrangling techniques for structured datasets	CO4
5	**Data Aggregation and Group Operations**			
5.1	Program using GroupBy mechanics and aggregation functions in Pandas	02	To perform grouped data analysis and aggregation operations	CO5
5.2	Program to implement split-apply-combine operations using Pandas	02	To apply split-apply-combine strategies for data analysis	CO5
5.3	Program to create pivot tables and perform cross tabulation operations	02	To summarize and analyse data using pivot tables and cross tabulation	CO5
5.4	Program to handle date and time data types using Pandas	02	To manipulate and analyse time-based datasets	CO5
6	**Data Visualization and Case Studies**			
6.1	Program to visualize categorical data using bar charts and pie charts	02	To create visual representations for categorical datasets	CO6

6.2	Program to visualize time series data using line plots	02	To analyse trends and patterns in time series datasets	CO6
6.3	Program to visualize multiple variables using scatter plots and heat maps	02	To analyse relationships among multiple variables using visualization techniques	CO6
6.4	Program to perform distribution and relationship analysis using histograms and box plots	02	To analyse data distribution and relationships among variables	CO6
6.5	Program to implement regression and classification case studies using Python libraries	02	To apply predictive analytics techniques for solving real-world problems	CO6
7	Mini Project	02	To develop a Python-based Data Science application using real-world datasets for data preprocessing, analysis, visualization, and predictive analytics using NumPy, Pandas, Matplotlib, Web APIs, databases, regression, and classification techniques	CO1, CO2, CO3, CO4, CO5, CO6
	Total No. of Practical Hours			**36**

Course Designer(s):

1. Dr.C.Mahadevi Assistant Professor Computer Applications cmids@tce.edu

26CACE0	DATABASE SYSTEMS LABORATORY	Category	L	T	P	Credit
		PCC	0	0	4	2

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

Cos	Course Outcome	TCE Proficiency Scale	Program Outcomes	SDG Addressed	SDG Level
CO1	Implement Entity Relationship and Relational models for the given real-world application.	TPS 3	PO1, PO2	SDG 04	02
CO2	Implement normalized database using various constraints.	TPS 3	PO1, PO3	SDG 09	02
CO3	Implement relational database and manipulate them using simple and complex queries in SQL	TPS 3	PO2, PO3,	SDG 09	02
CO4	Implement and use different database objects like Index, View, Sequence, abstract data types, Varray and Nested table using SQL	TPS 3	PO3, PO4	SDG 09	02
CO5	Implement sub programs like Procedure, Functions, Triggers and Package using Pl/SQL and manipulate the database through these programs	TPS 3	PO3, PO4,	SDG 09	02
CO6	Implement a complete database application using higher level language through JDBC /ODBC and NOSQL Database.	TPS 3	PO2, PO3,	SDG 09	02

Assessment Pattern: Cognitive Domain

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, post-lab	75	Conducted throughout the semester
Model Test	CO-wise (Cognitive, Psychomotor, Affective)	25	Mid/End semester internal test
Total Internal Marks		100	
End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	As directed by COE or as per curriculum

List of Experiments with CO Mapping

Sl. No.	Name of the Experiment	Level of Experiment	Objective(s)	COs Addressed
1.	Analyze the database application and create relational databases using normalization principles.	01	To analyse the given database application and design relational database tables using normalization principles to reduce data redundancy and maintain data consistency.	CO1
2.	Add the necessary integrity constraints to relational databases and practice with DCL commands.	01	To apply necessary integrity constraints to relational databases and practice Data Control Language (DCL) commands for maintaining data accuracy and database security.	CO2
3.	Manipulate the database using simple SQL Queries and practice with TCL commands such as COMMIT, ROLLBACK and SAVEPOINT commands	02	To manipulate databases using simple SQL queries and practice Transaction Control Language (TCL) commands such as COMMIT, ROLLBACK, and SAVEPOINT for transaction management.	CO3
4.	Manipulate the database using Complex SQL Queries	02	To develop the ability to manipulate and retrieve data from databases using complex SQL queries for effective database management and analysis.	CO3
5.	Practice with different database objects and complex data types such as BLOB, CLOB, NCLOB, BFILE	02	To understand and work with different database objects and complex data types such as BLOB, CLOB, NCLOB, and BFILE in database applications.	CO4
6.	Practice with PL/SQL blocks, programming constructs and composite data types in PL/SQL	03	To understand and implement PL/SQL blocks, programming constructs, and composite data types for developing efficient procedural database applications.	CO5
7.	Develop sub programs such as procedures, functions with cursors and demonstrate exceptions using PL/SQL.	03	To understand and implement PL/SQL procedures, functions, cursors, and exception handling techniques for effective database application development.	CO5
8.	Create Packages and Triggers using PL/SQL	04	To understand the concept of packages and triggers in PL/SQL. To create packages for modular and reusable database programming.	CO5
9.	Develop a web application with database connectivity	04	To develop a web application with database connectivity for performing dynamic data storage, retrieval, and management operations.	CO6
10.	Demonstrate NoSQL database models and execute queries.	04	To demonstrate NoSQL database models and execute queries for storing and retrieving unstructured	CO6

			and semi-structured data efficiently.	

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:

Dr.S.Vijayalakshmi, Professor,Department of Computer Applications, svlcse@tce.edu

26CACF0	DATA STRUCTURES LABORATORY	Category	L	T	P	Credits
		PCC	0	0	4	2

Preamble

Hands-on implementation and analysis of data structures form the core focus of this course. Students gain practical understanding of the working principles of various data structures and their application in solving simple computational problems. The course strengthens programming proficiency, logical reasoning, creativity, and analytical skills required for designing efficient solutions.

Prerequisite

- NIL

Course Outcomes

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

COs	Course Outcome 1 (CO1)	TCE Proficiency Scale	Program Outcomes	SDG Addressed	SDG Level
CO1	Implement basic array operations such as insertion, deletion, searching, and traversal to solve simple computational problems.	TPS 3	PO4, PO5	SDG 09	02
CO2	Implement linked list variants such as singly, doubly, and circular linked lists for dynamic data storage and manipulation.	TPS 3	PO4, PO5	SDG 09	02
CO3	Apply stack and queue operations using arrays to solve linear data structure-based problems.	TPS 3	PO4, PO5	SDG 09	02
CO4	Implement binary tree traversal techniques and binary search tree operations for hierarchical data processing.	TPS 3	PO4, PO5	SDG 09	02
CO5	Demonstrate AVL tree operations with balancing techniques to maintain efficient search performance.	TPS 3	PO4, PO5	SDG 09	02
CO6	Apply disjoint set operations using smart union and path compression techniques for efficient set representation and problem solving.	TPS 3	PO4, PO5	SDG 09	02

Assessment Pattern: Cognitive Domain

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, Post-lab	75	Conducted throughout the semester
Model Test	CO-wise (Cognitive, Psychomotor, Affective)	25	Mid/End semester internal test
Total Internal Marks		**100**	

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	As directed by COE or as per curriculum

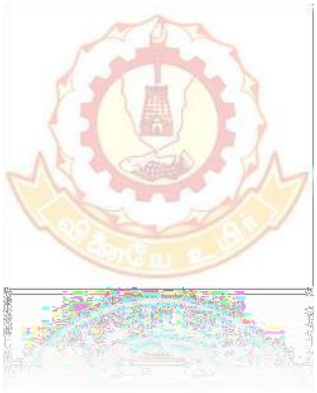
List of Experiments with CO Mapping

Sl. No.	Name of the Experiment	Level of Experiment	Objective(s)	COs Addressed
1.	Implementation of Arrays and basic array operations such as insertion, deletion, searching, and traversal	2	To implement array operations and understand sequential data storage and access.	CO1
2.	Implementation of Singly Linked List and operations such as insertion, deletion, searching, and traversal	2	To implement singly linked list operations for dynamic memory-based data handling.	CO1
3.	Implementation of Doubly Linked List with insertion and deletion operations	2	To create and manipulate doubly linked lists using forward and backward links.	CO2
4.	Implementation of Circular Linked List with insertion, deletion, and traversal operations	2	To implement circular linked list operations and understand circular data organization.	CO2
5.	Implementation of Stack using Arrays with push, pop, peek, and display operations	2	To implement stack operations and understand the LIFO principle.	CO3
6.	Implementation of Queue using Arrays with enqueue and dequeue operations	2	To implement queue operations and understand the FIFO principle.	CO3
7.	Implementation of Tree Traversal Operations such as inorder, preorder, postorder, and level-order traversal	2	To implement different tree traversal techniques for binary trees.	CO4
8.	Implementation of Binary Search Tree with insertion, deletion, and searching operations	2	To implement BST operations for efficient searching and hierarchical data organization.	CO4
9.	Implementation of AVL Tree with insertion, deletion, and balancing operations	2	To implement AVL tree operations and maintain height-balanced tree structures.	CO5

10.	Implementation of Disjoint Sets using smart union and path compression methods	2	To implement efficient disjoint set operations for set representation and union-find problems.	CO6

.Course Designer(s):

1. Dr.S.R. Mathu sudhanan, Assistant Professor, CA srmca@tce.edu



		Category	L	T	P	Credit
26CAHA0	**PROFESSIONAL COMMUNICATION**	EEC	1	0	2	2

Preamble

Professional Communication is an employability-enhancement course designed to equip second-year engineering students with professional and placement-oriented communication skills. The course integrates verbal aptitude, professional writing, group discussion, AI-assisted ethical communication practices, workplace etiquette, emotional intelligence, and interview readiness through experiential learning activities. It enables students to develop industry-ready communication competencies required for professional, multidisciplinary, and global workplace environments.

Prerequisite

- Nil

Course Outcomes

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

No.	Course Outcome	TPS Level	Pos Mapped	CO Weightage	SDG addressed	SDG Level
CO1	Apply verbal aptitude and AI-assisted professional writing skills to create effective workplace communication documents.	TPS3	5	30%	SDG 4	1
CO2	Implement group discussions, technical reporting, résumé building, and digital profile development to build career readiness skills.	TPS3	5	30%	SDG 4	1
CO3	Use listening, prompt engineering, and ethical AI communication practices in professional contexts.	TPS3	7	20%	SDG 4	1
CO4	Demonstrate professional etiquette, emotional intelligence, and interview skills for placement preparedness.	TPS3	8	20%	SDG 4	1

Assessment Pattern

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Verbal Aptitude (CO1-20), GD Practice (CO2-20), Report Writing (CO2-10), Listening Test (CO3-10), Interview Practice (CO4-15)	75	Conducted continuously through classroom and lab activities during the semester
Model Test	Cover Letter Preparation (CO1-10) and Resume Preparation (CO2 – 15)	25	End semester internal test
Total Internal Marks		100	No Common Continuous Assessment Test (CAT) will be conducted
End Semester Practical Examination	Verbal Aptitude (CO1-20), Cover Letter Preparation (CO1-10), GD Practice (CO2-20), Resume Preparation (CO2-10), Listening Test (CO3-20), Interview Practice (CO4-20)	100	Conducted as Practical Examination as per curriculum and COE guidelines

Syllabus

Module 1: Verbal Aptitude & Professional Writing

Verbal aptitude: vocabulary in context, error spotting/correction, sentence completion, and analogy — Business correspondence: professional email and job cover letters.

Module 2: Group Discussion Practice and Report & Resume Preparation

Group Discussion (GD): Dos and Don'ts, techniques and strategies — GD practice: live sessions, and understanding assessment rubrics — Professional Report Writing: Incident/Accident or Maintenance/Technical/Research Reports Format and Structure— ATS (Applicant Tracking System) optimized résumé writing, e-portfolio construction, and profile building in digital platforms.

Module 3: Listening Practice and AI Usage Essentials

Listening to placement and professional skills-related recordings and podcasts — AI Communication Tools: prompt engineering basics, critical and ethical use for drafting, editing, summarising, and tone adjustment; understanding AI limitations, and hallucinations.

Module 4: Placement and Professional Readiness

Professional etiquette — workplace conduct, dress codes, and personal branding — Netiquette: email, virtual meeting, and official social media pages — Emotional intelligence (EQ): self-awareness, self-regulation, empathy, and adaptability — Interview skills: STAR (Situation–Task–Action–Result) method and HR questions — Mock interviews: video recording using AI feedback tools (e.g., Yoodli, Interview Warmup), and peer debrief sessions.

List of Experiments with CO Mapping

Module	Sl. No.	Topics	Theory Periods	Lab Periods	CO
1	1.1	Verbal aptitude: vocabulary, error spotting/correction	1	1	CO1
	1.2	Sentence completion and analogy	1		CO1
	1.3	Professional email	1	1	CO1
	1.4	Cover letter writing practice		1	CO1
		Module 1 Total	**3**	**3**	

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

2	2.1	GD basics, strategies, and assessment rubrics	1		CO2
	2.2	GD practice		2	CO2
	2.3	Professional Report Writing Format and Structure	1		CO2
	2.4	ATS résumé and Digital profile creation & optimization	1	1	CO2
		Module 2 Total	**3**	**3**	
3	3.1	Listening to placement and professional skills-related recordings and podcasts		2	CO3
	3.2	Prompt engineering basics	1		CO3
	3.3	Ethical use of AI tools	1	1	CO3
	3.4	Understanding AI limitations and hallucinations		1	CO3
		Module 3 Total	**2**	**4**	
4	4.1	Professional etiquette		1	CO4
	4.2	Netiquette	1		CO4
	4.3	Emotional intelligence in the workplace	1		CO4
	4.4	STAR method, HR questions and Mock interviews with AI tools (e.g., Yoodli, Interview Warmup) & peer feedback		3	CO4
		Module 4 Total	**2**	**4**	
		TOTAL	**10**	**14**	**24**

Text Book

1. Rizvi, M. A. (2021). Effective technical communication. McGraw Hill.
2. Alred, G. J. (2023). Handbook of technical writing. Bedford/St. Martin's.
3. Raman, M., & Sharma, S. (2024). Technical communication: Principles and practice. Oxford University Press.
4. Elbow, P. (1998). Writing with power: Techniques for mastering the writing process (2nd ed.). Oxford University Press.

Reference Books & Web Resources

1. All India Council for Technical Education. (n.d.). Communication skills in English. AICTE.
2. Lesikar, R. V., & Flatley, M. E. (2011). Business communication: Connecting in a digital world (11th ed.). McGraw-Hill Education.
3. Purdue Online Writing Lab. (n.d.). APA formatting and style guide (7th edition). https://owl.purdue.edu/
4. British Council. (n.d.). LearnEnglish. https://learnenglish.britishcouncil.org/
5. LinkedIn. (n.d.). LinkedIn Learning. https://www.linkedin.com/learning/
6. Grammarly. (n.d.). Grammarly Blog. https://www.grammarly.com/blog/
7. edX. (n.d.). Communication courses. https://www.edx.org/learn/communication
8. Interview Warmup. https://www.interviewwarmup.com/
9. Yoodli. (n.d.). Yoodli AI communication coach. https://yoodli.ai/
10. Guwahati, IIT. (2026). *Shakshat Virtual Lab - Virtual English and Communication*. Iitg.ac.in. https://www.iitg.ac.in/cseweb/vlab/virtual-english/

Course Designers:

1. Dr. M. Sarppa Raje, Assistant Professor, Department of English, mseeng@tce.edu
2. Dr. P. Priyadharshini, Assistant Professor, Department of English, ppieng@tce.edu

		Category	L	T	P	Credit
26CACG0	**DESIGN AND ANALYSIS OF ALGORITHMS**	PCC	3	1	0	4

Preamble

Design and Analysis of Algorithms cultivates the ability to formulate elegant and efficient solutions to computational problems. The course emphasizes algorithmic strategies, complexity evaluation, and performance optimization, thereby strengthening analytical reasoning and structured problem-solving skills.

Prerequisite

- Data Structures

Course Outcomes

After the successful completion of the course, students will be able to:

No.	Course Outcome	TPS Level	Program Outcome	CO Weightage	SDG addressed	SDG Level
CO1	Solve recurrence relations and perform amortized analysis to evaluate algorithm complexity.	TPS3	PO1, PO2,	15%	SDG 04	1
CO2	Apply graph algorithms to solve problems and assess their efficiency.	TPS3	PO1, PO2,	15%	SDG 09	2
CO3	Use divide and conquer and dynamic programming techniques for solving computational problems.	TPS3	PO1, PO2,	20%	SDG 09	2
CO4	Apply solutions for optimization problems using greedy algorithm strategies.	TPS3	PO1, PO2	20%	SDG 09	2
CO5	Implement state space trees using backtracking and branch and bound techniques to solve problems.	TPS3	PO1, PO2,	15%	SDG 09	2
CO6	Apply the significance of NP-complete problems and approximation algorithms in computational problem solving.	TPS3	PO1, PO2	15%	SDG 09	2

Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	30	25			15
CO2	30	25			15
CO3	40	50			20
CO4			40	50	20
CO5			30	25	15
CO6			30	25	15
Total	**100**	**100**	**100**	**100**	**100**

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

Syllabus

Module: 01 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.

Module: 02 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 - FloydWarshall algorithm Network flow: Flow networks - Ford-Fulkerson method – Matching: Maximum bipartite matching.

Module: 03 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.

Module: 04 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.

Module: 05 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.

Learning Resources

Text Book

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, SartajSahni, SanguthevarRajasekaran "Computer Algorithms/C++" Orient Blackswan, 2nd Edition, 2019.

Reference Books & Web resources

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

Course Contents and Lecture Schedule

Module No	Topics	No of Periods
1.	**INTRODUCTION**	
1.1	Time and space complexity- Asymptotic Notations and its properties Best case, Worst case and average case analysis	2
1.2	Amortized Analysis: Aggregate, Accounting and Potential Method with example	2

1.3	Recurrence relation: substitution method, Master Theorem	2
1.4	Searching: linear search, binary search and Interpolation Search	2
1.5	Pattern search: The naïve string matching algorithm	2
1.6	Sorting: Insertion sort – heap sort	2
2	**GRAPH ALGORITHMS**	
2.1	Shortest path: Bellman-Ford algorithm	2
2.2	Dijkstra's algorithm	2
2.3	Floyd-Warshall algorithm	2
2.4	Network flow: Flow networks - Ford-Fulkerson method	2
2.5	Matching: Maximum bipartite matching	2
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	2
3.4	Greedy Technique: Elements of the greedy strategy - Activity-selection problem	2
3.5	Huffman Trees – Knapsack Problem	2
4	**STATE SPACE SEARCH ALGORITHMS**	
4.1	Backtracking: n-Queens problem - Hamiltonian Circuit Problem	2
4.2	Subset Sum Problem – Graph coloring problem	2
4.3	Branch and Bound: Solving 15-Puzzle problem - Assignment problem	2
4.4	Knapsack Problem - Travelling Salesman Problem	2
5	**NP-COMPLETE AND APPROXIMATION ALGORITHMS**	
5.1	Tractable and intractable problems: Polynomial time algorithms – Venn diagram representation	2
5.2	NP algorithms - NP-hardness and NP-completeness	2
5.3	Bin Packing problem - Problem reduction: TSP – 3- CNF problem	2
5.4	Approximation Algorithms: Travelling Salesman Problem	2
	Total Periods	48

Course Designer(s):

1. Dr.S.R. Mathu sudhanan, AP / Computer Applications srmca@tce.edu

		Category	L	T	P	Credit
26CACH0	**OBJECT ORIENTED PROGRAMMING**	PCC	3	1	0	4

Preamble

This syllabus is designed for Computer Applications students to help them understand Object-Oriented Programming and develop computer-based solutions in an effective and systematic manner. The syllabus highlights OOP concepts, functions, polymorphism, inheritance, and I/O operations. It aims to provide adequate depth in these topics so that students can apply the Object-Oriented approach to programming effectively. The modules in the syllabus focus on solving general problems through programming solutions. Collectively, the modules emphasize programming concepts, strategies, techniques, and their application toward the development of efficient programming solutions.

Prerequisite

- NIL

Course Outcomes

After the successful completion of the course, students will be able to:

No.	Course Outcome	TPS Level	Program Outcome(PO)	CO Weightage	SDG addressed	SDG Level
CO1	Apply object-oriented programs using methods and passing arrays, objects, and array of objects to them.	TPS 3	PO1	12	SDG 4	2
CO2	Demonstrate compile-time and run-time polymorphism using object-oriented programs.	TPS 3	PO1,PO3,	17	SDG 9	2
CO3	Implement the relationships between objects using inheritance hierarchies and aggregation.	TPS 3	PO2,PO3	17	SDG 9	2
CO4	Operate object-oriented programs to handle data using Java collections, files, and object serialization.	TPS 3	PO3,PO4	18	SDG 9	2
CO5	Solve object-oriented programs to handle exceptions.	TPS 3	PO3,PO4,PO5	18	SDG 4	2
CO6	Use object-oriented programs to demonstrate event-driven programming, concurrent programming, and network programming.	TPS 3	PO2,PO3,PO4	18	SDG 9	2

Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	30	30			15
CO2	40	40			20
CO3	30	30			15
CO4			35	40	20
CO5			35	40	15
CO6			30	20	15
Total	**100**	**100**	**100**	**100**	**100**

Syllabus

Module1: **Basics of Object-Oriented Programming:** Object-oriented programming and its benefits – Object-oriented programming concepts: Encapsulation, Information Hiding, Abstraction, Generalization/Specialization, and Polymorphism – Object-oriented design: finding the classes and their responsibilities – Object-oriented programming language: Java.

Module 2: **Objects and classes concepts:** 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

Module 3: **Polymorphism and Class Collaborations** Object Oriented Design: Class Collaborations – Aggregation –Composition –Chains of Inheritance – Overriding Super class methods – Abstract Classes and Abstract Methods – Interfaces

Module 4: **I/O Handling and Exception Handling:** Binary files – Random-Access files- Object serialization – Exception handing

Module 5: **Collection Framework in Java:** Introduction to java collections, Overview of java collection framework, commonly used collection classes- Array List, Vector, Hash table, Stack

Module 6: **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, Sepp Vanden Broucke, “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.

Reference Books & web resources

1. John G. Proakis, “Digital Communications”, McGraw Hill International Edition, Fourth Edition, 2001.
2. Paul Deitel and Harvey Deitel, “Java How to Program (Early Objects)”, Pearson, Eleventh Edition ,2017.
3. Herbert Schildt, “Java: The Complete Reference”, McGraw - Hill. Ninth Edition, 2014.
4. E.Balagurusamy, “Programming with Java”, McGraw-Hill, Fifth Edition, 2014.

https://nptel.ac.in/courses/106105191 - Programming In Java, IIT Kharagpur**,** Prof. Debasis Samanta

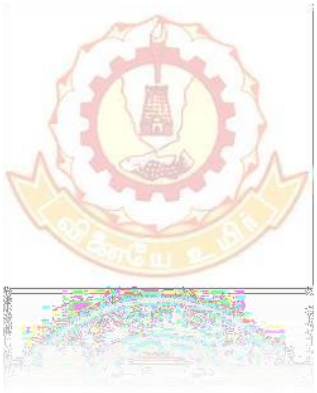
Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours
1	**Basics of Object Oriented Programming**	**8**
1.1	Object oriented programming and its benefits	1
1.2	Object oriented programming concepts: Encapsulation, Information Hiding, Abstraction, Generalization/Specialization and Polymorphism	2
1.3	Object oriented design: finding the Classes and their Responsibilities	2
1.4	Object oriented programming language: Java	2
1.5	Object oriented programming language: C++	1
2	**Classes and Objects**	**8**
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
2.6	Access Specifiers and Static Members	1
2.7	Wrapper Classes and Final Keyword	1
3	**Class Collaborations and Polymorphism**	**8**
3.1	Object oriented Design: Class Collaborations	1
3.2	Aggregation – Composition	1
3.3	Chains of Inheritance	2
3.4	Overriding Super Class Methods	1
3.5	Abstract Classes and Abstract Methods	1
3.6	Interfaces	1
3.7	Runtime Polymorphism and Dynamic Binding	1
4	**I/O Handling and Exception Handling**	**8**
4.1	Binary Files	1
4.2	Random-Access Files	1
4.3	Object Serialization	1
4.4	Exception Handling	2
4.5	Input and Output Streams	1
4.6	Buffered Reader and Buffered Writer	1
4.7	File Handling Applications	1
5	**Collection Framework**	**7**
5.1	Introduction to Java Collections	1
5.2	Overview of Java Collection Framework	1
5.3	Collection Classes – ArrayList, Vector	2

5.4	Collection Classes – Hashtable, Stack	2
5.5	Collection Applications and Iterators	1
6	**Event-Driven Programming, Concurrent Programming and Network Programming**	**9**
6.1	Frameworks	1
6.2	Text-related GUI Components, Other GUI Components	1
6.3	Handling Mouse Events and Button Events	2
6.4	Thread Life Cycle and Methods	1
6.5	Runnable Interface	1
6.6	Thread Synchronization	1
6.7	Basics of Network Programming	2
	Total Hours	**48**

Course Designer(s):

1. Dr.C.Mahadevi Assistant Professor Computer Applications cmids@tce.edu



22CACJ0	OPERATING SYSTEMS	Category	L	T	P	C
		PCC	3	1	0	4

Preamble

This course introduces the fundamentals of Operating Systems, and its components, it helps students understand system operations, scheduling, file systems, and disk allocation methods. The course also covers security concepts and protection mechanisms in modern operating systems.

Prerequisite

- Nil

Course Outcomes

After the successful completion of the course, students will be able to:

No.	Course Outcome	TPS Level	PO	CO Weightage	SDG addressed	SDG Level
CO1	Explain the fundamental concepts of operating systems, kernel and shell, system calls, and virtualization concepts.	TPS- 2	PO1	15%	SDG 4	1
CO2	Apply process scheduling algorithms to solve program in a multi-programmed operating system	TPS- 3	PO2	15%	SDG 9	2
CO3	Apply process synchronisation and deadlock and handling techniques to solve problems in a multi-programmed operating system.	TPS- 3	PO2	20%	SDG 9	2
CO4	Apply memory management techniques to solve problems related to multiprogramming and virtual memory.	TPS- 3	PO4	20%	SDG 9	2
CO5	Solve secondary storage problems using file allocation and free-space management techniques.	TPS- 3	PO3	15%	SDG 9	2
CO6	Apply disk scheduling algorithm and file system access control mechanisms for effective storage organization and protection	TPS- 3	PO7	15%	SDG 12	2

Assessment Pattern

CO	CAT1	Assignment 1	CAT2	Assignment 2	Terminal Exam
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	30	25			15
CO2	30	25			15
CO3	40	50			20
CO4			40	50	20
CO5			30	25	15
CO6			30	25	15
Total	100	100	100	100	100

Syllabus

Module 1: Introduction: Operating System Concepts and Functions - Types of Operating Systems - Operating System Services - OS Components and Architecture - System Calls and OS Operations - Kernel and Shell Commands - Virtual Machines and Virtualization Basics.

Module 2: Process Management: Process – Process states – PCB – Context switching. Threads – benefits of Threads –Multithreading concept. Process Scheduling – Types of Scheduler – Scheduling Criteria –Scheduling Algorithms - Pre-emptive and Non Pre-emptive, Process synchronization - solution to critical Section problem- Semaphores, Deadlock Detection, Deadlock Prevention, Deadlock Avoidance, Deadlock Recovery.

Module 3: Memory Management: Logical and Physical Addresses-Contiguous and Non Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction- Paging – Segmentation-Demand paging- Page Replacement algorithms- Memory Protection and Security Concepts.

Module 4: Storage Management: File System structure, Allocation methods, free space management, Disk Structure, Disk Scheduling, Swap-Space Management- File System Security and Access Control Mechanisms.

Module 5: Case Studies: Linux, Windows, Mobile Operating System, Security Features in Modern Operating Systems.

Text Book

Abraham Silberschatz, Greg Gagne, Peter B. Galvin, “Operating System Concepts”, 10th edition, Wiley, 2018.

Reference Books&Web resources

1. William Stallings, “Operating systems Internal and Design Principles”, 9 th edition, Pearson Education, Global edition, 2017.
2. Andrew Tanenbaum, “Modern Operating Systems”, 4th Edition, Pearson Education, Global edition, 2014.
3. H M Deital, P J Deital and D R Choffnes, “Operating Systems”, Third Edition, Pearson Education, (2004), Reprint 2012.
4. Jason Canon, ”Shell Scripting: How to Automate Command Line Tasks Using Bash Scripting and Shell Programming”, Linux Training Academy, 2015
5. Christine Bresnahan Richard Blum,” Linux Command Line and Shell Scripting Bible”, Third Edition, 2015,Wiley.
6. http://nptel.ac.in/courses/Webcoursecontents/IIScBANG/Operating%20Systems/New_index1.html
7. https://www.shellscript.sh/first.html
8. http://www.freeos.com/guides/lsst/

Course Contents and Lecture Schedule

Module No	Topics	No. of Periods
1 Introduction		
1.1	Operating System Concepts and Functions	01
1.2	Types of Operating Systems	01
1.3	Operating System Services	01
1.4	OS Components and Architecture	01
1.5	System Calls and OS Operations	01
1.6	Kernel and Shell Commands	02
1.7	Virtual Machines and Virtualization Basics	01
2 Process Management		
2.1	Process Concepts and Process States	01
2.2	PCB and Context Switching	01
2.3	Threads and Multithreading Concepts	01
2.4	Process Scheduling and Types of Scheduler	02
2.5	Scheduling Algorithms	02
2.6	Process Synchronization and Critical Section Problem	02
2.7	Semaphores	02
2.8	Deadlock, Prevention, Avoidance	02
2.9	Deadlock Detection, and Recovery	02
3 Memory Management		
3.1	Logical and Physical Addresses	01
3.2	Contiguous and Non-Contiguous Allocation	02
3.3	Fixed and Variable Partitioning	02
3.4	Fragmentation and Compaction	02
3.5	Paging and Segmentation	02
3.6	Demand Paging	01
3.7	Page Replacement Algorithms	02
3.8	Memory Protection and Security Concepts	02
4 Storage Management		
4.1	File System Structure	01
4.2	Allocation Methods and Free Space Management	02
4.3	Disk Structure and Disk Scheduling	02
4.4	Swap-Space Management	01
4.5	File System Security and Access Control Mechanisms	02
5 Case Studies		
5.1	Linux Operating System	01
5.2	Windows and Mobile Operating Systems	01
5.3	Security Features in Modern Operating Systems	01
	Total Periods	48

7. Course Designer(s):

Dr.P.Chitra, Professor , Department of CA, pccse@tce.edu

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

26CACK0	FULL STACK APPLICATION DEVELOPMENT	Category	L	T	P	C
		TCP-P	1	0	6	4

End Semester Exam: Practical

Preamble

This course aims at helping the students to design and develop their very own Web based Applications using open-source development platforms. Is to understand to basics JavaScript and importance of MERN and MERN stack.

Prerequisite

Basic Programming Language (Variables, Data Types, Functions, OOPS Concepts)

Basic understanding in any IDE

Course Outcomes

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

No.	Course Outcome	TPS	PO No	CO Weightage	SDG Addressed	SDG Level
CO1	Explain responsive web pages using HTML, CSS, and JavaScript.	TPS 2	PO1, PO2	10%	SDG 04	01
CO2	Develop frontend interfaces using ReactJS.	TPS 3	PO2, PO4	20%	SDG 09	01
CO3	Apply backend APIs using Node.js and Express.	TPS 3	PO2 PO4	20%	SDG 09	01
CO4	Apply MongoDB with web applications for persistent storage.	TPS 3	PO1, PO3	20%	SDG 09	01
CO5	Develop full-stack apps using MERN/MEAN stack and integrate Client server using AJAX	TPS 3	PO3, PO4	20%	SDG 09	01
CO6	Demonstrate sophisticated Full stack web applications by combining advanced web frameworks and technology.	TPS 3	PO3, PO8	10%	SDG 09	01

Assessment Pattern

CO	Theory		Practical		Terminal Exam (Practical)
	CAT1	CAT2	OCR	Model Test CO-wise	
	% of Marks	% of Marks	% of Marks	% of Marks	% of Marks
CO1	20		20	20	20
CO2	40		20	20	20
CO3	40		20	20	20
CO4		40	20	20	20
CO5		40	20	20	20
CO6		20	20	20	20
Total	**100**	**100**	**100**	**100**	**100**

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

Assessment Pattern: Practical Component

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, Post-lab	75	Conducted throughout the semester
Model Test	CO-wise(Cognitive, Psychomotor, Affective)	25	Rubrics covering all Domains of learning
Total Internal Marks		**100**	
End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	Rubrics covering all Domains of learning

Syllabus

Module:1 HTML, CSS and Client-side Scripting

Introduction to HTML5 - Headings, Linking, Images, Lists, Tables, Forms, New HTML5 Form Input Types, Page Structure Elements - Introduction to CSS - Inline Styles, Embedded Style Sheets, Linking External Style Sheets, Backgrounds, Element Dimensions, Box Model and Text Flow, Positioning, Media Types and Media Queries, Drop Down Menus Selectors, Text Shadows, Box Shadows- JavaScript - Introduction, Operators - Control Statements - Functions, Dialog boxes - Arrays - Objects - Document
Object Model - JavaScript Event Handling, Form Processing - Ajax – JSON

Module:2 React Framework

Environment Set up – JSX – React DOM – Props – State – Controlled Components – Custom Hooks – Reusable React Component – Conditional Rendering – Advanced and Impossible States – Fetching and Re-fetching – Async/Await – Styling in React- React Routers- Server Rendering - Authentication and
Authorization - Deployment – Mongoose and Redux libraries

Module:3 Web Server - Node JS

Getting Started with Node.js Installing Node.js, Working with Node Packages, Creating a Node.js Application - Using Events, Listeners, Timers, and Callbacks in Node.js - Implementing Event emitter, Implementing Callbacks - Accessing the File System from Node.js - Implementing HTTP Services in Node.js - Saving Time with Express - The Request and Response Objects - Form Handling - Sending
Client Data to Server, Form Handling with Express - Cookies and Sessions – Middleware Function Calls

Module:4 MongoDB

NoSQL Database - MongoDB Basics Documents, Collections, Database, Query Language, Installation, The Mongo Shell - MongoDB CRUD Operations - MongoDB Node.js Driver using Mongoose- Reactive Development Paradigm – Advanced Angular Architecture – Setting up Your Development Environment – Creating basic Angular App.

Text Book

1. Paul Deitel, Harvey Deitel and Abbey Deitel, “Internet and World Wide Web: How To Program”, 2018, 5th Edition, Pearson.
2. Vasan Subramanian, Pro MERN “Stack: Full Stack Web App Development with Mongo, Express, React, and Node”, 2019, 2nd Edition, Express.

Reference Books & web resources

1. Mark Lutz. "Learning Python", O'Reilly Media, Fifth Edition, 2013. Online Materials
2. Brad Dayley, Brendan Dayley, Caleb Dayley, “Node.js, MongoDB, and Angular Web Development”, 2017, 2nd Edition, Addison Wesley, Oreilly.

3. Ethan Brown, “Web Development with Node and Express”, 2019, 2nd Edition, O'Reilly Media Inc..
4. https://www.amazon.com/Murachs-Python-Programming-Michael-Urban/dp/1890774979
5. https://www.oreilly.com/library/view/core-python-applications/9780132779371/
6. https://github.com/halterman
7. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/
8. https://nptel.ac.in/courses/106106182/

Course Contents and Lecture Schedule

Module No	Topic	No. of Lecture	CO
1	**HTML, CSS and Client-side Scripting**	**8 hours**	
1.1	Introduction to HTML5 - Headings, Linking, Images, Lists, and Tables	1	CO1
1.2	New HTML5 Form Input Types, Page Structure Elements	1	CO1
1.3	Introduction to CSS - Inline Styles, Embedded Style Sheets, Linking External Style Sheets,	1	CO1
1.4	Backgrounds, Element Dimensions, Box Model and Text Flow, Positioning,	1	CO1
1-5	Media Types and Media Queries, Drop Down Menus Selectors, Text Shadows, Box Shadows-	1	CO1
1.6	JavaScript - Introduction, Operators - Control Statements	1	CO1
1.7	Functions, Dialog boxes - Arrays – Objects.	1	CO1
1.8	Document Object Model - JavaScript Event Handling, Form Processing - Ajax – JSON	1	CO1
2	**React Framework**	**7 hours**	
2.1	Environment Set up – JSX – React DOM – Props – State – Controlled Components – Custom Hooks.	2	CO2
2.2	Reusable React Component – Conditional Rendering	1	CO2
2.3	Advanced and Impossible States – Fetching and Re-fetching	1	CO2
2.4	Async/Await – Styling in React- React Routers	1	CO2
2.5	Server Rendering - Authentication and Authorization	1	CO3
2.6	Deployment – Mongoose and Redux libraries	1	CO3
3	**Web Server - Node JS**	**8 hours**	
3.1	Getting Started with Node.js Installing Node.js, Working with Node Packages, Objects - Form Handling -	2	CO3
3.2	Creating a Node.js Application - Using Events, Listeners, Timers, and Callbacks in Node.js -	1	CO3
3.3	Implementing Event emitter, Implementing Callbacks	1	CO4
3.4	Accessing the File System from Node.js - Implementing HTTP Services in Node.js -	1	CO4

3.5	Saving Time with Express - The Request and Response	1	CO4
3.6	Sending Client Data to Server, Form Handling with Express	1	CO4
3.7	Cookies and Sessions – Middleware Function Calls	1	CO4
4	**MongoDB**	**7 hours**	
4.1	NoSQL Database - MongoDB Basics Documents,	1	CO5
4.2	Collections, Database, Query Language,	1	CO5
4.3	Installation, The Mongo Shell, MongoDB CRUD Operations	1	CO5
4.4	Reactive Development Paradigm	1	CO6
4.5	Advanced Angular Architecture	1	CO6
4.6	Setting up your Development Environment	1	CO6
4.7	Creating basic Angular App.	1	CO6
	Total No. of Hours	**30**	

List of Experiments with CO Mapping

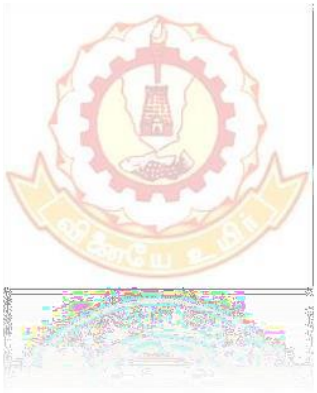
S.No	Topic	No. of Practical Hours	CO
1	Design interactive web pages using HTML5	1	CO1
2	Design responsive web pages using CSS	1	CO1
3	Implement Client-Side Validations and Dynamic Object Manipulations with Events using Javascript	2	CO1
4	Develop component-based User Interface using REACT JS	2	CO2
5	Develop Single Page Applications using Angular framework.	2	CO3
6	Design web applications with dynamic routing using Node JS, and Express Framework	2	CO3
7	Develop a three-tier web application model and data manipulations using Node Js, Express, and Mongo DB.	2	CO4
8	Implement Asynchronous Client, server, and data storage interactions using AJAX and JSON.	2	CO5
9	Develop a three-tier web application File upload and Email.	2	CO6
10	Develop web applications with Session and Cookies.	2	CO6
	Total	**18 hrs**	

Students will design and develop real-time web applications using full stack technologies. They will create databases, develop frontend and backend modules, execute SQL queries, and demonstrate the working application at the end of the lab. The sample applications can be:

- Agriculture Farm Management System
- News Portal Application
- Online Resume Builder
- IoT-Based Smart Home Dashboard
- Alumni Management System

Course Designer(s):

Dr. S.Vijayalakshmi , Professor, Department of Computer Applications, svlcse@tce.edu



26CACL0	OBJECT ORIENTED PROGRAMMING LABORATORY	Category	L	T	P	Credits
		PCC	0	0	4	2

Preamble

This syllabus is intended for the Computer Applications 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

- NIL

Course Outcomes

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

COs	CO Statement	TCE Proficiency Scale	Program Outcome	SDG Addressed	SDG Level
CO1	Apply object-oriented programs using methods and passing arrays, objects, and array of objects to them.	TPS 3	PO1	SDG 4	2
CO2	Demonstrate compile-time and run-time polymorphism using object-oriented programs.	TPS 3	PO1, PO3	SDG 9	2
CO3	Implement the relationships between objects using inheritance hierarchies and aggregation.	TPS 3	PO2, PO3, PO7	SDG 9	2
CO4	Operate object-oriented programs to handle data using Java collections, files, and object serialization.	TPS 3	PO3, PO4	SDG 9	2
CO5	Solve object-oriented programs to handle exceptions.	TPS 3	PO3, PO4, PO5	SDG 4	2
CO6	Use object-oriented programs to demonstrate event-driven programming, concurrent programming, and network programming.	TPS 3	PO2, PO3, PO4	SDG 9	2

Assessment Pattern: Cognitive Domain

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, Post-lab	75	Conducted throughout the semester
Model Test	CO-wise (Cognitive, Psychomotor, Affective)	25	Mid/End semester internal test
Total Internal Marks		**100**	

Passed in BoS Meeting on 23.05.2026

Approved in 71st Academic Council Meeting on 27.06.2026

End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	As directed by COE or as per curriculum

List of Experiments with CO Mapping

Sl. No.	Name of the Experiment	Level of Experiment	Objective(s)	COs Addressed
1.	Apply Object Oriented Program for passing arguments to a method and returning value from a method	02	To implement methods with argument passing and return values using object-oriented programming concepts.	CO1
2.	Implement Object Oriented Program for passing arrays and objects as arguments to method and returning objects from methods	02	To apply arrays, objects, and object-returning methods in Java applications.	CO1
3.	Demonstrate Object Oriented Program for method overloading and constructor overloading	02	To demonstrate compile-time polymorphism using method and constructor overloading techniques.	CO2
4.	Use aggregation and composition using object-oriented program	02	To implement object relationships using aggregation and composition concepts.	CO3
5.	Operate Object Oriented Program to demonstrate inheritance and overriding super class methods	02	To implement inheritance hierarchies and runtime polymorphism using method overriding.	CO2, CO3
6.	Prepare Object Oriented Program to demonstrate abstract base classes and abstract methods	02	To develop modular applications using abstract classes and abstraction concepts.	CO3
7.	Construct Object Oriented Program to demonstrate file handling and object serialization	02	To perform file operations and object serialization using Java I/O handling mechanisms.	CO4
8.	Solve Object Oriented Program for manipulation of data us ing Collections in Java	02	To manage and manipulate dynamic data using Java	CO4

			Collection Framework classes.	
9.	Compute Object Oriented Program to demonstrate exception handling	02	To implement exception handling mechanisms for secure and reliable Java applications.	CO5
10.	Develop event-driven programs using Java's delegation-based event model	02	To design event-driven GUI applications using Java event handling mechanisms.	CO6
11.	Apply concurrent programs using Java threads	02	To implement multithreading and concurrent execution using Java thread concepts.	CO6
12.	Implement network applications using Java sockets	02	To develop client-server applications using Java socket programming concepts.	CO6

Course Designer(s):

1. Dr.C.Mahadevi Assistant Professor Computer Applications cmids@tce.edu

26CACM0	SOFTWARE DEVELOPMENT LABORATORY	Category	L	T	P	Credits
		PCC	0	0	4	2

Preamble

This course enables students to gain practical exposure to software development practices used in the software industry. The course focuses on collaborative development workflows, AI-assisted software engineering, clean coding practices, software testing, and Agile-based project development.

Prerequisite

- NIL

Course Outcomes

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

COs	Course Outcome (CO)	TCE Proficiency Scale	Program Outcomes	SDG Addressed	SDG Level
CO1	Apply collaborative software development workflows using version control systems, repository management, branching strategies, and pull request mechanisms for team-based software projects.	TPS 3	PO4, PO5	SDG 4	02
CO2	Use AI-assisted development tools for software generation, debugging, code optimization, and productivity enhancement in software development environments.	TPS 3	PO4, PO5	SDG 9	02
CO3	Implement clean coding practices, refactoring techniques, and software design patterns to develop maintainable and scalable software applications	TPS 3	PO4, PO5	SDG 9	02
CO4	Implement clean coding practices, refactoring techniques, and software design patterns to develop maintainable and scalable software applications.	TPS 3	PO4, PO5	SDG 9	02
CO5	Apply software verification and validation techniques through unit testing, automated testing, static code analysis, and software quality assessment.	TPS 3	PO4, PO5	SDG 9	02
CO6	Demonstrate industry-oriented software solutions by integrating Agile practices, collaborative development workflows, AI-assisted engineering tools, testing methodologies, and deployment practices	TPS 3	PO4, PO5	SDG 9	02

Assessment Pattern: Cognitive Domain

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, Post-lab	75	Conducted throughout the semester

Model Test	CO-wise (Cognitive, Psychomotor, Affective)	25	Mid/End semester internal test
Total Internal Marks		**100**	
End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	As directed by COE or as per curriculum

List of Experiments with CO Mapping

Sl. No.	Name of the Experiment	Level of Experiment	Objective(s)	COs Addressed
1.	Configure a collaborative development environment using GitHub and perform repository management operations.	2	To understand repository creation, file management, commit operations, and collaborative software development setup using GitHub.	CO1
2.	Create and manage software project workflows using Git branching, pull requests, and merge conflict resolution.	2	To apply branching strategies, pull request mechanisms, and merge conflict resolution techniques in team-based software projects.	CO1
3.	Generate application modules using AI-assisted coding tools and evaluate generated code quality.	2	To utilize AI-assisted coding tools for generating software modules and assessing the readability, correctness, and quality of generated code.	CO2
4.	Perform AI-assisted debugging and code optimization for an existing software application.	2	To use AI-assisted tools for identifying errors, debugging code, and improving the performance and quality of an existing application.	CO2
5.	Create user stories, software requirements, UI prototype using Figma, and UML diagrams using Draw.io for a given case study.	2	To analyze user requirements and prepare user stories, software requirements, UI prototypes, and UML models for a given software problem.	CO3

6.	Refactor an existing application using clean coding standards and reusable design principles.	2	To apply clean coding standards and refactoring techniques to improve code readability, modularity, reusability, and maintainability.	CO3
7.	Implement commonly used software design patterns in a mini application.	2	To implement suitable software design patterns for developing structured, maintainable, and scalable mini applications.	CO4
8.	Develop reusable software components and document APIs using automated tools.	2	To develop reusable components and generate API documentation using automated tools for improving software maintainability and reusability.	CO4
9.	Write test cases and perform automated testing for software modules using Playwright.	2	To design test cases and execute automated tests using Playwright for verifying the functionality of software modules.	CO5
10.	Perform static code analysis and identify software quality issues using code analysis tools.	2	To analyze source code using static code analysis tools and identify quality issues, code smells, and possible improvements.	CO5

Course Designer(s):

1.Dr.S.R. Mathu sudhanan, AP / Computer Applications srmca@tce.edu

26CACN0	DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY	Category	L	T	P	Credits
		PCC	0	0	2	1

Preamble

The objective of this laboratory course is to equip students with the knowledge and skills to analyze algorithmic problems, design or select suitable data structures and algorithms, and implement efficient solutions that satisfy the specified problem constraints.

Prerequisite

- Data structures

Course Outcomes

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

COs	Course Outcome 1 (CO1)	TCE Proficiency Scale	Program Outcomes	SDG Addressed	SDG Level
CO1	Apply appropriate sorting and searching algorithms to solve computational problems and determine their time complexity.	TPS 3	PO4, PO5	SDG 09	02
CO2	Demonstrate the use of graph algorithms in solving real-world problems and evaluating their computational efficiency.	TPS 3	PO4, PO5	SDG 09	02
CO3	Implement divide-and-conquer algorithm design techniques to solve computational problems.	TPS 3	PO4, PO5	SDG 09	02
CO4	Apply dynamic programming techniques to develop efficient solutions for optimization and decision-making problems.	TPS 3	PO4, PO5	SDG 09	02
CO5	Implement greedy algorithm design techniques to solve optimization problems.	TPS 3	PO4, PO5	SDG 09	02
CO6	Apply the state space tree method to solve combinatorial and backtracking problems.	TPS 3	PO4, PO5	SDG 09	02

Assessment Pattern: Cognitive Domain

Type of Assessment	Component	Marks	Remarks
Continuous Assessment	Pre-lab, In-lab, Post-lab	75	Conducted throughout the semester
Model Test	CO-wise (Cognitive, Psychomotor, Affective)	25	Mid/End semester internal test
Total Internal Marks		**100**	
End Semester Examination	CO-wise (Cognitive, Psychomotor, Affective)	100	As directed by COE or as per curriculum

List of Experiments with CO Mapping

Sl. No.	Name of the Experiment	Level of Experiment	Objective(s)	COs Addressed
1.	Implement Linear Search and Binary search. Apply 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.	L3	To implement linear and binary search algorithms, compare their execution time for different input sizes, and analyze their time complexity using performance graphs.	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.	L3	To implement insertion and merge sort algorithms, compare their performance for different input sizes, and analyze their time complexity.	CO1
3.	For a given unweighted / uniform weighted connected graph, implement (a) BFS to find single source shortest path (b) DFS to detect cycle	L2	To implement Breadth First Search (BFS) and Depth First Search (DFS) algorithms for graph traversal, shortest path, and cycle detection.	CO2
4.	From a given vertex in a weighted connected graph, develop a program to find the shortest paths.	L3	To implement a shortest path algorithm for weighted graphs and evaluate its effectiveness in finding optimal paths.	CO2
5.	From a given vertex in a weighted connected graph, implement a program to find the minimum spanning tree.	L3	To implement a minimum spanning tree algorithm and construct an optimal spanning tree for a weighted graph.	CO2
6.	Compute the transitive closure of a given directed graph using Warshall's algorithm	L2	To implement Warshall's algorithm for determining the transitive closure of a directed graph.	CO2
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	L2	To implement the divide-and-conquer technique for efficiently finding the maximum and minimum elements in a dataset.	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.	L3	To implement merge sort and quick sort algorithms, compare their performance, and analyze their time complexity for different input sizes.	CO3
9.	Implement matrix chain multiplication using dynamic programming	L2	To implement dynamic programming for optimizing matrix chain multiplication and minimizing computation cost.	CO4
10.	Implementation of 0/1 knapsack problem using knapsack problem	L2	To implement the 0/1 knapsack algorithm using dynamic programming and obtain an optimal solution.	CO4
11.	Implementation of fractional knapsack problem using greedy algorithm and analyse its performance for various item selection strategies	L3	To implement the fractional knapsack algorithm using a greedy strategy and analyze its performance under different item selection criteria.	CO5
12.	Develop Huffman codes using greedy approach.	L2	To implement Huffman coding using the greedy approach for efficient data compression.	CO5
13.	Implement N Queens / Hamiltonian Circuit Problem / Subset Sum Problem / Graph coloring problem using Backtracking approach.	L3	To implement backtracking algorithms for solving combinatorial optimization and constraint satisfaction problems.	CO6
14.	Implement Assignment problem / Knapsack Problem / Travelling Salesman Problem using Branch and Bound technique	L3	To implement branch-and-bound techniques for solving optimization problems and evaluate their efficiency.	CO6

Course Designer(s):

1. Dr.S.R.Mathu sudhanan Assistant Professor , Computer Applications , srmca@tce.edu