Shree Warana Vibhag Shikshan Mandal's

WARANA UNIVERSITY,
WARANANAGAR

(A State Public University established under Section 3 (6) of MPUA, 2016)

।। विद्या सर्वस्य भूषणम् ।।

Warana University

Established:2025

Structure & Syllabus For
First Year of Master of Computer Application
F.Y. MCA Semester- I and II

UNDER

Faculty of Science & Technology

(As Per National Education Policy – 2020)

With Effect from Academic Year 2025-26 Onwards

Shree Warana Vibhag Shikshan Mandal's

TATYASAHEB KORE INSTITUTE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS), WARANANAGAR, KOLHAPUR

TKIET
Estd. 1983

Lead Institute of

WARANA UNIVERSITY, WARANANAGAR
(A State Public University)

Master of Computer Application (MCA)

Post Graduate (P.G.)

Under

Faculty of Science & Technology

From Academic Year 2025-26

Master of Computer Application

Structure and Syllabus under Autonomy as per NEP Policy 2020

Contents

Preface

The National Education Policy (NEP) 2020 has marked a significant transformation in India's education system, aiming to align higher education with the evolving needs of the 21st century. Warana University is firmly committed to the effective implementation of NEP 2020 in its true spirit, with a strong emphasis on holistic, multidisciplinary, and outcome-based education in accordance with the guidelines of the Government of Maharashtra.

We are pleased to present the First Year MCA Syllabus, thoughtfully designed in alignment with the core philosophy of NEP 2020, effective from the Academic Year 2025–26. This curriculum aims to provide students with a strong foundation in core computing areas such as programming, database management, software development, computer systems, and emerging information technologies, along with skill enhancement and co-curricular activities. The program is structured to develop competent IT professionals with strong technical knowledge, analytical thinking, and the adaptability required to meet dynamic industry and global technological demands.

Special emphasis has been given to experiential learning, where theoretical concepts are reinforced through practical laboratory work, project-based learning, and industry-oriented activities. The syllabus also promotes research, innovation, and entrepreneurship, encouraging students to engage in creative and investigative pursuits from the early stages of their academic journey.

We extend our sincere gratitude to the Chairpersons and members of the various Boards of Studies for their dedicated efforts in designing this forward-looking curriculum.

At Warana University, we believe this syllabus will empower our students with the critical thinking, creativity, and technical proficiency necessary to excel in their professional lives. Warana University looks forward to supporting students in this transformative and enriching academic journey.

Program Outcomes

Program Outcomes (POs) are clear, measurable statements that describe what students are expected to know, understand, and be able to do by the time they complete an academic program. They define the competencies, skills, and professional abilities that graduates should possess at the end of the program. In India, POs for Engineering Programs are formally prescribed and monitored by the National Board of Accreditation (NBA). NBA has defined the following three POs for a graduate of PG Engineering Program:

PO1: An ability to independently carry out research /investigation and development work to solve practical problems.

PO2: An ability to write and present a substantial technical Report/document.

PO3: Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program

Duration

- The full time M. Tech Program is a **2 years post graduate program**.
- The program is divided into **4 semesters**.

Eligibility

1. **Nationality:**
   The candidate must be an Indian National.
2. **Educational Qualification:**
   The candidate must have passed any recognized Bachelor's Degree of minimum three years duration (e.g., B.E., B.Tech., B.Sc., B.Com., B.A., B.Voc., BCA, etc.) awarded by a recognized University.
   - The candidate should have studied Mathematics at either 10+2 (HSC) level, or Graduation level.
   - The candidate must have secured at least 50% marks in aggregate in the qualifying examination.
   - For candidates belonging to Reserved Categories, Economically Weaker Section (EWS), and Persons with Disability (PwD) category belonging to Maharashtra State, the minimum required marks shall be 45% in aggregate.
3. **Entrance Examination:**
   The candidate must have obtained a non-zero positive score in the MAH-MCA-CET.

Medium of Instruction

- The medium of instruction, examinations, assignments, and project reports is English.

Abbreviations

Acronym	Full Form
ISE	In-Semester Examination
ISE -I	In-Semester Examination I
ISE-II	In-Semester Examination II
ESE	End Semester Examination
ISA	In Semester Assessment
POE	Practical Oral Examination
O	Oral Examination
L	Lecture
T	Tutorial
P	Practical
CH	Contact Hours
C	Credit
PCC	Program Core Courses
PE	Program Elective
OE	Open Elective
LC	Lab Course
SW	Seminar Work

Examination & Evaluation Pattern

Evaluation tools used for the evaluation of a student for each course is as follows:

For Theory Courses	In-Semester Examination (ISE) And End Semester Examination (ESE)
For Lab / Tutorial Courses	In-Semester Assessment (ISA) And / Or Practical and Oral Examination (POE)

Refer course structure for specific evaluation tools used for each course.

In-Semester Examination (ISE)

The In-Semester Examination (ISE) will be conducted at the departmental level. There will be two tests in each semester for every theory course: ISE-I and ISE-II.

- Each test will be of 40 marks.
- The duration of each test will be 1 hour and 30 minutes.

The total ISE marks will be calculated as the average of ISE-I and ISE-II. These rules may be modified from time to time as per the guidelines of the concerned regulatory authorities.

- ISE-I will cover Unit I and Unit II.
- ISE-II will cover Unit III and Unit IV.

  - **Minimum Passing Criteria**

    Students must score a minimum of 40% marks in the ISE. If a student fails to secure the minimum required marks, he/she must appear for a Make-up Examination.

    The Make-up Examination will be conducted in the same semester for:

    - Students who fail to secure minimum passing marks.
    - Students who were absent due to valid reasons such as medical issues, natural calamities, or participation in NSS, NCC, or similar activities (subject to verification of absence and recommendation from the Head of Department).

- **Special Provision**

  If a failed student appears for three tests (including the Make-up test) and scores more than 16 marks when calculating the average of the best two out of the three tests, the student will be awarded the minimum passing marks of 16 only.

  For students absent with valid reasons:

  - If absent in one test, the average of the attempted test and the Make-up test will be considered.
  - If absent in two tests, the decision will be taken after reviewing the reasons and based on the recommendation of the Head of Department.

End Semester Examination (ESE):

The End Semester Examination (ESE) will be conducted for 60 marks and will be based on the entire syllabus. The duration of each examination will be 2 hours.

Weightage of Units

The weightage of units in the ESE question paper will be as follows:

a) Units that are not covered in ISE-I or ISE-II will carry 30% weightage each.

b) Units that are covered in ISE-I and ISE-II will carry 10% weightage each.

Backlog Examination

Students who fail in the End Semester Examination (ESE) of either the odd or even semester within an academic year will be allowed to appear for the Backlog Examination, which will be conducted along with the regular ESE of the respective semester.

- **Re-Examination of ESE**

  A Re-Examination (Make-up Examination) for all courses (UG and PG), including both theory and laboratory courses, will be conducted once a year before the commencement of the odd semester of the next academic year.

  - A one-grade penalty will be applied to students appearing for the Make-up/Re-Examination.
  - However, no grade penalty will be applied if a student secures a ‘P’ grade in the Make-up/Re-Examination.
  - Grace marks will not be awarded for the Make-up/Re-Examination.
  - Exception: Grace marks may be considered if the student is appearing for the ESE for the first time.

- **Eligibility Criteria for ESE**

  To be eligible for the End Semester Examination (ESE), a student must:

  - Secure at least 40% marks in ISE and ISA of the concerned course.
  - Fulfil the attendance requirements as per the norms of Warana University, Warananagar.

  If a student does not meet these requirements, he/she will not be eligible to appear for the ESE.

Nature of Question Paper for ESE

Q. No.			Marks	BL	CO
1	**Attempt the following.**		**24**		
	a	Unit -1		II	1
	b	Unit -2		III	2
	c	Unit -3		IV	1
	d	Unit -4		I	1
2	**Attempt any Three of the following.**		**18**		
	a	Unit -5		VI	2
	b	Unit -5		II	3
	c	Unit -5		IV	3
	d	Unit -5		IV	3
3	**Attempt any Three of the following.**		**18**		
	a	Unit -6		IV	4
	b	Unit -6		III	4
	c	Unit -6		III	4
	d	Unit -6		III	4

In Semester Assessment (ISA):

ISA for laboratory courses will be conducted as a continuous assessment throughout the semester. The assessment will be based on the following:

1. Performance in laboratory work.
2. Submission of experiments in the form of a properly maintained journal or report.
3. Timely completion of assigned experiments.

4. Attendance in laboratory sessions.
5. Understanding of the experiments conducted, evaluated through methods such as quizzes, oral examinations, case studies, field work, surveys, open-book tests, model preparation, programming, projects, or any other criteria specified by the course teacher.

Practical Oral Examination (POE):

POE for laboratory courses will be conducted immediately after the end of the semester. The duration of the practical examination will be as specified in the curriculum structure. The POE will be conducted jointly by an Internal Examiner and an External Examiner.

The examination may be conducted in any one of the following ways:

1. **Oral Examination Only**

   Both the Internal and External Examiners will ask questions based on the practical content of the course to assess the student's practical knowledge.

2. **Practical Examination Only**

   Students will be required to perform a given experiment, complete a workshop task, prepare a drawing, or develop a computer program, as applicable. In this case, the student's performance will be evaluated by the External Examiner only.

3. **Practical and Oral Examination**

   Students will first perform a given practical task. This will be followed by an oral examination (viva voce) based on the practical content of the course. The student's performance will be evaluated jointly by both the Internal and External Examiners.

Grading System

The University follows a **10-Point Grading System** to evaluate student performance.

- **Conversion of Marks into Grades**
  In every semester, the marks you get in each subject (out of 100) are converted into **grade points** as per the table below. You need at least **40% marks** to pass a subject.

Marks Obtained (Out of 100)	Grade Point	Letter Grade	Meaning
Absent	0	AB	Absent
0 – 39	0	F	Fail
40 – 44	4	P	Pass
45 – 49	5	C	Average
50 – 59	6	B	Above Average
60 – 69	7	B+	Good
70 – 79	8	A	Very Good
80 – 89	9	A+	Excellent
90 – 100	10	O	Outstanding

Note:

1. If decimal marks are 0.5 or more, they will be rounded off to the next higher number. (Example: 59.5 will become 60)
2. For courses of 50 marks or 200 marks, marks will be converted proportionally to 100 marks before assigning grade points.

- **Calculation of Semester Grade Point Average (SGPA)**
  SGPA is calculated at the end of each semester**.** It shows your average performance in one semester.

$$SGPA = \frac{\sum(Credit \times Grade\ Point)\ for\ each\ course\ of\ a\ Semester}{\sum(Credits)\ for\ a\ Semester}$$

- **Calculation of Cumulative Grade Point Average (CGPA)**
  CGPA is calculated after completing multiple semesters. CGPA reflects the overall academic performance of the student in the program.

$$CGPA = \frac{\sum(Total\ Credits\ of\ a\ Semester \times SGPA\ of\ Respective\ Sememster)\ of\ all\ semesters}{\sum(Course\ Credits)\ of\ all\ Semesters}$$

Note:

1. The SGPA & CGPA shall be rounded off to 2 decimal points.

First Year Master of Computer Application

Curriculum Structure & Evaluation Scheme for Semester-I

Course Category	Course Code	Course Title	Teaching and Credit Scheme					Examination and Evaluation Scheme			
			L	T	P	C	CH	Component	Marks	Min. for Passing	
PCC	2501PMCAPCC101	Data Structure and Algorithms	4	-	-	4	4	ESE	60	24	40
								ISE	40	16	
	2501PMCAPCC102	Database Management Systems	4	-	-	4	4	ESE	60	24	40
								ISE	40	16	
	2501PMCAPCC103	Discrete Mathematics	3	-	-	3	3	ESE	60	24	40
								ISE	40	16	
	2501PMCAPCC103T	Discrete Mathematics - Tutorial	-	1	-	1	1	ISA	25	10	10
PE	2501PMCAPE104X	Program Elective - I	4	-	-	4	4	ESE	60	24	40
								ISE	40	16	
OE	2501PMCAOE105X	Open Elective – I	4	-	-	4	4	ESE	60	24	40
								ISE	40	16	
LC	2501PMCAPCC101P	Data Structure and Algorithms – Lab	-	-	4	2	4	ISA	25	10	30
								POE	50	20	
	2501PMCAPCC102P	Database Management Systems - Lab	-	-	4	2	4	ISA	25	10	30
								POE	50	20	
	2501PMCAOJT106P	Field Project – Lab	-	-	4	2	4	ISA	25	10	30
								O	50	20	
SW	2501PMCASW107T	Seminar	-	-	4	2	4	ISA	50	20	20
Total			**19**	**01**	**16**	**28**	**36**		**800**	**320**	**320**

Note : 'X' indicates the sequence number of PE/OE offered by the respective department.

First Year Master of Computer Application

List of Program Electives & Open Electives for Semester-I

First Year MCA (Semester – I)			
Course Code	**Program Elective - I**	**Course Code**	**Open Elective - I**
2501PMCAPE1041	Unix Shell Programming	2501PMCAOE1051	Cyber Security
2501PMCAPE1042	Computer Architecture	2501PMCAOE1052	Digital Marketing
2501PMCAPE1043	Cryptography & Network Security	2501PMCAOE1053	Design Thinking & Problem Solving

First Year Master of Computer Application

Curriculum Structure & Evaluation Scheme for Semester-II

Course Category	Course Code	Course Title	Teaching and Credit Scheme					Examination and Evaluation Scheme			
			L	T	P	C	CH	Component	Marks	Min. for Passing	
PCC	2501PMCA PCC201	Network Technology	3	-	-	3	3	ESE	60	24	40
								ISE	40	16	
	2501PMCA PCC201T	Network Technology - Tutorial	-	1	-	1	1	ISA	25	10	10
	2501PMCA PCC202	Software Project Management	4	-	-	4	4	ESE	60	24	40
								ISE	40	16	
	2501PMCA PCC203	Feature Engineering	3	-	-	3	3	ESE	60	24	40
								ISE	40	16	
	2501PMCA PCC204	Java Programming	3	-	-	3	3	ESE	60	24	40
								ISE	40	16	
PE	2501PMCA PE205X	Program Elective - II	4	-	-	4	4	ESE	60	24	40
								ISE	40	16	
LC	2501PMCA PCC204P	Java Programming - Lab	-	-	4	2	4	ISA	25	10	30
								POE	50	20	
	2501PMCA PCC206P	Python Programming - Lab	-	-	4	2	4	POE	50	20	20
SW	2501PMCA OJT207O	Mini Project	-	-	8	6	8	ISA	50	10	30
								POE	100	20	
Total			**17**	**01**	**16**	**28**	**34**		**800**	**320**	**320**

Note : 'X' indicates the sequence number of PE/OE offered by the respective department.

First Year Master of Computer Application

List of Program Electives for Semester-II

First Year MCA (Semester – II)	
Course Code	**Course Names**
2501PMCAPE2051	Data Mining and Data Warehousing
2501PMCAPE2052	Big Data Analytics
2501PMCAPE2053	Advanced Operating Systems

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC101 Course Name: Data Structure and Algorithms

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	04	ISE: 40 Marks ESE: 60 Marks

Prerequisites, if any:

Basic programming in C (variables, loops, arrays, functions), simple pointers (if C), basic math (log, Big-O), and logical problem-solving skills.

Course Objectives: The objective of the course is to

1) Understand fundamental techniques for evaluating algorithm efficiency
2) Explore key methods for data organization, including searching and sorting.
3) Gain knowledge of essential data structures required for problem solving.
4) Examine principles for managing hierarchical and networked data structures.
5) Develop skills in designing algorithms for complex problems.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Describe the fundamentals of algorithm performance analysis
CO2	Execute various searching and sorting algorithms to solve computational problems
CO3	Implement algorithms using linear data structures for efficient data handling
CO4	Assess the use of tree-based structures in hierarchical data organization

Course Description:

This course covers data structures and algorithm design & analysis. It focuses on performance evaluation, searching & sorting methods, and linear & non-linear data structures. Students will learn to assess algorithm efficiency and apply suitable strategies to solve complex problems, equipping them to design effective algorithms for real-world applications.

Course Content

Unit-1	**Performance Analysis of Algorithms**	**8 Hrs**

Definition of Algorithms; Basic Properties of Algorithms, Expressing Algorithms (Pseudo code and Flowcharts) Introduction to Algorithm Design Techniques Performance Analysis of Algorithms (Time and Space Complexity, Big O, Θ, Ω Notations) Introduction to Recursion (Basics of Recursive Functions; Examples of Simple Recursive Algorithms)

Unit-2	**Searching and Sorting**	**8 Hrs**

Introduction to Searching (Linear Search, Binary Search) Sorting Algorithms (Insertion Sort, Selection Sort, Merge Sort, Quick Sort) Analysis of Searching and Sorting Algorithms (Best, Average, Worst Case)

Unit-3	**Linear Data Structures**	**8 Hrs**

Linked Lists: Definition, Traversal, Searching, Insertion/Deletion (Singly, Doubly, Circular) Stack: Definition, Array & Linked Representation, Applications (Arithmetic Expressions, Polish Notation) Queue: Definition, Array & Linked Representation, Circular Queue, Priority Queue

Unit-4	**Non-Linear Data Structures - I**	**8 Hrs**

Trees: General Tree, Binary Tree, Binary Search Tree (BST) Operations on BSTs: insertion, deletion, traversal, searching Balanced Trees: AVL Trees,

Unit-5	**Non-Linear Data Structures - II**	**8 Hrs**

Graphs: Representation of Graphs (Adjacency List, Adjacency Matrix)
Graph Traversal Methods: Breadth-First Search (BFS), Depth-First Search (DFS)
Topological Sort Hashing: Hash Functions, Collision Handling, Static and Dynamic Hashing

Unit-6	Algorithm Design Approaches	8 Hrs
Greedy Algorithms (Introduction, e.g., Knapsack Problem, Minimum Spanning Tree) Backtracking (Introduction, e.g., N-Queens Problem)		

Learning Resources:

Text Books

1) Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education.
2) Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, *Fundamentals of Data Structures in C*, Universities Press.
3) Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, *Introduction to Algorithms* (3rd or 4th Edition), PHI / MIT Press.

Reference Books

1) Reema Thareja, *Data Structures Using C*, Oxford University Press.
2) Robert Sedgewick, *Algorithms in C* (Parts 1–5), Addison Wesley.

e-Books

1) https://docs.oracle.com/cd/E19059-01/stud.10/819-0493/OtherTools.html
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc22_cs71/preview[CMI]
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC102 Course Name: Database Management System

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours / Week	04	ISE: 40 Marks ESE: 60 Marks

Prerequisites, if any:

Some experience with spreadsheets can help visualize the data you'll be working with.

Course Objectives: The objective of the course is to

1) To understand the basic concepts of database management systems.
2) To understand the relational database design principles.
3) To Master the basics of SQL and construct queries using SQL.
4) To become Familiar with the basic issues of transaction processing and concurrency control.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Learn Basics of DBMS and RDBMS
CO2	Learn and practice data modelling using the entity-relationship and developing database designs.
CO3	Examine the use of Structured Query Language (SQL) and learn SQL syntax.
CO4	Utilize normalization techniques to normalize the database

Course Description:

This Course is designed to understand the Database concepts, RDBMS, SQL and Transaction Processing.

Course Content

Unit-1	**Basics of DBMS**	**8 Hrs**

Database Concept, Characteristics and architecture of DBMS, Database users, 3-tier architecture of DBMS-its advantages over 2-tier, Introduction of Parallel, Distributed Databases, Mobile databases and Cloud databases.

Unit-2	**Introduction to RDBMS**	**8 Hrs**

RDBMS, Entity introduction, characteristics, Comparison between DBMS and RDBMS, Generalization and Aggregation.

Unit-3	**SQL - I**	**8 Hrs**

Introduction to SQL, Features of SQL, Basic data types, SQL statements/commands, Set operations in SQL, order by and group by clause, between, in, like, create index.

Unit-4	**Data Constraints and Normalization**	**8 Hrs**

Primary key, Foreign key, Unique key, Null, Not Null, Default key.
Normalization, Functional dependency, types of normalization (1NF, 2NF, 3NF, BCNF).

Unit-5	**SQL - II**	**8 Hrs**

View and join command Nested queries, GRANT and REVOKE, Commit, Rollback, Save point. Join concept, Join Types, View.

Unit-6	**Concurrency Control and Transaction Management**	**8 Hrs**

Transaction processing and Concurrency, Concept of transaction processing, ACID properties, locking techniques, Timestamp based protocols, Granularity of data items, Deadlocks. Database Recovery & Backup. Programming (Introduction, All Pair Shortest Path).

Learning Resources:

Text Books

1) Introduction to database systems C. J. Date Pearsons Education 8th
2) Database system concept Korth, Silberschatz and Sudarshan MGH 5th

Reference Books

1) Fundamentals of Database Systems Elmasri Nava the Pearson Education5th
2) Object Oriented analysis& Design, Andrew High, Tata McGraw-Hill Publication
3) ORACLE PL/SQL Programming Scott Ulman TMH 9th
4) SQL, PL/SQL the programming language of Oracle Ivan Bayross BPB 4th
5) Advance Database Management System hakra bharati/DasguptaWileyDreamtech2011
6) Database Management systems Ramak rishnan & Gehrke, McGraw-Hill,3rd Ed.

e-Books

1) https://media.geeksforgeeks.org/courses/syllabus/5723a07ce6db1b2e7fe33b5db5f0d606.pdf
2) https://ncert.nic.in/vocational/pdf/jhde103.pdf

MOOC / NPTEL/YouTube Links

1) https://www.youtube.com/watch?v=S7YYYR61YdM
2) https://www.youtube.com/watch?v=YRnjGeQbsHQ

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC103 Course Name: Discrete Mathematics

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 03 Hours/Week	**03**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic Mathematics

Course Objectives: The objective of the course is to

1) Understand basic logic and set theory concepts.
2) Learn relations, functions, and their properties
3) Explore counting techniques and combinatory.
4) Study graphs and trees and their applications
5) Understand algebraic structures like groups and rings

Course Outcomes: After successful completion of the course, student will be able to

CO1	Describe logical reasoning and mathematical proofs.
CO2	Manipulate sets, relations, and functions efficiently.
CO3	Solve problems using combinatorial techniques.
CO4	Examine graphs and trees relevant to computing.

Course Description:

This course introduces the fundamental concepts of Discrete Mathematics essential for computer science and information technology.

Course Content

Unit-1	**Logic and Proof Techniques**	**8 Hrs**

Propositions, logical connectives, truth tables, Logical equivalences and laws, Predicates and quantifiers, Methods of proof: direct, indirect, contradiction, and mathematical induction

Unit-2	**Set Theory**	**8 Hrs**

Sets, subsets, power sets, Set operations and identities, Venn diagrams, Cartesian products

Unit-3	**Relations and Functions**	**8 Hrs**

Relations: definition, types (reflexive, symmetric, transitive, equivalence relations), Partial orderings, Functions: types (one-one, onto, bijection), composition, inverse functions

Unit-4	**Counting and Combinatory**	**8 Hrs**

Basic counting principles, Permutations and combinations, Binomial theorem, Pigeonhole principle

Unit-5	**Graph Theory**	**8 Hrs**

Graphs: types, terminology (degree, path, cycle, connectedness), Representations of graphs (adjacency matrix/list), Special graphs: bipartite, complete, regular, Trees: properties, spanning trees, minimum spanning tree basics.

Unit-6	**Algebraic Structures**	**8 Hrs**

Algebraic systems and examples, Groups: definition, examples, properties, cyclic groups, Rings and fields (basic introduction only)

Learning Resources:

Text Books

1) Discrete Mathematics and Its Applications by Kenneth H. Rosen
2) Discrete Mathematical Structures with Applications to Computer Science by J.PTremblay and R. Manohar

Reference Books

1) Elements of Discrete Mathematics by C.L. Liu, McGraw Hill
2) Discrete Mathematics by Richard Johnson baugh, Pearson
3) Mathematics for Computer Science by Eric Lehman, F. Thomson Leighton, and Albert R. Meyer (MIT Open Course Ware - Free)

e-Books

1) https://docs.oracle.com/cd/E19059-01/stud.10/819-0493/OtherTools.html
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc22_cs71/preview[CMI]
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC103T Course Name: Discrete Mathematics - Tutorial

Teaching Scheme	Credit	Evaluation Scheme
Tutorial: 01 Hours/Week	**01**	**ISA: 25 Marks** **POE: -**

Prerequisites, if any:

Course Objectives: The objective of the course is to

1) To develop a strong foundation in discrete mathematical structures such as sets, relations, functions, logic, and Boolean algebra used in computer science.
2) To enable students to analyze and solve computational problems using combinatorics, recurrence relations, and mathematical reasoning techniques.
3) To help students understand and apply graph theory concepts including graph representations, traversals, and path problems in real-world computing scenarios.
4) To enhance logical thinking and problem-modeling skills required for algorithm design, data structures, and advanced computer science subjects.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Apply set theory, logic, and Boolean algebra concepts to model and solve computational problems.
CO2	Analyze relations, functions, permutations, and combinations to solve discrete mathematical problems effectively.
CO3	Solve problems involving recurrence relations, recursion, and graph theory using appropriate mathematical techniques.
CO4	Demonstrate logical reasoning and problem-solving skills in applying discrete mathematics to computer science applications.

Course Description:

This tutorial on Discrete Mathematics introduces fundamental concepts such as sets, relations, functions, logic, combinatorics, and graph theory. It equips students with the mathematical tools and reasoning skills needed for computer science and problem-solving applications.

Course Content

Sr. No.	Topic of Practical / Experiment / Tutorial	Assigned Hours
1	Solving problems on set operations, Venn diagrams, and logical equivalences	2
2	Determining properties of relations (reflexive, symmetric, transitive) and mapping exercises for functions.	2
3	Problems on permutations, combinations, and the application of the Pigeonhole Principle.	2
4	Solving linear recurrence relations and writing recursive algorithms for sequences.	2
5	Exercises on graph representation, traversals (BFS, DFS), and finding Eulerian and Hamiltonian paths.	2
6	Simplifying Boolean expressions, Karnaugh maps, and designing basic logic circuits	2

Learning Resources:

Text Books

1. Discrete Mathematics and Its Applications by Kenneth H. Rosen
2. Discrete Mathematical Structures with Applications to Computer Science by J.PTremblay and R. Manohar

Reference Books

1) Elements of Discrete Mathematics by C.L. Liu, McGraw Hill
2) Discrete Mathematics by Richard Johnson baugh, Pearson
3) Mathematics for Computer Science by Eric Lehman, F. Thomson Leighton, and Albert R. Meyer (MIT Open Course Ware - Free)

e-Books

1) https://docs.oracle.com/cd/E19059-01/stud.10/819-0493/OtherTools.html
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc22_cs71/preview[CMI]
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPE1041 Course Name: Unix Shell Programming

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	04	ISE: 40 Marks ESE: 60 Marks

Prerequisites, if any:

Basic knowledge of operating systems and familiarity with command-line interfaces.

Course Objectives: The objective of the course is to

1) Understand the basics of UNIX operating system and its commands.
2) Master file system structure and file manipulation commands.
3) Learn shell programming fundamentals and scripting techniques.
4) Develop skills to write and debug shell scripts for automation.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Operate the UNIX system efficiently using commands and utilities.
CO2	Manage files, directories, and permissions in UNIX.
CO3	Write shell scripts using various programming constructs.
CO4	Automate system tasks and troubleshoot scripts.

Course Description:

This course teaches Unix/Linux commands and shell scripting to automate tasks and manage system operations efficiently.

Course Content

Unit-1	**Introduction to UNIX**	**6 Hrs**

History and features of UNIX, UNIX architecture and file system overview, Logging in, shells overview (Bourne, Korn, Bash), Basic UNIX commands (pwd, ls, cd, mkdir, rmdir, cat, more, file)

Unit-2	**File and Directory Management**	**8 Hrs**

File types and permissions, Commands: cp, mv, rm, chmod, chown, chgrp, Directory commands and navigation, Links: hard and symbolic, File system hierarchy.

Unit-3	**Process Management and Editors**	**6 Hrs**

Process concepts and states, Commands: ps, top, kill, bg, fg, jobs, File editors: vi editor basics, Searching files: grep, find, locate.

Unit-4	**Filters, Pipes, and Redirection**	**8 Hrs**

Standard input, output, and error, Redirection operators: >, >>, <, 2>, Filters: sort, uniq, head, tail, cut, paste, wc, Pipes and their usage.

Unit-5	**Shell Programming Basics**	**8 Hrs**

Introduction to shell scripting, Writing simple shell scripts

Variables, constants, and user input, Conditional statements: if, if-else, case, Looping constructs: for, while, until, Functions in shell scripts, Debugging scripts.

Unit-6	**Advanced Shell Scripting**	**9 Hrs**

Working with files and directories in scripts, Command line arguments and parameters, String operations and pattern matching, Handling signals and traps, Automating system tasks using shell scripts

Learning Resources:

Text Books

1) Stephen G. Kochan, Unix Shell Programming, 3rd Edition, Addison-Wesley, 2014.

Reference Books

1) Arnold Robbins and Nelson H. F. Beebe**,** Classic Shell Scripting: Hidden Commands that Unlock the Power of Unix, O'Reilly Media, 2005.
2) Christopher Negus, Linux Bible, 10th Edition, Wiley, 2020.

e-Books

1) https://sunilwanjarisvpcet.wordpress.com/wp-content/uploads/2021/12/unix-shell-programming.pdf
2) https://skkatariaandsons.com/view_book.aspx?productid=7873

MOOC / NPTEL/YouTube Links

3) https://www.youtube.com/watch?v=48zApVSX97s
4) https://www.youtube.com/watch?v=8c1BL5b47kg

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPE1042 Course Name: Computer Architecture

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic knowledge of computer organization, digital logic design, and programming fundamentals.

Course Objectives: The objective of the course is to

1) To understand the fundamental concepts and components of computer systems.
2) To analyze the design and organization of processors, memory, and input/output systems.
3) To study instruction sets, pipelining, and performance optimization techniques
4) To evaluate modern computer architectures and emerging trends in computing technology.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explain the basic structure and functioning of computer systems and their components.
CO2	Examine processor design, memory hierarchy, and input/output mechanisms.
CO3	Execute instruction set architecture and pipelining concepts to optimize performance.
CO4	Justify modern and emerging computer architectures for efficiency and scalability.

Course Description:

This course covers the design and organization of modern computer systems, including processors, memory, and I/O mechanisms.

Course Content

Unit-1	**Introduction to Computer Architecture**	**7 Hrs**

This unit introduces the basic concepts of computer architecture, including the structure and function of computer systems. It covers the history and evolution of computers, types of computers, and the role of hardware and software in system performance.

Unit-2	**Instruction Set Architecture (ISA)**	**8 Hrs**

Students learn about instruction sets, addressing modes, and instruction formats. The unit emphasizes how ISA defines the interface between hardware and software and its impact on processor design and efficiency.

Unit-3	**Processor Design and Organization**	**8 Hrs**

This unit focuses on processor components, including the arithmetic logic unit (ALU), control unit, and registers. Topics include CPU design, single-cycle and multi-cycle processors, and techniques for improving processing speed.

Unit-4	**Pipelining and Performance Optimization**	**8 Hrs**

Students study pipelining concepts, hazards, and techniques to enhance processor throughput. The unit also covers performance metrics, benchmarking, and strategies to minimize delays and maximize efficiency.

Unit-5	**Memory Hierarchy and Storage Systems**	**8 Hrs**

This unit explores memory organization, including cache, main memory, and secondary storage. It addresses memory access techniques, virtual memory, and methods to optimize data retrieval and storage performance.

Unit-6	Input/output Systems and Emerging Architectures	7 Hrs

The final unit examines input/output devices, interfaces, and communication with processors. It also discusses recent trends in computer architecture, such as multicore processors, parallel computing, and energy-efficient designs.

Learning Resources:

Text Books

1) David A. Patterson and John L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, 6th Edition, Morgan Kaufmann, 2020.

Reference Books

1) Stallings, W., *Computer Organization and Architecture: Designing for Performance*, 11th Edition, Pearson, 2020.
2) Mano, M. M., & Kime, C. R., *Logic and Computer Design Fundamentals*, 5th Edition, Pearson, 2017.

e-Books

1) https://sunilwanjarisvpcet.wordpress.com/wp-content/uploads/2021/12/unix-shell-programming.pdf
2) https://skkatariaandsons.com/view_book.aspx?productid=7873

MOOC / NPTEL/YouTube Links

1. https://onlinecourses.nptel.ac.in/noc25_cs83/preview
2. https://onlinecourses.nptel.ac.in/noc23_cs113/preview

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPE1043 Course Name: Cryptography & Network Security

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours / Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic Understanding of Computer Networks and Data Communication, Familiarity with Mathematics, especially Number Theory and Algebra.

Course Objectives: The objective of the course is to

1) Understand foundational concepts of cryptography and common encryption techniques.
2) Identify various types of network attacks and the principles of network security.
3) Analyze cryptographic algorithms and their applications for secure communication.
4) Learn methods for securing data integrity, confidentiality, and authentication in networks.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explain fundamental concepts of cryptography and network security.
CO2	Identify and describe different types of encryption techniques and their applications.
CO3	Demonstrate understanding of methods for ensuring data confidentiality, integrity, and authentication in network communications.
CO4	Examine common network attacks and explain strategies to protect against them.

Course Description:

This course introduces fundamental concepts of cryptography and network security, focusing on encryption, network vulnerabilities, and protective measures. Students will learn security protocols and methods to ensure data confidentiality, integrity, and authentication, and will be able to examine and respond to network security challenges.

Course Content

Unit-1	**Introduction to Cryptography and Mathematical Foundations**	**6 Hrs**

Fundamentals of Cryptography: definition, significance, applications, Cryptographic Paradigms: symmetric vs asymmetric, Mathematical Foundations: number theory, modular arithmetic, basic probability theory.

Unit-2	**Classical Cryptography and Symmetric Key Algorithms**	**8 Hrs**

Classical Cryptographic Systems: Caesar Cipher, Vigenère Cipher, substitution and permutation methods, Symmetric Key Cryptography: overview of symmetric encryption techniques, modern block ciphers like DES and AES, operational principles.

Unit-3	**Stream Ciphers and Hash Functions**	**6 Hrs**

Stream Ciphers: RC4, pseudo-random number generators, Hash Functions key characteristics, applications of SHA family, Message Authentication Codes - MACs)

Unit-4	**Introduction to Asymmetric Key Cryptography**	**8 Hrs**

Asymmetric Key Cryptography: fundamentals, comparison with symmetric systems, RSA Algorithm: Operational mechanisms, Diffie-Hellman Key Exchange: protocol significance in secure communications.

Unit-5	**Digital Signatures and Emerging Cryptographic Techniques**	**9 Hrs**

Digital Signatures: purpose, mechanisms, examples, Elliptic Curve Cryptography: principles, advantages, Blockchain Technology: introduction, cryptographic implications.

Unit-6	Network Security Fundamentals and Protocols	9 Hrs
Network Security Objectives: confidentiality, integrity, availability, Network Security Protocols: SSL/TLS, IPSec, VPNs, Security Infrastructure: firewalls, Intrusion Detection Systems - IDS, secure communication protocols, Additional Security Mechanisms: Pretty Good Privacy - PGP, Kerberos.		

Learning Resources:

Text Books

1) Douglas Stinson, "Cryptography Theory and Practice", 2nd Edition, Chapman & Hall/CRC.

Reference Books

1) William Stallings, "Cryptography and Network Security", Pearson Education.
2) Dr. B. B. Meshram, TCP/IP & Network Security, SPD Publication.

e-Books

1) https://mrcet.com/downloads/digital_notes/CSE/III%20Year/CSIT_III-II_CRYPTOGRAPHY%20AND%20NETWORK%20SECURITY%20DIGITAL%20NOTES%20(1).pdf

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc26_cs57/preview
2) https://nptel.ac.in/courses/106105162

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAOE1051 Course Name: Cyber Security

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic Computer and Internet Knowledge. Awareness of Common Cyber Threats.

Course Objectives: The objective of the course is to

1) Understand basic concepts and terms in computer security.
2) Identify cyber threats like fraud and stalking and learn protection methods.
3) Analyse attacks like DOS and malware, and ways to detect and remove them.
4) Learn hacker techniques and basics of penetration testing.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explain core concepts of computer security and common cyber threats
CO2	Identify and describe types of cybercrimes, including stalking, fraud, and malware
CO3	Contrast methods for detecting and protecting against cyber- attacks.
CO4	Justify the Indian cyber laws and discuss their importance in preventing cybercrimes.

Course Description:

This course covers fundamental concepts of cyber security and cyber laws, focusing on identifying cyber threats, understanding preventive measures, and exploring legal frameworks. Students will learn about types of cyber-attacks, protective strategies, and the basics of digital forensics, equipping them to navigate and respond to security challenges in the digital world responsibly.

Course Content

Unit-1	**Introduction to Computer Security**	**8 Hrs**

Introduction to Cyber Security, Types of Cyber Threats, Basic Security Terminology, Security Concepts and Approaches, Real-World Applications of Computer Security.

Unit-2	**Cyber Stalking, Fraud, and Abuse**	**8 Hrs**

Cyber Stalking, Internet Fraud, Identity Theft, Protecting Against Cyber Crime, Cyber Abuse.

Unit-3	**Denial of Service Attacks and Malware**	**8 Hrs**

Introduction to DOS, Illustrating an Attack, Types of Malwares (Viruses, Trojan Horses, Buffer Overflow, Spyware), Detecting and Eliminating Malware.

Unit-4	**Techniques Used by Hackers**	**8 Hrs**

Hacker Techniques, Reconnaissance Phase, Actual Attacks, Malware Creation, Basics of Penetration Testing.

Unit-5	**The Legal Perspectives of Cyber Crime**	**6 Hrs**

Importance of Cyber Laws, Indian IT Act Overview, Challenges in Indian Cyber Law, Cybercrime Scenario in India, Legal Procedures in Handling Cybercrime

Unit-6	**Introduction to Forensics**	**8 Hrs**

Basics of Digital Forensics, Evidence Collection on PC, System Logs, Data Recovery Techniques, Mobile Forensics, Case Studies / Practical Demonstration.

Learning Resources:

Text Books

1) Computer Security Fundamentals - Chuck East tom, Pearson, Third edition (Unit I to IV &VI).

Reference Books

1) Jason Luttgens, Matthew Pepe, Kevin Mandia, Incident Response & Computer Forensics, McGraw- Hill Osborne Media, 3rd edition, 2014.
2) Handbook of Applied Cryptography - Menezes, an Oorschot, and S.A. Vanstone.

e-Books

1) https://mrcet.com/pdf/Lab%20Manuals/IT/CYBER%20SECURITY%20(R18A0521).pdf

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc25_cs116/preview
2) https://onlinecourses.nptel.ac.in/noc25_cs117/preview

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAOE1052 Course Name: Digital Marketing

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic knowledge of marketing principles and familiarity with internet or social media usage.

Course Objectives: The objective of the course is to

1) To understand the fundamentals and scope of digital marketing in modern business.
2) To design and implement strategies for search engine optimization (SEO), social media, and content marketing.
3) To apply tools for online advertising, analytics, and campaign performance measurement.
4) To develop integrated marketing plans for brand promotion and customer engagement.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Implement digital marketing concepts and tools to real-world business scenarios.
CO2	Design and manage effective online marketing campaigns across multiple platforms.
CO3	Assess digital marketing metrics to assess and improve campaign performance.
CO4	Demonstrate skills in SEO, social media marketing, and content strategy development.

Course Description:

This course introduces the fundamentals of digital marketing, including SEO, social media, and analytics tools.

Course Content

Unit-1	**Introduction to Digital Marketing**	**6 Hrs**

fundamentals and scope of digital marketing in modern business, differences between traditional and digital marketing, overview of digital marketing channels including SEO, SEM, social media, content, email, and mobile marketing, roles, career opportunities, and emerging industry trends.

Unit-2	**Website Planning and Optimization**	**8 Hrs**

Website design principles and landing pages, content marketing strategy including blogs, videos, podcasts, and infographics, storytelling and content creation for engagement, content planning and editorial calendars.

Unit-3	**Search Engine and Content Marketing**	**8 Hrs**

On-page and off-page SEO techniques, keyword research, link building, and optimization tools, Google Ads and PPC campaigns, paid search strategies, monitoring SEO/SEM performance using analytics tools.

Unit-4	**Social Media Marketing**	**8 Hrs**

Major social media platforms such as Facebook, Instagram, LinkedIn, Twitter, YouTube, organic versus paid social strategies, community management and audience engagement, social media campaign planning, content scheduling, and analytics.

Unit-5	**Email and Mobile Marketing**	**6 Hrs**

Email campaign strategy and automation, mobile marketing including apps, push notifications, and SMS campaigns, personalization and segmentation techniques, metrics to track campaign effectiveness.

Unit-6	Web Analytics and Digital Strategy	8 Hrs
Digital marketing KPIs and metrics, tools for campaign performance measurement such as Google Analytics and social media insights, A/B testing, conversion rate optimization and reporting, designing integrated digital marketing campaigns aligned with brand promotion and customer engagement goals.		

Learning Resources:

Text Books

1) Philip Kotler, Hermawan Kartajaya, and Iwan Setiawan, *Marketing 5.0: Technology for Humanity*, Wiley, 2021.

Reference Books

1) Ryan, D., *Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation*, Kogan Page, 2022.
2) Chaffey, D., and Ellis-Chadwick, F., *Digital Marketing: Strategy, Implementation and Practice*, Pearson, 2023.

e-Books

1) https://www.digitalmarketer.com/digital-marketing/assets/pdf/ultimate-guide-to-digital-marketing.pdf?srsltid=AfmBOop2M19TEUV5PXqFi8K9-svRbQNmGczm-ym8UC6prneSsfshT7-O

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc26_mg06/preview
2) https://nptel.ac.in/courses/110105920

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAOE1053 Course Name: Design Thinking & Problem Solving

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic understanding of computers, familiarity with digital tools, and elementary problem-solving skills.

Course Objectives: The objective of the course is to

1) To develop computational and digital thinking skills for analyzing complex problems.
2) To apply structured problem-solving techniques using digital tools and technologies.
3) To enhance decision-making through logical reasoning, data analysis, and algorithmic approaches.
4) To foster creativity and innovation in designing effective digital solutions for real-world challenges.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Recall key concepts and stages of design thinking.
CO2	Explain user needs and problem definition using empathy and research.
CO3	Implement design thinking methods to develop practical solutions.
CO4	Evaluate feedback and refine solutions for improved outcomes.

Course Description:

This course is a human-centered approach to innovative problem-solving. It uses five stages: empathize, define, ideate, prototype, and test. This equips users with creative tools for real-world challenges.

Course Content

Unit-1	**Foundations of Design Thinking and Problem Solving**	**8 Hrs**

Introduction to design thinking, human-centered approach, stages of design thinking process, analytical versus design thinking, creativity and innovation, role of empathy and iteration, applications in computing and business innovation.

Unit-2	**Empathize – Understanding Users and Their Needs**	**7 Hrs**

User research methods, observation and interviews, empathy mapping, user journey mapping, persona development, identifying user needs and pain points, understanding user motivation and context.

Unit-3	**Define – Problem Identification and Framing**	**8 Hrs**

Data synthesis and insight generation, clustering and thematic analysis, defining problem statements, “How Might We” questions, user-centered problem framing, prioritizing design challenges, converting needs into opportunities.

Unit-4	**Ideate – Generating and Evaluating Ideas**	**8 Hrs**

Brainstorming techniques, mind mapping, lateral thinking, SCAMPER method, divergent and convergent thinking, creativity tools, idea evaluation and selection, feasibility and desirability analysis.

Unit-5	**Prototype – Developing and Refining Concepts**	**8 Hrs**

Prototyping principles, low-fidelity and high-fidelity prototypes, sketches and wireframes,

storyboarding, rapid prototyping tools, iterative refinement, feedback-driven design improvement, visualization of solutions.

Unit-6	Test and Implement – Validation and Real-World Integration	8 Hrs

User testing methods, usability evaluation, feedback analysis, iteration and improvement, linking design thinking with agile and lean approaches, implementation planning, case studies in IT and product design, project-based application of design thinking.

Learning Resources:

Text Books

1) Jeannette M. Wing, *Computational Thinking*, MIT Press, 2020.

Reference Books

1) Grover, S., & Pea, R., *Computational Thinking in K-12: A Review of the State of the Field*, Springer, 2018
2) Selby, C., & Woollard, J., *Computational Thinking: The Developing Definition*, Routledge, 2019.

e-Books

1) https://readings.design/PDF/Tim%20Brown,%20Design%20Thinking.pdf

MOOC / NPTEL/YouTube Links

1) https://nptel.ac.in/courses/110106124
2) https://onlinecourses.nptel.ac.in/noc26_me57/preview

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC101P Course Name: Data Structure and Algorithms - Lab

Teaching Scheme	Credit	Evaluation Scheme
Practical: 04 Hours/Week	**02**	**ISA: 25 Marks** **POE: 50 Marks**

Prerequisites, if any:

Computer Programming in C.

Course Objectives: The objective of the course is to

1) To understand fundamental techniques for evaluating algorithm efficiency
2) To explore key methods for data organization, including searching and sorting
3) Gain knowledge of essential data structures for problem solving.
4) To examine principles for managing hierarchical data structures

Course Outcomes: After successful completion of the course, student will be able to

CO1	Describe the fundamentals of algorithm performance analysis
CO2	Apply various searching and sorting algorithms to solve computational problems
CO3	Implement algorithms using linear data structures for efficient data handling
CO4	Analyze the use of tree-based structures in hierarchical data organization

Course Description:

This course covers algorithm design and analysis, focusing on performance evaluation, searching and sorting methods, and linear and non-linear data structures. Students will learn to assess algorithm efficiency and apply strategies to solve complex problems, equipping them to design effective algorithms for real-world applications.

Course Content

Sr. No.	Topic of Practical / Experiment / Tutorial	Assigned Hours
1	Implementation of Sorting Techniques – Part I (Insertion, Selection)	2
2	Implementation of Sorting Techniques – Part II (Merge, Quick)	2
3	Implementation of Stack operations (push, pop, display)	2
4	Implementation of Linear Queue operations (enqueue, de queue, display)	2
5	Implementation of Singly Linked List operations (create, insert, delete)	2
6	Implementation of Doubly Linked List operations (create, insert, delete)	2
7	Implementation of Tree operations (create, insert, delete, traversal)	2
8	Implementation of Binary Search Tree operations (create, insert, delete, search)	2
9	Implementation of Graph operations (representations, BFS/DFS, traversal)	2
10	Implementation of Sorting Techniques – Part I (Insertion, Selection)	2
11	Mini Project	10

Learning Resources:

Text Books

1) Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education.
2) Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, *Fundamentals of Data Structures in C*, Universities Press.
3) Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, *Introduction to Algorithms* (3rd or 4th Edition), PHI / MIT Press.

Reference Books

1) Reema Thareja**,** *Data Structures Using C*, Oxford University Press.
2) Robert Sedgewick**,** *Algorithms in C* (Parts 1–5), Addison Wesley.

e-Books

1) https://docs.oracle.com/cd/E19059-01/stud.10/819-0493/OtherTools.html
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc22_cs71/preview[CMI]
2) https://onlinecourses.swayam2.ac.in/cec22_cs13/preview[UoK]

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC102P Course Name: Database Management System - Lab

Teaching Scheme	Credit	Evaluation Scheme
Practical: 04 Hours/Week	**02**	**ISA: 25 Marks** **POE: 50 Marks**

Prerequisites, if any:

Computer knowledge, Data Structures and algorithms, Programming.

Course Objectives: The objective of the course is to

1) To expose the students to the fundamentals of Database Management System.
2) To understand the practical applicability of database management system concepts
3) To work on existing database systems, designing of database, creating relational database, analysis of table design
4) To expose the students to the fundamentals of Database Management System.

Course Outcomes: After successful completion of the course, student will be able to

CO1	To get practical knowledge on designing and creating relational database systems.
CO2	To implement various advanced queries execution such as relational constraints, joins, set operations, aggregate functions, trigger, views and embedded SQL.
CO3	To understand and apply transactions.
CO4	To be able to design and implement database applications on their own

Course Description:

This Course is designed to understand and apply various DBMS concepts.

Course Content

Sr. No.	Topic of Practical / Experiment / Tutorial	Assigned Hours
1	Model a database for a College Management System showing entities such as *Student, Faculty, Course,* and *Registration* with appropriate relationships, attributes, and cardinalities.	2
2	Given an Unnormalized Student Database, normalize it step-by-step up to 3NF to remove redundancy and anomalies.	2
3	Design and execute DDL queries for creating, altering, and dropping tables for a Library Management System.	2
4	Implement DML queries (inserting, deleting, updating, and displaying records) on the Library Management System database created in the previous practical to perform data operations on the Author, Book, and Member tables.	2
5	Consider two tables Students and Student_IT representing students enrolled in Computer Science and Information Technology departments. Perform SQL set operations to combine and compare their data.	2
6	Design two tables — Department and Employee — and demonstrate all SQL join operations to retrieve related data.	2
7	Create a table named Employee and demonstrate the use of aggregate functions along with GROUP BY and HAVING clauses.	2

8	Create and demonstrate the use of built-in and user-defined functions in SQL using an Employee table.	2
9	Write PL/SQL blocks to demonstrate predefined and user-defined exception handling.	2
10	Create and execute stored procedures to perform different operations such as inserting, updating, and displaying employee records.	2

Learning Resources:

Text Books

1) Introduction to database systems C.J.Date Pearsons Education8th
2) Database system concept Korth, Silber schatzand Sudarshan MGH5th

Reference Books

1) SQL,PL/SQL the programming language of OracleI van Bayross BPB4th
2) ORACLEPL/SQLProgrammingScottUlmanTMH9th

e-Books

1) https://media.geeksforgeeks.org/courses/syllabus/5723a07ce6db1b2e7fe33b5db5f0d606.pdf
2) https://ncert.nic.in/vocational/pdf/jhde103.pdf

MOOC / NPTEL/YouTube Links

1) https://www.geeksforgeeks.org/dbms/dbms
2) https://www.tutorialspoint.com/dbms/index.htm

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAOJT106P Course Name: Field Project - Lab

Teaching Scheme	Credit	Evaluation Scheme
Practical: 04 Hours/Week	**02**	**ISA: 25 Marks** **POE: 50 Marks**

Prerequisites, if any:

Computer Programming in C.

Course Objectives: The objective of the course is to

1) To apply theoretical knowledge to solve real-world problems through software development
2) To enhance skills in system design, coding, testing, and documentation.
3) To encourage teamwork, project planning, and time management.
4) To develop the ability to present and defend technical solutions effectively.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Identify and describe real-life or societal problems through field visits and observations.
CO2	Analyze the root causes and impacts of identified problems using suitable data collection and analytical methods.
CO3	Propose feasible and innovative solutions to the identified societal or community problems.
CO4	Communicate findings and proposed solutions effectively through written reports and oral presentations.

Course Description:

The Field Project aims to help students understand real-world societal problems through observation, interaction, and analysis. Students should visit a community, organization, or location, identify a problem, and propose possible solutions — implementation is *not compulsory*.

Field Project Guidelines

1.Objective

The Field Project aims to help students understand real-world societal problems through observation, interaction, and analysis. Students should visit a community, organization, or location, identify a problem, and propose possible solutions — implementation is *not compulsory*.

2.Project Stages

a) Topic Selection
- Choose a local or societal issue that can be studied through a field visit.
- The topic should relate to technology, environment, education, health, infrastructure, or community development.
- Get approval from your project guide or faculty before starting.

b) Field Visit and Data Collection
- Visit the selected site or community in small teams.
- Observe, interview, and collect data (photos, surveys, or notes).
- Record key findings about the problem and its causes.

c) Problem Analysis
- Analyze the data collected during the visit.
- Identify root causes and possible impacts of the problem.
- Discuss the issue with team members and mentor for better understanding.

d) Solution Proposal
- Suggest feasible, innovative, and sustainable solutions to address the problem.
- The solution can be technical, social, or managerial in nature.
- Justify how your solution could help the community or organization.

e) Report Preparation
- Prepare a well-structured report containing:
  - Title and team details

- Introduction and background
- Description of the field and identified problem
- Data and observations
- Analysis and discussion
- Proposed solution(s)
- Conclusion and references

f) Presentation

- Each team must present their findings and proposed solutions to the class or evaluation panel.
- Use PowerPoint slides or posters to support your presentation.

3.General Instructions

- Work in teams of 3–5 students.
- Maintain a record of visits and interactions.
- Give credit to all sources and people you consulted.
- Ensure ethical behaviour and respect during field visits.
- Avoid copying or using pre-existing project material.

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCASW107T Course Name: Seminar

Teaching Scheme	Credit	Evaluation Scheme
Practical: 04 Hours/Week	**02**	**ISA: 50 Marks** **POE: -**

Prerequisites, if any:

Computer Programming in C

Course Objectives: The objective of the course is to

1) To develop skills in researching and analyzing advanced topics in computer applications.
2) To enhance technical writing and documentation abilities.
3) To improve oral communication and presentation skills in a professional setting.
4) To encourage independent learning and awareness of recent technological trends.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explore cutting edge technology topics in computer applications.
CO2	Prepare a well-structured technical report based on thorough literature review.
CO3	Deliver an effective seminar presentation using appropriate tools and techniques.
CO4	Respond confidently to questions and engage in technical discussions.

Course Description:

This course enables students to explore advanced and emerging topics in computer applications with a focus on industry-oriented projects. Students conduct a literature survey, analyze recent technological trends, and prepare a structured technical seminar report. The course enhances research skills, technical writing, and documentation abilities. It also develops professional presentation and communication skills through seminar delivery and discussions. Continuous evaluation encourages independent learning, innovation, and awareness of current industry practices.

Nature of Seminar

1. The aim of this seminar is to make the students to study regarding industrial project. They are expected to go through the latest trend pertaining to computer and allied fields, to do the literature survey and deliver the seminar on their work done in an industrial project. The other important aim of the seminar is to encourage and develop the personality, aptitude and knowledge of the students

- Seminar work should be continually evaluated based on the contributions of an individual student, originality of the work, innovations brought in, research and developmental efforts, depth and applicability, etc.
- Three mid-term evaluations should be done, which includes presentations and demos of the work done.

Project Report Format:

1. Page Size: Trimmed A4
2. Top Margin: 1.00 Inch
3. Bottom Margin: 1.32 Inches
4. Left Margin: 1.5 Inches
5. Right Margin: 1.0 Inch
6. Para Text: Times New Roman 12 Point Font
7. Line Spacing: 1.5 Lines
8. Page Numbers: Right Aligned at Footer. Font 12 Point. Times New Roman
9. Headings: Times New Roman, 14 Point Bold Face
10. Certificate: All students should attach standard format of Certificate as described by the department. Certificate should be awarded to batch and not to individual student.

Certificate should have signatures of Guide, Head of Department and Principal/ Director.

11. Index of Report:
    a) Title Sheet
    b) Certificate
    c) Acknowledgement
    d) Table of Contents
    e) List of Figures
    f) List of Tables
12. References: References should have the following format
    a) For Books: “Title of Book”, Authors, Publisher, Edition
    b) For Papers: “Title of Paper”, Authors, Journal/Conference Details, Year

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC201 Course Name: Network Technology

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 03 Hours/Week	**03**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

1) Understanding of Computer Architecture.
2) Knowledge of Operating Systems.

Course Objectives: The objective of the course is to

1) Understand basic concepts and applications of data communication and networking.
2) Explore various LAN technologies and their importance.
3) Learn about the data link layer, including framing and error control.
4) Examine IPv4/IPv6 protocols and congestion control methods.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explain the fundamental concepts and applications of data communication and networking.
CO2	Identify and describe various LAN technologies and their roles in networking.
CO3	Demonstrate data link layer techniques for framing and error control in networking.
CO4	Differentiate IPv4/IPv6 protocols and congestion control methods for efficient data transfer.

Course Description:

This course covers data communication and networking concepts, focusing on network architecture, protocols, and technologies. Students will analyze LAN technologies, understand data link layer functions, and explore IPv4/IPv6 protocols, culminating in the evaluation of application layer protocols and basic network security practices.

Course Content

Unit-1	Introduction to Data Communication	8 Hrs

Introduction to Networking and Data communication: Need of Networking, Components of Data communication - sender, receiver, message, transmission media, Network Architecture-Client-Server and Peer to peer, Categories of Networks- LAN, WAN. MAN, Network topologies- Bus, Ring, Star, Mesh, Transmission Media - Guided Media -Twisted-Pair Cable, Coaxial Cable, Fiber-Optic Cable, Unguided Media: Radio Waves, Microwaves, Infrared and satellite communication,

Unit-2	Application Layer Protocols	6 Hrs

Network Models and Services: OSI reference model, TCP/IP reference model, Comparison of OSI and TCP/IP reference model, Introduction to Application Layer: Domain name system (DNS), Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol(SMTP), Telnet, File Transfer Protocol (FTP)

Unit-3	Presentation Layer and Session Layer Protocols	6 Hrs

Introduction to Presentation Layer, Services of Presentation Layer: Data encoding, Data encryption and data compression. Introduction to Session Layer, Services of session layer: Data Flow control, simplex, half-duplex, or full-duplex.

Unit-4	Transport layer and Network layer Protocols	8 Hrs

Transport layer - Transport Layer Primitives: listen, connect, send, receive, disconnect, Protocols: TCP, UDP, Network layer- IP Protocol and IP addressing, Connection oriented and connectionless services.

Unit-5	Routing algorithm	5 Hrs
Routing algorithm: Shortest path, Flooding, distance vector, Congestion control, Data link Layer-Data Link Layer protocols: Stop and Wait protocol, Sliding window protocol, Services of Data Link Layer: Framing, Error detection and correction, Flow control.		
Unit-6	**Network Vulnerabilities**	**6 Hrs**
Network Vulnerabilities: Introduction to Vulnerabilities and Threats, Threats intransit, Active/Passive and Passive attacks: Virus, Worm. Malware, Hacking, Cracking, Sniffing, Spoofing, Dos, DDos, Trojan Horse. Ransomware, Keyoggers, Rootkits.		

Learning Resources:

Text Books

1) A.Tanenbaum, Computer Networks, PHI Publication, 5th Edition, 2011.

Reference Books

1) Larry Peterson and Bruce Davie, Computer Networks: A Systems Approach, Morgan Kufman Publication, 5th Edition, 2012
2) S. Keshav, An Engineering Approach to Computer Networking, Addison-Wesley Professional.

e-Books

1) https://mrcet.com/downloads/digital_notes/CSE/III%20Year/COMPUTER%20NETWORKS%20NOTES.pdf
2) https://sandilands.info/sgordon/teaching/its413y09s2/protected/ITS413Y09S2L02-Network-Technologies.pdf

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.swayam2.ac.in/cec19_cs07/preview
2) https://onlinecourses.nptel.ac.in/noc22_ee61/preview [IIT Kharagpur]

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC201T Course Name: Network Technology - Tutorial

Teaching Scheme	Credit	Evaluation Scheme
Tutorial/Practical: 01 Hours/Week	**01**	**ISA: 25 Marks**

Prerequisites, if any:

1) Understanding of Computer Architecture
2) Knowledge of Operating Systems.

Course Objectives: The objective of the course is to

1) Understand basic concepts and applications of data communication and networking.
2) Explore various LAN technologies and their importance
3) Learn about the data link layer, including framing and error control.
4) Examine IPv4/IPv6 protocols and congestion control methods.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explain the fundamental concepts and applications of data communication and networking.
CO2	Identify and describe various LAN technologies and their roles in networking.
CO3	Demonstrate data link layer techniques for framing and error control in networking.
CO4	Differentiate IPv4/IPv6 protocols and congestion control methods for efficient data transfer.

Course Description:

This course covers data communication and networking concepts, focusing on network architecture, protocols, and technologies. Students will analyze LAN technologies, understand data link layer functions, and explore IPv4/IPv6 protocols, culminating in the evaluation of application layer protocols and basic network security practices.

Course Content

Sr. No.	Topic of Practical/Experiment/Tutorial	Assigned Hours
1	**Unit1:** Introduction to data communication concepts, transmission media and Network topologies	1 Hrs
2	**Unit2:** Exploration of OSI reference model, TCP/IP reference model.	1 Hrs
3	**Unit3:** Understanding Presentation Layer and Session Layer	1 Hrs
4	**Unit4:** Implementing protocols in networks technology.	1 Hrs
5	**Unit5:** Implementing routing algorithms in networks.	1 Hrs
6	**Unit6:** Exploring Network Vulnerabilities and hacking.	1 Hrs

Learning Resources:

Text Books

1) A. Tanenbaum, Computer Networks, PHI Publication, 5th Edition, 2011

Reference Books

1) Larry Peterson and Bruce Davie, Computer Networks: A Systems Approach, Morgan Kufman Publication, 5th Edition, 2012.
2) S.Keshav, An Engineering Approach to Computer Networking, Addison-Wesley Professional.

e-Books

1. https://mrcet.com/downloads/digital_notes/CSE/III%20Year/COMPUTER%20NETWORKS%20NOTES.pdf
2. https://sandilands.info/sgordon/teaching/its413y09s2/protected/ITS413Y09S2L02-Network-Technologies.pdf

MOOC / NPTEL/YouTube Links

1. https://onlinecourses.nptel.ac.in/noc22_ee61/preview
2. https://onlinecourses.swayam2.ac.in/cec19_cs07/preview

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC202 Course Name: Software Project Management

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Programming fundamentals, Algorithms, DBMS.

Course Objectives: The objective of the course is to

1) To introduce the principles and methodologies of software engineering.
2) To provide knowledge of software development life cycle models and their applications.
3) To develop skills in software requirements analysis and design.
4) To understand software testing strategies and quality assurance processes.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Apply software development life cycle models to design software systems.
CO2	Analyze and document software requirements effectively.
CO3	Design modular and structured software solutions using standard practices
CO4	Implement appropriate software testing techniques for quality assurance.

Course Description:

Software Engineering covers the principles and practices for designing, developing, testing, and managing high-quality software systems using structured engineering methods.

Course Content

Unit-1	**Introduction to Software Engineering**	**8 Hrs**

Introduction to Software Engineering, Software Development Life Cycle (SDLC), Software Process Models: Waterfall, Incremental, Spiral, Agile, Characteristics of Good Software, Software Engineering Ethics.

Unit-2	**Requirement Analysis**	**8 Hrs**

Types of Requirements: Functional and Non-functional, Requirements Elicitation Techniques, Requirements Analysis, Software Requirements Specification (SRS), Validation and Verification of Requirements.

Unit-3	**Software Design**	**8 Hrs**

Objectives of Input Design, Input Validations, Design of output:- Objectives of Output, Design Types Of Output, Coupling & Cohesion User Interface design: Elements of good design, design issues, features of modern GUI, error messages.

Unit-4	**Software Testing and Quality Assurance**	**8 Hrs**

Basics of Software Testing, Levels of Testing: Unit, Integration, System, Acceptance, Testing Techniques: Black-box, White-box, Test Planning and Test Case Design, Software Quality Assurance (SQA) Software Metrics and Measurement, Introduction to Automation Testing.

Unit-5	**Software Maintenance and Evolution**	**8 Hrs**

Introduction to Software Maintenance, Types of Maintenance: Corrective, Adaptive, Perfective, Preventive, Challenges in Software Maintenance, Impact Analysis and Change Management, Software Configuration Management in Maintenance.

Unit-6	**Advanced Software Engineering Concepts**	**8 Hrs**

Component-Based Software Engineering (CBSE), Service-Oriented Architecture (SOA), Software

Reuse, Software Reliability and Fault Tolerance, Introduction to Cloud-based Software Engineering, Dev*O*ps and Continuous Integration/Continuous Deployment (CI/CD).

Learning Resources:

Text Books

1) Software Engineering by R.S. Pressman, Tata McGraw-Hill.

Reference Books

1) An Integrated Approach to Software Engineering by Pankaj Jalote, Tata McGraw-Hill
2) Fundamentals of Software Engineering by Rajib Mall, PHI Learning
3) Software Engineering by Martin Shooman, McGraw-Hill
4) System Analysis and design and Introduction to Software Engineering by Parthsarathi, B.W. Khalkar, Everest Publishing House.

e-Books

1) https://engineering.futureuniversity.com/BOOKS%20FOR%20IT/Software-Engineering-9th-Edition-by-Ian-Sommerville.pdf
2) https://ebooks.lpude.in/computer_application/mca/term_2/DCAP405_SOFTWARE_ENGINEERING.pdf

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc24_cs119/preview#:~:text=Course%20certificate,Afternoon%20Session%202pm%20to%205pm.

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC203 Course Name: Feature Engineering

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 03 Hours/Week	**03**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic Mathematical statistics and probability, Discrete Mathematics.

Course Objectives: The objective of the course is to

1) Provide a strong foundation in the concepts, importance, and processes of feature engineering for machine learning applications.
2) Enable students to understand, apply, and evaluate various feature representation, selection, transformation, and learning techniques.
3) Develop the ability to design effective feature sets for structured, unstructured, and high-dimensional data to improve model performance.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Understand the role of feature engineering and apply techniques for feature understanding, cleaning, selection, construction, transformation, and learning to improve model performance.
CO2	Represent data effectively using numerical feature representations and apply scaling, normalization, binning, interaction features, and basic feature selection techniques.
CO3	Extract features from text and categorical data using bag-of-words, n-grams, TF-IDF, and appropriate encoding methods for large categorical variables.
CO4	Analyse the importance of feature selection and apply filter-based and greedy search methods such as recursive feature elimination and stepwise selection to reduce overfitting and improve generalization.
CO5	Apply feature transformation techniques, particularly Principal Component Analysis (PCA), for dimensionality reduction, whitening, and efficient feature representation while understanding their limitations.
CO6	Implement feature learning techniques using parametric and non-parametric approaches, including RBMs and word embeddings, and integrate learned features into machine learning pipelines for real-world applications.

Course Description:

This course introduces the principles and techniques of feature engineering, focusing on feature representation, selection, transformation, and learning to improve the performance of machine learning models on structured and unstructured data.

Course Content

Unit-1	**Introduction to Feature Engineering**	**8 Hrs**

Introduction of feature engineering, Evaluation of machine learning algorithms and feature engineering procedures, Feature understanding, Feature improvement – cleaning datasets, Feature selection – removing bad attributes, Feature construction, Feature transformation, Feature learning.

Unit-2	**Basics of Feature Representation**	**7 Hrs**

Scalars, Vectors, and Spaces, Dealing with Counts, Binarization, Quantization or Binning, Log Transformation, Feature Scaling or Normalization, Min-Max Scaling, Standardization (Variance Scaling), ℓ2 Normalization, Interaction Features, Feature Selection.

Unit-3	**Features of Text and Categorical Data**	**6 Hrs**

Bag-of-X: Turning Natural Text into Flat Vectors, Filtering for Cleaner Features, Atoms of Meaning: From Words to n-Grams to Phrases, Tf-Idf : A Simple Twist on Bag-of-Words, Putting It to the Test, Deep Dive, Encoding Categorical Variables, Dealing with Large Categorical Variables.

Unit-4	Feature Selection	6 Hrs
Importance of Feature Selection in Machine Learning, Goals of Feature Selection, Classes of Feature Selection Methodologies, Effect of Irrelevant Feature, Over fitting to Predictors and External Validation, Greedy Search Methods- Simple Filters, Recursive Feature Elimination, Stepwise Selection.		
Unit-5	**Feature Transformations**	**7 Hrs**
Intuition, Derivation, Linear Projection, Variance and Empirical Variance -Vector Formulation, General Solution of the Principal Components, Transforming Features, Implementing PCA, PCA in Action, Whitening and ZCA, Considerations and Limitations of PCA, Use Cases.		
Unit-6	**Feature Learning**	**9 Hrs**
Parametric assumptions of data, Non-parametric fallacy, feature learning algorithms, Reconstructing the data, The Bernoulli RBM, Extracting PCA components from MNIST, Extracting RBM components from MNIST, Using RBMs in a machine learning pipeline, Learning text features – word vectorizations, Word embeddings, Application of word embeddings – information Retrieval.		

Learning Resources:

Text Books

1) Sinan Ozdemir, Divya Susarla, "Feature Engineering Made Easy", Packt Publishing, ISBN 978-1-78728-760-0__
2) Alice Zheng & Amanda Casari, "Feature Engineering for Machine Learning: Principles and Techniques for data scientist", Oreilly.

Reference Books

1) Max Kuhn , Kjell Johnson, "Feature Engineering and Selection: A Practical Approach for Predictive Models" 1st Edition, Chapman & Hall/CRC Data Science Series, ISBN 13 978-1-138-07922-9.
2)

e-Books

1) https://www.repath.in/gallery/feature_engineering_for_machine_learning.pdf

MOOC / NPTEL/YouTube Links

1) https://www.mygreatlearning.com/academy/learn-for-free/courses/feature-en
2) https://www.pluralsight.com/paths/feature-engineering

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC204 Course Name: Java Programming

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 03 Hours/Week	**03**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic Mathematics

Course Objectives: The objective of the course is to

1) Understand Java basics, including classes, objects, methods, and data types.
2) Learn control structures, methods, and operators for building efficient programs.
3) Use arrays for managing collections of data in Java applications
4) Explore inheritance and polymorphism for code reusability and flexibility.
5) Handle errors using exception handling and gain an introduction to JavaScript for client-side scripting.

Course Outcomes: After successful completion of the course, student will be able to

CO1	**Explain** the fundamental concepts of Java, including classes, objects, methods, and data types.
CO2	**Develop** Java programs using control structures, operators, and methods.
CO3	**Implement** and manipulate arrays for handling collections of data in Java.
CO4	**Apply** the concepts of inheritance and polymorphism to achieve code reusability and flexibility in programs.

Course Description:

This course introduces Java programming, focusing on object-oriented concepts, control structures, data handling, and basic Java libraries. Students will learn to create applications using classes, inheritance, exception handling, and simple Java Script for web features, preparing them to build reliable Java applications.

Course Content

Unit-1	**Introduction to Java and Basic Concepts**	**8 Hrs**

Overview of Java, Java Class Libraries, Java Development Environment, memory concepts and arithmetic basics, introduction to classes, objects, methods, and instance variables, declaring classes and methods, using set and get methods, primitive vs. reference types, initializing objects with constructors.

Unit-2	**Control Statements and Methods**	**8 Hrs**

Control structures:'if', 'if-else', 'while', 'do-while', 'switch', 'break, and `Continue`, logical operators, program modules, static methods, and fields, declaring methods with parameters, method overloading, and Java API packages.

Unit-3	**Arrays and Data Handling**	**8 Hrs**

Declaring and creating arrays in Java, examples with arrays, passing arrays to methods, multidimensional arrays, variable-length arguments, and using command-line arguments.

Unit-4	**Object-Oriented Concepts: Inheritance and Polymorphism**	**8 Hrs**

Inheritance basics: super classes, subclasses, protected members, constructors in subclasses, relationship between classes, polymorphism: abstract classes, methods, and interfaces, final methods and classes.

Unit-5	**Exception Handling and Error Management**	**8 Hrs**

Introduction to exception handling, common exceptions (Arithmetic, Input Mismatch), Java exception hierarchy, using the `finally` block, and best practices for reliable error handling.		
Unit-6	**Introduction to JavaScript and Basic Applets**	**8 Hrs**
Overview of JavaScript for client-side scripting, JavaScript syntax basics: operators, comparisons, statements, loops, and events, objects and user-defined functions in JavaScript, basic validations, regular expressions, document object model, popovers, and windows.		

Learning Resources:

Text Books

1) Paul Deiteland Harvey Detail, Java : How to Program, Pearson's Publication, 9thEdition

Reference Books

1) Herbert Schildt, Java The Complete Reference, McGraw- Hill Publication, 9th Edition.
2) Patrick Niemeyer, Daniel Leuck, Learning Java, O'Reilly Media, 4th Edition, 2013.

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc22_cs47/preview
2) https://onlinecourses.swayam2.ac.in/aic20_sp13/preview
3) https://www.geeksforgeeks.org/java/java/
4) https://www.geeksforgeeks.org/java/java/

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPE2051 Course Name: Data Mining and Data Warehousing

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Database System, Programming Experience.

Course Objectives: The objective of the course is to

1) To understand the principles of Data warehousing and Data Mining.
2) To apply data mining techniques to extract meaningful knowledge from data repositories for analysis
3) To understand the Knowledge Discovery in Databases (KDD) process.
4) To describe data mining tasks and analyse commonly used techniques for each task.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Describe various tools of Data Mining and their techniques to solve the real time problems.
CO2	Demonstrate knowledge of Data Warehouse architecture and its implementation.
CO3	Identify and examine patterns discovered by association rule mining, classification, and their application techniques for problem-solving.
CO4	Use clustering methods and other data mining methodologies to analyze and interpret datasets.

Course Description:

This Course is designed to understand the concepts of data warehousing and data mining.

Course Content

Unit-1	Data Mining	8 Hrs

Introduction of Data Mining, Online Analytical Processing, Techniques used to mine the data, Market Basket Analysis, Limitations and challenges to DM. Foundation of Data mining, the roots of Data Mining, The
Approach to Data Exploration and Data Mining.

Unit-2	Data Warehouse	8 Hrs

Basic Concepts, Data warehouse architecture, The modern Data warehouse, Data Warehouse roles and structure, need of Data warehouse, The cost of Warehousing Data, Stores, Warehouses and Marts, Metadata, Metadata Extraction, Implementing Data Warehouse, Data Warehouse technologies.

Unit-3	Data Analysis and Visualization	8 Hrs

Data Analysis: Correlation, Covariance, Rank and Percentile, Histogram and Moving Average. Data Visualization with advance Charts: Stock Chart, Surface Chart, Donut Chart, Bubble Chart and Radar Chart.

Unit-4	Clustering	7 Hrs

What is clustering? Types of data, Partitioning Methods (k-Means, k-Medoids) Hierarchical Methods (Agglomerative, Divisive)

Unit-5	Association Rule Mining	7 Hrs

Association rules: Motivation for Association, Rule mining, Market Basket Analysis, Apriori Algorithm, FP tree Algorithm, Iceberg Queries

Unit-6	Web Mining	6 Hrs
Introduction, web content mining, web structure mining, web usage mining.		

Learning Resources:

Text Books

1) Data mining - Concepts & Techniques, Jiawei Han, Michelin Kamber, Jian Pei,3rd Ed.2012, MK publications.

Reference Books

1) Mastering Data Mining- Michael J. A. Berry, Gordon S. Linoff, 2nd Edition Wiley publications.
2) Fundamentals of Database Systems, Nava the and Elmasry, Addison Wesley, 2000

e-Books

1) https://www.lpude.in/SLMs/Master%20of%20Computer%20Applications/Sem_1/DECAP446_DATA_WAREHOUSING_AND_DATA_MINING.pdf

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc20_cs12/preview
2) https://onlinecourses.swayam2.ac.in/cec19_cs01/preview

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPE2052 Course Name: Big Data Analytics

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Computer Programming, Data Mining, Data Analytics and Data Visualization.

Course Objectives: The objective of the course is to

1. Understand the concepts Big Data and its technologies.
2. Use Big Data platforms for big data analysis.
3. Learn how to create compelling visualization stode liver business insights.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Demonstrate big data concepts and tools to manage and process large-scale datasets.
CO2	Describe structured and unstructured data using Hadoop and Spark frameworks.
CO3	Implement data analytics techniques for pattern discovery and decision-making.
CO4	Design big data solutions for performance, scalability, and real-world applications.

Course Description:

This Course is designed to understand the concepts of Big Data Analytics.

Course Content

Unit-1	**Introduction to Big Data**	**6 Hrs**

Big Data Concepts, Challenges Opportunities from Big Data Enterprise Information Management: New Approach to Enterprise Information Management for Big Data.

Unit-2	**Big Data in the enterprise**	**7 Hrs**

Capabilities needed for big data Big Data Implications for Industries Big Data Analytics Telecom/Banking/Retail/HealthCare/IT.

Unit-3	**Data Modelling**	**8 Hrs**

Understanding data integration Pattern Big Data Workload Design Approaches, Map- Reduce patterns, Algorithms and Use Cases. Introduction of No SQL, Database concepts: ACIDVs. BASE, Advantages, Shardingand Share Nothing, Architecture, No SQL Databases, Brewers CAP Theorem, Features and comparisons of few NOSQL Databases

Unit-4	**Hadoop Framework**	**8 Hrs**

Hadoop Architecture, History of Hadoop – Facebook, Dynamo, Yahoo, Google Components of Hadoop Framework: HDFS, MAP Reduce Introduction to Pig, Hive, Mahout Installation of Single Node cluster-installation of Java, Hadoop Configuration.

Unit-5	**Big Data Analytics Methodology**	**8 Hrs**

Big data Analytics Methodology- Analyse& Evaluate Business Cases Develop Business Hypothesis Analyse outcomes, Build & Prepare Data sets, Select & Build Analytical Model, Design for Big data Scale, build production ready System, setting Up the Big Data Analytics System, Gathering data, Measure & Monitor.

Unit-6	**Extracting Value from Big Data**	**7 Hrs**

Real-time Analytics, Apache Spark, In-Memory Data Grid for Real-time Analysis Real Time Processing, Use Case.

Learning Resources:

Text Books

1) Madhu Jagadeesh, Soumendra Mohanty, Harsha Srivatsa, "Big Data Imperatives: Enterprise Big Data Warehouse, BI Implementations and Analytics", 1st Edition, Apress(2013)

Reference Books

1) Frank J. Ohlhorst,"Big Data Analytics: Turning Big Data into Big Money", Wiley Publishers (2012)
2) Cristian Molaro, Surekha Parekh, Terry Purcell, "DB211: The Database for Big Data & Analytics", MC Press, (2013)

e-Books

1) https://bmsce.ac.in/Content/CS/Unit-5.pdf
2) https://jcer.in/jcer-docs/E-Learning/Digital%20Library%20/E-Books/Big%20Data%20Analytics%20by%20RaJ%20Kamal,%20Preethi%20Saxena,TMH.pdf

MOOC / NPTEL/YouTube Links

1) https://www.youtube.com/playlist?list=PLFW6lRTa1g82LxvN-OtWnPeEngSQrD3FR
2) https://www.youtube.com/watch?v=rvJgArru8dI

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPE2053 Course Name: Advanced Operating Systems

Teaching Scheme	Credit	Evaluation Scheme
Lectures: 04 Hours/Week	**04**	**ISE: 40 Marks** **ESE: 60 Marks**

Prerequisites, if any:

Basic understanding of operating system concepts such as processes, memory management, and file systems. Familiarity with programming and data structures.

Course Objectives: The objective of the course is to

1) To understand advanced concepts and design principles of modern operating systems.
2) To analyze distributed, parallel, and real-time operating system architectures.
3) To study resource management, synchronization, and fault tolerance techniques in advanced systems.
4) To explore emerging trends and research areas in operating system technologies.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Explain advanced operating system concepts, architectures, and design approaches.
CO2	Analyze synchronization, resource management, and scheduling techniques in complex operating systems.
CO3	Evaluate the working of distributed, parallel, and real-time operating systems.
CO4	Apply advanced operating system concepts to solve practical system-level problems.

Course Description:

This course covers advanced concepts of operating systems including distributed, parallel, and real-time systems, focusing on modern design techniques and resource management strategies. It also explores emerging trends and practical applications of advanced operating system technologies.

Course Content

Unit-1	**Advanced Operating System Structures and Design**	**6 Hrs**

Evolution of operating systems, Operating system architectures, Microkernel and monolithic kernel design, Layered and modular operating systems, Virtual machines and hypervisors, Case studies of modern operating systems.

Unit-2	**Advanced Process and Thread Management**	**7 Hrs**

Multithreading models, Thread libraries, Concurrency issues, Process synchronization techniques, Deadlock handling and prevention, Multiprocessor scheduling, Load balancing techniques.

Unit-3	**Distributed Operating Systems**	**8 Hrs**

Introduction to distributed systems, Distributed system architectures, Communication in distributed systems (RPC, RMI), Distributed file systems, Distributed synchronization, Fault tolerance and reliability.

Unit-4	**Parallel and Multicore Operating Systems**	**8 Hrs**

Parallel processing concepts, Multicore architectures, Parallel scheduling algorithms, Shared memory and message passing models, Performance issues in parallel systems, GPU computing basics.

Unit-5	**Real-Time and Embedded Operating Systems**	**8 Hrs**

Real-time system concepts, Types of real-time systems, Real-time scheduling algorithms, Resource management in real-time systems, Embedded operating systems, Case studies of RTOS.

Unit-6	Security and Emerging Trends in Operating Systems	7 Hrs

Operating system security concepts, Access control and authentication, Malware and intrusion detection, Virtualization security, Cloud operating systems, Containerization and modern OS trends.

Learning Resources:

Text Books

1) Silberschatz, A., Galvin, P. B., & Gagne, G., Operating System Concepts, Wiley.
2) Tanenbaum, A. S., & Bos, H., Modern Operating Systems, Pearson.

Reference Books

1) Stallings, W., Operating Systems: Internals and Design Principles, Pearson.
2) Rajkumar Buyya, Andrey Beloglazov, & Jemal Abawajy, Cloud Computing: Principles and Paradigms, Wiley.
3) Real-Time Systems by Jane W. S. Liu, Pearson.
4) Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts – Essentials, Wiley.
5) D. M. Dhamdhere, Operating Systems: A Concept-Based Approach, McGraw-Hill.

e-Books

1) https://techworldthink.github.io/MCA/Download/S2/El%20-%20OS/FULL/OS_MD_1.pdf
2) https://www.scribd.com/document/792943694/Advanced-Operating-Systems-EBook-1

MOOC / NPTEL/YouTube Links

1) https://onlinecourses.nptel.ac.in/noc22_cs80/preview
2) http://digimat.in/nptel/courses/video/106106144/L36.html

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC204P Course Name: Java Programming Lab

Teaching Scheme	Credit	Evaluation Scheme
Practical: 04 Hours/Week	**02**	**ISA: 25 Marks** **POE: 50 Marks**

Prerequisites, if any:

1) Basic knowledge of computer operations and software.
2) Understanding of fundamental programming concepts such as variables, data types, and algorithms.

Course Objectives: The objective of the course is to

1) Understand Java basics, including syntax, variables, and data types.
2) To explore key methods for data organization, including searching and sorting
3) Design and implement classes, objects, and inheritance in Java
4) Apply polymorphism and exception handling in object-oriented programming

Course Outcomes: After successful completion of the course, student will be able to

CO1	Describe the fundamentals of algorithm performance analysis
CO2	Use control structures such as loops, conditionals, and switch-case effectively
CO3	Implement algorithms using linear data structures for efficient data handling
CO4	Analyze the use of tree-based structures in hierarchical data organization

Course Description:

This course is designed to help students understand and implement the fundamentals of Java programming. It focuses on object-oriented concepts, control structures, arrays, inheritance, polymorphism, and exception handling to develop robust and efficient applications.

Course Content

Sr. No.	Topic of Practical / Experiment / Tutorial	Assigned Hours
1	Create a simple Java program to print a personalized welcome message.	2
2	Write a Java program to declare and initialize variables of different data types and display their values.	2
3	Create a Java class with `set` and `get` methods to manage the attributes of a "Student" object (name, age, and grade).	2
4	Implement a Java program to demonstrate the use of basic control statements (`if`, `if-else`, and `switch`) for checking conditions and displaying appropriate messages.	2
5	Write a Java program that uses a `while` loop to display numbers from 1 to 10 and calculate their sum.	2
6	Design a program that accepts two numbers and a basic operation (+, −, ×, ÷) from the user, using `switch` to perform the calculation.	2
7	Create a Java program to declare an array, populate it with values, and print each element using a `for` loop.	2
8	Write a Java program to calculate the average of values in an array and display the result.	2
9	Develop a simple Java program that demonstrates inheritance by creating a base class `Vehicle` and a derived class `Car`.	2
10	Create a Java program with a base class and subclass that illustrates method overriding, showing the concept of polymorphism.	2
11	Write a Java program that uses exception handling to manage divide-	2

	by-zero errors with a `try-catch` block.		
12	Develop a Java program that catches input mismatches when a user enters invalid data, demonstrating robust error handling.	2	

Learning Resources:

Text Books

1) Paul Deiteland Harvey Detail, Java: How to Program, Pearson's Publication, 9th Edition

Reference Books

1) Herbert Schildt, Java The Complete Reference, McGraw- Hill Publication, 9th Edition.
2) Patrick Niemeyer, Daniel Leuck, Learning Java, O'Reilly Media, 4th Edition, 2013.

e-Books

1) https://onlinecourses.nptel.ac.in/noc22_cs47/preview
2) https://onlinecourses.swayam2.ac.in/aic20_sp13/preview

MOOC / NPTEL/YouTube Links

1) https://www.geeksforgeeks.org/java/java/
2) https://www.geeksforgeeks.org/java/java/

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAPCC206P Course Name: Python Programming - Lab

Teaching Scheme	Credit	Evaluation Scheme
Practical: 04 Hours/Week	**02**	**ISA: --** **POE: 50 Marks**

Prerequisites, if any:

Basic Computer skills and mathematical skills.

Course Objectives: The objective of the course is to

1) Understand why Python is a useful scripting language for developers.
2) Understand how to design and program Python applications
3) Learn how to build and package Python modules for reusability.
4) Learn how to design object-oriented program swath Python classes.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Understand the Basic Syntax of Python Programming.
CO2	Understand and implement concepts of object-oriented methodology using Python.
CO3	Understand and design collections in Python.
CO4	To develop problem-solving skills and their implementation through Python.

Course Description:

This course is designed to understanding implement the fundamentals of Python programming

Course Content

Sr. No.	Topic of Practical / Experiment / Tutorial	Assigned Hours
1	Study of Basics of python language	2
2	Python variables declaration and data types	2
3	Write a program to create as simple calculator	2
4	Write a program to check Armstrong's number or not	2
5	Write a program to display prime numbers from inputted range.	2
6	Write a program to count the number of vowel sand constants in a string.	2
7	Write a program to implement a Python list.	2
8	Write a program to implement Python Dictionary	2
9	Write a program to implement set operations in python.	2

Learning Resources:

Text Books

1) Learning Python By MarkLutz, O'Reilly Publication
2) Programming with python, Ausers Book, Michael Dawson, Cengage Learning

Reference Books

1) Practical Programming: An introduction to Computer Science Using Python, second Edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf.
2) Python for Informatics: Exploring information, Charles Severance

e-Books
1) https://www.scribd.com/document/965650635/Advanced-Python-eBook-Extended
MOOC / NPTEL/YouTube Links
1) https://onlinecourses.nptel.ac.in/noc22_cs32/preview 2) https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_GUVI_01

Tatyasaheb Kore Institute of Engineering and Technology
First Year of Master in Computer Application (FYMCA)

Course Code: 2501PMCAOJT207O Course Name: Mini Project

Teaching Scheme	Credit	Evaluation Scheme
Practical: 08 Hours/Week	**04**	**ISA: 50 Marks** **POE: 100 Marks**

Prerequisites, if any:

1) Understanding of Software Development Life Cycle.

Course Objectives: The objective of the course is to

1) To apply theoretical knowledge to solve real-world problems through software development.
2) To enhance skills in system design, coding, testing, and documentation.
3) To encourage teamwork, project planning, and time management.
4) To develop the ability to present and defend technical solutions effectively.

Course Outcomes: After successful completion of the course, student will be able to

CO1	Design and develop a complete software application or system.
CO2	Apply software engineering principles to real-world project development.
CO3	Collaborate effectively in teams and manage project workflows.
CO4	Present technical work clearly through documentation and oral presentation.

Course Description:

The Mini Project course provides hands-on experience in applying theoretical knowledge to solve real-world computing problems. Students design and develop a small-scale project using suitable tools and technologies. The course enhances problem-solving, technical, and documentation skills while introducing the software development lifecycle.

Guide Lines for Project:

a) Project Selection
- Choose a relevant and feasible project that aligns with your specialization and current industry trends.
- The project must be original; plagiarism will lead to disqualification.

b) Project Proposal Submission
- Submit a project proposal including the title, objectives, scope, technology stack, and expected outcomes.
- The proposal must be approved by the assigned faculty guide.

c) Team Formation
- Projects can be done individually or in teams (preferably 2–3 members).
- Team members must contribute equally and document individual responsibilities.

d) Regular Progress Reviews
- Attend all scheduled review meetings and present progress updates.
- Maintain a project diary or logbook signed weekly by the faculty guide.

e) Technical Documentation
- Prepare proper documentation including system requirements, design diagrams, implementation, testing, and user manuals.
- Follow standard formatting and citation styles as instructed.

f) Final Submission Requirements
- Submit the source code, executable files, project report, and presentation slides.
- Ensure the report includes screenshots, test results, and references.

g) Viva Voce and Demonstration
- Demonstrate the working of the project and be prepared to answer technical questions during the final viva.
- Each team member must be able to explain the complete project.

h) Ethical Conduct
- Ensure ethical use of data and technology; avoid using copyrighted material without permission.
- Acknowledge all external tools, APIs, or frameworks used.