

ANDHRA KESARI UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

MASTER OF COMPUTER APPLICATION (MCA) COURSE

2026-27

AK UNIVERSITY – MCA Course Structure

MCA I Year – I Semester

S. No.	Course Code	Category	Course	Max Marks		Hours Per Week			Credits
				I	E	L	T	P	
1	MCA26101	C	Data Structures with C++	30	70	3	1	0	4
2	MCA26102	C	Database Management Systems	30	70	3	1	0	4
3	MCA26103	C	Operating Systems	30	70	3	1	0	4
4	MCA26104	CF	Probability and Statistics	30	70	3	1	0	4
5	MCA26105A	EF-I	1. Computer Organization	30	70	3	1	0	4
	MCA26105B		2. Digital Logic Design						
	MCA26105C		3. Discrete Mathematical Structures						
6	MCA26106	P	Data Structures lab	25	50	0	0	6	3
7	MCA26107	P	Database Management Systems Lab	25	50	0	0	6	3
8	MCA26108	SEC	Communication Skills	50	-	0	4	0	2
			TOTAL	**260**	**490**	**15**	**9**	**12**	**28**

MCA I Year – II Semester

S. No.	Course Code	Category	Course	Max Marks		Hours Per Week			Credits
				I	E	L	T	P	
1	MCA26201	C	Software Engineering	30	70	3	1	0	4
2	MCA26202	C	Computer Networks	30	70	3	1	0	4
3	MCA26203	C	Web Technologies	30	70	3	1	0	4
4	MCA26204	CF	Research Methodology & Intellectual Property Rights (IPR)	30	70	3	1	0	4
5	MCA26205A	EF-II	1. Programming and Problem-Solving using Python	30	70	3	1	0	4
	MCA26205B		2. Introduction to R Programming						
6	MCA26206	P	Python Programming / R Programming Lab	25	50	0	0	6	3
7	MCA26207	P	Web Technologies Lab	25	50	0	0	6	3
8	MCA26208	SEC	Seminar	50	-	0	4	0	2
9	MCA26209	MC	SWAYAM Courses	-	-	0	0	0	2*
			TOTAL	**260**	**490**	**15**	**9**	**12**	**28**

C – Mandatory Core
CF – Compulsory Foundation
P – Practical
EF – Elective Foundation
SEC – Skill Enhancement Course
MC – Mandatory Courses (MOOCs)

*__NOTE:__ The credits assigned for SWAYAM Course (MOOCs Mandatory Course) will not be counted for the GPA.

MCA II Year – III Semester

S. No.	Course Code	Category	Course	Max Marks		Hours Per Week			Credits
				I	E	L	T	P	
1	MCA26301	C	Data Mining and Big Data	30	70	3	1	0	4
2	MCA26302	C	Cloud Computing	30	70	3	1	0	4
3	MCA26303	C	Artificial Intelligence & Machine Learning	30	70	3	1	0	4
4	MCA26304A	E-I	1. Blockchain Technology	30	70	3	1	0	4
	MCA26304B		2. Cryptography & Network Security						
	MCA26304C		3. Cyber Security						
5	MCA26305A	E-II	1. WISER – Quantum Foundation and Applied Technologies	30	70	3	1	0	4
	MCA26305B		2. Mobile Computing with Android						
	MCA26305C		3. Microsoft Dynamics						
6	MCA26306	P	Data Mining and Big Data Lab	25	50	0	0	6	3
7	MCA26307	P	Cryptography & Network Security lab	25	50	0	0	6	3
8	MCA26308	SEC	Technical Report Writing	50	-	0	4	0	2
9	MCA26309	MC	SWAYAM Plus Courses	-	-	0	0	0	2*
			TOTAL	**260**	**490**	**15**	**9**	**12**	**28**

MCA II Year – IV Semester

S. No.	Course Code	Category	Course	Max Marks		Hours Per Week			Credits
				I	E	L	T	P	
1	MCA26401	SEC	Project Work	100	200	0	0	36	12
			TOTAL	**100**	**200**	**0**	**0**	**36**	**12**

C – Mandatory Core
CF – Compulsory Foundation
P – Practical
EF – Elective Foundation
SEC – Skill Enhancement Course
MC – Mandatory Courses (MOOCs)

*__NOTE:__ The credits assigned for SWAYAM Plus Courses (MOOCs Mandatory Course) will not be counted for the GPA.

SEMESTER - I

Semester	MCA26101: DATA STRUCTURES WITH C++	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	To explain the concepts of object-oriented programming
LO2	Understand how to apply object-oriented concepts like data abstraction, encapsulation and inheritance
LO3	Implement the concepts of pointers, constructors and destructors
LO4	Elucidate the implementation of the six data structures using C++

COURSE OUTCOMES (CO):

Students successfully completing this course will be able to:

CO1	Illustrate the implementation of linked lists and Recursion
CO2	Analyze search algorithms and hashing technique
CO3	Apply Stacks and Queues for real world tasks
CO4	Make use of trees and graphs in solving complex problems

Course Content:

UNIT-I

Software Engineering Principles and C++ Classes: Classes: Variable - Accessing Class members – Operators – Functions and Classes – Reference parameters and Class Objects – Implementation of member function – Constructors – Destructors; Data Abstraction, Classes and ADT – Information Hiding.

Pointers and Array based Lists: Pointer Data types and Pointer variables: Declaring Pointer Variables – Address of Operator – Dereferencing Operator - Classes, Structures and Pointer Variables – Initializing Pointer Variables – Dynamic Variables – Operators on Pointer Variables.

UNIT-II

Linked Lists: Linked List – Properties – Item Insertion and Deletion – Building a Linked List – Linked List as an ADT – Ordered Linked Lists – Doubly Linked Lists – Linked Lists with header and trailer nodes – Circular Linked Lists.

Recursion: Recursive Definitions – Problem solving using recursion – Recursion or iteration - Recursion and Backtracking: n- Queens Puzzle.

Search Algorithms: Search Algorithms: Sequential – Binary search – Performance of binary search – insertion into ordered list; Hashing: Hash functions – Collision Resolution – Hashing: Implementation using Quadratic Probing – Collision Resolution: Chaining.

UNIT-III

Stacks: Stack operations – Implementation of stacks as arrays – Linked implementation of stacks – Application of stacks.

Queues: Queues: Queue operations – Implementation of Queues as arrays; Linked implementation of Queues; Priority Queue; Application of Queues.

Sorting Algorithms: Selection Sort – Insertion Sort – Quick Sort – Merge Sort – Heap Sort.

UNIT-IV

Trees: Binary Trees – Binary Tree Traversal – Binary Search Tree – Non recursive Binary Tree Traversal Algorithms – AVL Trees.

Graphs: Graph Definitions and Notations – Graph Representation – Operations on graphs – Graph as ADT – Graph Traversals – shortest path Algorithm – Minimal Spanning Tree.

PRESCRIBED TEXTBOOK:

D.S.Malik, " Data Structures using C++ ", Cengage Learning India Edition (2008).

REFERENCE BOOKS:

1) Mark Allen Weiss, "Data structures and Algorithem Analysis in C++", Third Edition, Pearson Education (2008).

2) Adam Drozdek, "Data Structures and Algorithms in C++", Cengage Learning, India Edition.

Semester	MCA26102: DATABASE MANAGEMENT SYSTEMS	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	The purpose of a database management system (DBMS)
LO2	The role of the database administrator
LO3	Data consistency, data integrity, data redundancy and data independence
LO4	The concept of entity relationships and data normalization
LO5	The concept of a client/server database
LO6	The relevant advantages of a client/server database over a non-client/server database

COURSE OUTCOMES (CO):

Students successfully completing this module will be able to:

CO1	Explain about databases, different operations, queries performed for management system problems
CO2	Demonstrate the significance of ER-diagram in DBMS
CO3	Make use of different normalizations for database size reduction and removal of redundancy
CO4	Apply PL/SQL, SQL injection, procedures etc.

Course Content:

UNIT-I

Databases and Database Users: Introduction, Characteristics of the Database Approach, Actors on the Scene, Workers behind the scene, Advantages of the using the DBMS Approach.

Database System Concepts and Architecture: Data Models, Schemas and Instances, Three Schema architecture and Data Independence, Database Languages and Interfaces, Centralized and Client/Server Architecture for DBMS, Classification of Database Management Systems.

Disk Storage, Basic File Structures and Hashing: Introduction, Secondary Storage Devices, Buffering of Blocks, Placing file Records on Disk, Operations on Files, Files of Unordered Records, Files of Ordered Records, Hashing Techniques, Other Primary File Organizations, Parallelizing Disk Access using RAID Technology.

Indexing Structures for Files: Types of Single-Level Ordered Indexes, Multilevel Indexes and Dynamic Multilevel Indexes Using B-Trees and B + Trees, Indexes on Multiple Keys, Other Types of Indexes.

Data Modeling Using the ER Model: Conceptual Data models, Entity Types, Entity Sets, Attributes and Keys, Relationship types, Relationship sets, roles and structural Constraints, Weak Entity types, Relationship Types of Degree Higher than Two, Refining the ER Design for the COMPANY Database.

The Enhanced Entity-Relationship Model: Sub classes, Super classes and Inheritance, Specialization and Generalization, Constraints and Characteristics of Specialization and Generalization Hierarchies, Modeling of Union Types using Categories, An Example University ERR Schema, Design Choices and Formal Definitions.

UNIT-II

The Relational Data Model and Relational Database Constraints: Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions and Dealing with Constraint Violations.

The Relational Algebra and Relational Calculus: Unary Relational Operations: SELECT and PROJECT, Relational Algebra Operations from set Theory, Binary Relational Operations: JOIN and DIVISION, Additional Relational Operations, Examples, The Tuple Calculus and Domain Calculus.

SQL-99: Schema Definition, Constraints, Queries and Views: SQL Data Definitions and Data Types, Specifying Constraints in SQL, Schema Change Statements on SQL, Basic Queries in SQL, More Complex SQL Queries, INSERT, DELETE and UPDATE statements in SQL, Triggers and Views.

UNIT-III

Functional Dependencies and Normalization for Relational Databases: Informal Design Guidelines for Relation Schemas, Functional dependencies, Normal Forms Based in Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form.

Relational Database Design Algorithms and Further Dependencies: Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Multivalued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, Inclusion Dependencies, Other Dependencies and Normal Forms.

UNIT-IV

Introduction to Transaction Processing Concepts and Theory: Introduction to Transaction Processing, Transaction and System Concepts, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability, Characterizing schedules Based on Serializability.

Concurrency Control Techniques: Two Phase Locking Techniques for Concurrency Control, Concurrency Control Based on Timestamp Ordering, Multiversion Concurrency control techniques, Validation concurrency control Techniques, Granularity of Data Items and multiple Granularity Locking.

Distributed Databases and Client Server Architectures: Distributed Database Concepts, Data Fragmentation, Replication, and allocation Techniques for Distributed Database Design, Types of Distributed Database Systems, An Overview if 3 Tier Client Server Architecture.

PRESCRIBED TEXTBOOK:

RamezElmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson Education (2007)

REFERENCE BOOKS:

1) Peter Rob, Carlos Coronel, "Database Systems" – Design, Implementation and Management, Eigth Edition, Thomson (2008).

2) C.J. Date, A. Kannan, S. Swamynathan, "An Introduction to Database Systems", VII Edition of Pearson Education (2006).

3) Raman A Mata – Toledo, Panline K. Cushman, "Database Management Systems", Schaum's Outlines, TMH (2007).

4) Steven Feuerstein, "Oracle PL/SQL – Programming", 10th Anniversary Edition, OREILLY (2008).

Semester	MCA103: OPERATING SYSTEMS	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	Appreciating the role of an operating system
LO2	Making aware of the issues in management of resources like processor, memory and input-output
LO3	Selection of appropriate productivity enhancing tools or utilities for specific needs like filters or version control
LO4	Obtaining some insight into the design of an operating system.

COURSE OUTCOMES (CO):

Students successfully completing this module will be able to:

CO1	Explain what an operating system and the role is it plays
CO2	Infer high level understanding of the structure of operating systems, applications, and the relationship between them
CO3	How to gather knowledge of the services provided by operating systems
CO4	Compare the exposure to some details of major OS concepts.

Course Content:

UNIT-I:

Introduction: What Operating Systems Do – Computer System Orgranization – Computer system Architecture – Operating System Structure – Operating System Operations – Process Management – Memory Management – Storage Management – Protection and Security – Distributed Systems – Special purpose Systems – Computing Environments.

System Structure: Operating System Services – User Operating System Interface – System Calls – Types of System Calls – System Programs – Operating System Design and Implementation – Operating System Structure – Virtual Machine – Operating System Generation – System Boot.

Process Concept: Overview – Process Scheduling – Operations on Processes – Inter process Communication – Examples of IPC Systems – Communication in Client Server Systems.

UNIT-II:

Multithreaded Programming: Overview – Multithreading Models – Thread Libraries – Threading Issues – Operating System Examples.

Process Scheduling: Basic Concepts – Scheduling Criteria – Scheduling Algorithms – Multiple Processor Scheduling – Thread Scheduling.

Synchronization: Background – The Critical Section Problem – Peterson's solution – Synchronization Hardware – Semaphores – Classic Problem Of Synchronization – Monitors – Synchronization Examples – Atomic Transaction.

UNIT-III:

Deadlocks: System Model – Deadlock Characterization – Methods for Handling Deadlocks – Deadlock Prevention – Deadlock Avoidance – Deadlock Detection – Recovery from Deadlock.

Memory Management Strategies: Background – Swapping – Contiguous Memory Allocation – Paging – Structure of the Page Table – Segmentation – Example: The Intel Pentium.

Virtual Memory Management: Background – Demand Paging – Copy on Write – Page Replacement – Allocation of Frames – Thrashing.

UNIT-IV:

File System: File Concept – Access Methods – Directory Structure – File System Mounting – File Sharing – Protection.

Implementing File Systems: File System Structure – File System Implementation – Directory Implementation – Allocation Methods – Free Space Management – Efficiency and Performance – Recovery – Log structured File Systems.

Secondary Storage Structure: Overview of Mass – Storage Structure – Disk Structure – Disk Attachment – Disk Scheduling – Disk Management – Swap Space Management – RAID structure.

I/O Systems: Overview – I/O Hardware – Application I/O Interface – Kernal I/O Interface – Transforming I/O requests to Hardware Operations – Streams – Performance.

PRESCRIBED TEXTBOOK:

Abraham Silberschatz, Peter Baer Galvin, Greg Gagne. "Operating System Principles", Seventh Edition, Wiley.

REFERENCE BOOKS:

1) William Stallings, "Operating Systems – Internals and Design Principles", Fifth Edition, Pearson Education (2007)

2) Achyut S Godbole, "Operating Systems", Second Edition, TMH (2007).

3) Flynn/McHoes, "Operating Systems", Cengage Learning (2008).

4) Deitel & Deitel, "Operating Systems", Third Edition, Pearson Education (2008)

Semester	MCA104: PROBABILITY AND STATISTICS	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	Students will add new interactive activities to fill gaps that we have identified by analysing student log data
LO2	Collecting input from other college professors on where students typically have difficulties
LO3	Students will add new simulation-style activities to the course in Inference and Probability

COURSE OUTCOMES (CO):

Students successfully completing this module will be able to:

CO1	Show confidence in manipulating and drawing conclusions from data and provide them with a critical framework for evaluating study designs and results
CO2	Explain the basic notions of probability laws and develop them to the stage where one can begin to use
CO3	Explain the basic notions of probability laws and develop them to the stage where one can begin to use
CO4	Summarize the study of stochastic processes,

Course Content:

UNIT I:

Some probability laws: Axioms of Probability, Conditional Probability, Independence of the Multiplication Rule, Bayes' theorem

Discrete Distributions: Random Variables, Discrete Probability Densities, Expectation and distribution parameters, Binomial distribution, Poisson distribution, simulating a Discrete distribution,

Continuous distributions: continuous Densities, Expectation and distribution parameters, exponential distribution, Normal distribution, Weibull distribution and Reliability.

UNIT II:

Estimation: Point estimation, interval estimation and central limit theorem.

Inferences on the mean and the Variance of a distribution: Hypothesis Testing, significance testing, Hypothesis and significance test on the mean, Hypothesis tests on the Variance

Inferences on proportions: estimating proportions, testing hypothesis on a proportion, Comparing two proportions: estimation, comparing two proportions: hypothesis testing.

UNIT III:

Comparing two means and two variances: point estimation: independent samples, Comparing variances: the F-distribution,

Comparing means: variances equal,

Analysis of Variance: One-way classification fixed effects model, comparing variances, pair wise comparisons, randomized complete block design

UNIT IV:

Simple linear regression and correlation: model and parameter estimation, inferences about slope, inferences about intercept, Co-efficient of determination

Multiple linear regression models: least square procedures for model fitting, a matrix approach to least squares, interval estimation.

PRESCRIBED TEXTBOOK:

1) J Susan Milton and Jesse C. Arnold: "Introduction to Probability and Statistics", Fourth edition, TMH, (2007).

REFERENCE BOOK:

1) William Mendenhall, Robert J Beaver, Barbara M Beaver: Introduction to Probability and Statistics, Twelth edition, Thomson.

Semester	MCA26105A: COMPUTER ORGANIZATION	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	Help the students to develop and understand the nature and characteristics of the organization and design of the modern computer systems
LO2	Explain the students about the basic computer organization
LO3	Focus on the organization & operation of the CPU

COURSE OUTCOMES (CO):

Students successfully completing this module will be able to:

CO1	Explain the key concepts that are likely to be included in the design of any modern computer system
CO2	Make use of the basic metrics by which new and existing computer systems may be evaluated
CO3	Outline the impact that languages, their compilers and underlying operating systems have on the design of computer systems
CO4	How to evaluate the impact that peripherals, their interconnection and underlying data operations have on the design of computer systems
CO5	Demonstrate the techniques needed to conduct the design of a computer

Course Content:

UNIT-I

Digital Logic Circuits: Digital Computers, Logic Gates, Boolean Algebra, Map Simplification, Combinational Circuits, Flip-Flops, Sequential Circuits.

Digital Components: Integrated Circuits, Decoders, Multiplexers, Registers, Shift Registers, Binary Counters, Memory Unit.

Data Representation: Data Types, Complements, Fixed Point Representation, Floating Point Representation, Other Binary Codes, error Detection Codes.

UNIT-II

Register Transfer and Microoperations: Register Transfer Languages, Register Transfer, Bus and Memory Transfer, Arithmetic Micro Operations, Logic Micro Operations, Shift Micro Operations, Arithmetic Logic Shift Unit

Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory Reference Instructions, Input-Output and Interrupt.

UNIT-III

Micro programmed Control: Control Memory, Address Sequencing, Micro Program Example, Design of Control Unit.

Central Processing Unit: Introduction, General Register Organization, Stack Organization, Instruction Format, Addressing Modes, Data Transfer and Manipulation, Program Control.

UNIT-IV

Computer Arithmetic: Addition, Subtraction, Multiplication, Division Algorithms, Floating Point Arithmetic Operations.

Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associative Memory, Cache Memory.

PRESCRIBED TEXTBOOK:

1) M. Morris Mano, "Computer System Architecture", 3rd Edition, Pearson Education (2008).

REFERENCE BOOKS:

1) V. Rajaraman, T. Radha Krishnan, "Computer Organization and Architecture", PHI

2) Behrooz Parhami, "Computer Architecture", Oxford (2007)

3) ISRD group, "Computer Organization", ace series, TMH (2007)

4) William Stallings, "Computer Organization and Architecture – Designing for Performance", Pearson Education (2005)

5) P. Chakraborty, "Computer Architecture and Organization", Jaico Books (2008)

Semester	MCA26105B: DIGITAL LOGIC DESIGN	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	To understand basic number systems, codes and logical gates.
LO2	To understand the concepts of Boolean algebra
LO3	To understand the use of minimization logic to solve the Boolean logic expressions
LO4	To understand the design of combinational and sequential circuits
LO5	To understand the state reduction methods for Sequential circuits.
LO6	To understand the basics of various types of memories

COURSE OUTCOMES (CO):

Students successfully completing this module will be able to:

CO1	Explain number systems and codes.
CO2	Illustrate Boolean expressions using Minimization methods.
CO3	Demonstrate the sequential and combinational circuits.
CO4	Apply state reduction methods to solve sequential circuits.

Course Content:

UNIT-I

Binary Systems: Digital Systems, Binary Numbers, Number base conversions, Octal, Hexadecimal and other base numbers, complements, signed binary numbers, Floating point number representation, binary codes, Error detection and correction, binary storage and registers, binary logic.

Boolean algebra and logic gates: Basic theorems and properties of Boolean algebra, Boolean functions, canonical and standard forms, Digital Logic Gates.

UNIT-II

Gate–Level Minimization: The K-Map Method, Three-Variable Map, Four-Variable Map, Five-Variable Map, sum of products, product of sums simplification, Don't care conditions, NAND and NOR implementation and other two level implementations, Exclusive-OR function.

UNIT-III

Combinational Logic: Combinational Circuits (CC), Analysis procedure, Design Procedure, Combinational circuit for different code converters and other problems, Binary Adder- Subtractor, Decimal Adder, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers, Demultiplexers.

UNIT-IV

Synchronous Sequential Logic: Synchronous Sequential Circuits, Latches, Flip-flops, analysis of clocked sequential circuits, Registers, Shift registers, Ripple counters, Synchronous counters, other counters.

Asynchronous Sequential Circuits -Introduction, Analysis procedure, Circuits with latches, Design procedure, Reduction of state and follow tables, Race- free state assignment, Hazards.

UNIT-V

Memory: Introduction, Random-Access memory, Memory decoding, ROM, Programmable Logic Array, Programmable Array Logic, Sequential programmable devices.

Register Transfer and Micro operations - Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Microoperations, Logic Microoperations, Shift Microoperations, Arithmetic Logic Shift Unit.

PRESCRIBED TEXTBOOKS:

1) Digital Design, M. Morris Mano, M. D. Ciletti, 5th edition, Pearson. (Units I, II, III, IV, Part of Unit V)

2) Computer System Architecture, M. Morris Mano, 3rd edition, Pearson. (Part of Unit V)

REFERENCE BOOKS:

1) Switching and Finite Automata Theory, Z. Kohavi, Tata McGraw Hill.

2) Fundamentals of Logic Design, C. H. Roth, L. L. Kinney, 7th edition, Cengage Learning.

3) Fundamentals of Digital Logic & Micro Computer Design, 5TH Edition, M. Rafiquzzaman, John Wiley.

Semester	MCA26105C: DISCRETE MATHEMATICAL STRUCTURES	L	T	P	C
I		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

CO1	To extend student 's Logical and Mathematical maturity and ability to deal with abstraction
CO2	To introduce most of the basic terminologies used in computer science courses and application of ideas to solve practical problems.
CO3	Apply logical reasoning to solve a variety of problems.

COURSE OUTCOMES (CO):

Students successfully completing this module will be able to:

LO1	Show greater precision in logical argument and have gained core mathematical understanding of discrete mathematics.
LO2	Construct basic concepts of mathematical proof (direct proof, proof by contradiction, mathematical induction).
LO3	Solve standard logics With confidence
LO4	Apply complex mathematical expressions and general formulae to specific contexts
LO5	Utilise Boolean Algebra and solve problems.

Course Content:

UNIT-I:

The Foundations: Logic and Proofs: Propositional Logic – Propositional Equivalences – Predicates and Quantifiers – Nested Quantifiers – Rules of Inference – Introduction to Proofs – Proof Methods and Strategy

Basic Structures: Sets, Functions, Sequences and Sums: Sets – Set Operations – Functions – Sequences and Summations

The Fundamentals: Algorithms, The Integers and Matrices: Algorithms – The Growth of Functions – Complexity of Algorithms – The Integers And Divisions – Primes and Greatest Common Divisors – Integers and Algorithms – Applications of Number Theory – Matrices

Introduction and Recursion: Mathematical Induction – Strong Induction and Well-Ordering – Recursive Definitions and Structural Induction – Recursive Algorithms – Program Correctness

UNIT-II:

Counting: The Basics of Counting – The Pigeonhole Principle – Permutations and Combinations – Binomial Coefficients – Generalized Permutations and Combinations – Generating Permutations and Combinations

Advanced Counting Techniques: Recurrence Relations – Solving Linear Recurrence Relations – Divide and Conquer Algorithms and Recurrence Relations – Generating Functions – Inclusion – Exclusion – Applications of Inclusion & Exclusion

UNIT-III:

Relations: Relations and Their Properties – n-ary Relations and Their Applications – Representing Relations – Closures of Relations – Equivalence Relations – Partial Orderings

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

UNIT-IV:

Trees: Introduction to Trees – Applications of Trees – Tree Traversal – Spanning Trees – Minimum Spanning Trees

Boolean Algebra: Boolean Functions – Representing Boolean Functions – Logic Gates – Minimization of Circuits

PRESCRIBED TEXTBOOK:

1) Kenneth H Rosen, "Discrete Mathematics & its Applications", 6th Edition, McGraw-Hill (2007) Chapters: 1 to 10

REFERENCE BOOKS:

1) Ralph P. Grimaldi, B.V. Ramana, "Discrete and Combinational Mathematics", 5th Edition, Pearson Education (2008).

2) Swapan Kumar Sarkar, "A Textbook of Discrete Mathematics", S. Chand (2008).

3) D. S. Malik and M. K. Sen, "Discrete Mathematical Structures", Thomson (2006).

Semester	MCA26106: DATA STRUCTURES LAB	L	T	P	C
I		0	0	6	3

1) Write a program for implementing the operations on complex numbers using classes.

2) Program for finding the area of circle, rectangle and room using function overloading.

3) Program for finding the volume of box using constructor overloading.

4) Program for Sorting 'n' elements Using bubble sort technique.

5) Sort given elements using Selection Sort.

6) Sort given elements using Insertion Sort.

7) Sort given elements using Merge Sort.

8) Sort given elements using Quick Sort.

9) Implement the following operations on single linked list. (i) Creation (ii) Insertion (iii) Deletion (iv) Display

10) Implement the following operations on double linked list. (i) Creation (ii) Insertion (iii) Deletion (iv) Display

11) Implement the following operations on circular linked list. (i) Creation (ii) Insertion (iii) Deletion (iv) Display

12) Program for splitting given linked list.

13) Program for traversing the given linked list in reverse order.

14) Merge two given linked lists.

15) Implement Stack Operations Using Arrays.

16) Implement Stack Operations Using Linked List.

17) Implement Queue Operations Using Arrays.

18) Implement Queue Operations Using Linked List.

19) Implement Operations on Circular Queue.

20) Construct and implement operations on Priority Queue.

21) Implement Operations on double ended Queue.

22) Converting infix expression to postfix expression by using stack.

23) Write program to evaluate post fix expression.

24) Add two polynomials using Linked List.

25) Multiply Two polynomials using Linked List.

26) Construct BST and implement traversing techniques recursively.

27) Implement preorder traversal on BST non recursively.

28) Implement in order traversal on BST non recursively.

29) Implement post order traversal on BST non recursively.

30) Implement binary search techniques recursively.

Semester	MCA26107: DBMS LAB	L	T	P	C
I		0	0	6	3

Aim: Marketing Company wishes to computerize their operations by using following tables.

Table Name: Client Master

Description: This table stores the information about the clients.

Column Name	Data Type	Size	Attribute
Client_no	Varchar2	6	Primary Key and first letter should starts with 'C'
Name	Varchar2	10	Not null
Address1	Varchar2	10	
Address2	Varchar2	10	
City	Varchar2	10	
State	Varchar2	10	
Pincode	Number	6	Not null
Bal due	Number	10,2	

Table Name: Product master

Description: This table stores the information about products.

Column Name	Data Type	Size	Attribute
Product_no	Varchar2	6	Primary Key and first letter should starts with 'P'
Description	Varchar2	10	Not null
Profit_percent	Number	2,2	Not null
Unit_measure	Varchar2	10	
Qty_on_hand	Number	8	
Record_lvl	Number	8	
Sell_price	Number	8,2	Not null, can't be 0
Cost_price	Number	8,2	Not null, can't be 0

Table Name: salesman_master

Description: This table stores the salesmen working in the company

Column Name	Data Type	Size	Attribute
Salesman_id	Varchar2	6	Primary Key and first letter should starts with 'S'
Name	Varchar2	10	Not null
Address1	Varchar2	10	
Address2	Varchar2	10	
City	Varchar2	10	
State	Varchar2	10	
Pincode	Number	6	Not null
Sal_amt	Number	8,2	Should not null and zero
Target_amt	Number	6,2	Should not null and zero

Column Name	Data Type	Size	Attribute
Remarks	Varchar2	10	

Table Name: sales_order

Description: This table stores the information about orders

Column Name	Data Type	Size	Attribute
S_order_no	Varchar2	6	Primary Key and fisrt char is 'O'
S_order_date	Date		
Client_no	Varchar2	6	Foreign key
Delve_address	Varchar2	20	
Salesman_no	Varchar2	6	Foreign key
Delve_type	Varchar2	1	Delivery: part(P)/Full(F) and default 'F'
Billed_yn	Char	1	
Delve_date	Date		Can't be less than the s_order_date
Order_status	Varchar2	10	Values in 'IN PROCESS', 'FULFILLED', 'BACK ORDER', 'CANCELLED'

Table Name: sales_order_details

Description: This table stores the information about products ordered

Column Name	Data Type	Size	Attribute
S_order_no	Varchar2	6	Primary key, foreign key references sales_order table
Product_no	Varchar2	6	Primary key, foreign key references product_master table
Qty_ordered	Number	8	
Qty_disp	Number	8	
Product_rate	Number	10,2	

Table Name: challan_master

Description: This table stores the information about challans made for orders.

Column Name	Data Type	Size	Attribute
Challan_no	Varchar2	6	Primary key, first two letters must start with 'CH'
S_order_no	Varchar2	6	Foreign key references sales_order
Challan_date	Date		
Billed_yn	Char	1	Values in 'Y', 'N' default 'N'

Table Name: Challan_Details

Description: This table stores the information about challan detail.

Column Name	Data Type	Size	Attribute
Challan_no	Varchar2	6	Primary key, foreign key references challan_master table
Product_no	Varchar2	6	Primary key, foreign key references product_master table
Qty_disp	Number	4,2	Not null



SOLVE THE FOLLOWING QUERIES BY USING ABOVE TABLES.

1) Retrieve the list of names and cities of all the clients.
2) List the various products available from product_master.
3) Find out the clients who stay in a city whose second letter is 'a'.
4) Find the list of all clients who stay in the city ' CHENNAI' or 'DELHI'.
5) List all the clients located at 'CHENNAI'.
6) Print the information from sales order as the order the places in the month of January.
7) Find the products with description as 'Floppy Drive' and 'Pen drive'.
8) Find the products whose selling price is grater than 2000 and less than or equal to 5000.
9) Find the products whose selling price is more than 1500 and also find the new selling price as original selling price * 15.
10) Find the products in the sorted order of their description.
11) Divide the cost of product '540 HDD' by difference between its price and 100.
12) List the product number, description, sell price of products whose description begin with letter 'M'.
13) List all the orders that were cancelled in the month of March.
14) Count the total number of orders.
15) Calculate the average price of all the products.
16) Determine the maximum and minimum product prices.
17) Count the number of products having price grater than or equal to 1500.
18) Find all the products whose quantity on hand is less than reorder level.
19) Find out the challan details whose quantity dispatch is high.
20) Find out the order status of the sales order, whose order delivery is maximum in the month of March.
21) Find out the total sales made by the each salesman.
22) Find the total revenue gained by the each product sales in the period of Q1 and Q2 of year 2006.
23) Print the description and total qty sold for each product.
24) Find the value of each product sold.
25) Calculate the average qty sold for each client that has a maximum order value of 1,50,000.
26) List the products which has highest sales.
27) Find out the products and their quantities that will have to deliver in the current month.
28) Find the product number and descriptions of moving products.
29) Find the names of clients who have purchased 'CD DRIVE'.
30) List the product numbers and sales order numbers of customers having quantity ordered less than 5 from the order details for the product '1.44 Floppies'.
31) Find the product numbers and descriptions of non-moving products.
32) Find the customer names and address for the clients who placed the order '019001 '.

Semester	MCA26108: COMMUNICATION SKILLS	L	T	P	C
I		0	4	0	2

COURSE DESCRIPTION:

This course develops the communication skills required for academic work, professional interaction, and employment in the information technology industry. It focuses on listening, speaking, reading, writing, interpersonal communication, and presentation skills. Students practice communicating technical ideas clearly to technical and non-technical audiences through individual and collaborative activities.

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

1. **LO1:** Develop effective verbal, non-verbal, and written communication skills.
2. **LO2:** Strengthening listening comprehension and reading skills for academic and professional purposes.
3. **LO3:** Enable students to participate confidently in discussions, presentations, and interviews.
4. **LO4:** Develop professional correspondence and workplace communication skills.
5. **LO5:** Promote teamwork, cultural sensitivity, and responsible digital communication.

COURSE OUTCOMES (CO):

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

- **CO1:** Identify communication barriers and apply appropriate strategies to overcome them.
- **CO2:** Interpret spoken and written information and prepare accurate summaries.
- **CO3:** Communicate ideas clearly through presentations, discussions, and professional conversations.
- **CO4:** Prepare effective emails, résumés, application letters, and other workplace documents.
- **CO5:** Demonstrate appropriate interpersonal skills, professional etiquette, and teamwork.

Course Content:

UNIT-I: Fundamentals of Communication

Meaning, process, purpose, and types of communication; verbal and non-verbal communication; formal and informal communication; communication barriers and methods of overcoming them; audience and purpose; clarity, conciseness, correctness, completeness, and courtesy; body language, posture, gestures, facial expressions, and eye contact; cross-cultural communication and professional etiquette.

UNIT-II: Listening and Speaking Skills

Active listening; listening for main ideas and supporting details; note-taking; identifying assumptions and seeking clarification; pronunciation, word stress, intonation, fluency, and intelligibility; self-introduction; professional conversations; telephone and video-conference etiquette; explaining a technical concept; giving instructions; asking and answering questions; impromptu speaking.

UNIT-III: Reading and Professional Writing

Skimming, scanning, intensive reading, and critical reading; understanding technical articles and professional documents; vocabulary development in context; summarizing and paraphrasing; paragraph organization and coherence; appropriate tone and language; professional emails, requests, enquiries, application letters, résumés, meeting agendas, and minutes; proofreading and editing.

UNIT-IV: Interpersonal and Team Communication

Interpersonal communication; empathy and assertiveness; giving and receiving constructive feedback; group discussions; teamwork and collaborative problem-solving; negotiation and conflict resolution; communication with clients and colleagues; conducting meetings; communicating project progress; professional conduct in online collaboration and social media.

UNIT-V: Presentations and Employability Skills

Planning and organizing presentations; introduction, development, and conclusion; effective use of slides and visual aids; explaining software solutions and project demonstrations; adapting content to the audience; handling questions; managing presentation anxiety; interview preparation; technical and human-resource interviews; mock interviews; group-discussion practice.

SUGGESTED PRACTICAL ACTIVITIES

1. Deliver a self-introduction and a short professional introduction.
2. Listen to a technical talk and prepare a structured summary.
3. Explain a computing concept to a non-technical audience.
4. Participate in a group discussion on contemporary technology topic.
5. Prepare a professional email, résumé, and application letter.
6. Conduct a simulated project meeting and prepare its minutes.
7. Present a software application or academic project.
8. Participate in a mock interview and peer-feedback exercise.

SUGGESTED ASSESSMENT

Assessment may include listening and reading tasks, written assignments, group discussions, presentations, mock interviews, and a portfolio of professional communication activities. Evaluation should consider clarity, organization, language accuracy, audience awareness, participation, and professional conduct.

SEMESTER – II

Semester	MCA26201: SOFTWARE ENGINEERING	L	T	P	C
II		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	The need for software engineering, its different life cycles and different phases
LO2	Measuring cost, efforts, time and team management etc.,
LO3	Testing and maintenance techniques of big projects and
LO4	Different risks and its management systems
LO5	Learn about quality management.

COURSE OUTCOMES (CO):

At the end of the course, the student will be able to

CO1	Describe software engineering, layered technology and process framework
CO2	Evaluate the different process models and choose the best model for their project
CO3	Understand the different development practices and their advantages
CO4	Explain software testing approaches, software tactics and metrics for process and project domains
CO5	Analyze estimation techniques, quality management and formal methods

Course Content:

UNIT-I:

Introduction to Software Engineering: The Evolving Role of Software, Software, The Changing Nature of Software, Legacy Software: The Quality of legacy software, Software Evolution, Software Myths.

A Generic View of Process: Software Engineering-A Layered Technology, A Process Frame Work, The capability Maturity Model Integration (CMMI), Process Patterns, Process Assessment, Personal and Team Process Models: Personal Software Process (PSP), Team Software Process (TSP), Process Technology, Product and Process.

Process Models: Prescriptive Models, The Waterfall Model, Incremental Process Models: The Incremental Model, The RAD Model, Evolutionary Process Model: Prototyping, The Spiral Model, The Concurrent Development Model, Specialized Process Models: Component Based Development, The formal Methods Model, The Unified Process.

An Agile View of Process: What is Agility? What is Agile Process? Agile Process Models: Extreme Programming, Adaptive Software Development, Dynamic Systems Development Method, Scrum, Crystal, Feature Driven Development, Agile Modeling.

UNIT-II

Software Engineering Practice: Software Engineering Practice, communication practices, Planning Practices, Modeling Practices, Construction Practices, and Deployment.

System Engineering: Computer Based Systems, the System Engineering Hierarchy, Business Process Engineering: An Overview, System Modeling.

Building the Analysis Model: Requirement Analysis, Analysis Modeling Approaches, Data Modeling Concepts, Object Oriented Analysis, Scenario Based Modeling, Flow Oriented Modeling, Class Based Modeling, Creating a Behavioral Model.

Design Engineering: Design within the context of Software Engineering, Design Process and Design Quality, Design Concepts, The Design Model, Pattern Based Software Design.

UNIT-III

Testing Strategies: A strategic Approach to Software Testing, Strategic Issues, and Test Strategies for conventional Software, Testing Strategies for Object Oriented Software, Validation Testing, System Testing, the Art of Debugging.

Testing Tactics: Software Testing Fundamentals, Black Box and White Box Testing, White Box Testing, Basis Path Testing, Control Structure Testing, Black Box Testing, Object Oriented Testing Methods, Testing Methods Applicable at the class level, InterClass Test Case Design, Testing for Specialized Environments, Architectures and Applications, Testing Patterns.

Project Management: The Management Spectrum, the People, The Product, The Process, The Project, The W5HH Principles.

Metrics for Process and Projects: Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality, Integrating Metrics within Software Process, Metrics for Small Organizations, Establishing a Software Metrics Program.

UNIT-IV

Estimation: Observations on Estimations, The project planning process, Software Scope and Feasibility, Resources, Software Project Estimation, Decomposition Techniques, Empirical Estimation Models, Estimations for Object Oriented Projects, Specialized Estimation Techniques, The Make/Buy Decision

Quality Management: Quality Concepts, Software Quality Assurance, Software Reviews, Formal Technical Reviews, Formal Approaches to SQA, Statistical Software Quality Assurance, Software Reliability, The ISO 9000 Quality Standards, the SQA Plan

Formal Methods: Basic Concepts, Object Constraint Language (OCL), The Z specification language, The Ten Commandments for Formal Methods.

Cleanroom Software Engineering: The Cleanroom Approach, Functional Specification, Cleanroom Design, Cleanroom Testing.

PRESCRIBED TEXTBOOK:

1) Roger S Pressman, "Software Engineering–A Practitioner's Approach", Sixth Edition, TMH International.

REFERENCE BOOKS:

1) Sommerville, "Software Engineering", Seventh Edition Pearson Education (2007)

2) S.A.Kelkar, "Software Engineering – A Concise Study", PHI.

3) Waman S.Jawadekar, "Software Engineering", TMH.

4) Ali Behforooz and Frederick J.Hudson, "Software Engineering Fundamentals", Oxford (2008).

Semester	MCA26202: COMPUTER NETWORKS	L	T	P	C
II		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	Understanding the state-of-the-art in network protocols, architectures, and applications
LO2	Examining and studying different protocols in OSI and TCP/IP
LO3	Understanding network addressing, mapping etc.
LO4	Understanding error control, flow control, packet recovery etc.
LO5	Understanding internetworking of devices

COURSE OUTCOMES (CO):

At the end of the course, the student will be able to

CO1	Analyze basic taxonomy and terminology of the computer networking area.
CO2	Describe the configuration and design of a small network
CO3	Explain about research areas and future internets research fields
CO4	Learn components and rules of communications
CO5	Construct and implement layer protocols within an environment

Course Content:

UNIT-I

Introduction: Uses of Computer Networks: Business Application, Home Applications, Mobile Users – Social Issues. Network Hardware: Local Area Networks – Metropolitan Area Networks – Wide Area Networks – Wireless Networks – Home Networks – Internetworks. Network Software: Protocol Hierarchies – Design Issues for the Layers – Connection Oriented and Connectionless Services – Service Primitives – The relationship of Services to Protocols. Reference Models: The OSI Reference Model – The TCP/IP Reference Model – A Comparison of OSI and TCP/IP reference Model – A Critique of the OSI Model and Protocols – A Critique of the TCP/IP reference model. Example Networks: The Internet – Connection Oriented Networks: x.25, Frame Relay, and ATM – Ethernet – Wireless LANs Network Standardization: Who's who in the Telecommunication World – Who's who in the International Standards World – Who's who in the Internet Standards World?

Physical Layer: Guided Transmission Media: Magnetic Media – Twisted Pair – Coaxial Cable – Fiber Optics

Data Link Layer: Data Link Layer Design Issues: Services Provided to the Network Layer – Framing – Error Control – Flow Control. Error Detection and Correction: Error correcting Codes – Error Detecting Codes. Elementary Data Link Protocols: An unrestricted Simplex Protocol – A simplex Stop- and – wait Protocol – A simplex Protocol for a Noisy channel. Sliding Window Protocols: A one-bit sliding Window Protocol – A Protocol using Go Back N – A Protocol using selective Repeat. Example Data Link Protocols: HDLC – The Data Link Layer in the Internet.

UNIT-II

The Medium Access Control Sublayer: Ethernet: Ethernet Cabling – Manchester Encoding – The Ethernet MAC sublayer Protocol – The Binary Exponential Backoff Algorithm – Ethernet Performance – Switched Ethernet – Fast Ethernet – Gigabit Ethernet – IEEE 802.2: Logical Link Control – Retrospective on Ethernet.

Wireless Lans: The 802.11 Protocol Stack - The 802.11 Physical Layer - The 802.11 MAC sublayer Protocol - The 802.11 Frame Structure. Bluetooth: Bluetooth Architecture – Bluetooth Applications – The Bluetooth Protocol Stack – The Bluetooth Radio Layer – The Bluetooth Baseband Layer – The Bluetooth L2CAP layer – The Bluetooth Frame Structure. Data Link Layer Switching: Bridges from 802.x to 802.y – Local Internetworking – Spanning Tree Bridges – Remote Bridges – Repeaters, Hubs, Bridges, Switches, Routers and Gateways – Virtual LANs.

UNIT-III

The Network Layer: Network Layer Design Issues: Store – and Forward Packet Switching – Services provided to the Transport Layer – Implementation of Connectionless Services – Implementation of Connection Oriented Services – Comparison Of Virtual Circuit and Datagram subnets. Routing Algorithms: The Optimality Principle – Shortest Path Routing – Flooding – Distance Vector Routing – LinkState Routing – Hierarchical Routing – Broadcast Routing – Multicast Routing – Routing for Mobile Hosts. Internet Working: How Networks

Differ – How Networks can be connected – Concatenated Virtual Circuits – Connectionless Internetworking – Tunneling – Internet work Routing – Fragmentation. The Network Layer in the Internet: The IP Protocol – IP address – Internet Control Protocols – OSPF – The Internet Gateway Routing Protocol – BGP – The Exterior Gateway Routing Protocol.

UNIT-IV:

The Transport Layer: The Transport Service: Services provided to the Upper Layers – Transport Services Primitives – Berkeley Sockets. Elements of Transport Protocols: Addressing – Connection Establishment – Connection Release – Flow Control and Buffering – Multiplexing – Crash Recovery. The Internet Transport Protocols: UDP Introduction to UDP – Remote Procedure Call – The Real Time Transport Protocol. The Internet Transport Protocols: TCP Introduction to TCP – The TCP Service Model – the TCP Protocol – The TCP segment header – TCP connection establishment – TCP connection release – Modeling TCP connection management- TCP Transmission Policy – TCP congestion Control – TCP Timer Management – Wireless TCP and UDP – Transactional TCP.

UNIT-V:

The Application Layer: DNS: The Domain Name System: The DNS Name Space – Resource Records – Name Servers. Electronic Mail: Architecture and Services – The User Agent – Message Formats – Message Transfer – Final Delivery. The World Wide Web: Architecture Overview – Static Web Documents – Dynamic Web Documents – HTTP – The Hyper Text Transfer Protocol – Performance Enhancements – The Wireless Web. Multimedia: Introduction to Digital Audio – Audio Compression – Streaming Audio – Internet Radio – Voice Over IP – Introduction to Video – Video Compression – Video on Demand.

PRESCRIBED TEXTBOOK:

1) Andrew S. Tanenbaum, "Computer Networks", Fourth Edition, PHI.

REFERENCE BOOKS:

1) James F.Kurose, Keith W.Ross, "Computer Networking", Third Edition, Pearson Education

2) Behrouz A Forouzan, "Data Communications and Networking", Fourth Edition, TMH (2007)

3) Michael A. Gallo, William M. Hancock, "Computer Communications and Networking Technologies", Cengage Learning (2008)

Semester	MCA26203: WEB TECHNOLOGIES	L	T	P	C
II		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	Design and develop Web applications
LO2	Create web pages using HTML, DHTML and Cascading Styles sheets.
LO3	Analyze and build interactive web applications using JSP and Servlets.
LO4	Design and develop Web applications
LO5	Designing Enterprise based applications by encapsulating an application's business logic

COURSE OUTCOMES (CO):

At the end of the course, the student will be able to

CO1	Explain the technologies used in web applications.
CO2	Demonstrate HTML5, CSS, JavaScript coding for web applications
CO3	Design creative websites using object-based scripting concepts
CO4	Learn to access databases through Java programs, using Java Database Connectivity (JDBC)
CO5	Create dynamic web pages, using Servlets and JSP

Course Content:

UNIT I

Java Basics: Java buzzwords, Review of OOP concepts, dynamic binding, abstract classes and methods, interfaces, Packages.

GUI Programming with JAVA: Event Handling, Applets, Swing - Introduction to Swing, Swing vs. AWT,MVC architecture, Hierarchy for Swing components, Containers, JFrame, JApplet, JWindow, JDialog, JPanel, A simple swing application, Overview of several swing components, Layout management - Layout manager types - border, grid, flow, box.

UNIT II

HTML: Common Tags: List, Tables, images, forms, Frames, Cascading Style Sheets;

Java Script: Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script.

XML: Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX

UNIT III

JDBC: Introduction to JDBC - Connections - Internal Database Connections - Statements - Results Sets - Prepared Statements - Callable Statements.

Network Programming and RMI: why networked Java - Basic Network Concepts - looking up Internet Addresses - URLs and URIs - UDP Datagrams and Sockets - Remote Method Invocation.

Unit -IV

Web Servers and Servlets: Tomcat web server, Introduction to Servlets: Lifecycle of a Servlet, JSDK, The Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization parameters. The javax.servlet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues.

Introduction to JSP: The Problem with Servlet. The Anatomy of a JSP Page, JSP Processing. JSP Application Design with MVC Setting Up and JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat

PRESCRIBED TEXTBOOKS:

1) The Complete Reference: Java, Herbet Schildt, 7th Edition, McGraw Hill.
2) Java Programming with JDBC; Donald Bales, O'Reilly
3) Web Technologies - a computer science perspective, Jeffrey C. Jackson, Pearson, 2007.

REFERENCE BOOKS:

1) Java Network Programming, elliotte Rusty Harold, 3rd Edition
2) Java Server Pages - Hans Bergsten, SPD O'Reilly
3) Robert W. Sebesta, "Programming the World Wide Web", Third Edition, Pearson Education (2007).
4) Anders Moller and Michael schwartzbach, "An Introduction to XML and Web Technologies", Addison Wesley (2006)
5) Chris Bates, "Web Programming-Building Internet Applications", Second Edition, Wiley (2007).

Semester	MCA26204: RESEARCH METHODOLOGY & INTELLECTUAL PROPERTY RIGHTS (IPR)	L	T	P	C
II		3	1	0	4

Course Description and Purpose:

The aim of this course is to develop research bent of mind (spirit of inquiry) and impart research skills to all Post graduate students. It also encompasses the series of research methodology contents: from problem formulation, to design, to data collection, analysis, reporting and dissemination. This course also covers intellectual property rights (IPR) and is intended to equip students with conceptual understanding of current scenario of IPR, and the practical issues encountered in filing patents, trademarks and copyrights.

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

- LO1: To understand some basic concepts of research and its methodologies
- LO2: To develop an understanding of the basic framework of research processes.
- LO3: To develop an understanding of various research designs and techniques.
- LO4: To identify various sources of information for literature review and data collection.
- LO5: Ability to write a research Proposal, report and thesis
- LO6: To demonstrate knowledge and understanding of IPR Filing and Rights

COURSE OUTCOMES (CO):

Upon successful completion of the course, the student will be able to:

- CO-1: Understand some basic concepts of research and its methodologies
- CO-2: Identify appropriate research topics
- CO-3: Select and define appropriate research problems and parameters
- CO-4: Demonstrate the ability to choose methods appropriate to research aims and objectives
- CO-5: Have adequate knowledge on measurement & scaling techniques
- CO-6: Have basic awareness of data analysis-and hypothesis testing procedures

Course Content:

UNIT I

Foundations of Research

Meaning of Research – Definitions of Research – Motivation in Research – General Characteristics of Research– Criteria of Good Research – Types of Research – Research Process – Research Methods vs. Methodology – Defining and Formulating the Research Problem – Review of Literature – Approaches to Critical Literature Review – Importance of Literature Review in Identifying Research Gaps and Defining a Problem – Development of Working Hypothesis.

UNIT II

Research Design, Sampling Concepts, and Data Collection Methods

Meaning, Significance and Characteristics of Good Research Design – Types of Research Design: Exploratory, Conclusive Research and Experimental – Sampling Theory: Types of Sampling and Errors in Sampling – Data Collection: Types of Data – Data Collection Methods and Techniques for Primary and Secondary Data.

UNIT III

Measurement & Scaling Techniques, Hypothesis Formulation and Testing, Overview of Data Analysis and Report Writing

Basic measurement scales – Reliability & Validity – Definition and Types of Hypothesis – Hypothesis Formulation and Testing Procedure – Overview of Data Analysis: Methods, Process and Types – Report Writing: Significance of Report Writing, Different Steps in Writing Report, Layout of the Research Report, Types of Reports, Oral Presentation, Mechanics of Writing a Research Report, Precautions for Writing Research Reports – How to Write a Research Proposal, Research Ethics, Conflict of Interest and Plagiarism.

UNIT IV

Intellectual Property Rights (IPR)

Definition and Nature and Features of Intellectual Property Rights (IPR) – Types of Intellectual Property Rights – Procedure for Grants of Patents – Rights of a Patent – Scope of a Patent Rights

–Licensing and Transfer of Technology – Why protection of intellectual property is important?

–Enforcement of IPR – Infringement of IPR.

UNIT V

Indian and International Scenario and New Developments in IPR

IPR Developments in India for the past Five Years – Development of IPR Laws in India – International Cooperation on IPR – New Developments in IPR – Administration of Patent System –International Patent protection – Case Studies in Indian and Global Contexts.

PRESCRIBED TEXTBOOKS:

- Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002, An introduction to Research Methodology, RBSA Publishers.
- Cohen, L. Lawrence, M., & Morrison, K. (2005), Research Methods in Education (5thedition). Oxford: Oxford University Press.
- Kothari, C.R., 1990, Research Methodology: Methods and Techniques, New Age International.
- Dornyei, Z. (2007). Research Methods in Applied Linguistics. Oxford: Oxford University Press.

Anthony, M., Graziano, A.M. and Raulin, M.L., 2009, Research Methods: A Process of Inquiry, Allyn and Bacon.

Semester	MCA26205A: PROGRAMMING AND PROBLEM-SOLVING USING PYTHON	L	T	P	C
II		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	Install and run the Python interpreter
LO2	Create and execute Python programs
LO3	Understand the concepts of file I/O
LO4	Be able to read data from a text file using Python
LO5	Plot data using appropriate Python visualization libraries

COURSE OUTCOMES (CO):

At the end of the course, the student will be able to

CO1	Demonstrate understanding of modern version control tools.
CO2	Exhibit facility with a Linux command line environment.
CO3	Demonstrate understanding of the role of testing in scientific computing and write unit tests in Python.
CO4	Use command line tools to write and edit code.
CO5	Develop publication-ready graphics from a dataset.

Course Content:

UNIT I

Introduction: The Process Of Computational Problem Solving, Python Programming Language. Python Data Types: Expressions, Variables and Assignments, Strings, List, Objects and Classes, Python Standard Library

Imperative Programming: Python programs, Execution Control Structures, User-Defined Functions, Python Variables and Assignments, Parameter Passing.

UNIT II

Text Files: Strings, Formatted Output, Files, Errors and Exception Handling

Execution and Control Structures: if Statement, for Loop, Two Dimensional Lists, while Loop, More Loop Patterns, Additional Iteration Control Statements

Containers and Randomness: Dictionaries, Other Built-in Container Types, Character Encoding and Strings, Module random, Set Data Type.

UNIT III

Object Oriented Programming: Fundamental Concepts, Defining a New Python Class, User-Defined Classes, Designing New Container Classes, Overloaded Operators, Inheritance, User-Defined Exceptions

Namespaces: Encapsulation in Functions, Global versus Local Namespaces, Exception Control Flow, Modules and Namespaces.

Objects and Their Use: Software Objects, Turtle Graphics, Modular Design: Modules, Top-Down Design, Python Modules

Recursion: Introduction to Recursion, Examples of Recursion, Run Time Analysis, Searching, Iteration Vs Recursion, Recursive Problem Solving, Functional Language Approach.

UNIT IV

Graphical User Interfaces: Basics of tkinter GUI Development, Event-Based tkinter Widgets, Designing GUIs, OOP for GUI,

The Web and Search: The World Wide Web, Python WWW API, String Pattern Matching, Database Programming in Python

PRESCRIBED TEXTBOOK:

1) Ljubomir Perkovic, "Introduction to Computing Using Python: An Application Development Focus", Wiley, 2012.

REFERENCE BOOKS:

1) Charles Dierbach, "Introduction to Computer Science Using Python: A Computational Problem-Solving Focus", Wiley, 2013.

Semester	MCA26205B: INTRODUCTION TO R PROGRAMMING	L	T	P	C
II		3	1	0	4

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

LO1	The basics of statistical computing and data analysis
LO2	How to implement data structure in R
LO3	R loop functions and debugging tools
LO4	Lists and Vectors concepts in R
LO5	Factors and levels in R

COURSE OUTCOMES (CO):

At the end of the course, the student will be able to

CO1	Explain critical R programming concepts
CO2	Demonstrate how to install and configure RStudio
CO3	Apply OOP concepts in R programming
CO4	Explain the use of data structure and loop functions
CO5	Apply various concepts to write programs in R

Course Content:

UNIT I

Introduction: Overview of R, R data types and objects, reading and writing data, sub setting R Objects, Essentials of the R Language, Installing R, Running R, Packages in R, Calculations, Complex numbers in R, Rounding, Arithmetic, Modulo and integer quotients, Variable names and assignment, Operators, Integers, Factors, Logical operations

UNIT II

Control structures, functions, scoping rules, dates and times, Introduction to Functions, preview of Some Important R Data Structures, Vectors, Character Strings, Matrices, Lists, Data Frames, Classes.

Vectors: Generating sequences, Vectors and subscripts, Extracting elements of a vector using subscripts, Working with logical subscripts, Scalars, Vectors, Arrays, and Matrices, Adding and Deleting Vector Elements, Obtaining the Length of a Vector, Matrices and Arrays as Vectors Vector Arithmetic and Logical Operations, Vector Indexing, Common Vector Operations

UNIT III

Lists: Creating Lists, General List Operations, List Indexing Adding and Deleting List Elements, Getting the Size of a List, Extended Example: Text Concordance Accessing List Components and Values Applying Functions to Lists, Data Frames, Creating Data Frames, Accessing Data Frames, Other Matrix-Like Operations

UNIT IV

Factors and Levels, Common Functions Used with Factors, Working with Tables, Matrix/Array-Like Operations on Tables, Extracting a Sub table, Finding the Largest Cells in a Table, Math Functions, Calculating a Probability, Cumulative Sums and Products, Minima and Maxima, Calculus, Functions for Statistical Distributions

PRESCRIBED TEXTBOOKS:

1) Roger D. Peng, "R Programming for Data Science", 2012

2) Norman Matloff, "The Art of R Programming- A Tour of Statistical Software Design", 2011

REFERENCE BOOKS:

1) Jared P. Lander, "R for Everyone - advanced analytics and Graphics", 2nd Edition, Addison-Wesley.

2) Paul Teetor, "R Cookbook", 2nd Edition, O'Reilly publications.

3) Garrett Grolemund, Hadley Wickham, "Hands-On Programming with R: Write Your Own Functions and Simulations", 1st Edition, 2014.

Semester	MCA26206: PYTHON PROGRAMMING LAB	L	T	P	C
II		0	0	6	3

LAB CYCLE

SIMPLE PROGRAMS

1) Write a program using print Pascal triangle.
2) Write a program to find out the roots of the quadratic equations.
3) Write a program to display the Fibonacci series using generators.
4) Write a program to check the given number is palindrome or not.
5) Write a program to find the sum of digits of a given number
6) Write a Python program to calculate $X = \frac{1}{2!} + \frac{2}{4!} + \frac{4}{8!} + \frac{8}{16!}$
7) Write a Python program to remove the punctuations from a string.
8) Write a Python program to implement the simple calculator.
9) Write a Python program to print the lower and upper triangles of a matrix.
10) Write a Python program to merge two mails.

FUNCTIONS

1) Write a recursive Python function that has a parameter representing a list of integers and returns the maximum stored in the list.
2) Write a recursive Python function to that generates the top n prime numbers in the range 1 to 1000.
3) Write a python function to calculate the multiplication of two matrices.
4) Write a Python function to reverse the given string.
5) Write a Python function that takes an integer n and a character c, returns a string and displays as “xxxxx” (Ex: the length of the retuned string is 5, then the output as XXXXX)
6) Write Python function that the search the given number in the list of numbers by using binary search.
7) Write a Python function to convert the given decimal number into binary number by using recursion.
8) Write a Python function to sort the list of records in a file.

GUI PROGRAMS

1) Construct a GUI application to generate the employee pay slip
2) Construct a GUI application to generate a Bar Graph for a excel data
3) Construct a GUI application to perform the Arithmetic operations
4) Read Input Values through input window
5) Choose choice and Operation through following windows
6) Display the result in Message Box.

Semester	MCA26206: R PROGRAMMING LAB	L	T	P	C
II		0	0	6	3

1) Download and install R-Programming environment and install basic packages using install.packages() command in R.

2) Learn all the basics of R-Programming (Data types, Variables, Operators etc.)

3) Implement R-Loops with different examples.

4) Learn the basics of functions in R and implement with examples.

5) Implement data frames in R. Write a program to join columns and rows in a data frame using cbind() and rbind() in R.

6) Implement different String Manipulation functions in R.

7) Implement different data structures in R (Vectors, Lists, Data Frames)

8) Write a program to read a csv file and analyze the data in the file in R

9) Create pie charts and bar charts using R.

10) Create a data set and do statistical analysis on the data using R.

Semester	MCA26207: WEB TECHNOLOGIES LAB	L	T	P	C
II		0	0	6	3

1) Write a Java Program to define a class, describe its constructor, overload the constructors and instantiate its object
2) Build and run "Celsius Converter" sample application using swings
3) Develop an applet that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked
4) Develop and demonstrate a HTML document that illustrates the use of external style sheets, ordered list, table, borders, padding, color, and the tag.
5) Create a form with the following specifications:
   a) Our form uses frames, one to hold the links bar at the top of the browser window.
   b) Other is a larger frame that provides the main view.
   c) The links bar should contain 5 links, which when clicked, should display the appropriate HTML file in the larger frame.
6) Create a webpage with the following using html
   a) Embed an image in web page
   b) Fix the hot spots
   c) Show all the related information when a hot spot is clicked on the map
7) Write a JavaScript code to find factorial of N. (Use recursive function)
8) Write a JavaScript code block using arrays and generate the current date in words, this should include the day, month and year.
9) Create a web page using two image files, which switch between one another as the mouse pointer moves over the images. Use the onMouseOver and onMouseOut event handlers.
10) Design an XML document to store information about a student in an engineering college affiliated to ANU. The information must include college id, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
11) Create an XML document, which contains 10 users' information. Implement a program, which takes User Id as an input and returns the user details by taking the user information from the XML document
12) write a java program to illustrate java to database connectivity using JDBC
13) Write a program to print Fibonacci numbers using RMI.
14) write a java servlet program to conduct online examinations and to display student mark list available in a database
15) Create a java program to create an airline reservation service and a travel agent and the travel agent is searching for an airline using web services and database.

Semester	MCA26208: SEMINAR	L	T	P	C
II		0	4	0	2

COURSE DESCRIPTION:

The Seminar provides an opportunity for students to study a focused topic in computer applications, examine relevant literature, and present their understanding through a technical report and oral presentation. It develops independent learning, critical analysis, technical communication, and the ability to respond to questions using evidence.

The seminar should go beyond reproducing information from a single source. Students should compare approaches, identify applications and limitations, and explain the significance of the selected topic.

LEARNING OBJECTIVES (LO):

1. LO1: Encourage independent study of contemporary and emerging areas in computer applications.
2. LO2: Develop the ability to locate, evaluate, and synthesize technical literature.
3. LO3: Strengthen analytical thinking and evidence-based comparison.
4. LO4: Improve technical presentation, report-writing, and discussion skills.
5. LO5: Promote academic integrity and responsible use of information sources.

COURSE OUTCOMES (CO):

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

- **CO1:** Select and define a relevant technical topic with an appropriate scope.
- **CO2:** Identify and critically evaluate credible literature and technical resources.
- **CO3:** Compare methods, technologies, applications, and limitations related to the topic.
- **CO4:** Prepare a structured seminar report with appropriate citations and references.
- **CO5:** Deliver an organized technical presentation and defend its content during discussion.

INDICATIVE TOPIC AREAS

Topics may be selected from Artificial Intelligence, Machine Learning, Data Science, Generative AI, Natural Language Processing, Computer Vision, Cybersecurity, Cloud Computing, Distributed Systems, Internet of Things, Software Engineering, Database Technologies, Human–Computer Interaction, or other relevant areas approved by the Department.

Examples include explainable machine learning, retrieval-augmented generation, privacy-preserving analytics, cloud-native applications, secure software development, edge computing, and responsible AI.

SEMINAR PROCEDURE

1. **Topic Selection:** Each student shall propose a focused topic in consultation with a faculty supervisor. The topic should be appropriate to postgraduate study and sufficiently distinct from routine classroom coverage.
2. **Proposal Submission:** The student shall submit a brief proposal stating the topic, rationale, objectives, scope, and initial references.
3. **Literature Study:** The student shall consult relevant research papers, books, standards, official documentation, and other credible sources.
4. **Progress Review:** The student shall present progress to the supervisor and incorporate feedback.

5. **Analysis and Preparation:** The student shall organize the study around key concepts, comparisons, applications, limitations, and possible future directions.
6. **Report Submission:** A structured seminar report shall be submitted in the prescribed departmental format.
7. **Presentation and Discussion:** The student shall deliver an oral presentation followed by questions and discussion.

SUGGESTED SEMINAR REPORT STRUCTURE

- Title page
- Abstract
- Introduction and motivation
- Objectives and scope
- Background and essential concepts
- Review of relevant literature
- Technical discussion and comparative analysis
- Applications, limitations, and challenges
- Conclusions and future directions
- References
- Appendices, where necessary

EXPECTED DELIVERABLES

A topic proposal, literature summary, seminar report, presentation slides, and an oral presentation. A small demonstration, experiment, or case study may be included where appropriate; original research results are not mandatory.

SUGGESTED ASSESSMENT

Evaluation may consider the relevance and scope of the topic, quality of literature review, depth of understanding, critical analysis, report organization, citation practices, presentation quality, responses to questions, and completion of progress reviews.

ACADEMIC INTEGRITY

All borrowed ideas, text, figures, tables, and data shall be acknowledged. Any use of AI-assisted tools shall comply with University policy. Students remain responsible for verifying sources, checking factual accuracy, and explaining the submitted work independently.



Semester	MCA26209: SWAYAM COURSES	L	T	P	C
II		0	0	0	2*

*__NOTE:__ The credits assigned for SWAYAM Courses (MOOCs Mandatory Course) will not be counted for the GPA.

COURSE DESCRIPTION

This course component develops practical and employment-oriented skills through approved SWAYAM Plus courses. Preference shall be given to offerings related to Artificial Intelligence, Machine Learning, Data Science, Data Analytics, and associated software tools.

Students shall undertake a course that provides suitable practical learning through exercises, case studies, projects, or tool-based activities. The emphasis is on applying knowledge to clearly defined problems and demonstrating the skills acquired.

LEARNING OBJECTIVES (LO):

The course is designed to meet the objectives of:

1. LO1: Develop practical skills relevant to contemporary computing roles.
2. LO2: Strengthen the application of AI, ML, and Data Science tools.
3. LO3: Introducing students to professional workflows and problem-solving practices.
4. LO4: Encourage the creation of demonstrable work and learning portfolios.
5. LO5: Promote responsible, continuous, and employment-oriented learning.

COURSE OUTCOMES (CO):

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

- **CO1:** Use the tools and techniques introduced in the selected course.
- **CO2:** Complete relevant practical exercises or case studies.
- **CO3:** Apply acquired skills to a defined computing or data-analysis problem.
- **CO4:** Evaluate the quality, limitations, and responsible use of the resulting solution.
- **CO5:** Demonstrate learning through a portfolio, report, presentation, or viva voce.

RECOMMENDED AREAS FOR COURSE SELECTION

1. **Applied AI and ML:** Implementing and evaluating machine learning solutions using appropriate programming tools.
2. **Data Analytics and Visualization:** Preparing datasets, analyzing information, creating visualizations, and communicating findings.
3. **Python for AI and Data Science:** Practical programming, data processing, and use of analytical libraries.
4. **Generative AI Applications:** Effective use and evaluation of generative tools, output verification, and responsible application development.
5. **Applied Deep Learning:** Introductory practical work in image, text, or other suitable data domains.
6. **Deployment and Data Workflows:** Introductory model deployment, cloud-based analytical workflows, or related practices appropriate to the student's level.

The detailed instructional content shall be determined by the approved course. The Department should favor offerings with clear learning outcomes, suitable practical content, and verifiable assessment or completion requirements.

COURSE SELECTION AND APPROVAL

- Students shall select a course with prior departmental approval.
- The selected course should match the students' prerequisites and the intended learning outcomes of the MCA programme.
- Course workload, provider, assessment arrangements, certificate requirements, software access, and any applicable fees shall be reviewed before approval.
- Where both SWAYAM and SWAYAM Plus components occur in the programme, students should choose complementary courses.
- The same learning activity or certificate shall not be counted twice.

SUGGESTED PRACTICAL WORK

Depending on the selected course, students may undertake one of the following:

- Prepare and analyze a dataset using Python.
- Create an analytical dashboard and explain its findings.
- Build and evaluate a classification or regression model.
- Compare alternative machine learning approaches.
- Develop a small AI-enabled application.
- Demonstrate a text-analysis or image-analysis workflow.
- Document a practical case study using relevant tools.

Where predictive modelling is involved, students should use an appropriate evaluation method and avoid data leakage. Work involving personal or sensitive information shall follow applicable institutional requirements.

EXPECTED DELIVERABLES

- Evidence of enrolment and successful completion
- A brief description of the skills acquired
- Practical exercises, notebooks, source code, dashboards, or other relevant artefacts
- A short application report explaining the problem, method, results, and limitations
- A demonstration or viva-voce as prescribed by the Department

Where the selected course does not include a practical submission itself, the Department may prescribe a proportionate application exercise aligned with its learning outcomes.

ASSESSMENT AND RECOGNITION

Evaluation should consider successful completion of the approved course and the student's ability to demonstrate the skills acquired. Any departmental assessment, grade conversion, or credit recognition shall follow the University's approved scheme. A provider certificate shall not automatically establish equivalence to an MCA credit-bearing course.

RECOMMENDED COURSES AND LEARNING RESOURCES

Students are encouraged to select courses related to Artificial Intelligence, Machine Learning, Data Science, and their programming or mathematical foundations. The following courses are indicative options, subject to availability and prior departmental approval:

1. **Python for Data Analytics, Machine Learning and Generative AI: A Comprehensive Guide**
   Recommended for developing foundations in Python programming, data analysis, machine learning, and generative AI.
   Course link: https://onlinecourses.swayam2.ac.in/imb26_mg130/preview
2. **Data Science Using Python**
   Recommended for learning data preparation, analysis, visualization, and modelling using Python.
   Course link: https://onlinecourses.swayam2.ac.in/ini24_cs02/preview
3. **SWAYAM Course Catalogue**
   Students may identify other suitable courses in Artificial Intelligence, Machine Learning, Data Science, Deep Learning, Natural Language Processing, Computer Vision, Probability, Statistics, and related areas.
   Course catalogue: https://swayam.gov.in/search_courses

SELECTION AND COMPLETION GUIDELINES

The listed links refer to course offerings and may remain accessible after enrolment closes. The Department shall verify the current offering, prerequisites, syllabus, workload, assessment arrangements, and certification requirements before approving enrolment. Where a listed course is unavailable, an equivalent course in an approved subject area may be selected.

Learning materials, assignments, readings, and references prescribed by the selected course shall form the primary learning resources. Students shall complete the requirements specified by the course provider and submit the evidence prescribed by the Department. Credit recognition, where applicable, shall follow the University's approved regulations.