



# **Eklavya University**

**SESSION**

**2023-24**

**M.C.A. II SEMESTER**

**SYLLABUS**

**OF**

**Computer Application And Information  
Technology Department**

**School of Basic and Applied Sciences**

# EKLAHYA UNIVERSITY, DAMOH (M.P.)

## Scheme of Examination MCA II Sem

For batch admitted in Academic Session 2023-24

### Subject wise distribution of marks and corresponding credits

S.No.	Subject Code	Subject Name	Maximum Marks Allotted					Total Marks	Contact Periods Per week			Total Credits
			Theory Slot			Practical Slot			L	T	P	
			End Sem.	Mid term Examination	Quiz/ Assignment/ Attendance	End Sem	Lab Work/ sessional					
1	MCAPL20S201	OBJECT ORIENTED PROGRAMMING	60	30	10	-	-	100	3	1	-	4
2	MCAPL20S202	OPERATING SYSTEMS	60	30	10	-	-	100	3	1	-	4
3	MCAPL20S203	DATABASE MANAGEMENT SYSTEMS	60	30	10	-	-	100	3	1	-	4
4	MCAPL20S204	DATA STRUCTURES & ANALYSIS OF ALGORITHMS	60	30	10	-	-	100	3	1	-	4
5	MCAPL20S205	CYBER SECURITY	60	30	10	-	-	100	3	1	-	4
6	MCAPL20S206	OBJECT ORIENTED PROGRAMMING LAB	-	-	-	120	80	200	-	-	8	8
7	MCAPL20S207	DBMS AND DS LAB	-	-	-	30	20	50	-	-	2	2
<b>Total</b>			300	150	50	150	100	750	15	5	10	30

Induction programme of three weeks (MC): Physical activity, Creative Arts, Universal Human Values, Literary, Proficiency Modules, Lectures by Eminent People, Visits to local Areas, Familiarization to Dept./Branch & Innovations.

*CD*  
01/06/23  
Nidhi

*Murli*  
01/08/23

*RS*  
01/06/23

*Prakruti*  
01/06/23

*Prabhu*  
01/06/23

*Deepak*  
1/6/23

<b>Class</b>		<b>Master of Computer Application (MCA) I YEAR</b>	
<b>Semester/Year</b>		<b>II Semester</b>	
<b>Subject &amp; Subject Code</b>		<b>OBJECT ORIENTED PROGRAMMING - MCAPL20S201</b>	
<b>Max. Marks</b>		<b>60</b>	
<b>Credit</b>		<b>Total Credits</b>	
L	T	P	4
3	1	0	

**Course Objectives:**

1. To Implement Object Oriented Programming.
2. To Learn the Syntax and Semantics of the C++ Programming Language.
3. To Learn how to Implement Copy Constructors and Class Member Functions.
4. To Learn how to Overload Functions and Operators in C++.
5. To Learn how Containment and Inheritance Promote Code Reuse in C++.
6. To Learn how to Use Exception Handling in C++ Programs.

**Course Outcome:**

1. Explain Concepts and Advantages of Object Oriented Programming.
2. Apply and implement the concepts of the Object -Oriented paradigms to analyze, design and develop the solutions of real world problems using the Principles of information Hiding.
3. Demonstrate the Advanced Features of C++ Specifically Stream I/O, , Templates and Operator Overloading and overriding

**Student Learning Outcomes (SLO):**

1. To understand how C++ improves C with object-oriented features.
2. To learn how to write inline functions for efficiency and performance.
3. To learn the syntax and semantics of the C++ programming language.
4. To learn how to design C++ classes for code reuse.

Unit	Syllabus	Periods
UNIT - I	Introduction: Object Oriented Programming: objects, classes, Abstraction, Encapsulation, Inheritance, Polymorphism, OOP in Java, Characteristics of Java, The Java Environment, Java Source File Structure, and Compilation. Fundamental Programming Structures in Java: Defining classes in Java, constructors, methods, access specifiers, static members, Comments, Data Types, Variables, Operators, Control Flow, Arrays.	8
UNIT - II	Inheritance, Interfaces, and Packages: Inheritance: Super classes, sub classes, Protected members, constructors in sub classes, Object class, abstract classes and methods. Interfaces: defining an interface, implementing interface, differences between classes and interfaces and extending interfaces, Object cloning, inner classes. Packages: Defining Package, CLASSPATH Setting for Packages, Making JAR Files for Library Packages, Import and Static Import Naming Convention For Packages, Networking java.net package.	8
UNIT - III	Exception Handling, I/O: Exceptions: exception hierarchy, throwing and catching exceptions, built-in exceptions, creating own exceptions, Stack Trace Elements. Input / Output Basics: Byte streams and Character streams, Reading and Writing, Console Reading and Writing Files.	8

*Shel*  
01/06/23

*WMP*  
01.06.23

*Deep*  
1/6/23

*Prashant*  
01/06/23  
Nidhi

UNIT - IV	Multithreading and Generic Programming: Differences between multi-threading and multitasking, thread life cycle, creating threads, synchronizing threads, Inter-thread communication, daemon threads, thread groups. Generic Programming: Generic classes, generic methods, Bounded Types: Restrictions and Limitations.	8
UNIT - V	Event Driven Programming: Graphics programming: Frame, Components, working with 2D shapes, Using colors, fonts, and images. Basics of event handling: event handlers, adapter classes, actions, mouse events, AWT event hierarchy. Introduction to Swing: layout management, Swing Components: Text Fields, Text Areas, Buttons, Check Boxes, Radio Buttons, Lists, choices, Scrollbars, Windows Menus and Dialog Boxes.	8

**Text Books :-**

- 1 Herbert Shield, "C++ The Complete Reference "TMH Publication
- 2 Ashok Kamthane , Object-Oriented Programming with ANSI and Turbo C++
- 3 R. Lafore, 'Object Oriented Programming C++

**References Books:**

- 1 Herbert Schildt, "Java The complete reference", McGraw Hill Education, 8th Edition, 2011.
- 2 Cay S. Horstmann, Gary Cornell, "Core Java Volume -I Fundamentals", Prentice Hall,
- 3 Steven Holzner, "Java Black Book", Dreamtech.
- 4 Balagurusamy E, " Programming in Java", McGraw Hill
- 5 Balagurusamy E, " Programming in Java", McGraw Hill
- 6 Khalid Mughal, "A Programmer's Guide to Java SE 8 Oracle Certified Associate (OCA) AddisonWesley.

*[Signature]*  
01/06/23

*[Signature]*  
01-06-23

*[Signature]*  
01/06/23

*[Signature]*  
01/06/23

*[Signature]*  
01/06/23

*[Signature]*

(2)

*[Signature]*  
1/6/23

*[Faint signatures and dates at the bottom of the page]*

<b>Class</b>		<b>Master of Computer Application (MCA) I YEAR</b>	
<b>Semester/Year</b>		<b>II Semester</b>	
<b>Subject &amp; Subject Code</b>		<b>OPERATING SYSTEMS - MCAPL20S202</b>	
<b>Max. Marks</b>		<b>60</b>	
<b>Credit</b>		<b>Total Credits</b>	
<b>L</b>	<b>T</b>	<b>P</b>	<b>4</b>
<b>3</b>	<b>1</b>	<b>0</b>	
<b>Course Objectives:</b>			
<ol style="list-style-type: none"> <li>1. To Understand the Services Provided by Operating System</li> <li>2. To Understand the Working and Organization of Process and its Scheduling and Synchronization.</li> <li>3. To Understand the Concept of Deadlock.</li> <li>4. To Understand Different Approaches to Memory Management Techniques.</li> <li>5. To Understand the Structure and Organization of the File System</li> </ol>			
<b>Course Outcome:</b>			
<ol style="list-style-type: none"> <li>1. Identify and describe the Services Provided by Operating Systems.</li> <li>2. Understand and Solve Problems Involving Process Control, Mutual Exclusion Synchronization and Deadlock.</li> <li>3. Apply Various Approaches of Memory Management Techniques</li> <li>4. Analyze Various Operating System Approaches in Linux and Windows</li> </ol>			
<b>Student Learning Outcomes (SLO):</b>			
<ol style="list-style-type: none"> <li>1. High-level understand what is an operating system and the role it plays.</li> <li>2. A high-level understanding of the structure of operating systems, applications, and the relationship between them.</li> <li>3. Some knowledge of the services provided by operating systems.</li> <li>4. Exposure to some details of major OS concepts.</li> </ol>			
<b>Unit</b>	<b>Syllabus</b>		<b>Periods</b>
UNIT - I	Introduction: Operating System Structure- Layered structure, System Components, Operating system functions, Classification of Operating systems- Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multi process Systems, Multithreaded Systems, Operating System services, Reentrant Kernels, Monolithic and Microkernel Systems.		8
UNIT - II	Concurrent Processes: Process Concept, Principle of Concurrency, Producer / Consumer Problem, Mutual Exclusion, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, Test and Set operation, Classical Problem in Concurrency- Dining Philosopher Problem, Sleeping Barber Problem, Inter Process Communication models and Schemes, Process generation.		8
UNIT - III	CPU Scheduling: Scheduling Concepts, Performance Criteria, Process States, Process Transition Diagram, Schedulers, Process Control Block (PCB), Process address space, Process identification information, Threads and their management, Scheduling Algorithms, Multiprocessor Scheduling. Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.		8

*at*  
01/06/23

*Shah*  
01/06/23

*MM*  
01.06.23

(3)

*PSM*  
01/06/23

*Sharma*  
01/06/23

*Chokhane*  
01/06/23

*Nidhi*

UNIT - IV	Memory Management: Basic bare machine, Resident monitor, Multiprogramming with fixed partitions, Multiprogramming with variable partitions, Protection schemes, Paging, Segmentation, Paged segmentation, Virtual memory concepts, Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing, Cache memory organization, Locality of reference.	8
UNIT - V	I/O Management and Disk Scheduling: I/O devices, and I/O subsystems, I/O buffering, Disk storage and disk scheduling, RAID. File System: File concept, File organization and access mechanism, File directories, and File sharing, File system implementation issues, File system protection and security.	8

**Text Books : -**

- 1 William Stallings, "Computer Organization and Architecture", Prentice Hall of India, Sixth Edition.
- 2 A. Tannenbaum, "Structured Computer Organization", Pearson Education, 2002.
- 3 Patterson & Hennessy, "Computer Organization and Design", Morgan Kaufmann, 2007

**References Books:**

- 1 Silberschatz, Galvin and Gagne, "Operating Systems Concepts", Wiley Publication.
- 2 Sibsankar Halder and Alex A Arvind, "Operating Systems", Pearson Education.
- 3 Harvey M Dietel, "An Introduction to Operating System", Pearson Education.
- 4 William Stallings, "Operating Systems: Internals and Design Principles", 6th Edition, Pearson Education.
- 5 Harris, Schaum's Outline Of Operating Systems, McGraw Hill

Handwritten signatures and dates in blue ink:

- Signature: [unclear], Date: 01/06/23

④

Faint handwritten signatures and dates at the bottom of the page, including "01/06/23" and "01/06/23".

<b>Class</b>		<b>Master of Computer Application (MCA) I YEAR</b>	
<b>Semester/Year</b>		<b>II Semester</b>	
<b>Subject &amp; Subject Code</b>		<b>DATABASE MANAGEMENT SYSTEMS - MCAPL20S203</b>	
<b>Max. Marks</b>		<b>60</b>	
<b>Credit</b>		<b>Total Credits</b>	
<b>L</b>	<b>T</b>	<b>P</b>	<b>4</b>
<b>3</b>	<b>1</b>	<b>0</b>	
<b>Course Objectives:</b>			
<ol style="list-style-type: none"> <li>1. To create and manipulate a database using SQL.</li> <li>2. To know database administration basics and practice commands.</li> <li>3. To Understand File Organization and Indexing in Database.</li> <li>4. To Acquire the Knowledge of Query Evaluation to Monitor the</li> <li>5. To Know Parallel, Distributed, Object relational and XML database basics.</li> </ol>			
<b>Course Outcome:</b>			
<ol style="list-style-type: none"> <li>1. Understand and describe the basic concepts and terminology of Database Management System</li> <li>2. Apply query language commands using MySQL.</li> <li>3. Understand internal storage mechanism, File Organization and Indexing in Database.</li> <li>4. Know Basic Concepts in parallel, distributed, object relational and XML database.</li> </ol>			
<b>Student Learning Outcomes (SLO):</b>			
<ol style="list-style-type: none"> <li>1. have a broad understanding of database concepts and database management system software</li> <li>2. have a high-level understanding of major DBMS components and their function.</li> <li>3. be able to model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model.</li> <li>4. be able to write SQL commands to create tables and indexes, insert/update/delete data, and query data in a relational DBMS.</li> <li>5. be able to program a data-intensive application using DBMS APIs.</li> </ol>			
<b>Unit</b>	<b>Syllabus</b>		<b>Periods</b>
UNIT - I	Introduction: Overview, Database System vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence and Database Language and Interfaces, Data Definitions Language, DML, Overall Database Structure. Data Modeling Using the Entity Relationship Model: ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Extended ER Model, Relationship of Higher Degree.		8

*[Signature]*  
01/08/23

*[Signature]*  
01/08/23

*[Signature]*  
01/08/27

*[Signature]*  
01/08/23

*[Signature]*  
01/08/23

*[Signature]*  
01/08/23

*[Signature]*

UNIT - II	Relational data Model and Language: Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra, Relational Calculus, Tuple and Domain Calculus. Introduction to SQL: Characteristics of SQL, Advantage of SQL. SQL Data Type and Literals. Types of SQL Commands. SQL Operators and their Procedure. Tables, Views and Indexes. Queries and Sub Queries. Aggregate Functions. Insert, Update and Delete Operations, Joins, Unions, Intersection, Minus, Cursors, Triggers, Procedures in SQL/PL SQL	8
UNIT - III	Data Base Design & Normalization: Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependence, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design	8
UNIT - IV	Transaction Processing Concept: Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Checkpoints, Deadlock Handling. Distributed Database: Distributed Data Storage, Concurrency Control, Directory System	8
UNIT - V	Concurrency Control Techniques: Concurrency Control, Locking Techniques for Concurrency Control, Time Stamping Protocols for Concurrency Control, Validation Based Protocol, Multiple Granularity, Multi Version Schemes, Recovery with Concurrent Transaction, Case Study of Oracle.	8

**Text Books : -**

- 1 Fundamentals of Database Systems 7th Edition
- 2 Database Management Systems, 3rd Edition
- 3 Database System Concepts 7th Edition

**References Books:**

- 1 Korth, Silbertz, Sudarshan, " Database Concepts", McGraw Hill.
- 2 Date C J, "An Introduction to Database Systems", Addison Wesley.
- 3 Elmasri, Navathe, "Fundamentals of Database Systems", Addison Wesley.
- 4 O'Neil, "Databases", Elsevier Pub.
- 5 Ramakrishnan, "Database Management Systems", McGraw Hill.
- 5 Leon & Leon, "Database Management Systems", Vikas Publishing House.
- 7 Bipin C. Desai, " An Introduction to Database Systems", Gagotia Publications.
- 8 Majumdar & Bhattacharya, "Database Management System", McGraw Hill.

*[Handwritten signature]*  
01/06/23

*[Handwritten signature]*

6

<b>Class</b>		<b>Master of Computer Application (MCA) I YEAR</b>	
<b>Semester/Year</b>		<b>II Semester</b>	
<b>Subject &amp; Subject Code</b>		<b>DATA STRUCTURES &amp; ANALYSIS OF ALGORITHMS - MCAPL20S204</b>	
<b>Max. Marks</b>		<b>60</b>	
<b>Credit</b>		<b>Total Credits</b>	
<b>L</b>	<b>T</b>	<b>P</b>	<b>4</b>
<b>3</b>	<b>1</b>	<b>0</b>	

**Course Objectives:**

1. Understand the concepts of data structures and algorithm design methods.
2. Choose the appropriate data structure such as arrays, linked lists, stacks, queues, hash tables, trees, heaps and graphs and writing programs for these solutions.
3. Solve problems of Searching and Sorting.
4. Understand the complexity of algorithm.

**Course Outcome:**

1. Understand the concept and usage of data types, dynamic memory management and data structures.
2. Understand various data structures like arrays, linked lists, stacks, queues, hash tables, trees, heaps and graphs.
3. Apply algorithms for Searching and Sorting Methods.
4. Develop and analyze Algorithms and calculate their Complexities.
5. Understand the applications of various data structures.

**Student Learning Outcomes (SLO):**

1. Designs and analyzes simple algorithms.
2. Defines the meaning of iterative and recursive algorithms.
3. Calculates the running time of iterative algorithms.
4. Discusses basic ADTs such as stacks, queues, and trees
5. Discusses simple hashing schemes for searching.

<b>Unit</b>	<b>Syllabus</b>	<b>Periods</b>
UNIT - I	Introduction to data structure: Data, Entity, Information, Difference between Data and Information, Data type, Build in data type, Abstract data type, Definition of data structures, Types of Data Structures: Linear and Non-Linear Data Structure, Introduction to Algorithms: Definition of Algorithms, Difference between algorithm and programs, properties of algorithm, Algorithm Design Techniques, Performance Analysis of Algorithms, Complexity of various code structures, Order of Growth, Asymptotic Notations. Arrays: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D, 2-D Array Application of arrays, Sparse Matrices and their representations. Linked lists: Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition Subtraction & Multiplications of Single variable.	8

*Handwritten signature and date: 01/08/23*

*Handwritten signature and date: 01/08/23*

*Handwritten signature and date: 01/06/23*

*Handwritten signature and date: 01/06/23*

*Handwritten signature and date: 01/06/23*

UNIT - II	Stacks: Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Iteration and Recursion- Principles of recursion, Tail recursion, Removal of recursion Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. Queues: Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue. Searching: Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing.	8
UNIT - III	Sorting: Insertion Sort, Selection Sort, Bubble Sort, Heap Sort, Comparison of Sorting Algorithms, Sorting in Linear Time: Counting Sort and Bucket Sort. Graphs: Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List, Adjacency. Graph Traversal: Depth First Search and Breadth First Search, Connected Component.	8
UNIT - IV	Trees: Basic terminology used with Tree, Binary Trees, Binary Tree Representation: Array Representation and Pointer (Linked List) Representation, Binary Search Tree, Complete Binary Tree, A Extended Binary Trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Constructing Binary Tree from given Tree Traversal, Operation of Insertion, Deletion, Searching & Modification of data in Binary Search Tree. Threaded Binary trees, Huffman coding using Binary Tree, AVL Tree and B Tree.	8
UNIT - V	Divide and Conquer with Examples Such as Merge Sort, Quick Sort, Matrix Multiplication: Strassen's Algorithm Dynamic Programming: Dijkstra Algorithm, Bellman Ford Algorithm, Allpair Shortest Path: Warshal Algorithm, Longest Common Sub-sequence Greedy Programming: Prims and Kruskal algorithm.	8

**Text Books :-**

- 1 Data Structure and Algorithm
- 2 Introduction to Algorithms by Thomas H. Cormen
- 3 Data Structure Using "C"

**References Books:**

- 1 Cormen T. H., Leiserson C. E., Rivest R. L., and Stein C., "Introduction to Algorithms", PHI.
- 2 Horowitz Ellis, Sahni Sartaj and Rajasekharan S., "Fundamentals of Computer Algorithms", 2nd Edition, Universities Press.
- 3 Dave P. H., H.B. Dave, "Design and Analysis of Algorithms", 2nd Edition, Pearson Education.
- 4 Lipschutz S., "Theory and Problems of Data Structures", Schaum's Series.
- 5 Goyal K. K., Sharma Sandeep & Gupta Atul, "Data Structures and Analysis of Algorithms", HP Hamilton.
- 6 Lipschutz, Data Structures With C - SIE - SOS, McGraw Hill
- 7 Samanta D., "Classic Data Structures", 2nd Edition Prentice Hall India.
- 8 Reema Thareja, Data Structures using C, Oxford Univ. Press

*Nishu*

*01/06/23*

*01/06/23*

**8**

*01/06/23*

*01/06/23*

*01/06/23*



UNIT - III	Introduction to E-Commerce , Threats to E-Commerce, Electronic Payment System, e- Cash, Credit/Debit Cards. Digital Signature, Cryptography Developing Secure Information Systems, Application Development Security, Information Security Governance & Risk Management, Security Architecture & Design Security Issues in Hardware, Data Storage & Downloadable Devices, Physical Security of IT Assets - Access Control, CCTV, Backup Security Measures.	8
UNIT - IV	Security Policies- Why policies should be developed, Policy Review Process, Publication and Notification Requirement of policies, Types of policies – WWW policies, Email Security policies, Corporate Policies, Sample Security Policies. Case Study – Corporate Security	8
UNIT - V	Information Security Standards-ISO, IT Act, Copyright Act, IPR. Cyber Crimes , Cyber Laws in India; IT Act 2000 Provisions, Intellectual Property Law, Copy Right Law , Semiconductor Law and Patent Law , Software Piracy and Software License.	8

**Text Books :-**

- 1 Mayank Bhushan, Rajkumar Singh Rathore, Aatif Jamshed, Fundamentals of Cyber Security
- 2 Mark Merkow , Information Security: Principles and Practices, Pearson Education
- 3 Information Security Policy & Implementation Issues, NIIT, PHI

**References Books:**

- 1 The Art of Exploitation (2nd Ed.)
- 2 The Art of Invisibility
- 3 The Code Book
- 4 Practical Malware Analysis

*[Signature]*  
6/06/23

*[Signature]*  
1/6/23

*[Signature]*  
01/06/23

*[Signature]*  
01/06/23

*[Signature]*  
01/06/23  
N-dh

Class		Master of Computer Application (MCA) I YEAR	
Semester/Year		SEMESTER — II	
Subject & Subject Code		OBJECT ORIENTED PROGRAMMING – MCAPL20S206	
Max. Marks		120 (ETE) + 80 (IA) = 200	
Credit		Total Credits	
L	T	P	8
0	0	8	

**PRACTICAL LIST**

1. Use Java compiler and eclipse platform to write and execute java program.
2. Creating simple java programs,
3. Understand OOP concepts and basics of Java programming.
4. Create Java programs using inheritance and polymorphism.
5. Implement error-handling techniques using exception handling and multithreading.
6. Understand the use of java packages.
7. File handling and establishment of database connection.
8. Develop a calculator application in java.
9. Develop a Client Server Application.
10. Develop GUI applications using Swing components.

1/6/23  
 1/6/23  
 01/06/23  
 01/06/23  
 01/06/23  
 11

<b>Class</b>		<b>Master of Computer Application (MCA) I YEAR</b>	
<b>Semester/Year</b>		<b>SEMESTER — II</b>	
<b>Subject &amp; Subject Code</b>		<b>DBMS AND DS – MCAPL20S207</b>	
<b>Max. Marks</b>		<b>30 (ETE) + 20 (IA) = 50</b>	
<b>Credit</b>		<b>Total Credits</b>	
<b>L</b>	<b>T</b>	<b>P</b>	<b>2</b>
<b>0</b>	<b>0</b>	<b>2</b>	

**PRACTICAL LIST**

**DBMS**

1. Installing oracle/ MYSQL.
  2. Creating Entity-Relationship Diagram using case tools.
  3. Writing SQL statements Using ORACLE /MYSQL:
    - a. Writing basic SQL SELECT statements.
    - b. Restricting and sorting data.
    - c. Displaying data from multiple tables.
    - d. Aggregating data using group function.
    - e. Manipulating data.
    - F. Creating and managing tables.
  4. Normalization.
  5. Creating cursor.
  6. Creating procedure and functions.
  7. Creating packages and triggers.
  8. Design and implementation of payroll processing system.
  9. Design and implementation of Library Information System.
  10. Design and implementation of Student Information System.
- Automatic Backup of Files and Recovery of Files.

**DS**

Program in C or C++ for following:

1. To implement addition and multiplication of two 2D arrays.
2. To transpose a 2D array.
3. To implement stack using array
4. To implement queue using array.
5. To implement circular queue using array.
6. To implement stack using linked list.
7. To implement queue using linked list.
8. To implement BFS using linked list.
9. To implement DFS using linked list.
10. To implement Linear Search.
11. To implement Binary Search.
12. To implement Bubble Sorting.
13. To implement Selection Sorting.
14. To implement Insertion Sorting.
15. To implement Merge Sorting.
16. To implement Heap Sorting.
17. To implement Matrix Multiplication by strassen's algorithm 18. Find Minimum Spanning Tree using Kruskal's Algorithm

Handwritten signatures and dates at the bottom of the page, including dates like 01/10/23, 1/6/23, and 01/06/23.