# DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY
# UTTAR PRADESH, LUCKNOW

## EVALUATION SCHEME & SYLLABUS
## First Year
## FOR

# MASTER OF COMPUTER APPLICATIONS (MCA)
# (Two Year Course)

## Based On
## NEP2020
## (Effective from the Session: 2024-25)

# MCA (MASTER OF COMPUTER APPLICATIONS)
## MCA FIRST YEAR, 2024-25

## SEMESTER-I

| S.No. | Subject Code | Subject Name | Periods | | | Sessional | | | ESE | Total | Credit |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | CT | TA | Total | | | |
| 1. | BMC101 | Fundamental of Computers & Emerging Technologies | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 2. | BMC102 | Problem Solving using C | 3 | 1 | 0 | 20 | 10 | 30 | 70 | 100 | 4 |
| 3. | BMC103 | Principles of Management & Communication | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 4. | BMC104 | Discrete Mathematics | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 5. | BMC105 | Computer Organization & Architecture | 3 | 1 | 0 | 20 | 10 | 30 | 70 | 100 | 4 |
| 6. | BMC151 | Problem Solving using C **Lab** | 0 | 0 | 4 | 30 | 20 | 50 | 50 | 100 | 2 |
| 7. | BMC152 | Computer Organization & Architecture **Lab** | 0 | 0 | 3 | 30 | 20 | 50 | 50 | 100 | 2 |
| 8. | BMC153 | Professional Communication **Lab** | 0 | 0 | 2 | 30 | 20 | 50 | 50 | 100 | 2 |
| 9. | BMC106 | Cyber Security* | 2 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 0 |
| | | **Total** | | | | | | **330** | **570** | **900** | **23** |

**CT**: Class Test **TA**: Teacher Assessment **L/T/P**: Lecture/ Tutorial/ Practical

*** Qualifying Non-credit Course**

## SEMESTER-II

| S.No. | Subject Code | Subject Name | Periods | | | Sessional | | | ESE | Total | Credit |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | CT | TA | Total | | | |
| 1. | BMC201 | Web Technology | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 2. | BMC202 | Object Oriented Programming | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 3. | BMC203 | Operating Systems | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 4. | BMC204 | Database Management Systems | 3 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 3 |
| 5. | BMC205 | Data Structures & Analysis of Algorithms | 4 | 0 | 0 | 20 | 10 | 30 | 70 | 100 | 4 |
| 6. | BMC251 | Web Technology **Lab** | 0 | 0 | 3 | 30 | 20 | 50 | 50 | 100 | 2 |
| 7. | BMC252 | Object Oriented Programming **Lab** | 0 | 0 | 3 | 30 | 20 | 50 | 50 | 100 | 2 |
| 8. | BMC253 | DBMS **Lab** | 0 | 0 | 2 | 30 | 20 | 50 | 50 | 100 | 1 |
| 9. | BMC254 | Data Structures & Analysis of Algorithms **Lab** | 0 | 0 | 3 | 30 | 20 | 50 | 50 | 100 | 2 |
| 10. | BVA251 | Sports and Yoga****** | 0 | 0 | 3 | - | 100 | - | - | - | 0 |
| | | **Total** | | | | | | **350** | **550** | **900** | **23** |

**CT**: Class Test **TA**: Teacher Assessment **L/T/P**: Lecture/ Tutorial/ Practical

****Non-credit Course**

# Syllabus

# MCA 1st Year
# IInd Semester

# MCA (MASTER OF COMPUTER APPLICATIONS)
# FIRST YEAR SYLLABUS
# SEMESTER-II

| BMC201 : WEB TECHNOLOGY | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to:** | | |
| CO 1 | Apply the knowledge of HTML and CSS to develop web application and analyze the insights of internet programming to implement complete application over the web. | K3, K6 |
| CO 2 | Understand, analyze and apply the role of JavaScript in the workings of the web and web applications. | K2, K3 |
| CO 3 | Understand, analyze and build dynamic web applications using servlet and JSP. | $K_2$, $K_3$ |
| CO 4 | Develop Spring-based Java applications using Java configuration, XML configuration, annotation-based configuration, beans and their scopes, and properties. | $K_2$, $K_4$, $K_6$ |
| CO 5 | Develop web application using Spring Boot and RESTful Web Services | $K_3$, $K_6$ |
| **DETAILED SYLLABUS** | | **3-1-0** |
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Web Page Designing:** Introduction and Web Development Strategies, History of Web and Internet, Protocols Governing Web, HTML-Introduction, HTML Tags, HTML-Grouping Using Div & Span, HTML-Lists, HTML-Images, HTML- Hyperlink, HTML-Table, HTML-Iframe, HTML-Form, Introduction of CSS, CSS Syntax, External Style Sheet using < link >, Multiple Style Sheets, Value Lengths and Percentages, CSS-Selectors, CSS-Box Model, Floats, Clear, Introduction to Bootstrap. | **08** |
| **II** | **Scripting:** Introduction to JavaScript, Creating Variables in JavaScript, Creating Functions in JavaScript, UI Events, Returning Data from Functions, Working with Conditions, looping in JavaScript, Block Scope Variables, Working with Objects, Creating Object using Object Literals, Manipulating DOM Elements with JavaScript | **08** |
| **III** | **Web Application development using JSP & Servlets:** Servlet Overview and Architecture, Interface Servlet and the Servlet Life Cycle, Handling HTTP get Requests, Handling HTTP post Requests, Redirecting Requests to Other Resources, Session Tracking, Cookies, Session Tracking with Http Session. Java Server Pages (JSP): Introduction, Java Server Pages Overview, A First Java Server Page Example, Implicit Objects, Scripting, Standard Actions, Directives, Custom Tag Libraries. | **08** |

| | | |
|---|---|---|
| **IV** | **Spring:** Spring Core Basics-Spring Dependency Injection concepts, Introduction to Design patterns, Factory Design Pattern, Strategy Design pattern, Spring Inversion of Control, AOP, Bean Scopes-Singleton, Prototype, Request, Session, Application, WebSocket, Auto wiring, Annotations, Life Cycle Call backs, Bean Configuration styles | **08** |
| **V** | **Spring Boot:** Spring Boot- Spring Boot Configuration, Spring Boot Annotations, Spring Boot Actuator, Spring Boot Build Systems, Spring Boot Code Structure, Spring Boot Runners, Logger, BUILDING RESTFUL WEB SERVICES, Rest Controller, Request Mapping, Request Body, Path Variable, Request Parameter, GET, POST, PUT, DELETE APIs, Build Web Applications | **08** |

**Suggested Readings:**


1. Burdman J., "Collaborative Web Development – Strategies and Best practices for Web Teams", Addison-Wesley.
2. Xavier C, "Web Technology & Design", New Age international Publishers.
3. Bayross I., "Web Technologies", BPB Publications.
4. Schieldth H., "The Complete Reference – HTML & CSS", McGraw Hill.
5. Bergsten H., "Java Server Pages", SPD O' Reilly.
6. Walls C., "Spring Boot in Action", Manning Publications.
7. Bakliwal S., "Hands-on Application Development using Spring Boot", BPB Publications.

| BMC202 : OBJECT ORIENTED PROGRAMMING | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO 1 | List the significance and key features of object-oriented programming and modeling using UML | $K_4$ |
| CO 2 | Construct basic structural, behavioral and architectural models using object oriented software engineering approach. | $K_6$ |
| CO 3 | Integrate object-oriented modeling techniques for analysis and design of a system. | $K_4$, $K_5$ |
| CO 4 | Use the basic features of data abstraction and encapsulation in C++ programs. | $K_4$ |
| CO 5 | Use the advanced features such as Inheritance, polymorphism and virtual function in C++ programs. | $K_3$, $K_4$ |
| **DETAILED SYLLABUS** | | **3-1-0** |
| **Unit** | **Topic** | **Proposed Lecture** |
| **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 specifies, static members, Comments, Data Types, Variables, Operators, Control Flow, Arrays. | **08** |
| **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. | **08** |
| **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. | **08** |
| **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. | **08** |
| **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. | **08** |

**Suggested Readings:**

1. Schildt H., "Java - The complete Reference", McGraw Hill Education.
2. Horstmann C. S. and Cornell G., "Core Java Volume I Fundamentals", Prentice Hall.
3. Holzner S., “Java - Black Book”, Dreamtech.
4. Balagurusamy E., “Programming in Java”, Tata McGraw Hill.
5. Mughal K., “A Programmer’s Guide to Java SE 8 Oracle Certified Associate (OCA)”, Addison-Wesley.

| **BMC203 : OPERATING SYSTEMS** | | |
|---|---|---|
| **Course Outcome (**CO**)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO 1 | Explain main components, services, types and structure of Operating Systems. | $K_2$ |
| CO 2 | Apply the various algorithms and techniques to handle the various concurrency control issues. | $K_3$ |
| CO 3 | Compare and apply various CPU scheduling algorithms for process execution. | $K_2$ |
| CO 4 | Identify occurrence of deadlock and describe ways to handle it. | $K_3$ |
| CO 5 | Explain and apply various memory, I/O and disk management techniques. | $K_5$ |
| **DETAILED SYLLABUS** | | **3-0-0** |
| **Unit** | **Topic** | **Proposed Lecture** |
| **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. | **08** |
| **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. | **08** |
| **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. | **08** |
| **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. | **08** |
| **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. | **08** |

**Suggested Readings:**

1. Silberschatz A., Galvin P. B. and Gagne G., "Operating Systems Concepts", Wiley Pub.
2. Halder S. and Arvind A. A "Operating Systems", Pearson Education.
3. Dietel H. M, " An Introduction to Operating System", Pearson Education.
4. Stallings W., "Operating Systems: Internals and Design Principles", Pearson Education.
5. Harris J.A., "Operating Systems (Schaum's Outlines)", McGraw Hill Education.

| **BMC204 : DATABASE MANAGEMENT SYSTEMS** | | |
|---|---|---|
| **Course Outcome (**CO**)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO 1 | Describe the features of a database system and its application and compare various types of data models. | $K_2$ |
| CO 2 | Construct an ER Model for a given problem and transform it into a relation database schema. | $K_5$, $K_6$ |
| CO 3 | Formulate solution to a query problem using SQL Commands, relational algebra, tuple calculus and domain calculus. | $K_5$, $K_6$ |
| CO 4 | Explain the need of normalization and normalize a given relation to the desired normal form. | $K_2$, $K_3$ |
| CO 5 | Explain different approaches of transaction processing and concurrency control. | $K_2$ |
| **DETAILED SYLLABUS** | | **3-0-0** |
| **Unit** | **Topic** | **Proposed Lecture** |
| **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. | **08** |
| **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 | **08** |
| **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 | **08** |
| **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 | **08** |
| **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 | **08** |

| | | |
|---|---|---|
| | Transaction, Case Study of Oracle. | |

**Suggested Readings:**


1. Silbertschatz A., Korth H. and Sudarshan S., “Database Concepts”, McGraw Hill.
2. Date C. J., “An Introduction to Database Systems”, Addision Wesley.
3. Elmasri R. and Navathe S., “Fundamentals of Database Systems”, Pearson Education.
4. O’Neil P., "Databases", Elsevier Publications.
5. Ramakrishnan R. Gehrke J., "Database Management Systems", McGraw Hill.
6. Leon A. and Leon M., “Database Management Systems”, Vikas Publishing House.
7. Desai B.C., “An Introduction to Database Systems”, Galgotia Publications.
8. Majumdar A. K. and Bhattacharya P., “Database Management System”, Tata McGraw Hill.

| **BMC205 : DATA STRUCTURES & ANALYSIS OF ALGORITHMS** | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO 1 | Explain the concept of data structure, abstract data types, algorithms, analysis of algorithms and basic data organization schemes such as arrays and linked lists. | $K_2$ |
| CO 2 | Describe the applications of stacks and queues and implement various operations on them using arrays and linked lists. | $K_3$ |
| CO 3 | Describe the properties of graphs and trees and implement various operations such as searching and traversal on them. | $K_3$ |
| CO 4 | Compare incremental and divide-and-conquer approaches of designing algorithms for problems such as sorting and searching. | $K_4$ |
| CO 5 | Apply and analyze various design approaches such as Divide-and-Conquer, greedy and dynamic for problem solving. | $K_4$ |
| **DETAILED SYLLABUS** | | **4-0-0** |
| **Unit** | **Topic** | **Proposed Lecture** |
| **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.<br>**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.<br>**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. | **08** |
| **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.<br>**Queues:** Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.<br>**Searching:** Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & | **08** |

| | | |
|---|---|---|
| | Collision resolution Techniques used in Hashing. | |
| **III** | **Sorting:** Insertion Sort, Selection Sort, Bubble Sort, Heap Sort, Comparison of Sorting Algorithms, Sorting in Linear Time: Counting Sort and Bucket Sort.<br>**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. | **08** |
| **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. | **08** |
| **V** | Divide and Conquer with Examples Such as Merge Sort, Quick Sort, Matrix Multiplication: Strassen's Algorithm Dynamic Programming: Dijikstra Algorithm, Bellman Ford Algorithm, All- pair Shortest Path: Warshal Algorithm, Longest Common Sub-sequence Greedy Programming: Prims and Kruskal algorithm. | **08** |

**Suggested Readings:**


1. Cormen T. H., Leiserson C. E., Rivest R. L. and Stein C., "Introduction to Algorithms", PHI.
2. Horowitz E., Sahni S. and Rajasekharan S., "Fundamentals of Computer Algorithms", Universities Press.
3. Dave P. H. and Dave H. B, "Design and Analysis of Algorithms", Pearson Education.
4. Lipschuts S., "Theory and Problems of Data Structures Schaum's Series)", Tata McGraw-Hill.
5. Goyal K. K., Sharma S. and Gupta A., "Data Structures and Analysis of Algorithms", HP Hamilton.
6. Samanta D., "Classic Data Structures", Prentice Hall India.
7. Goodrich M. T. and Tomassia R., "Algorithm Design: Foundations, Analysis and Internet Examples", John Wiley and sons.
8. Sridhar S., "Design and Analysis of Algorithms", Oxford Univ. Press.
9. Aho A., Ullman J. and Hopcroft J., "Design and Analysis of algorithms", Pearson Education.
10. Neapolitan R. and K. Naimipour, "Foundations of Algorithms", Jones and Bartlett Student Edition.

| **BMC251 : WEB TECHNOLOGY LAB** | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO1 | Design web pages using HTML, DHTML and Cascading Styles sheets. | $K_2$ |
| CO2 | Develop a dynamic web pages using JavaScript. | $K_3$ |
| CO3 | Develop an interactive web applications using JSP. | $K_3$ |
| CO4 | Create web applications using Spring Boot. | $K_2$ |

1. Create a simple webpage using HTML.
2. Create a HTML page, which has properly aligned paragraphs with image along with it.
3. Write a program to display list of items in different styles.
4. Use frames to Include Images and Videos.
5. Add a Cascading Style sheet for designing the web page.
6. Design a dynamic web page with validation using JavaScript.
7. Write a program using JavaScript to demonstrate the concept of built-in array methods.
8. Write a program using JavaScript to demonstrate the concept of nested functions.
9. Write programs using JavaScript for Web Page to display browsers information.
10. Write a program using JavaScript to merge property of two objects.
11. Write a program using JavaScript to include a JS file into another JS file.
12. Develop a Servlet to validate user name and password stored in database. Display authorized user is she/he is authorized else display unauthorized user.
13. Write JSP & Servlet program to store student details sent from registration form in to database table.
14. Write appropriate JSP pages to insert, update and delete data in student table in a single application with proper linking of JSP pages and session management.
15. Write a java program/servlet application to connect to a database and extract data from the table containing employee's information and display them.
16. Write program to demonstrate the concept of spring and spring boot.
17. Create REST Service for an Education Site.
18. Use the Spring Boot Starter Web to Create a Web Application.

**Note: The instructor may add/delete/modify experiments, wherever he/she feels in a justified manner.**

| BMC252 : OBJECT ORIENTED PROGRAMMING LAB | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO1 | Use the Concept of Data Abstraction and Encapsulation in C++ programs. | $K_3$ |
| CO2 | Design and Develop C++ program using the concept such as polymorphism, virtual function, exception handling and template. | $K_3$ |
| CO3 | Apply object-oriented techniques to analyze, design and develop a complete solution for a given problem. | $K_3$ |

1. Use Java compiler and eclipse platform to write and execute java program.
2. Write programs to implement conditional statements and looping constructs.
3. Write programs to implement basic input / output operations.
4. Write programs using inheritance and polymorphism.
5. Write programs to implement error-handling techniques using exception handling and multithreading.
6. Write programs to demonstrate the use of java packages.
7. Write programs to demonstrate the concept of file handling and establishment of database connection.
8. Write program to develop a calculator application in Java.
9. Write program to develop a Client Server Application.
10. Write program to develop GUI applications using Swing components.

**Note: The instructor may add/delete/modify experiments, wherever he/she feels in a justified manner.**

| BMC253 : DATABASE MANAGEMENT SYSTEMS LAB | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO1 | Use the Concept of Data Abstraction and Encapsulation in C++ programs. | $K_6$ |
| CO2 | Write SQL commands to query a database. | $K_3$ |
| CO3 | Write PL/SQL programs for implementing stored procedures, stored functions, cursors, trigger and packages. | $K_6$ |

1. Install oracle/ MYSQL.
2. Create Entity-Relationship Diagram using case tools.
3. Write SQL statements Using ORACLE /MYSQL:
    a) Write basic SQL SELECT statements.
    b) Restrict and sort data.
    c) Display data from multiple tables.
    d) Aggregate data using group function.
    e) Manipulate data.
    f) Create and manage tables.
4. Normalization.
5. Create cursor.
6. Create procedure and functions.
7. Create packages and triggers.
8. Design and implement payroll processing system.
9. Design and implement Library Information System.
10. Design and implement Student Information System.
11. Automatic Backup of Files and Recovery of Files

**Note: The instructor may add/delete/modify experiments, wherever he/she feels in a justified manner.**

| **BMC254 : DATA STRUCTURES & ANALYSIS OF ALGORITHMS LAB** | | |
|---|---|---|
| **Course Outcome (CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course, the student will be able to** | | |
| CO1 | Write and execute programs to implement various searching and sorting algorithms. | $K_3$ |
| CO2 | Write and execute programs to implement various operations on two-dimensional arrays. | $K_3$ |
| CO3 | Implement various operations of Stacks and Queues using both arrays and linked lists data structures. | $K_3$ |
| CO4 | Implement graph algorithm to solve the problem of minimum spanning tree | $K_3$ |

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.

**Note: The instructor may add/delete/modify experiments, wherever he/she feels in a justified manner.**

## BVA251 : SPORTS AND YOGA

**Objective of the Course:**

- To maintain mental and physical wellness upright and develop ability in the students to cope up with the stress arising in the life.
- To create space in the curriculum to nurture the potential of the students in sports/games/yoga etc.
- To introduce a practice oriented introductory course on the subject. More involved / advanced course may come up in subsequent years of study

**Syllabus/ Guidelines**

**Part A: Sports/Games**

Some form of Athletics would be compulsory for all students, unless restricted due to medical / physical reasons. In addition to this, student has to opt for at least one game out of the remaining mentioned below.

A fair theoretical knowledge and a reasonable amount of field / site practice of the chosen games will be essential.

1. Athletics **Compulsory**
2. Volleyball
3. Basketball
4. Handball
5. Football
6. Badminton
7. Kabaddi
8. Kho-kho
9. Table tennis
10. Cricket

**Part B: Yoga**

**a. Introduction of Yoga**

Introduction of Yoga, Origin of Yoga, Aims and Objective of Yoga, Patanjali Yoga darshan, Hath yoga, Gheranda Samhita, Karm yoga, Gyan yoga.

**b. Asanas, Pranayam and Meditation Practices**

Meaning of Asanas, Objective of Asanas, rules and regulations of Asanas and Pranayams, Types of Yogasana

Yogic postures: Standing Posture, Sitting posture, Supine posture, Prone posture, balancing Postures, Pranayam according to Patanjali and Hath Yoga, Meditaion Mudras

**c. Science of Yoga**

Physiological effects of Asanas- Paranayama and meditation, stress management and yoga, Mental health and yoga practice, Health and Personality Development.

**General Guidelines**

1. Institutes must assign minimum of three periods in the Time Table for the activities of Sports/Yoga.
2. Institutes must provide field/facility and offer a minimum of five choices of the Games/Sports.
3. Institutes are required to provide sports instructor / yoga teacher to mentor the students.
4. Student must be made familiar with the terminologies, rules/regulations, dimension/ marking of the play field/area and general knowledge of national/ international level facts/figures related to the chosen game.

**Assessment:**

The Institute must assign coordinator / subject teacher for the subject, for every batch/group of the students, who would be responsible for coordinating the required activities and keep watch on the level of student's participation in the chosen game.

Coordinator/mentor would be responsible for the award of the sessional marks based upon following components.

| | | |
|---|---|---|
| I. | Level of understanding and general awareness | ( 20 %) |
| II. | Involvements in the Practice Sessions | (50 %) |
| III. | Regularity, Sincerity and Discipline | (20 %) |
| IV. | Participation in University level / District level / State level / National Level events | (10 %) |