# Scheme of Instructions & Syllabi

# of

# Master of Computer Applications
# First Year

(Effective from session 2020-21)

**Department of Computer Applications**

**INVERTIS UNIVERSITY**
**Bareilly-243123 U.P.**

# STUDY AND EVALUATION SCHEME
# Master of Computer Applications
(Effective from session 2020-2021)

## SEMESTER I, YEAR I

| S.N. | Course Code | Subjects | L+T+P | Scheme | | Total | Credit |
|---|---|---|---|---|---|---|---|
| | | | | CA | EE | | |
| **THEORY** | | | | | | | |
| **1** | MCA106 | Object Oriented Programming Concepts | 3+1+0 | 30 | 70 | 100 | 4 |
| **2** | MCA107 | Advanced Computer Architecture | 3+1+0 | 30 | 70 | 100 | 4 |
| **3** | MCA108 | Advanced Database Management Systems | 3+1+0 | 30 | 70 | 100 | 4 |
| **4** | MCA109 | Advanced Computer Networks | 3+1+0 | 30 | 70 | 100 | 4 |
| **5** | MCA110 | Advanced Data Structure and Algorithms | 3+1+0 | 30 | 70 | 100 | 4 |
| **PRACTICAL / PROJECTS** | | | | | | | |
| **7** | MCA153 | Object Oriented Programming Concepts Lab | 0+0+4 | 15 | 35 | 50 | 2 |
| **8** | MCA154 | Advanced Database Management Systems Lab | 0+0+4 | 15 | 35 | 50 | 2 |
| **9** | MCA155 | Advanced Data Structure and Algorithms Lab | 0+0+4 | 15 | 35 | 50 | 2 |
| **TOTAL** | | | **15 5 12** | **195** | **455** | **650** | **26** |

## SEMESTER II, YEAR I

| S.N. | Course Code | Subjects | L+T+P | Scheme | | Total | Credit |
|---|---|---|---|---|---|---|---|
| | | | | CA | EE | | |
| **THEORY** | | | | | | | |
| **1** | MCA206 | Advanced Java Programming | 3+1+0 | 30 | 70 | 100 | 4 |
| **2** | MCA207 | Design & Analysis of Algorithms | 3+1+0 | 30 | 70 | 100 | 4 |
| **3** | MCA208 | Advanced Operating Systems | 3+1+0 | 30 | 70 | 100 | 4 |
| **4** | MCA209 | Advanced Software Engineering | 3+1+0 | 30 | 70 | 100 | 4 |
| **5** | MCA210 | Web Technologies | 3+1+0 | 30 | 70 | 100 | 4 |
| **PRACTICAL / PROJECTS** | | | | | | | |
| **7** | MCA253 | Java Programming Lab | 0+0+4 | 15 | 35 | 50 | 2 |
| **8** | MCA254 | Web Technologies Lab | 0+0+4 | 15 | 35 | 50 | 2 |
| **9** | MCA255 | Seminar | 2+0+0 | 0 | 0 | 50 | 2 |
| **TOTAL** | | | **15 5 12** | **195** | **455** | **650** | **26** |

| Program Outcomes (POs) | | |
|---|---|---|
| **PO1** | Computational Knowledge | Understand and apply mathematical foundation, computing and domain knowledge for the conceptualization of computing models from defined problems. |
| **PO2** | Problem analysis | Ability to identify, critically analyze and formulate complex computing problems using fundamentals of computer science and application domains. |
| **PO3** | Design / Development of Solutions | Ability to transform complex business scenarios and contemporary issues into problems, investigate, understand and propose integrated solutions using emerging technologies |
| **PO4** | Conduct Investigations of Complex Computing Problems | Ability to devise and conduct experiments, interpret data and provide well informed conclusions. |
| **PO5** | Modern Tool Usage | Ability to select modern computing tools, skills and techniques necessary for innovative software solutions. |
| **PO6** | Professional Ethics | Ability to apply and commit professional ethics and cyber regulations in a global economic environment. |
| **PO7** | Life-long Learning | Recognize the need for and develop the ability to engage in continuous learning as a Computing professional. |
| **PO8** | Project Management and Finance | Ability to understand, management and computing principles with computing knowledge to manage projects in multidisciplinary environments. |
| **PO9** | Communication efficacy | Communicate effectively with the computing community as well as society by being able to comprehend effective documentations and presentations. |
| **PO10** | Societal & Environmental Concern | Ability to recognize economical, environmental, social, health, legal, ethical issues involved in the use of computer technology and other consequential responsibilities relevant to professional practice. |
| **PO11** | Individual & Team Work | Ability to work as a member or leader in diverse teams in multidisciplinary environment. |
| **PO12** | Innovation and Entrepreneurship | Identify opportunities, entrepreneurship vision and use of innovative ideas to create value and wealth for the betterment of the individual and society. |

| MCA106:Object Oriented Programming Concepts | |
|---|---|
| **Teaching Scheme**<br>Lectures: 3 hrs/Week<br>Tutorials: 1 hr/Week<br><br>Credits: 4 | **Examination Scheme**<br>Unit Test -12Marks<br>Teachers Assessment - 6Marks<br>Attendance – 12 Marks<br>End Semester Exam – 70 marks |

**Prerequisite**: -

1. Computer Fundamentals
2. Principles of computer programming
3. Basic mathematical formulas.

**Course Objectives:**

1. Be able to write a C++ program to solve a well specified problem.
2. Understand a C++ program written by someone else.
3. Be able to debug and test C++ programs;
4. To make the students understand the features of object oriented principles.
5. Familiarize them with virtual functions, templates and exception handling.
6. To make the students to develop applications using C++.

**Detailed Syllabus**

| |
|---|
| **UNIT I (10 Hours)**<br>**Introduction to OOP:** Basic concepts of OOPs, Advantages of OOP, characteristics of object-oriented languages, Object, Classes, Encapsulation, Data Abstraction, Inheritance, Polymorphism, Dynamic binding, Message Passing, keywords, identifiers, data types, manipulators, **O**perators in C++, Operator Precedence, Typecast operator, Control structures, Loops. |
| **UNIT II (6 Hours)**<br>**Functions:** Function Prototyping, Call by reference, Return by Reference, Default and Constant Arguments, Inline Function, functions Overloading, Friend and virtual Functions, static function. |
| **UNIT III (10 Hours)**<br>**Objects and classes:** Specifying class & object, Arrays as class member data, Arrays of objects, Constructors and Destructors, objects as function arguments. **Operator Overloading: O**verloading Unary & Binary operators, |
| **UNIT IV (10 Hours)**<br>**Inheritance: introduction, defining deriv**ed classes, overriding member functions, Single Inheritance, multilevel Inheritance, multiple Inheritance, Hierarchical Inheritance, Virtual Base Class.<br>F**iles and Streams:** Introduction, classes for file stream operations, opening and closing files, file pointers and their manipulations, Error Handling, command-line Arguments. |
| **UNIT V (10 Hours)**<br>**Object Modeling:** Objects and classes, links and association, generalization and inheritance, aggregation, abstract class, multiple inheritance, Meta data, candidate keys, constraints. **Dynamic Modeling:** Events and states, operations, nested state diagrams and concurrency, advanced dynamic modeling concepts, a sample dynamic model. |
| **UNIT VI (10 Hours)**<br>**Functional Modeling:** Data flow diagram, specifying operations, constraints, a sample functional model. OMT, examples and case studies to demonstrate methodologies, comparisons of methodologies, SA/SD, JSD. |

**Text and Reference Books**


1. Object Oriented Programming with C++, E. Balaguruswamy, 4th Edition.
2. Object Oriented Programming in C++, Robert Lafore, Sams, Dec., 2001.
3. C++ Programming, D. Ravichandran, TMH, 2nd Edition, Dec. 2002.
4. Mastering C++, Venugopal, TMH, September, 1997.


## Course Outcomes:

| |
|---|
| 1. Understanding the concept and recognize the basic terminology used in computer programming. |
| 2. Students will be able to apply the computer programming techniques to solve practical problems. |
| 3. Students will be able to understand the concepts and implementation of class , constructors and destructors. |
| 4. Students are able to learn C++ data types, memory allocation/deallocations, functions and pointers. |
| 5. Use different data structures and create / manipulate basic data files and developing applications for real world problems. |
| 6. Students are able to apply object oriented programming concepts to software problems in C++ Outcome(s) |

| MCA107: Advanced Computer Architecture | |
|---|---|
| **Teaching Scheme**<br>Lectures: 3 hrs/Week<br>Tutorials: 1 hr/Week<br><br>Credits: 4 | **Examination Scheme**<br>Unit Test -12Marks<br>Teachers Assessment - 6Marks<br>Attendance – 12 Marks<br>End Semester Exam – 70 marks |

**Prerequisite**: - Basic knowledge of computer

**Course Objectives:**

1. To familiarize with digital computer building blocks.
2. To introduce working of a computer at instruction level.
3. To know various processor design.
4. To know the basics of working of I/O operations of a computer.
5. To be familiar with various types of computer memory.
6. To know different memory management techniques.

**Detailed Syllabus**

| |
|---|
| **Introduction:** Logic Gates, Adders, Subtractors, Multiplexer, Decoder, Encoder, IEEE standard for Floating pointnumbers, Register Transfer Language and notations, Tri- state Buffer, Bus structure, Arithmetic, Logical & Shift Micro operation. |
| **UNIT II (10 Hours)**<br>**Processing Unit:** Fundamental Concepts: Micro instruction, Performing arithmetic or Logic Micro operation, Fetching and Storing of a Word in Memory. Execution of Complete Instruction, Microprogram sequencing, Multiple-Bus organization. |
| **UNIT III (10 Hours)**<br>**Processor Design:** General register organization, Control Word, Stack Organization, Instruction Format, 0,1,2,3 Address Instructions, Addressing Modes, Data transfer & Manipulations Instructions, Reduced Instruction Set Computer. **I/O organization:** Input-Output Interface, Handshaking, Direct Memory Access |
| **UNIT IV (10 Hours)**<br>**Memory Organization:** RAM, ROM, Boot Strap Loader, Cache Memory Mapping Functions, **Virtual Memory:** Virtual Memory: address space and Memory space, Address Mapping using Pages, associative Memory Page Table, Page Replacement, Page Replacement algorithm: Least Recently Used, First in First out, Optimal, Interleaving, Hit Ratio. |
| **UNIT V (10 Hours)**<br>Pipelining Review - basic concept of pipeline and two different types of hazards. • Pipeline CPI • Processor Pipeline Hazards • Computer Architecture & Tech Trends • Processor Speed, Cost, Power • Measuring Performance • Benchmarks Standards • Iron Law of Performance • Moore's Law • Amdahl's Law • Lhadma's Law • Gustafson's law |
| **UNIT VI (10 Hours)**<br>**SIMD Architecture-** Introduction, Parallel Processing, classification of Parallel Processing, Fine-Grained SIMD Architecture, coarse-Grained SIMD Architecture, MIMD Architecture, RAID |

**Text and Reference Books**

1. Computer System Architecture, M. Morris Mano, PHI, 2002,5th Edition
2. Computer Organization, Vravice, Zaky&Hamacher, TMH Publication, 2001, 3rd Edition
3. Structured Computer Organization, Tannenbaum, PHI, 2008, 2nd Edition.
4. Computer Organization, Stallings, PHI, 2002, 7th Edition.

**Course Outcomes:**

After completing the course, students will be able to:

| |
|---|
| 1. Know various components of a digital computer. |
| 2. Design basic computer instructions |
| 3. Propose a new processor design. |
| 4. Understand the working of input and output devices and device controller. |
| 5. Understand computer memory hierarchy |
| 6. Implement paging and segmentation in computer memory. |

| MCA108: Advanced Database Management System | |
|---|---|
| **Teaching Scheme** | **Examination Scheme** |
| Lectures: 3 hrs/Week | Class Test -12Marks |
| Tutorials: 1 hr/Week | Teachers Assessment - 6Marks |
| | Attendance – 12 Marks |
| Credits: 4 | End Semester Exam – 70 marks |

**Prerequisite**: - Computer Organization, Operating System, Data Structure, Mathematics

**Course Objectives:**

The objectives of this course are
1. Understand values of Data.
2. Understand significant role of DBMS.
3. Understand need for normalizing a Database.
4. Understand problems with unnecessary duplication of data.
5. Understand concepts of transaction
6. Understand concepts of concurrent transactions

**Unit-1 (6 Hours)**

**Introduction Database Systems:** An overview of database management system, Database System Vs File System, Database system concepts and architecture, data models schema and instances, data independence and data base language and interfaces, Data definitions language, DML, Overall Database Structure.

**Unit-II (10 Hours)**

**Data Modeling using Relational Data Model:** Modeling Techniques-Different Types of Models. Hierarchical Database, Network Database, and Relational Database. Relational data model-Codd's Rules, Concept of Domain, Tuple,and Cardinality. Introduction to ERD-ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation.

**Unit-III (10 Hours)**

**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.

**Unit-IV (10 Hours)**

**Structured Query Language:** Features of SQL, SQL *PLUS, SQL V/s SQL *PLUS, Rules for SQL, SQL Delimiters, Components of SQL.**Constraints:** Data constraints, Types of data constraints: UNIQUE, NOT NULL at column level, CHECK, NULL value constraint

**PL/SQL:** Basic Introduction, Advantages of PL/SQL, The generic PL/SQL block, Literals, Variables, Constants, Comparisons, Comments. **Control Structure:** Conditional Control, Iterative Control and Sequential Control. **PL/SQL Transaction:** Oracle Transactions, Cursor, Types of Cursor: Implicit cursor, Explicit cursor.

**Unit-V (10 Hours)**

**Transaction Processing Concepts:** Transaction system, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery. Concurrency Control-Concurrency control, Protocols for concurrency control-locking, Time stamping, validation based protocol. Multiple granularity, Multi-version schemes, Recovery with concurrent transaction.

**Unit-VI (10 Hours)**

**Modern Database Systems:** Transaction Processing in Distributed system, data fragmentation, Replication and allocation techniques for distributed system, overview of concurrency control and recovery in distrusted database. Parallel databases, multimedia databases, spatial and temporal databases, data warehousing and data mining, deductive databases.

**Text and Reference Books**


**1.** Database System Concepts, Henry Korth , A. Silberschatz, 5th Edition, 2005.
**2.** An Introduction to Database System, Bipin Desai, Galgotia Publications, 1991.
**3.** SQL, PL/SQL the Programming Language of Oracle, Ivan Bayross, BPB Publications, 4th Edition.
**4.** Schaum's Outline of "Fundamental of Relational Databases", Ramon A. Mata, Pauline K. Cushman, McGraw Hill, December, 2006.


**Course Outcomes:**

| |
|---|
| 1. Acquire knowledge of handling large volume of data. |
| 2. Acquire skills to deal with Real life database implementation. |
| 3. Response off faster queries and serve as many users as possible concurrently. |
| 4. Attain the capability to represent various real life problem domains using logic based techniques and use this to perform inference or planning. |
| 5. Fit with any Database project in industry after completion of degree. |

| MCA109: Advanced Computer Networks | |
|---|---|
| **Teaching Scheme**<br>Lectures: 3 hrs/Week<br>Tutorials: 1 hr/Week<br><br>Credits: 4 | **Examination Scheme**<br>Class Test -12Marks<br>Teachers Assessment - 6Marks<br>Attendance – 12 Marks<br>End Semester Exam – 70 marks |

**Pre-requisites:** Data Communication and Computer Network, INTERNET, Router

**Course Objectives:**

1. To discuss and explain about basics of data communication and networking concepts .Explain how the data link layer prepares data for transmission les and list the component parts of a Layer
2. To discussed the medium access control and to create a logical design and physical design of a simple Ethernet LAN
3. Describe how routers use next-hop addresses to determine the path that packets need to take to reach their destinations and their describe the IP addressing structure
4. Explain the difference between TCP and UDP and describe how TCP and UDP function are worked Describe and application layer for using end user application such as DNS, SMTP and Telnet etc

| |
|---|
| **UNIT 1**<br>**Introduction –** Data Communication, Data encoding and Modulation, Broadband and Baseband transmission, The Internet Today, Protocols and Standards, Internet Standards Network topologies design ,Connecting Devices, Network Types, OSI Reference Model, TCP/IP Protocol Suite |
| **UNIT 2**<br>**Data Link Layer:** Error Detection and Correction, Techniques, CRC and Hamming Code. Flow Control and Error Control Techniques: Stop and Wait, Sliding Window, Go-Back-N, Selective Repeat Protocol. Ethernet, Ethernet frame, Addressing. |
| **UNIT 3**<br>**Media Access Sub layer**-Media Access Sub layer, ALOHA Protocol, Overview of protocol, Channel allocation, WLAN protocol, CSMA, CSMA/CD, CSMA/CA and CDMA Protocol, Wireless LAN |
| **UNIT 4**<br>**Network Layer:** Network Layer - Point - to Pont Networks, Routing Protocols, TCP / IP, IP packet, IP address, IPv4 &IPv6. Bluetooth, Cellular telephony, IP Addresses: Class Addressing, |
| **UNIT 5**<br>**Transport Layer:** TCP, & UDP protocol, Routing Protocols, Static & Dynamic Routing, Routing Table and Routing Module, Socket Interface: Definitions, Byte Ordering, Address Transformation, Byte Manipulation Function, Information about Remote Host, Socket System Calls, Connectionless Iterative Server, |
| **UNIT-6**<br>**Application Layer:** Network Management and SNMP Multimedia and Data Compression, Electronic Mail: SMTP and MIME, DNS, TELNET & Rlogin, FTP, TFTP, SMTP, HTTP, WWW, RTP, BOOTP & DHCP, DNS, TELNET & Rlogin. |

| Text and Reference Books |
|---|
| 1. Behrouz A Forouzan: TCP/IP Protocol Suite, 4th Edition,2010, TMH |
| 2. Douglas E Comar: TCP/IP Protocol, 6th Edition,2008, Pearson Education |
| 3. Behrouz A Forouzan: Data Communication and Networking, 4th Edition,2006, TMH |
| 4. Richard Stevens: TCP/IP Illustrated Vol 1: The Protocols, 1st Edition,2006, Pearson Education, India. |

**Course Outcomes:**

After completing the course, students will be able to:

| |
|---|
| 1. Recognize and Describe about the working of Computer Networks and Illustrate reference models with layers, protocols and interfaces. |
| 2. Illustrate data link layer for using different error Control techniques |
| 3. Examine problems of a computer networks related techniques for CSMA/CD, Aloha, Ethernet and WLAN |
| 4. Students will understand for network layer internetworking technologies, Routing, IP Addressing and routing protocol for using shortest path for destination |
| 5. Students will understand TCP/IP implementation |
| 6. Students will understand the end user application for such domain name system , HTTP, UDP Telnet and SMTP etc |

| MCA110: Advanced Data Structure and Algorithms | |
|---|---|
| **Teaching Scheme** | **Examination Scheme** |
| Lectures: 3 hrs/Week | Class Test -12Marks |
| Tutorials: 1 hr/Week | Teachers Assessment - 6Marks<br>Attendance – 12 Marks |
| Credits: 4 | End Semester Exam – 70 marks |

**Prerequisite**: -

1. Familiarity with the fundamentals of C or other programming language
2. A solid background in mathematics, including probability, set theory

**Course Objectives:**

1. Understand various data structures like array, linked list.
2. Implement operations like insertion, deletion and traversing mechanism on various data structures.
3. Implement Linear and Non-Linear data structures.
4. Implement sorting/searching technique.
5. Understand and implement advance data structure using Non-Linear data structure.
6. Determine and analyze the complexity of given algorithms.

**UNIT I (10 Hours)**

**Introduction to Algorithm Design and Data Structures:** Abstract data types, Fundamental and derived data types. Representation, Primitive data structures. Algorithm Definition, Analysis of Algorithm, Comparison of Algorithms. Top Down and bottom up Approaches, Complexity- time and space. Structured approach to programming.

**Arrays:** Representation of Arrays (Single and Multidimensional arrays), Address calculation using column and row major ordering, Operations on Arrays. Application of arrays- Matrix Multiplication, Sparse matrix.

**UNIT II (10 Hours)**

**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack , Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion, Tower of Hanoi

**Queues:** Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.

**UNIT III (10 Hours)**

**Linked lists:** Array Implementation and Dynamic Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition, Generalized Linked List

**UNIT IV (10 Hours)**

**Trees:** Basic terminology, Binary Trees, Binary Tree Representation: Array Representation and Dynamic Representation, Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Array and Linked Representation of Binary trees, Tree Traversal algorithms: In order, Preorder and Post order, Threaded Binary trees, Traversing Threaded Binary trees, Huffman algorithm

**UNIT V (6 Hours)**

**Searching :** Sequential search, Binary Search, Comparison and Analysis Internal Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Two Way Merge Sort, Heap Sort, Radix Sort, Practical consideration for Internal Sorting.

**Search Trees:** Binary Search Trees (BST), Insertion and Deletion in BST, Complexity of Search

Algorithm, AVL trees, Introduction to m-way Search Trees, B Trees & B+ Trees .
**Hashing:** Hash Function, Collision Resolution Strategies Storage Management: Garbage Collection and Compaction.

**UNIT VI (10 Hours)**
Graphs: Terminology, Sequential and linked Representations of Graphs: Adjacency Matrices, Adjacency List, Adjacency Multi list, Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Transitive Closure and Shortest Path algorithm: Warshal Algorithm and Dijkstra Algorithm, Introduction to Activity Networks

**Text and Reference Books**


1. Data Structures and Program Design in C, R.L. Kruse, B.P. Leung and C. L. Tondo, PHI, 2008.
2. Data Structures, Seymour Lipscutz, Mcgraw Hill Publication, 2009
3. Data structures using C, Aaron M.Tenanbaum, Pearson education, 2004.
4. Data structure through C, Yashvant Kanetkar, BPB Publication, 2006.


**Course Outcomes:**

1. Solving problems and simulate the insertion and deletion by using DS methods.
2. Understanding the concept and recognize the basic terminology used in computer programming.
3. Write, Compile and Debug programs in C language and use different data types for writing the programs.
4. Design programs connecting decision structures, loops and functions.
5. Understand the dynamic behavior of memory by the use of pointers
6. Use different data structures and create / manipulate basic data files and developing applications for real world problems.

| MCA206:Advanced Java Programming | |
|---|---|
| **Teaching Scheme**<br>Lectures: 3 hrs/Week<br>Tutorials: 1 hr/Week<br><br>Credits: 4 | **Examination Scheme**<br>Class Test – 12 Marks<br>Teachers Assessment – 6 Marks<br>Attendance – 12 Marks<br>End Semester Exam – 70 Marks |

**Prerequisite:** C Programming, and OOPs Concepts.

**Course Objectives:**

1. To understand the basic concepts of Java, Importance of Classes & objects along with Method overloading and overriding.
2. To understand the conditional construction, arrays as well as Packages.
3. To learn the Exception Handling and I/o file handling with buffer reader and scanner class.
4. To understand importance of Multi-threading and AWT that respond to different user events.
5. To learn experience of Java swing and JDBC.
6. To understand Java beans and Java servlets for web development.

**Detailed Syllabus:**

| |
|---|
| **Unit-1**<br>**Introduction of Java**: Features of Java Language, Platform Independency, JVM, Byte-code, Operator, Data type, Variables, Robustness.<br><br>**OOPS:** Object, Class, Classifications, Methods & classes, Inheritance, Static and non Static methods, Call by Value, Call by Reference, Method Overloading, Method Overriding, Abstraction, Interface, Polymorphism, Inner Class & Anonymous Classes, Abstract Class. |
| **Unit-2**<br>**Conditional Construct in Java:** if, if else, nested if else, if else ladder, Ternary Operator, Switch.<br>**Array:** Introduction of arrays, Understanding and working with single, double dimensional arrays, Initialization of array, Linear and Binary Search.<br>**Packages and Exception Handling:** Data Encapsulation, Concept of Package, creating package, Importing packages, Child Packages. |
| **Unit-3**<br>**Exception Handling:** Exceptions & Errors, Types of Exception, Control Flow in Exceptions, Use of try, catch, finally, throw, throws in Exception Handling. Checked and Un-Checked Exceptions.<br>**I/O and File Handling:** Buffered Reader class, InputStreamReader class, Scanner class, Creating File, Reading File and Writing File |
| **Unit-4**<br>**Multi Threading:** Understanding Threads, Needs of Multi-Threaded Programming, Solution of Producer consumer problem by Multi Thread, Thread Life-Cycle, Thread Priorities, Synchronization of Thread.<br>**AWT:** Introduction to AWT, AWT controls, Layout managers, Menus, Images. |
| **Unit-5**<br>**Java Swing:** Creating a Swing Applet and Application, Programming using Panes, Labels, Text fields, Buttons, Toggle buttons, Checkboxes, Radio Buttons, Scroll Panes, Scroll Bars, Lists, Combo box, Progress Bar, Menus and Toolbars, Layered Panes, Tabbed Panes, Split Panes, Layouts, Windows, Dialog Boxes.<br>**JDBC:** The connectivity Model, JDBC/ODBC Bridge, java.sql package, connectivity to remote database, navigating through multiple rows retrieved from a database. |

**Unit-6**

**Java Beans:** Application Builder tools, The bean developer kit (BDK), Developing a simple bean, The Java Beans API.

**Java Servlets:** Servlet basics, Servlet API basic, Life cycle of a Servlet, Running Servlet, Debugging Servlets, Thread-safe Servlets, HTTP Redirects, Cookies, Introduction to Java Server pages (JSP).

**Text and Reference Books:**


1. The Complete Reference: Java, Herbert Schieldt, TMH, 7th Edition 2006
2. Programming in JAVA, E. Balagurusamy, TMH, 2nd Edition 2007
3. Object Oriented Modeling and Design, James Rumbaugh et al, PHI, 4th Edition 2003
4. Object Oriented Analysis & Design with Application, Booch Grady, Pearson Education, New Delhi, 3rd Edition, 2006.


**Course Outcomes:**

After completing the course, students will be able to:

1. Implement Object Oriented programming concept using basic syntaxes of controls Structures, strings and function for developing skills of logic building activity.
2. Demonstrates how to achieve reusability using inheritance, interfaces and packages and describes faster application development can be achieved.
3. Demonstrate understanding and use of different exception handling mechanisms and concept of multithreading for robust faster and efficient application development.
4. Demonstrate understanding and use of multi threading and AWT.
5. Identify, Design & develop complex Graphical user interfaces using Java Swing classes.
6. Demonstrates how to implement Java Beans and Java Servlets.

| MCA207: Design and Analysis of Algorithm | |
|---|---|
| **Teaching Scheme**<br>Lectures: 4 hrs/Week<br>Tutorial: 1 hr/Week<br><br>Credits: 4 | **Examination Scheme**<br>Class Test -12Marks<br>Teachers Assessment - 6Marks<br>Attendance – 12 Marks<br>End Semester Exam – 70 marks |

**Prerequisite:** - C Programming Concepts, Data Structure Concepts, Discrete Mathematics concepts.

**Course Objectives:**

1. To analyze the asymptotic performance of algorithms.
2. To analyze of Advanced Data Structure Concepts.
3. To analyze Greedy and Dynamic Programming Concepts and its application
4. To analyze concepts of Graphs.
5. To analyze Branch and Bound and Backtracking Concepts and its applications.
6. To analyze Deterministic and Non deterministic Problem.

**UNIT I (10 Hours)**
**Introduction:** Algorithm, Pseudo code for expressing algorithms, Performance Analysis Of algorithm-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Recurrences and their solutions, Amortized analysis.

**Divide and Conquer:** General method, applications-Binary search, Quick sort, Merge sort, Heap Sort, Strassen's matrix multiplication.

**UNIT II (9 Hours)**
**Advanced Data Structure:** Red Black Tree, Binomial Heap, B tree, Fibonacci Heap. **Disjoint Sets:** disjoint set operations, union and find algorithms, spanning trees, connected components and biconnected components.

**UNIT III (10 Hours)**
**Greedy method:** General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees.

**Dynamic Programming:** General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, Travelling sales person problem.

**UNIT IV (10 Hours)**
**Graph Algorithm:** Graph Algorithms, BFS, DFS, Minimum Spanning Tree, Kruskal's Algorithms, Prim's Algorithms, Single Source Shortest Path, All pair Shortest Path, Maximum flow.

**UNIT V (9 Hours)**
**Backtracking:** General method, applications-n-queen problem, graph colouring, Hamiltonian cycles.
**Branch and Bound:** General method, applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution.

**UNIT VI (8 Hours)**
NP-Hard and NP-Complete problems: Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem.

**Text and Reference Books:**

| |
|---|
| Introduction to Algorithms, Thomas H CormenLeiserson et al, PHI, 2nd Edition 2001 |
| Computer Algorithms: Introduction to Design and Analysis, Sara Baase and Allen Van Gelder, Pearson Education, 3rd Edition 2000 |
| Algorithm Design, Jon Kleinberg and Eva Tardos, Pearson Education, 1st Edition 2005 |
| The Design and analysis of Algorithms, A V Aho et al, Pearson Education, 3rd Edition 2007 |
| Fundamentals of computer Algorithm,Ellis Horowitz,SartajSahni and Rajasekharam,Galgotia Publication 2009 |

**Course Outcomes:**

After completing the course, students will be able to:

| |
|---|
| 1. Understand Asymptotic Notation. |
| 2. Understand Advanced Data Structure Concepts and searching concepts. |
| 3. Understand the Concepts of Greedy Methods and Dynamic Programming methods and solve problem related with its. |
| 4. Understand the concepts of Graph. |
| 5. Understand the concepts of Backtracking and Branch and bound Concepts and solve problem related with its. |
| 6. Understand the Concepts of NP hard and NFA DFA Concepts. |

| MCA208: Advanced Operating Systems | |
|---|---|
| **Teaching Scheme** | **Examination Scheme** |
| Lectures: 3 hrs/Week | Class Test -12Marks |
| Tutorials: 1 hr/Week | Teachers Assessment - 6Marks<br>Attendance – 12 Marks |
| Credits: 4 | End Semester Exam – 70 marks |

**Prerequisite**: - Basic Computer Concepts

**Course Objectives:**

1. Understand the services of an operating system provides to its users and system itself.
2. Apply various CPU scheduling algorithms and recognize the classic synchronization problems
3. Compare methods for handling deadlocks and apply various memory management techniques.
4. Describe file systems.
5. To understand the disk scheduling.
6. Security issues in system

**Detailed syllabus:**

| |
|---|
| **UNIT I**<br>**Introduction:** Definition of operating systems, Computer System architecture: single Processor, Multi-Processor, Clustered Systems. Operating system structure, Dual Mode Operating system Operations, Distributed System, Operating system services, System calls, system programs, Design Goals, Layered Approach. |
| **UNIT II**<br>**Process Management:** Process concept, Process scheduling, Cooperating processes, Threads, Inter-process communication, CPU Scheduling: Scheduling Queues, Schedulers, Context Switch, CPU scheduling criteria, Scheduling algorithms, Multiple-processor scheduling. |
| **UNIT III**<br>**Process Synchronization and Deadlocks:** The Critical-Section problem, Peterson's solution, Semaphores, Classical problems of synchronization, Critical regions, Deadlocks-System model, Characterization: Necessary Conditions, Resource allocation Graph. Deadlock prevention, Avoidance and Detection, Recovery from deadlock. |
| **UNIT IV**<br>**Storage management:** Memory Management-Basic Hardware, Logical and Physical Address Space, Swapping, Fragmentation, Non Contiguous Memory allocation, Contiguous Memory allocation, Paging: Basic concept, allocation algorithm, Relocation, Protection. Segmentation: Basic concept, allocation algorithm, Relocation, Protection. Segmentation with paging, Virtual Memory, Demand paging, Page replacement algorithms, Allocation of frames, Thrashing: Cause of Thrashing, Working set Model. |
| **UNIT V**<br>**File concept, access methods, and Directory implementation:** Linear List, Hash Table. Disk structure, Disk scheduling methods, Disk management: Disk Formatting, Boot Block, Bad Block. Interrupt, Direct Memory Access. |
| **UNIT VI**<br>**Security & Case Study:** Protection and Security-Goals of protection, Domain of protection, Access matrix, Implementation of access Matrix, The Security problem, Authentication, One Time passwords, Program threats, System threats, Threat Monitoring, Encryption. |

**Text and Reference Books**


1. "Operating system concepts", Galvin, TMH, IV, 2006
2. "Operating system concepts & Design", Milankovic, AddisionWesely, 2010.
3. "Operating System", Madnic,TMH, 1997
4. "Operating System", A.s. Godbole, TMH, 2001.
5. "Operating System", W.Stallings, Printice Hall, VI, 2007


**Course Outcomes:**

After completing the course, students will be able to:

| |
|---|
| 1. Experiment with various CPU scheduling algorithms with the understanding of operating system concepts |
| 2. Explain the need for process coordination |
| 3. Apply the various memory management strategies |
| 4. Illustrate the various file management strategies |
| 5. Explain about disk management |

| MCA 209: Advanced Software Engineering | |
|---|---|
| **Teaching Scheme** | **Examination Scheme** |
| Lectures: 3 hrs/Week | Class Test -12Marks |
| Tutorials: 1 hr/Week | Teachers Assessment - 6Marks |
| | Attendance – 12 Marks |
| Credits: 4 | End Semester Exam – 70 marks |

**Prerequisite: -**

1. Familiarity with the fundamentals of system analysis and design
2. Basic terminologies used in software development.

**Course Objectives:**

1. It aims to develop a broad understanding of the discipline of software engineering.
2. It seeks to complement this with a detailed knowledge of techniques for the analysis and design of complex software intensive systems.
3. It aims to set these techniques in an appropriate engineering and management context.

**Detailed Syllabus**

**UNIT I (10 Hours)**

**Introduction to Software and Software Engineering:** Software Characteristics and Applications, Software Engineering a Layered Technology, Software Process.

**UNIT II (10 Hours)**

**Software Life Cycle Models:** Classical Waterfall Model, Iterative Waterfall Model, Prototyping Model, Evolutionary Model, RAD Model , Spiral Model, Agile Software Development Model, Comparison of different Life Cycle Models.

**UNIT III (10 Hours)**

**Software Project Management:** Project Planning, Project size estimation-LOC and FP Metric, Project Estimation Technique-COCOMO Model, Project Scheduling-WBS, Gantt chart, PERT Chart, Activity Network and Critical Path Method, Risk Management, Software Configuration Management.

**UNIT IV (10 Hours)**

**Requirement Engineering:** Requirement Gathering, Requirement Analysis-ERD, DFD, Data Dictionary, Decision Tree, Decision Table, SRS Document, Characteristics of good SRS Document, SRS Verification and Validation.

**UNIT V (6 Hours)**

**Software Design:** Characteristics of good Software Design, Cohesion and Coupling. Function Oriented Design: Structured Analysis. Object Oriented Design: OOPS Concepts, UML and USE Case Model.

**UNIT VI (10 Hours)**

**Testing and Implementation:** What is Testing and Debugging, Design of Test Cases, Unit Testing, Integration Testing, White Box and Black Box Testing, System Testing, McCabe's Cyclomatic Complexity, System Testing. Software Reliability Models, SQA, SEI/CMM, CASE. Software Maintenance Models.

**Text and Reference Books**


1. Software Engineering, Roger S Pressman, Tata McGraw Hill, 6th Edition 2005
2. Fundamentals of Software Engineering, Rajib Mall, PHI, 3rd Edition 1997
3. Software Engineering, I. Sommerville, Pearson Education, 8th Edition 2007
4. Software Engineering Concepts, R Fairley, Tata McGraw Hill, 4th Edition 1997

**Course Outcomes:**

| |
|---|
| 1. Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility. |
| 2. Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment. |
| 3. An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics |
| 4 an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors |
| 5. an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives |
| 6. an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions. |

| MCA 210: Web Technology | |
|---|---|
| **Teaching Scheme**<br>Lectures: 3 hrs/Week<br>Tutorials: 1 hr/Week<br><br>Credits: 4 | **Examination Scheme**<br>Class Test -12Marks<br>Teachers Assessment - 6Marks<br>Attendance – 12 Marks<br>End Semester Exam – 70 marks |

**Course Objectives:**

The course content enables students to:

1. Understand best technologies for solving web client/server problems
2. Analyze and design real time web applications
3. Use Java script for dynamic effects and to validate form input entry
4. Analyze to Use appropriate client-side or Server-side applications

| |
|---|
| **UNIT I (6 Hours)**<br>**Introduction:** Introduction to web, protocols governing the web, web development strategies, Web applications, web project, web team. |
| **UNIT II (10 Hours)**<br>**Web Page Designing using HTML**: Structure of HTML page, link, list, table, images, frames, forms, CSS; DHTML |
| **UNIT III (10 Hours)**<br>**XML:** DTD, XML schemes, presenting and using XML |
| **UNIT IV (10 Hours)**<br>**Java script:** Introduction, documents, forms, statements, functions, objects; event and event handling; introduction to AJAX, VB Script |
| **UNIT V (10 Hours)**<br>**Server Side Programming:** Introduction to active server pages (ASP), ASP.NET, java server pages (JSP), JSP application design, tomcat server, JSP objects, declaring variables, and methods, debugging, sharing data between JSP pages, Session, Application: data base action , development of java beans in JSP, introduction to COM/DCOM. |
| **UNIT VI (10 Hours)**<br>**PHP (Hypertext Preprocessor):** Introduction, syntax, variables, strings, operators, if-else, loop, switch, array, function, form ,mail, file upload, session, error, exception, filter, PHP-ODBC.<br>Web Page Designing using HTML: Structure of HTML page, link, list, table, images, frames, forms, CSS; DHTML |

| **Text and Reference Books** |
|---|
| 1.Heywood J.B., ―Internal combustion Engine Fundamentals‖, McGraw Hill, 1988 |
| 2.ObertE.F.,―InternalcombustionEngineandAirPollution‖,IntextEducationalPub,1974 |
| 3.Ganesan V.,―Internal combustion Engines‖, 6 th Ed.Tata Mc Graw Hill Publishing Co. Domkundwar V.M. ―Internal CombustionEngines‖- |
| 4.Mathur M.C.,Sharma R.D.,―Internal combustion engines‖,8th Ed.; Dhanpat Rai publication.,2003 PulkrabekW, ―Engineering Fundamentals Of Internal Combustion Engine‖, Prentice Hall, 1997 |

**Course Outcomes:**

| |
|---|
| 1. Choose, understand, and analyze any suitable real time web application. |
| 2. Integrate java and server side scripting languages to develop web applications. |
| 3. To develop and deploy real time web applications in web servers and in the cloud. |
| 4. Extend this knowledge to .Net platforms. |