AWADHESH PRATAP SINGH UNIVERSITY
REWAA (MP) 486003

CBCS
CURRICULAM & SYLLABUS
(as per unified ordinance no. 14 of MP universities)
for
MASTER OF COMPUTER APPLICATION (MCA)
(AICTE Approved)
w.e.f. Session 2020-21

Course code : 060

www.apsurewa.ac.in
A. P. S. UNIVERSITY, REWA (MP)
MASTER OF COMPUTER APPLICATION (MCA)
SCHEME OF EXAMINATION(w.e.f. Session 2020-21)

Course code : 060

FIRST SEMESTER

<table>
<thead>
<tr>
<th>Paper Code</th>
<th>Paper Name</th>
<th>Course Type</th>
<th>Credit</th>
<th>Theory Marks Max(Min)</th>
<th>CCE Marks Max(Min)</th>
<th>Total Marks (Min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10601</td>
<td>Computer Organization</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>10602</td>
<td>Mathematical Foundation of Computer Science</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>10603</td>
<td>Operating System</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>10604</td>
<td>DBMS</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>10605</td>
<td>Problem Solving using C &amp; C++</td>
<td>GE</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>10606</td>
<td>Lab I – DBMS</td>
<td>LAB</td>
<td>2</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>10607</td>
<td>Lab II – Prog. in C &amp; C++</td>
<td>LAB</td>
<td>2</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>10608</td>
<td>Comprehensive Viva</td>
<td>Viva</td>
<td>4</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td></td>
<td>28</td>
<td></td>
<td></td>
<td>800</td>
</tr>
</tbody>
</table>

SECOND SEMESTER

<table>
<thead>
<tr>
<th>Paper Code</th>
<th>Paper Name</th>
<th>Course Type</th>
<th>Credit</th>
<th>Theory Marks Max(Min)</th>
<th>CCE Marks Max(Min)</th>
<th>Total Marks (Min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20601</td>
<td>Data Structure and Analysis of Algorithms</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>20602</td>
<td>Software Engineering</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>20603</td>
<td>Computer Graphics &amp; Visualization</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>206041</td>
<td>Elective I : (Any one from the following)</td>
<td>DCE</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>206042</td>
<td>• Artificial Intelligence</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Cloud Computing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20605</td>
<td>Java Programming &amp; Technologies</td>
<td>GE</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>20606</td>
<td>Lab I – Computer Graphics</td>
<td>LAB</td>
<td>2</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>20607</td>
<td>Lab II – Java</td>
<td>LAB</td>
<td>2</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>20608</td>
<td>Comprehensive Viva</td>
<td>Viva</td>
<td>4</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td></td>
<td>28</td>
<td></td>
<td></td>
<td>800</td>
</tr>
</tbody>
</table>
### THIRD SEMESTER

<table>
<thead>
<tr>
<th>Paper Code</th>
<th>Paper Name</th>
<th>Course Type</th>
<th>Credit</th>
<th>Theory Marks Max(Min)</th>
<th>CCE Marks Max(Min)</th>
<th>Total Marks (Min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>30601</td>
<td>Compiler Design</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>30602</td>
<td>Computer Networking &amp; Internet</td>
<td>CC</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>306031</td>
<td>Elective II : : (Any one from the following)</td>
<td>DCE</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>306032</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>306033</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>306034</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>306041</td>
<td>Elective III : : (Any one from the following)</td>
<td>DCE</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>306042</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>306043</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>306044</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30605</td>
<td>Web Applications Development</td>
<td>GE</td>
<td>4</td>
<td>60(21)</td>
<td>40(20)</td>
<td>100</td>
</tr>
<tr>
<td>30606</td>
<td>Lab I – Based on Elective III</td>
<td>LAB</td>
<td>2</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>30607</td>
<td>Lab II – Web Applications Development</td>
<td>LAB</td>
<td>2</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>30608</td>
<td>Comprehensive Viva</td>
<td>Viva</td>
<td>4</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>800</td>
</tr>
</tbody>
</table>

**TOTAL CREDITS : 28+28+28+24=108**  
**Grand Total : 800+800+800+600=3000**

CC = Core Course,  GE = Generic Elective, DCE = Discipline Centric Elective

### FOURTH SEMESTER

<table>
<thead>
<tr>
<th>Paper Code</th>
<th>Paper Name</th>
<th>Course Type</th>
<th>Credit</th>
<th>Theory Marks Max(Min)</th>
<th>CCE Marks Max(Min)</th>
<th>Total Marks (Min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>40601</td>
<td>Major Project / Dissertation Evaluation</td>
<td>CC</td>
<td>16</td>
<td></td>
<td></td>
<td>400(200)</td>
</tr>
<tr>
<td>40602</td>
<td>Major Project / Dissertation Internal Evaluation</td>
<td>CC</td>
<td>4</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td>40603</td>
<td>Comprehensive Viva</td>
<td>Viva</td>
<td>4</td>
<td></td>
<td></td>
<td>100(50)</td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>600</td>
</tr>
</tbody>
</table>

CC = Core Course,  GE = Generic Elective, DCE = Discipline Centric Elective

**TOTAL CREDITS : 28+28+28+24=108**  
**Grand Total : 800+800+800+600=3000**
Programme Objectives (POs):

Master of Computer Applications (MCA) is a full-time four-semester course, which includes one semester of project work in the fourth semester. The objective of MCA programme is to impart quality education in Computer Science and its applications, so that students are well prepared to face the challenges of the highly competitive computer industry. The course structure ensures overall development of the student, while concentrating on imparting technical skills required for computer/IT profession.

Programme Specific Outcomes (PSOs):

The programme is designed to

PSO1: enable the students to apply the computing and soft skills acquired in the MCA program for designing and developing innovative applications for the betterment of the society.

PSO2: Identify, formulate, research literature, and solve complex computing problem searching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

PSO3: provide exposure to techniques that would enable the students to design, implement and evaluate IT solutions.

PSO4: Demonstrate knowledge and understanding of the computing and management principles and apply these to one’s own work, as a member and leader in a team to manage projects and in multidisciplinary environments.

PSO5: Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.

PSO6: Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practices.

PSO7: To enable the students to meet the challenges of research and development in computer science and applications.
Course Outcomes:
On completion of this course, the student will be able to:

CO1: describe the basic organization of computer hardware.

CO2: represent and manipulate data – number systems, conversion between different number systems, perform binary arithmetic.

CO3: learn Boolean algebra, the language that simplifies communication in the world of computers.

CO4: understand formal logic, and to reason/infer interesting outcomes; formally prove validity and soundness of a statement.

CO5: design simple combinational and sequential logic circuits - flip-flops, counters, shift registers, adders, subtractor, multiplexer, de-multiplexer, and Arithmetic/Logic unit.

CO6: design simple programs in assembly language.

UNIT I
Computer Organization: Digital and Analog computers, Major components of a digital computer, Basic concepts of IT, concepts of Data & Information, data processing, history of computers (generations, type of languages), I/O devices, Storage devices, Software’s Types & its uses, Binary Number System and Conversion, Complements, Signed Binary Numbers, Binary Codes, Error detecting Code, Introduction of Assembler, Compiler & Interpreters

UNIT II
Computer Arithmetic: Binary representation of Negative Integers using 2’s complement and Signed magnitude representation, Fixed point Arithmetic operations on Positive and Signed (Negative) Integers like addition, subtraction, multiplication, Booth’s algorithm for multiplication, Division of positive and negative binary numbers. Boolean Algebra and Logic Gates: Basic Definitions, Basic Theorems and properties of Boolean algebra, Boolean Functions, Digital Logic gates.

UNIT III

UNIT IV
Introduction of 8085 Microprocessor: Architecture of 8085 processor, Register Architecture: Accumulator, Register and Flag Register, Addressing Modes: Direct memory addressing mode and Register direct Addressing Mode, Memory addressing capability of a CPU, Word length of a computer, Processing speed of a CPU, Working of CPU, Buses, Block Diagram of 8085

UNIT V
Introduction to Assembly Language Programming: Various Instructions Classifications: instruction Format, Opcode, Operand and Hex code. Instruction Operation Status, Various Instruction Sets: Data Transfer Group Instructions, Arithmetic Group Instructions, Logical Group Instruction, Branch Group Instructions: Conditional , Unconditional and Machine control Instructions.

Books Recommended:
1. Microprocessor Architecture, Programming and Applications with 8085/8080 by Ramesh S. Gaonkar.
Reference Book(s):
2. Microprocessor and Its applications by R Theagragajan,S Dhanapal
MCA I Sem-Course Code 10602: Mathematical Foundation of Computer Science

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to gain fundamental knowledge of:

CO1: mathematical structures (sets, relations and functions), and will be able to model real world situations mathematically.
CO2: principles of proportions & lattices.
CO3: Groups, Graphs and their applications in Computer Science.
CO4: discrete Numeric function and Recurrence relation.
CO5: growth of functions asymptotically.

UNIT-I
Sets, Relations and Functions: Sets, Subsets, Power sets, Complement, Union and Intersection, Demorgan’s law, Cartesian products, Relations, relational matrices, properties of relations, equivalence relation, functions, Injection, Surjection and Bijective mapping, Composition of functions, the characteristic functions and Mathematical induction

UNIT-II
Proportions & Lattices: Proposition & prepositional functions, Logical connections, Truth-values and Truth Table, the algebra of prepositional functions-the algebra of truth values-Applications (switching circuits, Basic Computer Components); Partial order set, Hasse diagrams, upper bounds, lower bounds, Maximal and minimal element, first and last element, Lattices, sub lattices, Isotonicity, distributive inequality, Lattice homomorphism, lattice isomorphism, complete lattice, complemented lattice, distribution lattice

UNIT-III
Groups: Binary Composition, Algebraic Structure, Algebraic properties or Group axioms, Monoid, Semigroup, Groupoid, Groups, Abelian Groups, Finite and Infinite Group, Integral power of an element, order of an element of a group, Transformations, Permutation and permutations group, Cyclic permutation, Even and odd permutations, Subgroups of a group, Cosets, Lagrange theorem, Cyclic groups, Normal subgroups

UNIT-IV
Graphs: Finite graphs, incidence and degree, isomorphism, sub graphs and union of graphs, connectedness, walk, paths, and circuits Eulerian graphs, tree properties of trees, pendant vertices in tree, center of tree, spanning trees and cut vertices, binary tree, matrix representation of graph, incidence and adjacency matrix and their properties, applications of graphs in computer science

UNIT-V
Discrete Numeric function and Recurrence relation: Introduction to discrete numeric functions and generating functions, introduction to recurrence relations and recursive algorithms, linear recurrence relations with constant coefficients, homogeneous solutions, particular solutions and total solutions

Books Recommended:

### Course Outcomes:

On completion of this course, the student will be able to:

**CO1**: describe basic functions of an Operating System.

**CO2**: distinguish between different types of Operating Systems so as to use each of them most efficiently in the respective application areas.

**CO3**: describe different techniques for managing computer resources like CPU, memory, file and devices.

**CO4**: implement simple algorithms for managing computer resources.

### UNIT-I

Introduction: Evolution of operating systems (History of evolution of OS with the generations of computers), Types of operating systems, Multitasking, Timesharing, Multithreading, Multiprogramming and Real time operating systems, Operating system concepts and structure, Layered Operating Systems, Monolithic Systems. Processes: The Process concept, The process control block, Systems programmer's view of processes, Operating system services for process management, Scheduling algorithms and types.

### UNIT-II

Memory Management: Memory management without swapping or paging, Concepts of swapping and paging, Page replacement algorithms namely Least recently used, Optimal page replacement, Most recently used, Clock page replacement, First in First out (This includes discussion of Belady’s anomaly and the category of Stack algorithms), Modeling paging algorithms, Design issues for paging system, Segmentation, Segmented Paging, Paged Segmentation

### UNIT-III

Inter-process Communication and Synchronization: The need for inter-process synchronization, Concept of mutual exclusion, binary and counting semaphores, hardware support for mutual exclusion, queuing implementation of semaphores, Classical problems in concurrent programming, Dining Philosopher’s problem, Bounded Buffer Problem, Sleeping Barber Problem, Readers and Writers problem, Critical section, critical region and conditional critical region, Monitors and messages. Deadlocks: Concepts of deadlock detection, deadlock prevention, deadlock avoidance; Banker’s Algorithm

### UNIT-IV

File System: File systems, directories, file system implementation, security protection mechanisms; Input/output: Principles of I/O Hardware: I/O devices, device controllers, direct memory access, Principles of I/O software: Goals interrupt handlers, device drivers, and device independent I/O software. **User space I/O Software.** Disks: Disk hardware, Disk scheduling algorithms (namely First come first serve, shortest seek time first, SCAN, C-SCAN, LOOK and C-LOOK algorithms) Error handling, track-at-a-time caching, RAM Disks. Clocks: Clock hardware, memory-mapped terminals, I/O software.

### UNIT-V


### Books Recommended:

MCA I Sem - Course Code 10604 : Data Base Management System
Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to:

CO1: understand basic database concepts, including the structure and operation of the relational data model.
CO2: apply logical database design principles, including E-R diagrams, conversion of ER diagrams to relations.
CO3: understand the concepts of integrity constraints, relational algebra, relational domain & tuple calculus, data normalization.
CO4: construct simple and moderately advanced database queries using Structured Query Language (SQL).
CO5: understand the emerging fields in database.

UNIT-I
Introduction: Advantage of DBMS approach, various view of data, data independence, schema and subschema, primary concepts of data models, Database languages, transaction management, Database administrator and users, data dictionary, overall system architecture. ER model: basic concepts, design issues, mapping constraint, keys, ER diagram, weak and strong entity sets, specialization and generalization, aggregation, inheritance, design of ER schema, reduction of ER schema to tables.

UNIT-II
Domains, Relations and Keys: domains, relations, kind of relations, relational database, various types of keys, candidate, primary, alternate and foreign keys. Relational Algebra & SQL: The structure, relational algebra with extended operations, modifications of Database, idea of relational calculus, basic structure of SQL, set operations, aggregate functions, null values, nested sub queries, derived relations, views, modification of Database, join relations, DDL in SQL, Introduction of Database Design, SDLC, DDLC, Automated Design Tools

UNIT-III
Functional Dependencies and Normalization: basic definitions, trivial and non trivial dependencies, closure set of dependencies and of attributes, irreducible set of dependencies, introduction to normalization, non loss decomposition, FD diagram, first, second, third Normal forms, dependency preservation, BCNF, multivalued dependencies and fourth normal form, Join dependency and fifth normal form.

UNIT-IV
Database Integrity: general idea. Integrity rules, domain rules, attribute rules, relation rules, Database rules, assertions, triggers, integrity and SQL. Transaction, concurrency and Recovery: basic concepts, ACID properties, Transaction states, implementation of atomicity and durability, concurrent executions, basic idea of serializability, basic idea of concurrency control, basic idea of deadlock, failure classification, storage structure types, stable storage implementation, data access, recovery and atomicity- log based recovery, deferred database modification, immediate database modification, checkpoints. Distributed Database: basic idea, distributed data storage, data replication, data fragmentation horizontal, vertical and mixed fragmentation

UNIT-V
Emerging Fields in DBMS, Object oriented Databases, Data Warehousing, Database on www, multimedia Databases-difference with conventional DBMS, issues, similarity based retrieval, continuous media data, multimedia data formats, video servers. Storage structure and File organizations, Indexing, Network and hierarchical models, DBTG model, implementations, tree structure diagram, implementation techniques, comparison of the three models, Basics of Decision Support System, Introduction and Installation of MYSQL, SQLLITE, MSSQL, MongoDB, PHP Introduction, Installing PHP, PHP and MYSQL, Case Study : Database Design of an Internet Bookshop

Books Recommended:
4. B.C. Desai. “An introduction to Database systems” BPB
5. Raghurama Krishnan “Database Systems” TMH
MCA I Sem - Paper Code-10605 : Problem Solving using C & C++

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to:
CO1: learn basic concepts of problem solving using programming language.
CO2: apply procedure oriented & object-oriented paradigm for problem solving.
CO3: select a suitable programming construct and in-built data structure for the given problem.
CO4: design, develop, document, and debug modular programs.

UNIT–I
C language programming: Flowchart, Algorithm, Introduction to C language, Simple I/O functions (scanf, printf, gets, puts, getchar, getche, getch); Data types in C, Assignment statement, Arithmetic, Operators, Precedence of operators. Control structure: The if-else statements, nesting of if-else, switch statement, Loops: while and do-while loop, the for loop, Functions: User defined functions, Returning a value from a function, Local and Global variables, Parameters, Type declaration of a function, Functions with more than one parameters, Storage classes

UNIT–II
Arrays & Pointers: Declaration and initialization; the break and continue statement; String and Character arrays, operations with arrays; searching in array (linear and binary). Sorting an array, String & String functions: sprintf, strcpy, scanf, strcat, stlen, malloc, strcmp. Two dimensional array. Pointers: The concept of pointers, pointer arithmetic, passing pointers as parameters, pointer & arrays, Pointer to pointers, Array of pointers to strings.

UNIT–III
Structures: The concept of structure, Initializing, Arrays of structures, Arrays within structures, Structures within Structures, passing structures to function; unions.
Files: Files in ‘C’, Modes for files; Functions used in files (getc, putc, fopen, fclose, fscanf, fread, fprintf, fseek, ftell, rewind), text versus binary files; command line arguments; Preliminaries of C preprocessor Directives, (#define, #undef, #include, #ifdef, ifndef, #endif, #else, #if).

UNIT–IV
Introduction to OOP: Basic concepts of OOP: Object, Classes, Inheritence, Polymorphism, Reusability; Benefits & applications of OOP, Characters used in C++. Basic data types, user defined data types, use of conditional and looping statements in C++. Arrays in C++. Reference variable, Functions: prototypes, default arguments, const arguments in functions, Inline functions, call by value, call by reference, function overloading.
Classes and objects: Declaring a class, defining an object, data hiding and encapsulation, public and private data members & functions, friend function. Pointer to data member, pointer to member function and pointer to object, virtual function.

Unit-V
Constructors & Destructors: Parametrized constructors, multiple constructor in a class, copy constructors, object as function arguments, returning objects, the this pointer, memory allocation for objects. Operator Overloading: Unary and binary operators. Inheritance: Inheritance and derivation, single, multilevel, multiple, hierarchical & hybrid inheritance, Overriding functions, virtual function; Manipulators, managing output with manipulators, user defined manipulators with arguments; Streams: C++ streams, stream classes; Files: Classes for file stream operations, file I/O with streams.

Books Recommended:
1. Gottfried, Programming with C, TMH
2. E. Balagurusamy, Programming in ANSI C, TMH
3. Y. Kanetkar, Let us C, BPB
4. Y. Kanetkar, Let us C++, BPB
5. E. Balagurusamy, Object Oriented Programming with C++, TMH
MCA II Sem-Paper Code-20601 : Data Structure & Analysis of Algorithms

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to:
CO1: identify best suited data structure for the problem at hand.
CO2: identify the programming constructs to optimize the performance of the data structure in different scenarios.
CO3: describe the algorithm design techniques: iteration, divide and conquer, dynamic programming, greedy approach algorithms.
CO4: analyse the strengths and weaknesses of each technique.
CO5: identify and apply technique(s) suitable for simple applications.
CO6: appreciate that certain problems are too hard to admit fast solutions

Unit – I
Data Structure: types, operations on data structures, Algorithm analysis, time space complexities; Stack: Contiguous implementation of stack, PUSH & POP, applications of stack: Various polish notations – infix, prefix, postfix, conversion using stack; Queue: implementation of queue, operations on queue, priority queue, Linear queue and circular queue, various operation on queue.

Unit – II
General List: list and its contiguous & linked implementation, its drawback; singly linked list-operations on it; doubly linked list-operations on it; circular linked list; applications of linked list,
Trees: Definition – height, depth, order degree, etc; Binary Tree, complete binary tree, implementation of Binary tree, Tree traversal algorithms – preorder, inorder & post order, Binary search tree, operations on binary tree, application of binary tree.

Unit – III
Graph: related definition, implementation of graph, traversal algorithms - depth first search, breadth first search; minimum spanning tree, shortest path algorithms, Searching: Sequential search, binary search, indexed sequential search, Hashing, hash methods, collisions & its resolution techniques. Sorting: bubble sort, selection sort, heap sort, insertion sort and tree sort

Unit – IV
Divide and Conquer: Structure of divide and conquer algorithm, Merge sort, Quick sort; Asymptotic Notation; Greedy Method: Overview of the greedy paradigm examples of exact optimization solution(minimum cost spanning tree), Approximate solution (Knapsack problem), Single source shortest paths. Branch and bound; 0/1 Knapsack problem, Traveling Salesman Problem.

UNIT- V

Books Recommended :
1. Kruse R.L. Data Structures and Program Design in C; PHI
3. Trembly “Introduction to Data Structure with Applications”.
4. TennenBaum A.M. & others: Data Structures using C & C++; PHI
6. Ullman ”Analysis and Design of Algorithm” TMH
MCA II Sem-Paper Code-20602 : Software Engineering Methodologies

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of the course, the student is expected to:

CO1: demonstrate an understanding of software engineering layered technology and software process models that provide a basis for the software development lifecycle.

CO2: describe software/system requirements and understand the processes involved in the discovery and documentation of these requirements.

CO3: practice system modeling techniques and object-oriented design for software development.

CO4: test software using verification and validation, static analysis, reviews, inspections, and audits.

CO5: appreciate software project management that includes project planning, project estimation techniques, risk management, quality management, and configuration management.

CO6: work as an individual and/or in team to develop and deliver quality software.

UNIT - I
System concepts and Information system environment: The system concept, characteristics of system, elements of system, The System Development Life Cycle, The Role of System Analyst. Introduction system planning & initial investigation, various information gathering tools, feasibility study, structures tools of system analysis, various methods of process design, form design methodologies, introduction to information system testing, quality assurance.

UNIT - II

UNIT-III

UNIT-IV

UNIT-V

Books Recommended:
3. Ian Sommerville : Software Engineering 6/e (Addison-Wesley)
Course Outcomes:
After the completion of this course, students will be able to:
CO1: understand the concept of image formation as realized by human visual System.
CO2: illustrate the digitization process of images and related algorithms for drawing basic geometric figures in the 2D display devices.
CO3: describe architecture of basic Input/ Output devices and their underlying working principles along with various primitives for drawing shapes.
CO4: apply fundamental mathematics in producing spatial 3D-image of an object in an inherently 2D display device.
CO5: understand the basics of OpenGL API and to manipulate graphics & images.

UNIT-I

UNIT-II
Attributes of output primitives, line style, color and intensity, Area filling algorithms, Scan line algorithm, boundary fill, flood fill algorithm, Antialiasing techniques. Two dimensional transformations; translation, scaling, rotation, reflection sheering, composite transformation, transformation commands, character generation.

UNIT-III
Viewing coordinates, Window, view port, clipping. Window to view port transformation, line clipping algorithm; Cohen Sutherland, polygon clipping; Sutherland Hodgman algorithm, 3D clipping: Normalized view volumes, view port clipping, clipping in homogeneous coordinates. Illumination model: Light sources, diffuse reflection specular reflection, reflected light, intensity levels, surface shading; phong shading gouraud shading, color models like RGB, YIQ, CMY, HSV etc.

UNIT-IV
3-D Viewing: Three-dimensional concepts, 3D display techniques, 3D representation polygon & curved surfaces. Design of curves & surfaces- Bezier’s Method, B-spline method, 3D transformation transition, scaling, composite transformation rotation about arbitrary axis, projections: Parallel & Perspective, Hidden surface and line removal; back face removal, depth buffer and scan line methods.

UNIT-V

Books Recommended:
Course Outcomes:
On completion of this course, the student will be able to:
CO1: describe various approaches to Artificial Intelligence.
CO2: design intelligent agents.
CO3: describe and apply concepts, methods, and theories of search, heuristics, games, knowledge representation, planning.
CO4: acquire basics knowledge of Natural language processing.
CO5: understand the limitations of Artificial Intelligence techniques.

UNIT-I
General Issues and Overview of AI: The AI problems, what is an AI technique, Characteristics of AI applications. Introduction to LISP programming: Syntax and numeric functions, Basic list manipulation functions, predicates and conditionals, input output and local variables, iteration and recursion, property lists and arrays.

UNIT-II

UNIT-III
Knowledge Representations: First order predicate calculus, skolemization, resolution principle & unification, interface mechanisms, horn's clauses, semantic networks, frame systems and value inheritance, scripts, conceptual dependency.

UNIT-IV
Natural Language processing: Parsing techniques, context free grammar, recursive transition nets (RNT), augmented transition nets (ATN), case and logic grammers, symantc analysis. Game playing: Minimax search procedure, alpha-beta cutoffs, additional refinements. Planning: Overview an example domain the block world, component of planning systems, goal stack planning, non linear planning.

UNIT-V

Books Recommended:
Course Outcomes:
On completion of this course, the student will be able to:
CO1 : understand the architecture and infrastructure of cloud.
CO2 : learn the resource virtualization technique.
CO3 : build the appropriate file system and database.
CO4 : understand cloud security and challenges.
CO5 : evaluate third party cloud services for a real world problem.

Unit-I
Introduction: Historical development, Vision of Cloud Computing, Characteristics of cloud computing as per NIST, Cloud computing reference model, Cloud computing environments, Cloud services requirements, Cloud and dynamic infrastructure, Cloud Adoption and rudiments. Overview of cloud applications: ECG Analysis in the cloud, Protein structure prediction, Gene Expression Data Analysis, Satellite Image Processing, CRM and ERP, Social networking.

Unit-II

Unit-III
Cloud Management & Virtualization Technology: Resiliency, Provisioning, Asset management, Concepts of Map reduce, Cloud Governance, High Availability and Disaster Recovery. Virtualization: Fundamental concepts of compute, storage, networking, desktop and application virtualization. Virtualization benefits, server virtualization, Block and file level storage virtualization, Hypervisor management software, Infrastructure Requirements, Virtual LAN(VLAN) and Virtual SAN(VSAN) and their benefits.

Unit-IV

Unit-V

Books Recommended:
MCA II Sem-Paper Code-20605 : JAVA Programming & Technologies

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to:

**CO1**: understand the object-oriented concepts – Classes, Objects, Inheritance, Polymorphism – for problem solving.
**CO2**: design, implement, document, test, and debug a Java application consisting of multiple classes.
**CO3**: handle program exceptions.
**CO4**: handle input/output through files.
**CO5**: create Java applications with graphical user interface (GUI).

UNIT-I
The Java Environment: History of Java; Comparison of Java and C++; Java as an object oriented language; Basic idea of application and applet; Basics: Data types; Operators- precedence and associativity; Type conversion; Java control statements; arrays; memory allocation and garbage collection in java. Object Oriented Programming in Java: Class & Object; Packages; scope and lifetime; Access specifies; Constructors; Copy constructor; this pointer; finalize () method; arrays; Memory allocation and garbage collection in java, keywords Inheritance : Inheritance basics, method overriding, dynamics method dispatch, abstract classes.

UNIT-II
Interfaces : defining an interface, implementing & applying interfaces, extending interfaces. Multithreading and Exception Handling: Basic idea of multithreaded programming; The lifecycle of a thread; Creating thread with the thread class and runnable interface; Thread synchronization; Thread scheduling; Producer-consumer relationship; Daemon thread, Selfish threads; Basic idea of exception handling; The try, catch and throw; throws Constructor and finalizers in exception handling; Exception Handling.

UNIT-III
Applets: the class hierarchy for applets; Life cycle of applet; HTML Tags for applet. The AWT: The class hierarchy of window fundamentals; The basic user interface components Label, Button, Check Box, Radio Button, Choice menu, Text area, Scroll list, Scroll bar; Frame; Layout managers; The Java Event Handling Model, ignoring the event, Self contained events, Delegating events; The event class hierarchy; The relationship between interface, methods called, parameters and event source; Adapter classes; Event classes.

UNIT-IV
Input/Output : Exploring Java I/O., stream classes The Byte stream : Input stream, output stream, file input stream, file output stream, print stream, Random access file, the character streams, Buffered reader, buffered writer, print writer, serialization. JDBC: JDBC ODBC bridge; The connectivity model; The driver manager; Navigating the resultset object contents; java.sql Package; The JDBC exception classes; Connecting to Remote database.

UNIT-V
Networking & RMI: Java Networking : Networking Basics : Socket, Client server, reserved sockets, proxy servers, Inet address, TCP sockets, UDP sockets ; RMI for distributed computing; RMI registry services; Steps of creating RMI Application and an example. Collections: The collections framework, collection interfaces, collection classes.

Books Recommended :
2. Deitel “Java- How to Program;” Pearson Education, Asia
3. Horstmann & Cornell “Core Java 2” (Vol I & II ) , Sun Microsystems
4. Ivan Bayross “Java 2.0” : BPB publications
5. Ivor Horton’s “Beginning Java 2, JDK 5 Ed., Wiley India
Course Outcomes:
On completion of this course, the student will be able to:
CO1: describe how different phases of a compiler work.
CO2: understand formal languages and automata.
CO3: implement top down and bottom up parsing algorithms.
CO4: use compiler tools like lex for implementing syntax directed translator.
CO5: learn implementation of block structure languages.

Unit I
Compilers & Interpreters: aspects of compilation, structure of compiler, compilation of expression compilation of control structures, interpreters. Software tool. Linker & Loaders: Relocation & linking concepts, design of linkers, self relocating programs, linking for overlays, loaders: A two pass loader scheme.

Unit II

Unit III
Context free grammar, derivation of parse tree, capabilities of CFGs, normal form for CFG, Pushdown automata, relation of PDA with CFG, capabilities of CFG, Parser, Shift reduce parsing, operator precedence parsing top down parsing, Predictive parsing, LR parser, the canonical collection of LR(0) items, constructing SLR parsing table, constructing canonical LR parsing table, constructing LALR parsing table.

Unit IV
Syntax direct translation schemes, implementation of syntax directed translators, intermediate code, postfix notation, parse tree and syntax tree, three address code, quadruples and triples, translations of assignment statement, Boolean expression, statements that alter the flow of control, cost fix translations, translation with top down parser.

Unit V
Symbol table, the contents of symbol tables, data structure for symbol tables, representing scope information, run time storage administration, implementation of a simple stack allocation schemes, implementation of block structure languages, storage for block structured languages.

Books Recommended:
1. Principal of compiler design by Alfred V. Aho, Jeffrey D. Ulman.
Course Outcomes:
On completion of this course, the student will be able to:
CO1: learn the basics of Computer network Technologies.
CO2: understand the fundamentals of types of transmission mediums and interfacing standards along with the current edge of the data communication techniques.
CO3: learn flow control and error control techniques and Computer Network protocols at Conceptual level.
CO4: learn WAN and TCP/IP.
CO5: learn the architecture & protocols of email and www.

UNIT-I

UNIT-II
Data Security and Integrity: Parity Checking Code, Cyclic redundancy checks (CRC), Hamming Code, Protocol Concepts, Basic flow control, Sliding window protocol - Go-Back-N protocol and selective repeat protocol, Protocol correctness- Finite state machine

UNIT-III

UNIT-IV

UNIT-V

Books Recommended:
2. Forouzan “Data Communication and Networking 3ed”, TMH
MCA III Sem-Paper Code-306031 : Cryptography & Network Security

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to:
CO1: learn classical encryption techniques and block cipher modes of operation.
CO2: implement a symmetric and asymmetric cryptographic methods
CO3: learn Message authentication and Hash functions.
CO4: describe the role and implementation of digital signatures.
CO5: understand IP security, Web security and system security.

UNIT-I
Classical Encryption Techniques: Symantec Cipher model, substitution Techniques, transposition techniques, rotor machines, steganography. Block Ciphers and the Data Encryption standards: Simplified DES, block cipher principles, the data encryption standard, the strength of DES, block cipher design principles, block cipher modes of operation, Triple DES.

UNIT-II
Confidentiality using symmetric encryption: Placement of Encryption function, traffic confidentiality, key distribution; Public key Encryption, Public key cryptography and RSA: Principles of Public key cryptosystems, the RSA algorithm; Key Management other public key cryptosystems: Key management, Diffie-Hallman key exchange algorithm.

UNIT-III

UNIT-IV

UNIT-V

Books Recommended:
5. Roberta Bragg “ Mark Rhodes, Ousley & Keith Strassberg Network Security : The Complete
MCA III Sem-Paper Code-306032 : Mobile Computing
Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Outcomes:
On completion of this course, the student will be able to:
CO1 : learn multiple access technology for Wireless Communication.
CO2 : understand the concept of mobile data communication.
CO3 : learn Digital Cellular Systems and Standards.
CO4 : describe Components and working of Wireless LAN.
CO5 : understand Bluetooth technology & WLL architecture.

UNIT-I
Overview of OSI Model: Significance of layered Model, PDUs, SDUs, IDUs, Higher layer Protocols, Switching and Components : Introduction, Applications, history of wired & wireless Communication systems; Radio Transmission : frequencies, signal propagation, antenna, types of modulation, FHSS, DSSS; Multiple Access technology for Wireless Communication : FDMA, TDMA, CDMA Cellular System : Introduction, types.

UNIT-II
Mobile Data Communication : Cellular Telephony; Structure, Fading, Small scale fading, Multi-path Fading, Speech Coding, Error Coding and Correction, Hand off Management, Switching and authentication, MTSO interconnections, frequency hopping, frequency reuse; Circuit Switched Data Services & Packet Switched Data Services on Cellular Networks, Personal Communication Systems (PCS) Architecture, Digital Enhanced Cordless Telecommunications (DECT), Personal Access Communications System (PACS).

UNIT-III

UNIT-IV

UNIT-V
Introduction to. Bluetooth technology. Wireless in Local Loop (WLL) architecture, products. Satellite as a switch, Components of VSAT system, VSAT topologies access schemes.

Books Recommended :
1. Jochen Schiller "Mobile Communication", Pearson Education
2. Yi -Bing Lin and Imrich Chlamtac "Wireless and Mobile Network Architectures", Wiley India.
3. Raj Pandaya "Mobile and Personal Communication System & Services".
Course Outcomes:
On completion of this course, the student will be able to:

CO1 : understand quality management processes.
CO2 : understand the importance of standards in the quality management process and role of SQA function in an organization.
CO3 : gain knowledge of software quality assurance.
CO4 : understand the need and purpose of software testing.
CO5 : learn the five views of software quality.

Unit-I
Introduction : Software Quality, Role of testing, v & v, objectives and issues of testing, Testing activities and levels, Sources of Information for Test Case Selection, White-Box and Black-Box Testing, Test Planning and Design, Monitoring and Measuring Test Execution, Test Tools and Automation, Test Team Organization and Management; Unit Testing: Concept, Static Unit Testing, Defect Prevention, Dynamic Unit Testing, Mutation Testing, Debugging.

Unit-II

Unit-III

Unit-IV

Unit-V

Books Recommended:
MCA III Sem-Paper Code-306034 : Internet Of Things
Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Objectives:
On completion of this course, the student will be able to:
CO1 : Understand the IOT Terminology and Technology.
CO2 : Describe IOT applications.
CO3 : Analyze Protocol standardization for IOT.
CO4 : Perform an analysis of IOT security issues.
CO5: Identify the role of cloud computing in IOT.

Unit-I
Introduction – Digital Electronics, Logical gates and its working, Types of sensors: Temperature sensor (LM35,RTD,Thermocouple), Light sensor(photodiode, optocoupler), Distance and range sensor (IR,LVDT), Accelerometer sensor, Touch screen sensor.

Unit-II

Unit-III
IoT Smart X Application - Smart Cities, Smart Energy & Smart Grid, Smart Mobility & transport, Smart Home, Smart Building & Infrastructure, Smart Factory & Manufacturing, Smart Health, Smart Logistics & Retail, Embedded suite for IoT Physical device – Arduino / Raspberry Pi Interfaces, Hardware requirement of Arduino / Pi, Connecting remotely to the Arduino /Raspberry Pi, GPIO Basics

Unit-IV
Protocols in IOT: RFID: Introduction, Principle of RFID, Components of an RFID system, RFID Protocols & NFC protocols, CoAP, XMPP, AMQP, MQTT
Resource Management In The Internet Of Things: Clustering, Software Agents, Clustering Principles in an Internet of Things Architecture, Design Guidelines, Software Agents for Object Representation, Data Synchronization.

Unit-V
Internet of things Challenges: Vulnerabilities of IoT, Security, Privacy & Trust for IoT, Security requirements Threat analysis, Use cases and misuse cases, Introduction to cloud computing, Role of Cloud Computing in IoT, Cloud-to-Device Connectivity, Cloud data management, cloud data monitoring, Cloud data Exchange.

Books Recommended :
Course Objectives:
On completion of this course, the student will be able to:
CO1: learn .NET Technology.
CO2: understand the Visual Basic fundamentals.
CO3: describe the classes, interfaces & arrays.
CO4: learn creation of window forms & controls.
CO5: understand file handling and graphics in VB.

UNIT-I
Introduction to .NET Technology, Introduction to VB.NET, Building VB.net Application, IDE Dot.NET, Evolution of Dot.net Framework, Keywords, Statement, Variables, Enumerable, Constant, Data Types, Conversion, Operators, Comments, Decision Making, Looping, Array, Handling Strings, Strings Function.

UNIT-II

UNIT-III
Classes and Objects: Types, Field, Properties, Methods and Events, Class vs Object, Members Overloading, Overriding, Creating Class, Object, Structure & Modules, Accessing Modifiers, Shadowing, Creating Interfaces, Polymorphism, Early and Late binding, Multiple Interface, Using MyBase and MyClass Keyword, Inheritance based Polymorphism and Interface Based Polymorphism.

UNIT-IV
Window Forms: Creating Window Forms, Controls to Form, Setting Title bars, Dialog Boxes, Handling Mouse Events, Handling Key Press Events, Controls Classes: Textbox, Rich Textbox, Labels, Link Labels, Buttons, Checkbox, Panels, Group Boxes, Radio Button, Drop Down, List Boxes, Combo Box, Scroll Bars, Pickers, Tool Tips, Timers, Menu, Min & Max Button, Image, Toolbars, Popup Menu, Setup Dialog, Progress bar, Status Bar, Tab Controls.

UNIT-V
File Handling: File Opening and Creating, Writing Files, Reading Binary Data, Directory class, Files Class, Graphics: Using Brush class, Using Pen Class, Graphics Class, Data Access and ADO.NET: Creating Data set, Populating Dataset, Displaying data in Grids, Data access using Data Adapters Controls, Binding Data to Controls, Using Data Views.

Books Recommended:
4. Alex Homer, Dave Sussman “Professional ASP.NET1.1” Wiley Dreamtech
5. Bill Evzen,Bill Hollis “Professional VB.NET 2003” Wiley Dreamtech
7. Chris Ullman, Kauffman “Beg. ASP.NET1.1 with VB.NET 2003” Wiley Dreamtech
Course Objectives:
On completion of this course, the student will be able to:
CO1 : understand different data types used in python.
CO2 : get better understanding of different types of control structures.
CO3 : use different data structures for different problem domains.
CO4 : apply different object oriented features for solving real world problems.
CO5 : develop different web based applications.

UNIT -I
Installing Python, Simple program using Python, Expressions and Values, Variables and Computer Memory, error detection, Multiple line statements, Designing and using functions, functions provided by Python, Tracing function calls in memory model, omitting return statement. Working with Text: Creating Strings of Characters, Using Special Characters in Strings, Creating a Multiline String, Printing Information, Getting Information from the Keyboard.

UNIT-II
A Boolean Type , Choosing Statements to Execute, Nested If Statements, Remembering the Results of a Boolean Expression Evaluation, A Modular Approach to Program Organization, Importing Modules, Defining Your Own Modules, Testing Code Semi automatically Grouping Functions Using Methods: Modules, Classes, and Methods, Calling Methods the Object-Oriented Way, Exploring String Methods, Underscores.

UNIT-III

UNIT-IV
Files: Kinds of files, Opening a File, Techniques for Reading Files, Files over the Internet, Writing Files and Writing Algorithms That Use the File-Reading Techniques, Multiline Records. Storing Data Using Other Collection Types: Storing Data Using Sets, Storing Data Using Tuples, Storing Data Using Dictionaries, Inverting a Dictionary, Using the In Operator on Tuples, Sets, and Dictionaries, Comparing Collections.

UNIT-V

Books Recommended :
2. John Zelle, Python Programming: An Introduction to Computer Science
4. O’Relly, Head First Python: A Brain-Friendly Guide, by Paul Barry
MCA III Sem-Paper Code-306043 : Data Warehousing and Mining

Course Objectives:
On completion of this course, the student will be able to:
CO1 : learn the data mining functionalities.
CO2 : understand and exhibit the basics of data warehousing and multi-dimensional modeling.
CO3 : describe data preprocessing.
CO4 : understand classification, clustering, frequent pattern analysis and regression.
CO5 : learn cluster analysis and DM tools.

UNIT – I
Motivation, importance, Data type for Data Mining : relational Databases, Data Warehouses, Transactional databases, advanced database system and its applications, Data mining Functionalities: Concept/Class description, Association Analysis classification & Prediction, Cluster Analysis, Outlier Analysis, Evolution Analysis, Classification of Data Mining Systems, Major Issues in Data Mining.

UNIT – II
Data Warehouse and OLAP Technology for Data Mining : Differences between Operational Database Systems and Data Warehouses, a multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Data Cube Technology.

UNIT- III
Data Preprocessing: Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation. Data Mining Primitives, Languages, and System Architectures, Concept Description: Characterization and Comparison, Analytical Characterization.

UNIT – IV
Mining Association Rules in Large Databases: Association Rule Mining: Market Basket Analysis, Basic Concepts, Mining Single-Dimensional Boolean Association Rules from Transactional Databases: the Apriori algorithm, Generating Association rules from Frequent items, Improving the efficiency of Apriori, Mining Multilevel Association Rules, Multidimensional Association Rules, Constraint-Based Association Mining.

UNIT – V
Classification & Prediction and Cluster Analysis: Issues regarding classification & prediction, Different Classification Methods, Prediction, Cluster Analysis, Major Clustering Methods, Applications & Trends in Data Mining: Data Mining Applications, currently available tools.

Books Recommended:
2. Berson “Dataware housing, Data Mining & DLAP, @004, TMH.
Course Objectives:
On completion of this course, the student will be able to:
CO1: learn Big data and its characteristics.
CO2: understand best practices for Big data Analytics and Integration tools.
CO3: describe data modeling.
CO4: learn elementary data analysis.
CO5: understand basics of visualization.

UNIT-I
Introduction of big data, Big data characteristics - Volume, Veracity, Velocity, and Variety – Data, Appliance Challenges and Issues, Case for Big data, Big data sources, Features of data, Evolution of Big data, Best Practices for Big data Analytics and Integration tools.

UNIT-II
Introduction to Data Modeling. Data Models Used in Practice: Conceptual data models, Logical data models, Physical data models, Common Data Modeling Notations, How to Model Data: Identify entity types, Identify attributes, Apply naming conventions, Identify relationships, Apply data model patterns, Assign keys, Normalize to reduce data redundancy.

UNIT-III

UNIT-IV
Introduction to Bayesian Modeling: Bayes Rule, Probabilistic Modeling Introduction to Predictive Analytics: Simple Linear regression, Multiple Linear regression, Logistic Linear Regression.

UNIT-V
Visualization: History of Visualization, Goals of Visualization, Scientific Visualization, Information Visualization, Visual Analytics, Impact of visualization Introduction to Data Processing, Map Reduce Framework, Hadoop, HDFS, S3, Hadoop Distributed file systems, Apache Mahout, Hive, Sharding, Hbase, Impala, Case studies: Analyzing big data with twitter, Big data for Ecommerce, Big data for blogs.

Books Recommended:
MCA III Sem-Paper Code-30605 : Web Applications Development

Credit 4; Theory Max/Min(60/21), CCE Max/Min(40/20)

Course Objectives:
On completion of this course, the student will be able to:
CO1 : understand web architecture.
CO2 : learn HTML & CSS.
CO3 : apply different modern technologies used for real-time client server application.
CO4 : develop different attractive and interactive web pages.
CO5 : learn basics of android application deployment environment.

UNIT - I

UNIT - II

UNIT - III
JSP Basics: JSP lifecycle, Directives, scripting elements, standard actions, implicit objects. Connection of JSP and Servlet with different database viz. Oracle, MS-SQL Server, MySQL. java.sql Package. Querying a database, adding records, deleting records, modifying records. Type of Statement.

UNIT - IV
Separating Business Logic and Presentation Logic, Building and using JavaBean. Session handling in JSP, Types of errors and exceptions handling. Introduction to Web Services, MVC Architecture, Struts and Hibernate.

UNIT - V

Books Recommended :