

**Faculty of Science**  
**SATAVAHANA UNIVERSITY - KARIMNAGAR**  
**MCA - MASTER OF COMPUTER APPLICATIONS – (2018-19)**  
Under Choice Based Credit System

**SU - M.C.A - I SEMESTER**

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 101T	Computer Organization & Architecture	4	--	20	80	100	4	3 Hrs
MCA 102T	Business Systems and Applications	4	--	20	80	100	4	3 Hrs
MCA 103T	Computer Programming with C	4	--	20	80	100	4	3 Hrs
MCA 104T	Discrete Mathematical Structure	4	--	20	80	100	4	3 Hrs
MCA 105T	Business English and communication	4	--	20	80	100	4	3 Hrs
MCA 106P	Micro Programming and Architecture – LAB	--	4	10	40	50	2	3 Hrs
MCA 107P	C Programming – LAB	--	4	10	40	50	2	3 Hrs
MCA 108P	Business Presentation and Language – LAB	--	4	10	40	50	2	3 Hrs
MFCE 101T*	Professional Communication	2	--	10	40	50	2	2 Hrs
<b>TOTAL</b>		<b>22</b>	<b>12</b>	<b>140</b>	<b>560</b>	<b>700</b>	<b>28</b>	

**SU- M.C.A - II SEMESTER**

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 201T	Data Communication & Computer Networks	4	--	20	80	100	4	3 Hrs
MCA 202T	Programming in Python	4	--	20	80	100	4	3 Hrs
MCA 203T	Data Structures Using C++	4	--	20	80	100	4	3 Hrs
MCA 204T	Data Base Management Systems	4	--	20	80	100	4	3 Hrs
MCA 205T	Modern Operating Systems	4	--	20	80	100	4	3 Hrs
MCA 206P	Python – LAB	--	4	10	40	50	2	3 Hrs
MCA 207P	Data Structures Using C++ - LAB	--	4	10	40	50	2	3 Hrs
MCA 208P	DBMS – LAB	--	4	10	40	50	2	3 Hrs
MFCF 201T*	Fundamentals of Computers & Office Automation	2	--	10	40	50	2	2 Hrs
<b>TOTAL</b>		<b>22</b>	<b>12</b>	<b>140</b>	<b>560</b>	<b>700</b>	<b>28</b>	

\*Every student must pass this paper since it is mandatory. However the credits will not included in the calculation of SGPA and CGPA

### SU - M.C.A - III SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 301T	Linux Programming	4	--	20	80	100	4	3 Hrs
MCA 302T	Software Engineering	4	--	20	80	100	4	3 Hrs
MCA 303T	Java Programming	4	--	20	80	100	4	3 Hrs
MCA 304T	Business Management Logic & Design	4	--	20	80	100	4	3 Hrs
MCA 305T	R Programming	4	--	20	80	100	4	3 Hrs
MCA 306P	Linux – LAB	--	4	10	40	50	2	3 Hrs
MCA 307P	Software Engineering – LAB	--	4	10	40	50	2	3 Hrs
MCA 308P	Java Programming – LAB	--	4	10	40	50	2	3 Hrs
MFCE 301T*	Seminar	2	--	50	--	50	2	---
<b>TOTAL</b>		<b>22</b>	<b>12</b>	<b>180</b>	<b>520</b>	<b>700</b>	<b>28</b>	

### SU - M.C.A - IV SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 401T	Web Technologies	4	--	20	80	100	4	3 Hrs
MCA 402T	Data Mining	4	--	20	80	100	4	3 Hrs
MCA 403T	Network Security	4	--	20	80	100	4	3 Hrs
MCA 404T	<b>ELECTIVE-I</b> A. Object Oriented Analysis and Design B. Distributed Systems	4	--	20	80	100	4	3 Hrs
MCA 405T	<b>ELECTIVE-II</b> A. Design and Analysis of Algorithm B. Soft Computing	4	--	20	80	100	4	3 Hrs
MCA 406P	Web Technologies – LAB	--	4	10	40	50	2	3 Hrs
MCA 407P	Data Mining – LAB	--	4	10	40	50	2	3 Hrs
MCA 408P	Network Security – LAB	--	4	10	40	50	2	3 Hrs
MFCE 401T*	Seminar	2	--	50	--	50	2	---
<b>TOTAL</b>		<b>22</b>	<b>12</b>	<b>180</b>	<b>520</b>	<b>700</b>	<b>28</b>	

### SU - M.C.A - V SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 501T	Programming in C#	4	--	20	80	100	4	3 Hrs
MCA 502T	Big Data Analytics	4	--	20	80	100	4	3 Hrs
MCA 503T	Mobile Application Development	4	--	20	80	100	4	3 Hrs
MCA 504T	<b>ELECTIVE-I</b> C. Automata Theory D. Mobile Computing	4	--	20	80	100	4	3 Hrs
MCA 505T	<b>ELECTIVE-II</b> C. Information Retrieval Systems D. Internet of Things	4	--	20	80	100	4	3 Hrs
MCA 506P	Programming in C# – LAB	--	4	10	40	50	2	3 Hrs
MCA 507P	Big Data Analytics – LAB	--	4	10	40	50	2	3 Hrs
MCA 508P	Mobile Application Development – LAB	--	4	10	40	50	2	3 Hrs
MFCE 501T*	Seminar	2	--	50	--	50	2	---
<b>TOTAL</b>		<b>22</b>	<b>12</b>	<b>180</b>	<b>520</b>	<b>700</b>	<b>28</b>	

### SU - M.C.A - VI SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits
		Theory	Practical	Internal	University	Total	
MCA 601	Seminar	--	--	100	--	100	4
MCA 602	Project Work Review – I	--	--	200	--	200	8
MCA 603	Project Viva Voce	--	--	--	200	200	16
<b>TOTAL</b>				<b>300</b>	<b>200</b>	<b>500</b>	<b>28</b>

## **MCA101T: COMPUTER ORGANIZATION AND ARCHITECTURE**

### **UNIT- I**

Digital Logic circuits: Digital computers, Logic Gates, Boolean Algebra, Map Simplification, Combinational circuits, Flip Flops, Sequential Circuits.

Digital Components: Integrated Circuits, Decoder, Multiplexers, Registers, Shift Registers, Binary Counter, Memory Unit.

Data Representation: Data Types, Complements, Fixed and Floating Point Representation, Other binary codes and error Detection codes.

### **UNIT- II**

Register Transfer and Micro operations: Register Transfer languages, Register transfer, Bus and Memory Transfer, Arithmetic Micro operations, Logic Micro operations, Shift micro operations and Arithmetic logic shift unit.

Basic Computer Organization and Design: Instruction codes, computer registers, computer instructions, timing and control, instruction cycles, memory reference instructions, input, output and interrupts, design of accumulator logic.

### **UNIT- III**

Programming the Basic Computer : Introduction, Machine language, Assembly language, the assembler, programming arithmetic and logic operations, subroutines, and input-output, programming.

Micro Programmed Control: Control memory, Addressing Sequencing, Micro Program example, design of Control Unit.

Central Processing Unit: Introduction, General Register organization, stack organization, Instruction formats, Addressing Modes, data Transfer and manipulation, program control, RISC.

### **UNIT- IV**

Computer Arithmetic: Addition and subtraction, Multiplication algorithms, Division Algorithms, Floating point arithmetic operations, decimal arithmetic unit, and decimal arithmetic operations.

Input-Output Organization: Peripheral Devices, I/O output interface, Asynchronous data transfer, modes of transfer, priority interrupt, DMA, Input output processor, Serial communication.

Memory Organization: Memory Hierarchy, Main Memory, Cache Memory

### **Books:**

- 1.COMPUTER SYSTEM ARCHITECTURE, MORRIS MANO, PHI
- 2.COMPUTER ORGANIZATION, HAMACHER, MGH
3. COMPUTER ARCHITECTURE, CARTER, SCHAUM OUTLINE SERIES, TMH
- 4.SYSTEM ARCHITECTURE, BUAD, VIKAS
- 5.THE FUNDAMENTALS OF COMPUTER ORGANIZATION, RAJA RAO, SCITECH
- 6.COMPUTER ORGANIZATION & DESIGN, PAL CHOWDHURY, PHI

## **MCA102T: BUSINESS SYSTEMS AND APPLICATIONS**

### **UNIT- I**

Use of computers for managerial applications, Technology issues and data and information processing in organizations, Introduction to Information Systems, shift in Information system thinking, latest trends in Information Technology

### **UNIT- II**

Computer Based Information Systems- office automation systems, decision making and MIS, transaction processing systems, decision support system, Group Decision Support, Executive Information systems, DSS generator

### **UNIT- III**

Overview on Artificial Intelligence based systems, end user computing, distributed data processing, Knowledge Management, Business system.

Deciding on IS architecture, IT leadership & IS strategic planning, IS strategy and Effects of IT on competition

### **UNIT- IV**

ERP, re-engineering work processes for IT applications, Business Process Redesign, Knowledge engineering and data warehouse.

Books:

1. Management Information System, O'Brien, TMH, 5th Ed.
2. Management Information System, Kelkar, PHI
3. Management Information System, Jawadekar, TMH
4. Business Information Systems, Munish Kumar, VIKAS
5. ERP: Concepts & Practice, Garg, 2nd Ed, PHI

## **MCA103T: COMPUTER PROGRAMMING WITH C**

### **UNIT- I**

Overview of C, Constants, variables & data types, Operators and expressions, Managing input and output operators

### **UNIT- II**

Decision-making and branching / Looping. Arrays, handling of character Strings, Functions.

### **UNIT- III**

Structures and unions, Pointers, Searching and sorting, file management in C, Dynamic memory allocations in relation to array (Use malloc(), calloc(), realloc(), free() )

### **UNIT- IV:**

Overview of Pre-processor statements, Program through Command Line Arguments

BOOKS:

1. PROGRAMMING WITH C, GOTTFRIED, TMH
2. C THE COMPLETE REFERENCE, SCHILDT, TMH
3. PRACTICAL C PROGRAMMING, 3<sup>RD</sup> ED, O'UALLINE, SPD/O'REILLY
4. A FIRST COURSE IN PROGRAMMING WITH C, JEYAPOOVAN, VIKAS
5. THE C ANSWER BOOK, TONDO, 2<sup>ND</sup> ED, PHI

## **MCA104T: DISCRETE MATHEMATICAL STRUCTURE**

### **UNIT- I**

Set Theory foundation mapping (bijective, surjective, injective), Relations-equivalence, Poset, Lattice, Mathematical induction, Propositional logic, Logical equivalence.

### **UNIT- II**

Permutation and combinations. Generating functions, Recurrence relations. Concepts of Graph Theory, sub-graphs, cyclic graphs. Trees, spanning trees, binary trees.

### **UNIT- III**

Algorithms- Kruskal's , Prim's , Dijkstra's , Flyod's ,Warshall's, DFS, BFS. Isomorphism, Homomorphism of Graphs.

### **UNIT- IV**

Finite automata – Construction & Conversion of NFA, DFA, State minimization, Mealy M/C, Moore M/C. Definition Of Grammars – Type 0,1,2,3. Fuzzy sets – basic properties

Books:

- 1.Theory of Computer Science, Mishra & Chandrasekharan, PHI
- 2.Discrete Mathematics for Comp. Scientists & Mathematicians, Mott, Kandel & Baker, PHI
- 3.Discrete Mathematical Structure, C.L.Liu, TMH
- 4.Discrete Mathematical Structure, G.S.RAO, New Age International
- 5.Discrete Mathematics With Applications, Rosen, TMH, 5th Ed
6. Discrete Mathematics, Ash & Ash, MH.
7. Discrete Mathematical Structure, Somasundaram, PHI

## **MCA105T: BUSINESS ENGLISH AND COMMUNICATION**

### **UNIT- I**

This should cover general and technical writing, oral communications and listening skills: letter writing, technical report writing, and business communication.

### **UNIT- II**

Expression: Practical communication skill development, business presentation with multimedia, speaking skill, prepared speech, extempore speech ,Reading skill: comprehension test

### **UNIT- III**

Writing: precise, technical/business letter, organization of writing material, poster Presentation,

### **UNIT- IV**

Writing technical document, preparing software user manual, preparing project documentation.

BOOKS:

- 1.BUSINESS CORRESPONDENCE & REPORT WRITING, SHARMA, TMH
- 2.BUSINESS COMMUNICATION STRATEGIES, MONIPALLY, TMH
- 3.ENGLISH FOR TECHNICAL COMMUNICATION, LAXMINARAYANAN, SCITECH
4. BUSINESS COMMUNICATION, KAUL, PHI
- 5.COMMUNICATION SKILL FOR EFFECTIVE MGMT., GHANEKAR, EPH

## **MCA106P    MICRO PROGRAMMING AND ARCHITECTURE – LAB**

### I. Digital Systems:

1. Verification of The Logic Gates, Flip – Flops
2. Encoder/ Decoder, Mux / Demux
3. Design of Counters

### II. Microprocessors– 8085, Interfacing and Assembly Language programs:

1. Addition, Subtraction, Multiplication and Division
2. Find Ones and Two's complement of a number
3. Find Square and Square root of a number
4. Find Largest and Smallest Number in Data Array / from a Series of Numbers
5. To Arrange a Data / a Series of Numbers in Ascending and Descending Order
6. Write a Delay Subroutine Program using One, Two or More Registers
7. Measurement of Physical Quantities – Temperature, Speed of a Motor
8. Interfacing a Stepper Motor, with forward and Reverse Steps
9. To Generate a Square Wave of Pulse

### III. Micro processors – 8086

1. Addition, Subtraction, Multiplication and Division of the given Operands.
2. Find Largest and Smallest Number in Data Array / from a Series of Numbers
3. To Arrange a Data Array / a Series of Numbers in Ascending and Descending order
4. Display the Message “ .....” on the Monitor, String Operations
5. Write a Program to generate a Delay of 100 ms, 100 sec, 1 minute and 10 minutes
6. To Generate Waveforms of – Saw tooth, Triangular, Pulse, Sine, by Interfacing DCA  
By using – 8086 Kit, DEBUG of DOS utility, MASM or TASM

### IV. Micro controllers – 8051

1. Setting up UMPS and getting a program to assemble
2. Arithmetic Operations
3. Direct and Indirect Bank Register Addressing
4. RAM Direct addressing
5. Subroutines
6. State Machines

Experiments with Universal Microprocessor Program Simulator (UMPS)

## **MCA107P    C PROGRAMMING LAB**

### **Write a C program to**

1. Finding the maximum and minimum of given set of numbers.
2. Finding the roots of Quadratic equation.
3. sin x and cos x values using series expansion
4. Conversion of binary to decimal, octal, hexa-decimal and vice versa
5. Generating a Pascal triangle
6. Program using recursion-factorial, Fibonacci, GCD, Quick sort and merge sort
7. Matrix addition and multiplication using arrays
8. Programs for bubble sort, selection sort, insertion sort
9. Program on linear search and binary search
10. Functions of string manipulations
11. Finding the no of characters, words and lines from given text file
12. Program to open a file and copy the contents of it into another file.

## **MCA108P BUSINESS PRESENTATION AND LANGUAGE – LAB**

**Exercise –I:** Introduction to Phonetics –Speech Sounds –Vowels and Consonants.  
Ice-Breaking activity and JAM session Articles, Prepositions, Word formation-Prefixes & Suffixes, Synonyms & Antonyms.

**Exercise –II:** Structure of Syllables -Past Tense Marker and Plural Marker –Weak Forms and Strong Forms -Consonant Clusters.  
Situational Dialogues –Role-Play-Expressions in Various Situations –Self-introduction and Introducing Others –Greetings –Apologies –Requests –Social and Professional Etiquette - Telephone Etiquette. Concord (Subject in agreement with verb) and Words often misspelt-confused/misused.

**Exercise -III:** Minimal Pairs-Word accent and Stress Shifts-Listening Comprehension.  
Descriptions-Narrations-Giving Directions and guidelines. Sequence of Tenses, Question Tags and One word substitutes.

**Exercise –IV:** Intonation and Common errors in Pronunciation.  
Extempore-Public Speaking Active and Passive Voice, –Common Errors in English, Idioms and Phrases

**Exercise –V:** Neutralization of Mother Tongue Influence and Conversation Practice.  
Information Transfer-Oral Presentation Skills Reading Comprehension and Job Application with Resume preparation



## **MCA201T: DATA COMMUNICATIONS AND COMPUTER NETWORKS**

### **UNIT- I**

Computer Networks Fundamentals: Overview, Network Hardware, Network Software, Reference models–OSI Model, TCP/IP Reference Model, Comparison of OSI and TCP/IP Reference Model, Example Networks, Network Standardization.

Physical Layer: Guided Transmission Media, Wireless Transmission, Multiplexing, Switching.

Data Link Layer: Design Issues, Error Detection and Correction, Data Link Layer Protocols, Sliding Window Protocol

### **UNIT– II**

Multiple Access Sub layer: ALOHA, CSMA, Collision Free Protocols, Ethernet, Wireless LAN-802.11, Data Link Layer Switching–Repeaters, Hubs, Bridges, Switches, Routers, Gateways.

Network Layer: Design Issues, Routing Algorithms–Shortest path, Flooding, Distance Vector Routing, Link state Routing, Hierarchical, Broadcast Routing, Multicast Routing; Congestion Control Algorithms.

### **UNIT–III**

Internetworking: Tunneling, Internetwork Routing, Fragmentation, IPv4 VsIPv6Protocol, IP Addresses, CIDR, Internet Control Protocols–IMCP, ARP, RARP, DHCP.

Transport Layer: Services provided to the upper layers, Transport Protocols, Overview of Congestion Control.

### **UNIT–IV**

The Internet Transport Protocols: Introduction to UDP&RPC, Real Time Transport Protocols, The Internet Transport Protocols–TCP, TCP Service Model, TCP protocol, TCP Segment Header, TCP Connection Establishment, TCP Connection Release, Modelling TCP Connection Management, TCP Sliding Window, TCP Time Management, TCP Congestion Control.

Application Layer: DNS, TELNET, E-Mail, FTP, HTTP,SSH, Overview of WWW.

### **TEXT BOOK**

1. DATA COMMUNICATIONS AND NETWORKING BY BEHROUZ A. FOROUZAN (TATA Mc.GrawHill)
2. ANDREW S. TANENBAUM, DAVID J WETHERALL, COMPUTER NETWORKS, (5E)

### **REFERENCE BOOKS**

1. BUSINESS DATA COMMUNICATION & NETWORKS By - FITZ GERALD (Jhon Wiley)
2. DATA & COMPUTER COMMUNICATIONS – W STALLINGS (PEARSON, PHI)
3. COMPUTER COMMUNICATIONS & NETWORKING TOPOLOGIES – MA GALLO, V.M.HANCOCK (THOMSON)
4. DATA COMMUNICATION & COMPUTER NETWORKS – R. AGARWAL, BB TIWARI
5. COMPUTER NETWORKS – AS TANENBAUM (PHI)
6. COMPUTER NETWORKS – BLACK (PHI)
7. UNDER STANDING COMMUNICATIONS & NETWORKS – WA SHAY (THOMSON)

## **MCA 202T: PROGRAMMING IN PYTHON**

### **UNIT - I**

Introduction to Python, Installing Python. How a Program Works, Using Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output. Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables. Repetition Structures: Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops. Data types and Expressions: Strings, Assignment and Comments, Numeric Data Types and Character Sets, Expressions, Functions and Modules. Control Statements: Definite Iteration, Formatting Text for Output, Selection, Conditional Iteration.

### **UNIT - II**

Functions: Introduction, Defining and Calling a Void Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions-Generating Random Numbers, The math Module, Time module, Storing Functions in Modules.

Strings and Text Files: Accessing Characters and Substrings in a String, Strings and Number System, String Methods, Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings, File and Exceptions: Introduction to File Input and Output, Using Loops to Process Files, Processing Records, Exceptions.

### **UNIT - III**

Lists, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two-Dimensional Lists, Tuples Sequences, Tuples. Dictionaries and Sets: Dictionaries, Sets, Serializing Objects. Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms.

Design with Classes: Classes and Objects, Classes and Functions, Classes and Methods, Working with Instances, Inheritance and Polymorphism. Object-Oriented Programming: Procedural and Object-Oriented Programming, Classes, techniques for Designing Classes.

### **UNIT - IV**

Graphical User Interfaces: Behavior of terminal based programs and GUI-based programs, Coding simple GUI-based programs, other useful GUI resources. GUI Programming: Graphical User Interfaces, Using the tkinter Module, Display text with Label Widgets, Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes, Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons.

### **TEXT BOOK**

1. TONY GADDIS, STARTING OUT WITH PYTHON (3E)

### **REFERENCE BOOKS**

1. KENNETH A. LAMBERT, FUNDAMENTALS OF PYTHON
2. JAMES PAYNE, BEGINNING PYTHON USING PYTHON 2.6 AND PYTHON 3
3. PAUL GRIES, PRACTICAL PROGRAMMING: AN INTRODUCTION TO COMPUTER SCIENCE USING PYTHON 3
4. CHARLES DIERACH, INTRODUCTION TO COMPUTER SCIENCE USING PYTHON
5. CLINTON W. BROWNLEY, FOUNDATIONS FOR ANALYTICS WITH PYTHON

## **MCA203T: DATASTRUCTURES USING C++**

### **UNIT - I**

C++ Programming Concepts: Introduction, functions and parameters, reference parameters, Parameter passing, function overloading, function templates, Exceptions-throwing an exception and handling an exception, arrays, pointers, new and delete operators, class and object, access specifiers, friend functions, constructors and destructor, Operator overloading, class templates, Inheritance and Polymorphism, Recursion, Testing and debugging

Basic Concepts - Data objects and Structures, Algorithm Specification-Introduction, Data Abstraction, Performance analysis- time complexity and space complexity, Asymptotic Notation- Big O, Omega and Theta notations, Complexity Analysis Examples, Introduction to Linear and Non Linear data structures.

### **UNIT – II**

Representation of single, two dimensional arrays, sparse matrices-array and linked representations.

Linear list ADT-array representation, vector representation and linked representation, Singly Linked Lists- Operations-Insertion, Deletion, Circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations- Insertion, Deletion- Applications- Bin Sort, Radix sort and Union find problem

Stack ADT, definition, array and linked implementations, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definition, array and linked Implementations, Circular queues-Insertion and deletion operations.

### **UNIT- III**

Trees – definition, terminology, Binary trees-definition, Properties of Binary Trees, Binary Tree ADT, representation of Binary Trees-array and linked representations, Binary Tree traversals, Threaded binary trees, Priority Queues –Definition and applications, Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

Searching - Linear Search, Binary Search, Hashing-Introduction, hash tables, hash functions, Overflow Handling, Comparison of Searching methods.

Sorting-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Heap Sort, Merge sort, Comparison of Sorting methods.

### **UNIT – IV**

Graphs–Definitions, Terminology, Applications and more definitions, Properties, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph Search methods - DFS and BFS, Complexity analysis.

Search Trees-Binary Search Tree ADT, Definition, Operations- Searching, Insertion and Deletion, Balanced search trees-AVL Trees-Definition and Examples only, B-Trees-Definition and Examples only, Red-Black Trees-Definitions and Examples only, Comparison of Search Trees.

### **TEXT BOOKS:**

1. DATA STRUCTURES, ALGORITHMS AND APPLICATIONS IN C++, SARTAJSAHNI, UNIVERSITIES PRESS.
2. DATA STRUCTURES AND ALGORITHMS IN C++, ADAM DROZDEK

### **REFERENCE BOOKS:**

1. DATA STRUCTURES WITH C++, J. HUBBARD, SCHAUUM'S OUTLINES, TMH.
2. DATA STRUCTURES AND ALGORITHMS IN C++, M.T. GOODRICH, R. TAMASSIA AND D. MOUNT, WILEY INDIA.
3. DATA STRUCTURES AND ALGORITHM ANALYSIS IN C++, 3RD EDITION, M. A. WEISS, PEARSON.
4. CLASSIC DATA STRUCTURES, D. SAMANTA, 2ND EDITION, PHI.

## **MCA204T: DATABASE MANAGEMENT SYSTEMS**

### **UNIT - I**

Database Systems: Introducing the Database and the DBMS, Problems with File system Data management, The database System Environment, DBMS functions. Data Models: The importance of Data models, Data model basic building blocks, Business rules, The evolution of Data models, Degrees of Data abstraction

The Relational Database model: Logical view of data, Keys, Integrity rules, Relational set operators, Relationships within the Relational Database, Codd's Relational Database rules

### **UNIT – II**

ER modelling: Entities, Attributes, Relationships, connectivity and cardinality, Dependence, Relationships, Developing an ER Diagram, Normalization of Database tables- Need for Normalization, The Normalization process, Improving the design, Higher level Normal Forms, Denormalization.

Advanced Data Modelling: The Extended ER model- entity, Entity clustering, Entity Integrity: selecting primary keys.

### **UNIT – III**

SQL: Introduction to SQL, Data definition commands, Data Manipulation commands, Select queries, Advanced Data Definition Commands, Advanced Select queries, creating view, Joining Database tables.

Advanced SQL: Relational Set operators, Join operators, subqueries and correlated queries, SQL functions, Procedural SQL, Embedded SQL.

### **UNIT – IV**

Transaction Management and concurrency control- What is a Transaction?-Concurrency control, Concurrency control with Locking methods, Concurrency control with Time stamping methods, Concurrency control with Optimistic methods, Transaction recovery.

Database Administration- The evolution of the Database Administration Function, The database Environment's Human component-Managerial role, Technical role, Database Administration tolls

### **TEXT-BOOK**

1. PETER ROB,CARLOS CORONEL -DATABASE SYSTEMS -DESIGN,IMPLEMENTATION AND MANAGEMENT

### **REFERENCE BOOKS**

- 1.SHARON ALLEN, EVAN TERRY, BEGINNING RELATIONAL DATA MODELING
- 2.JEFFREY A. HOFFER, V. RAMESH, HEIKKI TOPI, MODERN DATABASE MANAGEMENT
- 3.RAGHU RAMAKRISHNAN, JOHANNES GEHRKE, DATABASE MANAGEMENT SYSTEMS
- 4.RAMEZ ELMASRI, SHAMKANT B. NAVATHE, FUNDAMENTALS OF DATABASE SYSTEMS
- 5.ABRAHAM SILBERSCHATZ, HENRY F. KORTH, S. SUDARSHAN, DATABASE SYSTEM CONCEPTS

## **MCA 205T: MODERN OPERATING SYSTEMS**

### **UNIT - I**

**INTRODUCTION:** Operating system as extended Machine and Resource Manager, Computer-System Architecture,

Types of operating systems: Mainframe, Server, Multiprocessor, Personal computer, Handheld, Embedded, Sensor Node , Real time, Smart card ,Operating-System Structure- Monolithic system, layered system, Microkernel, Client server Model, Virtual Machines Operating-System concepts: Processes, Address space, Files, Input and output, Protection and shell

System calls: Process Management, File Management and Directory Management.

**PROCESS & THREADS:** Process- Model, Creation, termination, hierarchies, states, Implementation of process. Thread-usage, Classical thread model, POSIX Thread Implementation of threading User space and in the Kernel, Hybrid Implementation

### **UNIT – II**

Inter process Communication- Race condition, Critical region, Mutual exclusion, Sleep and waiting, semaphores, Mutexes, Monitors, Message passing, Barriers.

**CPU SCHEDULING:** Introduction, Categories of scheduling- Batch, Interactive, Real time, Scheduling ,Thread Scheduling,

### **UNIT- III**

**MEMORY MANAGEMENT:** Memory Abstraction: Addressing, Swapping, Virtual Memory: Paging, Page Tables, Speeding Up Pages, Page replacement algorithms, segmentation. Design and implementation issues for paging systems

**FILE SYSTEMS:** File Concept, Access Methods, Directories, File-System Implementation, File system Management and Optimization,

Disks, clocks, User interface, Thin clients, power managements

### **UNIT- IV**

**DEADLOCKS:** Introduction, Deadlock detection and Recovery, Deadlock avoidance, Deadlock prevention

**PROTECTION AND SECURITY:** The security Environment, Basics of cryptography, Protection Mechanism, Authentication, , Insider attacks, Exploiting code bugs, Malware, Defences,

### **TEXT-BOOK**

1. ANDREW S. TANENBAUM, MODERN OPERATING SYSTEMS
2. ABRAHAM SILBERSCHATZ, PETER BAER GALVIN, GREG GAGNE, OPERATING SYSTEM CONCEPTS (9E)

### **REFERENCE BOOKS**

1. DHANANJAY M. DHANDHERE, OPERATING SYSTEMS-A CONCEPT BASED APPROACH
2. WILLIAM STALLINGS, OPERATING SYSTEMS-INTERNAL AND DESIGN PRINCIPLES
3. THOMAS W. DOEPPNER, OPERATING SYSTEMS IN DEPTH

## MCA 206P: PYTHON LAB

1. Write a program that prompt the user to enter his or her name, age, height and assigns the user's input to the variables n,a,m respectively then print the values hold by the variables.
2. Write a program to find the largest three integers using if-else and conditional operator.
3. Write a program to find the factors of a number.
4. Write a program with a loop that asks the user to enter a series of positive numbers. The user should enter a negative number to signal the end of the series. The program should display the numbers in order and their sum.
5. Write a program to reverse a given number.
6. Write a program to display the multiplication table.
7. Write a program to find the Gravitational force acting between two objects.
8. Write a program to find the product of two matrices  $[A]_{m \times p}$  and  $[B]_{p \times r}$
9. Write a program to find the product of two matrices  $[A]_{m \times p}$  and  $[B]_{p \times r}$
10. Write recursive and non-recursive functions to find GCD of two integers, factorial of positive integer, Fibonacci Sequence up to given number n , convert Decimal to Binary.
11. Write a program to display two random numbers that are to be added, such as: 247 + 129, the program should allow the student to enter the answer. If the answer is correct, a message of congratulations should be displayed. If the answer is incorrect, a message showing the correct answer should be displayed.
12. Write recursive and non-recursive functions to display prime number from 2 to n.
13. Assume a file containing a series of integers is named numbers.txt and exists on the computer's desk. Write a program that reads all of the numbers stored in the file and calculates their total.
14. Write a program that writes a series of random numbers to a file from 1 to n and display.
15. Write a program to create file, write the content and display the contents of the file with each line preceded with a line number (start with 1) followed by a colon.
16. Write a program to generate random numbers from 1 to 20 and append them to the List.
17. Write a program that reads the contents of text file. The program should create a dictionary in which the keys are the individual words found in the file and the values are the number of times each word appears.
18. Write a program that opens a specified text file and then displays a list of all the unique words found in the file. (Store each word as an element of a set.)
19. Write a program to analyze the contents of two text files using set operations.
20. Write a program to find the area of a rectangle using classes.
21. Write a class named person with data attributes for a person's name, address, and telephone number. Next, write a class named customer that is subclass of the person class.
22. Write a program to implement the inheritance and dynamic polymorphism.
23. Write a GUI program that converts Celsius temperatures to Fahrenheit temperatures.
24. Write a GUI program that displays your details when a button is clicked.

Note: Programs are to be practiced on the basis of topics covered in corresponding theory paper. Handle the exceptions raised from file operations.

## MCA 207P: DATA STRUCTURES THROUGH C++ LAB

1. Write a C++ program that uses functions to perform the following:
  - a) Create a singly linked list of integers.
  - b) Delete a given integer from the above linked list.
  - c) Display the contents of the above list after deletion.
2. Write a template based C++ program that uses functions to perform the following:
  - a) Create a doubly linked list of elements.
  - b) Delete a given element from the above doubly linked list.
  - c) Display the contents of the above list after deletion.
3. Write a C++ program that uses stack operations to convert a given infix expression into its postfix equivalent, Implement the stack using an array.
4. Write a C++ program to implement a double ended queue ADT using an array, using a doubly linked list.
5. Write a C++ program that uses functions to perform the following:
  - a) Create a binary search tree of characters.
  - b) Traverse the above Binary search tree recursively in preorder, in order and post order,
6. Write a C++ program that uses function templates to perform the following:
  - a) Search for a key element in a list of elements using linear search.
  - b) Search for a key element in a list of sorted elements using binary search.
7. Write a C++ program that implements Insertion sort algorithm to arrange a list of integers in ascending order.
8. Write a template based C++ program that implements selection sort algorithm to arrange a list of elements in descending order.
9. Write a template based C++ program that implements Quick sort algorithm to arrange a list of elements in ascending order.
10. Write a C++ program that implements Heap sort algorithm for sorting a list of integers in ascending order.
11. Write a C++ program that implements Merge sort algorithm for sorting a list of integers in ascending order
12. Write a C++ program to implement all the functions of a dictionary (ADT) using hashing.
13. Write a C++ program that implements Radix sort algorithm for sorting a list of integers in ascending order
14. Write a C++ program that uses functions to perform the following:
  - a) Create a binary search tree of integers.
  - b) Traverse the above Binary search tree non recursively in inorder.
15. Write a C++ program that uses functions to perform the following:
  - a) Create a binary search tree of integers.
  - b) Search for an integer key in the above binary search tree non recursively.
  - c) Search for an integer key in the above binary search tree recursively.

## MCA 208P: DBMS LAB

### PART A: SQL PROGRAMMING

1. Consider the following schema for a Library Database:

BOOK(*Book\_id, Title, Publisher\_Name, Pub\_Year*)

BOOK\_AUTHORS (*Book\_id, Author\_Name*) PUBLISHER (*Name, Address, Phone*)

BOOK\_COPIES (*Book\_id, Branch\_id, No-of\_Copies*)

BOOK\_LENDING (*Book\_id, Branch\_id, Card\_No, Date\_Out, Due\_Date*)

LIBRARY\_BRANCH (*Branch\_id, Branch\_Name, Address*)

Write SQL queries to

- Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch etc.
- Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun2018
- Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
- Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
- Create a view of all books and its number of copies that are currently available in the Library.

2. Consider the following schema for Order Database:

SALESMAN (*Salesman\_id, Name, City, Commission*)

CUSTOMER (*Customer\_id, Cust\_Name, City, Grade, Salesman\_id*)

ORDERS (*Ord\_No, Purchase\_Amt, Ord\_Date, Customer\_id, Salesman\_id*)

Write SQL queries to

- Count the customers with grades above Bangalore's average.
- Find the name and numbers of all salesmen who had more than one customer.
- List all salesmen and indicate those who have and don't have customers in their cities (Use UNION operation.)
- Create a view that finds the salesman who has the customer with the highest order of a day.
- Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

3. Consider the schema for Movie Database:

ACTOR (*Act\_id, Act\_Name, Act\_Gender*)

DIRECTOR (*Dir\_id, Dir\_Name, Dir\_Phone*)

MOVIES (*Mov\_id, Mov\_Title, Mov\_Year, Mov\_Lang, Dir\_id*)

MOVIE\_CAST (*Act\_id, Mov\_id, Role*)

RATING (*Mov\_id, Rev\_Stars*)

Write SQL queries to

- List the titles of all movies directed by 'SS Rajamouli'.
- Find the movie names where one or more actors acted in two or more movies.
- List all actors who acted in a movie before 2010 and also in a movie after 2015 (use JOIN operation).



- d) Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.
- e) Update rating of all movies directed by 'Kodi Ramakrishna' to 5.

4. Consider the schema for College Database:

STUDENT (*USN, SName, Address, Phone, Gender*)  
 SEMSEC (*SSID, Sem, Sec*)  
 CLASS (*USN, SSID*)  
 SUBJECT (*Subcode, Title, Sem, Credits*)  
 IAMARKS (*USN, Subcode, SSID, Test1, Test2, Test3, FinallA*)

Write SQL queries to

- a) List all the student details studying in fourth semester 'C' section.
- b) Compute the total number of male and female students in each semester and in each section.
- c) Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
- d) Calculate the FinallA (average of best two testmarks) and update the corresponding table for all students.
- e) Categorize students based on the following criterion: If FinallA = 17 to 20 then CAT = 'Outstanding'
- f) If FinallA = 12 to 16 then CAT = 'Average' If FinallA < 12 then CAT = 'Weak'
- g) Give these details only for 8th semester A, B, and C section students.

5. Consider the schema for Company Database:

EMPLOYEE (*SSN, Name, Address, Sex, Salary, SuperSSN, DNo*)  
 DEPARTMENT (*DNo, DName, MgrSSN, MgrStartDate*)  
 DLOCATION (*DNo, DLoc*)  
 PROJECT (*PNo, PName, PLocation, DNo*)  
 WORKS\_ON (*SSN, PNo, Hours*)

Write SQL queries to

- a) Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.
- b) Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percentage
- c) Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department
- d) Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs.6,00,000.

## **PART-B : PL/SQL PROGRAMMING**

1. PL/SQL Program To Add Two Numbers
2. PL/SQL Program for Prime Number
3. PL/SQL Program to Find Factorial of a Number
4. PL/SQL Program to Print Table of a Number
5. PL/SQL Program for Reverse of a Number
6. PL/SQL Program for Fibonacci Series
7. PL/SQL Program to Check Number is Odd or Even
8. PL/SQL Program to Reverse a String
9. PL/SQL Program for Palindrome Number
10. PL/SQL Program to Swap two Numbers
11. PL/SQL Program for Armstrong Number
12. PL/SQL Program to Find Greatest of Three Numbers

## **MCA 301T: LINUX PROGRAMMING**

### **UNIT - I**

Linux Utilities-File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities.

Sed-Scripts, Operation, Addresses, Commands, awk-Execution, Fields and Records, Scripts, Operation, Patterns, Actions, Associative Arrays, String and Mathematical functions, System commands in awk, Applications.

Shell programming with Bourne again shell(bash)- Introduction, shell responsibilities, pipes and Redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

### **UNIT - II**

Files and Directories- File Concept, File types, File System Structure, file metadata-Inodes, kernel support for files, system calls for file I/O operations- open, create, read, write, close, lseek, dup2, file status information-stat family, file and record locking- fcntl function, file permissions - chmod, fchmod, file ownership-chown, lchown, links-soft and hard links - symlink, link, unlink. Directories-Creating, removing and changing Directories-mkdir, rmdir, chdir, obtaining current working directory-getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir functions.

Process - Process concept, Layout of a C program image in main memory. Process environment-environment list, environment variables, getenv, setenv, Kernel support for process, process identification, process control - process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process, system call interface for process management-fork, vfork, exit, wait, waitpid, exec family, Process Groups, Sessions and Controlling Terminal, Differences between threads and processes.

### **UNIT - III**

Signals - Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise, alarm, pause, abort, sleep functions.

Interprocess Communication - Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes-creation, IPC between related processes using unnamed pipes, FIFOs- creation, IPC between unrelated processes using FIFOs(Named pipes), differences between unnamed and named pipes,popen and pclose library functions. Message Queues- Kernel support for messages, APIs for message queues, client/server example.

### **UNIT - IV**

Semaphores-Kernel support for semaphores, APIs for semaphores, file locking with semaphores. Shared Memory- Kernel support for shared memory, APIs for shared memory, shared memory example. Sockets- Introduction to Berkeley Sockets, IPC over a network, Client-Server model, Socket address structures (unix domain and Internet domain), Socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs-Single Server-Client connection, Multiple simultaneous clients, Socket options-setsockopt and fcntl system calls, Comparison of IPC mechanisms.

### **TEXT BOOKS:**

1. Unix System Programming using C++, T. Chan, PHI.
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.
3. Unix Network Programming, W. R. Stevens, PHI.

**REFERENCE BOOKS:**

1. Beginning Linux Programming, 4th Edition, N. Mathew, R. Stones, Wrox, Wiley India Edition.
2. Unix for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson.
3. System Programming with C and Unix, A. Hoover, Pearson.
4. Unix System Programming, Communication, Concurrency and Threads, K. A. Robbins, Pearson Education.
5. Unix shell Programming, S. G. Kochan and P. Wood, 3rd edition, Pearson Education.
6. Shell Scripting, S. Parker, Wiley India Pvt. Ltd.
7. Advanced Programming in the Unix Environment, 2nd edition, W. R. Stevens and S. A. Rago, Pearson Education.
8. Unix and Shell Programming, B. A. Forouzan and R. F. Gilberg, Cengage Learning.
9. Linux System Programming, Robert Love, O'Reilly, SPD.
10. C Programming Language, Kernighan and Ritchie, PHI.

## **MCA 302T: SOFTWARE ENGINEERING**

### **UNIT-I**

**SOFTWARE ENGINEERING:** The Nature of Software, Changing Nature of Software, Defining the Discipline, Software Process, Software Engineering Practice.

**THE SOFTWARE PROCESS:** A Generic Process Model, Defining a Framework Activity, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, Unified Process, Personal and Team

Process Models. Defining Agility, Agile Process, Extreme Programming, Psychology of Software Engineering, Software Team Structures, Software Engineering Using the Cloud, Global Teams.

### **UNIT – II**

**REQUIREMENTS:** Core Principles of Modeling, Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Analysis Model, Requirements Analysis, UML Models That Supplement the Use Case, Identifying Analysis Classes, Specifying Attributes, Defining Operations, Class-Responsibility-Collaborator Modeling, Associations and Dependencies, Analysis Packages.

**DESIGN CONCEPTS:** Design within the Context of SE, Design Process, Design Concepts, Design Model, Software Architecture, Architectural Styles, Architectural Considerations, Architectural Design, Component, Designing Class-Based Components, Conducting Component-Level Design, Component-Based Development, User Interface Design Rules.

### **UNIT – III**

**QUALITY MANAGEMENT:** Quality, Software Quality, Software Quality Dilemma, Achieving Software Quality, Defect Amplification and Removal, Reviews, Informal Reviews, Formal Technical Reviews, Elements of Software Quality Assurance, SQA Tasks, Goals, and Metrics, Software Reliability, A Strategic Approach to Software Testing, Test Validation Testing, System Testing, Debugging, Software Testing Fundamentals, White-Box Testing, Black-Box Testing, Path Testing, Control Structure Testing, Object-Oriented Testing Strategies & Methods, Security Engineering Analysis, Security Assurance, Security Risk Analysis.

### **UNIT – IV**

Software Configuration Management, SCM Process, Product Metrics for Requirements Model, Design Model, Source Code, Testing and Maintenance.

**MANAGING SOFTWARE PROJECTS:** The Project Management Spectrum, W5HH Principle, Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality, Integrating Metrics within the Software Process, Software Project Estimation, Decomposition Techniques, Project Scheduling – basics, scheduling, Software Risks, Risk Mitigation, Monitoring, and Management, Software Maintenance, Software Reengineering, Reverse Engineering, Forward Engineering.

### **TEXT BOOK**

1. ROGER S PRESSMAN, B R MAXIM, SOFTWARE ENGINEERING–A PRACTITIONER'S APPROACH (8E)

### **REFERENCE BOOKS**

1. SOFTWARE ENGINEERING BY GHEZZI (PHI)
2. SOFTWARE ENGINEERING FUNDAMENTALS BY BEHFOROZ AND HUDSON OXFORD UNIVERSITY PRESS

## **MCA 303T: JAVA PROGRAMMING**

### **UNIT-I**

OBJECT ORIENTED METHODOLOGY-1: Paradigms of Programming Languages, Evolution of OO Methodology, Basic Concepts of OO Approach, Comparison of Object Oriented and Procedure Oriented Approaches, Benefits of OOPs, Introduction to Common OO Language, Applications of OOPs.

OBJECT ORIENTED METHODOLOGY-2: Classes and Objects, Abstraction and Encapsulation, Inheritance, Method Overriding and Polymorphism.

Java Language Basics: Introduction to Java, Basic Features, Java Virtual Machine Concepts, Primitive Data Types and Variables, Java Operators, Expressions, Control Statements and Arrays.

### **UNIT-II**

OBJECT ORIENTED CONCEPTS: Class and Objects-- Class Fundamentals, Creating objects, Assigning object reference variables; Introducing Methods, Static methods, Constructors, Overloading constructors; This Keyword; Using Objects as Parameters, Argument passing, Returning objects, Method overloading, Garbage Collection, The Finalize ( ) Method.

INHERITANCE AND POLYMORPHISM: Inheritance Basics, Access Control, Multilevel Inheritance, Method Overriding, Abstract Classes, Polymorphism, Final Keyword.

### **UNIT-III**

PACKAGES: Defining Package, CLASSPATH, Package naming, Accessibility of Packages, using Package Members.

INTERFACES: Implementing Interfaces, Interface and Abstract Classes, Extends and Implements together.

EXCEPTIONS HANDLING: Exception, Handling of Exception, Using try-catch, Catching Multiple Exceptions, Using finally clause, Types of Exceptions, Throwing Exceptions, and Writing Exception Subclasses.

### **UNIT-IV**

MULTI-THREADING: Introduction, The Main Thread, Java Thread Model, Thread Priorities, Synchronization in Java, Inter thread Communication.

I/O IN JAVA: I/O Basics, Streams and Stream Classes, the Predefined Streams, Reading from, and Writing to, Console, Reading and Writing Files, Random Access File.

STRINGS AND CHARACTERS: Fundamentals of Characters and Strings, the String Class, String Operations, String Methods, String Buffer Class and Methods.

### **TEXT BOOK**

1. THE COMPLETE REFERENCE JAVA, TMH PUBLICATION.

### **REFERENCE BOOKS**

1. PROGRAMMING JAVA - DECKER&HIRSH FIELD VIKAS PUBLISKING (2001) (THOMSON LEARNING ) (SECOND EDITON)
2. INTRODUCTION TO JAVA PROGRAMMING - Y.DANIEL LIANG PHI(2002)
3. OBJECT ORIENTED PROGRAMMING THROUGH JAVA 2 BY - THAMUS WU (Mc.Graw Hill)
4. JAVA 2 - DIETEL & DIETEL (PEARSON EDUCATION)
5. INTRODUCTION TO JAVA – BALA GURU SWAMY
6. INTRODUCTION TO PROGRAMMIND & OOD USING JAVA – JAINO NINE & FA HOSCH (JOHN WILEY)
7. STARTING OUT WITH JAVA – JONY GADDIS (DREAM TECH PRESS)

## **MCA 304T: BUSINESS MANAGEMENT**

### **UNIT - I: Introduction to Organisation and Management**

Organisation: - Definition and nature of organisations – characteristics- Types of organisations: Formal and informal - Management: Definition and Characteristics. Management as an Art, Science and Profession. Concept of efficiency and effectiveness – Functions of Management: Planning, Organizing, Staffing, Leading, and controlling – Approaches: An Overview of Classical, Human Relations and Systems Approaches- Fayol's Principles of Management.

### **UNIT - II: Planning and Decision Making**

Types of Plans – MBO - Decision Making: Decision Making Process, Types of Decisions - Decision Making Models: Classical, Administrative and Political - Group Decision Making Techniques: Brain Storming, Delphi, Nominal Group Technique, Lotus Blossom Technique

### **UNIT - III: Organizing**

Departmentation: Product, Functional, Matrix and other bases for Departmentation, Team based, Virtual, Boundary-less and Learning Organization - Authority and Responsibility: Bases for authority, power and responsibility, distinctions - Delegation and Decentralization: Concept, Barriers and Overcoming barriers to delegation, factors pointing to decentralization - Span of Management: Factors influencing Span, Span and Organisation Structure - Line and Staff: Concepts and conflicts, resolution of conflicts.

### **UNIT - IV : Leadership and Controlling**

Direction: Elements - Motivation: Nature of Motivation, motivators, financial and non-financial.- Leadership: Nature, Trait and Charismatic approach, leadership styles, Ohio and Michigan studies, Managerial Grid – Communication: Process, types and barriers. – Coordination: Nature, methods and techniques. – Controlling: Control process, types-Problems, Requisites of Good Control System.

### **TEXT BOOKS**

1. Wehrich Heinz, Cannice V Mark and Koontz Harold, 2008, Management: A Global and Entrepreneurial Perspective, Tata McGraw Hill Publishing Company Limited, New Delhi.
2. Cullen B John and Praveen K Parboteeah, 2005, International Management: A Strategic Perspective, Cengage Learning India Private Limited, New Delhi.

### **REFERENCE BOOKS**

1. Daft L.Richard, 2006, The New Era of Management, Thomson Corporation, New Delhi.
2. Robbins P Stephen and Decenzo A David, 2006, Fundamental of Management: Essential Concepts and Applications, Pearson Education, New Delhi, p 53.
3. Sridhara Bhat, 2005, Management and Behavioural Process: Text and Cases, Himalaya Publishing House, Hyderabad.
4. Satya Raju, R and Parthasarathy, A, 2005, Management: Text and Cases, Prentice Hall of India Private Limited, New Delhi.
5. Hill W L Charles and McShane L Steven, 2008, Principles of Management, Tata McGraw Hill Company Limited, New Delhi.

## **MCA 305T: R PROGRAMMING**

### **UNIT – I**

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

### **UNIT – II**

Control structures, functions, scoping rules, dates and times, Introduction to Functions, preview of Some Important R Data Structures, Vectors, Character Strings, Matrices, Lists, Data Frames, Classes Vectors: Generating sequences, Vectors and subscripts, Extracting elements of a vector using subscripts, Working with logical subscripts, Scalars, Vectors, Arrays, and Matrices, Adding and Deleting Vector Elements, Obtaining the Length of a Vector, Matrices and Arrays as Vectors Vector Arithmetic and Logical Operations, Vector Indexing, Common Vector Operations

### **UNIT – III**

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

### **UNIT – IV**

FACTORS AND TABLES, Factors and Levels, Common Functions Used with Factors, Working with Tables, Matrix/Array-Like Operations on Tables , Extracting a Subtable, Finding the Largest Cells in a Table, Math Functions, Calculating a Probability, Cumulative Sums and Products, Minima and Maxima, Calculus, Functions for Statistical Distributions  
OBJECT-ORIENTED PROGRAMMING: S Classes, S Generic Functions, Writing S Classes, Using Inheritance, S Classes, Writing S Classes, Implementing a Generic Function on an S Class, visualization, Simulation, code profiling, Statistical Analysis with R, data manipulation

### **TEXT BOOKS**

1. R Programming for Data Science by Roger D. Peng
2. The Art of R Programming by Prashanth singh, Vivek Mourya, Cengage Learning India.



## **MCA 306P: LINUX PROGRAMMING LAB**

1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
2. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
6. Write a shell script to list all of the directory files in a directory.
7. Write a shell script to find factorial of a given integer.
8. Write an awk script to count the number of lines in a file that do not contain vowels.
9. Write an awk script to find the number of characters, words and lines in a file.
10. Write a C program that makes a copy of a file using standard I/O and system calls.
11. Implement in C the following Linux commands using System calls a) cat b) mv
12. Write a C program to list files in a directory.
13. Write a C program to emulate the Unix ls -l command.
14. Write a C program to list for every file in a directory, its inode number and file name.
15. Write a C program that redirects standard output to a file.Ex: ls > f1.
16. Write a C program to create a child process and allow the parent to display "parent" and the child to display "child" on the screen.
17. Write a C program to create a Zombie process.
18. Write a C program that illustrates how an orphan is created.
19. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex:- ls -l | sort
20. Write C programs that illustrate communication between two unrelated processes using named pipe (FIFO File).
21. Write a C program in which a parent writes a message to a pipe and the child reads the message.
22. Write a C program (sender.c) to create a message queue with read and write permissions to write 3 messages to it with different priority numbers.
23. Write a C program (receiver.c) that receives the messages (from the above message queue as specified in (22)) and displays them.
25. Write a C program that illustrates suspending and resuming processes using signals.

## **MCA 307P: SOFTWARE ENGINEERING LAB**

1. Study of case tool Requirements
2. Implementation of requirements engineering activities such as elicitation, validation, management using  
Case tools  
Analysis and Design
3. Implementation of Analysis and design using case tools
4. Study and usage of software project management tools such cost estimates and scheduling
5. Documentation generators –Study and practice of Documentation generators
6. Data Modeling using automated tools
7. Practice reverse engineering and re engineering using tools
8. Exposure towards test plan generators, test case generators, test coverage and software metrics.

9. Meta modeling and software life cycle management.

Case Studies:

10. Structure charts, Data Flow Diagrams, Decision tables and ER diagrams for

1. Banking System
2. Railway Reservation System
3. Hotel management system
4. Inventory Control System
5. Library management system

Note: The teacher should define the boundaries for the above case study problems and make the practice of problems mentioned.

### **MCA 308P: JAVA PROGRAMMING LAB**

1. Write a Java Program to define a class, define instance methods and overload them and use them for dynamic method invocation.
2. Write a Java Program to demonstrate use of sub class
3. Write a Java Program to demonstrate use of nested class.
4. Write a Java Program to implement array of objects.
5. Write a Java program to practice using String class and its methods.
6. Write a Java program to practice using String Buffer class and its methods.
7. Write a Java Program to implement inheritance and demonstrate use of method overriding.
8. Write a Java Programs Illustrating the Implementation of Various forms of Inheritance. (Ex. Single, Hierarchical, Multilevel inheritance....)
9. Write a Java Program which illustrates the implementation of multiple Inheritance using interfaces in Java.
10. Write a Java Program illustrates the implementation of abstract class.
11. Write a Java program to demonstrate use of implementing interfaces.
12. Write a Java program to implement the concept of threading by implementing Runnable Interface
13. Write a Java program to implement the concept of Exception Handling using predefined exception.
14. Write a Java Program to Create Multiple Threads in Java.
15. Write a Java Program which illustrate the manipulation of strings. a. Ex. 1. Sorting an array of Strings.
16. Write a Java Program that reads on file name from the user and displays the contents of file.
17. Write a Java Program that displays the no. of characters, lines & words in a text file.
18. Write a Java Program to display the contents of file along with a line number before each line.
19. Write a Java Program to read & write the data using Random Access File.

## **MCA 401T: WEB TECHNOLOGIES**

### **UNIT – I**

**HTML-** Basic HML, The document body, Text, Hyperlinks, Adding More Formatting, Lists, Using Color and Images, Images, Tables, Frames, Forms-Toward Interactivity Cascading Style sheets – Introduction, Inline Styles, Embedded Style Sheets, Linking external sheets, Backgrounds, text flow and box model.

### **UNIT – II**

**Introduction to PHP:** Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc.

Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc ., on text and binary files, listing directories

### **UNIT – III**

**XML:** Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML Parsing XML Data - DOM and SAX Parsers in java.

### **UNIT – IV**

**JDBC:** JDBC Overview, Architecture, Types of JDBC Drivers, DriverManager, Database Connection Statement, ResultSet Transaction, DataBaseMetadata, ResultSetMetadata and Aggregate functions, PreparedStatement, CallableStatement, Connection to various back ends.

**Introduction to Servlets:** Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions.

### **TEXT BOOKS**

1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill

### **REFERENCES**

1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
2. Java Server Pages –Hans Bergsten, SPD O'Reilly

## **MCA 402T: DATA MINING**

### **UNIT – I**

**Introduction to Data Mining:** Why data mining? What is data mining? What kinds of data can be mined? What kinds of patterns can be mined? Which technologies are used? Major issues in data mining. Getting to Know Your Data: data objects and attribute types, basic statistical description of data, data visualization, measuring data similarity and dissimilarity. Data Processing: an overview, data cleaning, data integration, data reduction, data transformation and data discretization.

### **UNIT – II**

**Data Warehousing and Online Analytical Processing (OLAP):** basic concepts of data warehouse, data warehouse modelling—data cube and OLAP, data warehouse design and usage, data warehouse implementation, data generalization by attribute-oriented. Data Cube Technology: data cube computation preliminary concepts, data cube computation methods, processing advanced kinds of queries by exploring cube technology, multidimensional data analysis in cube space. Mining Frequent Patterns, Associations, and Correlations: basic concepts, frequent itemset mining methods, mining various kinds of association rules, from association mining to correlation analysis, constraint-based association mining.

### **UNIT – III**

**Classification:** Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, classification by backpropagation, support vector machines, associative classification, lazy learners, other classification methods. Cluster Analysis: basic concepts of cluster analysis, partitioning methods, hierarchical methods, density-based methods, evaluation of clustering.

### **UNIT – IV**

**Outlier Detection:** outliers and outlier analysis, outlier detection methods, statistical approaches, proximity-based approaches, clustering-based approaches, classification-based approaches. Data Mining Trends and Research Frontiers: mining complex data types, other methodologies of data mining, data mining applications, data mining and society, data mining trends.

### **TEXT-BOOK**

1. JIAWEI HAN, MICHELINE KAMBER, JIAN PEI, DATA MINING – CONCEPTS AND TECHNIQUES (3E)

### **REFERENCE BOOKS**

1. ARUN K PUJARI, DATA MINING TECHNIQUES
2. PAWEŁ CICHOSZ, DATA MINING ALGORITHMS: EXPLAINED USING R
3. NONG YE, DATA MINING – THEORIES, ALGORITHMS, AND EXAMPLES
4. PANG-NING TAN, M STEINBACH, V KUMAR, INTRODUCTION TO DATA MINING
5. S. SUMATHI, S.N. SIVANANDAM, INTRODUCTION TO DATA MINING AND ITS APPLICATIONS

## **MCA 403T: NETWORK SECURITY**

### **UNIT – I**

**Overview of Network Security:** Computer Security Concepts, the OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, a Model for Network Security. Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines, Steganography. Block Ciphers and the Data Encryption Standard: Traditional Block Cipher Structure, the Data Encryption Standard (DES), A DES Example, Strength of DES. Block Cipher Operation: Double DES, Triple DES, Electronic Code Book, Cipher Block Chaining Mode, Cipher Feedback Mode, Output Feedback Mode, Counter Mode.

### **UNIT – II**

**Advanced Encryption Standard (AES):** The Origins AES, AES Structure, AES Round Functions, AES Key Expansion, an AES Example AES Implementation. Pseudorandom Number Generation and Stream Ciphers: Principles of Pseudorandom Number Generation, Pseudorandom Number Generators, Pseudorandom Number Generation using BlockCipher, StreamCiphers-RC4. Public-Key Cryptography and RSA: Principles of Public-Key Cryptosystems, the RSA Algorithm. Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption and Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Diffie-Hellman Key Exchange.

### **UNIT – III**

**Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Secure Hash Algorithm (SHA) & MD5 Algorithm. Message Authentication Codes: Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MACs, MACs Based on Hash Functions: HMAC, MACs Based on Block Ciphers: DAA and CMAC. Digital Signatures: Digital Signatures, NIST Digital Signatures Algorithm.

### **UNIT – IV**

**Transport-Level Security:** Web Security Considerations, Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTPS, Secure Shell (SSH), E-Mail Security: Pretty Good Privacy, S/MIME. IP Security: IP Security Overview, IP Security Architecture, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange. Intruders, Virus and Firewalls: Intruders, Intrusion Detection, Password Management, Virus and Related Threats, Countermeasures, Firewall Design Principles, Types of Firewalls.

### **TEXT-BOOK**

1. WILLIAM STALLINGS, CRYPTOGRAPHY AND NETWORK SECURITY – PRINCIPLES AND PRACTICE (6E)

### **REFERENCE BOOKS**

1. ZHENFU CAO, NEW DIRECTIONS OF MODERN CRYPTOGRAPHY
2. DOUGLAS R. STINSON, CRYPTOGRAPHY THEORY AND PRACTICES
3. TOM ST DENIS, SIMON JOHNSON, CRYPTOGRAPHY FOR DEVELOPERS
4. A. MENEZES, P. VAN OORSCHOT, S. VANSTONE, HANDBOOK OF APPLIED CRYPTOGRAPHY
5. HENK C.A. VAN TILBORG, SUSHIL JAJODIA, ENCYCLOPEDIA OF CRYPTOGRAPHY AND SECURITY
6. KEITH M. MARTIN, EVERYDAY CRYPTOGRAPHY–FUNDAMENTAL PRINCIPLES AND APPLICATIONS
7. CHWAN-HWA WU, J. DAVID IRWIN, INTRODUCTION TO COMPUTER NETWORKS AND CYBER SECURITY
8. SAIFUL AZAD, AL-SAKIB KHAN PATHAN, PRACTICAL CRYPTOGRAPHY-ALGORITHMS AND IMPLEMENTATIONS USING C++

## **MCA 404T(A): OBJECT ORIENTED ANALYSIS AND DESIGN**

### **UNIT – I**

**Complexity:** The structure of complex systems, the inherent complexity of software, the five attributes of a complex system, organized and disorganized complexity, bringing order to chaos, on designing complex systems.

The object model: the evolution of the object model, foundations of the object model, elements of the object model, applying the object model.

### **UNIT – II**

**Classes and Objects:** the nature of an object, relationships among objects, the nature of a class, relationships among classes, the interplay of classes and objects, on building quality classes and objects. Classification: the importance of proper classification, identifying classes and objects, key abstractions and mechanisms. Notation: unified modeling language (UML), package diagrams, component diagrams, deployment diagrams.

### **UNIT – III**

**Notation:** use case diagrams, activity diagrams, class diagrams, sequence diagrams, interaction overview diagrams, composite structure diagrams, state machine diagrams, timing diagrams, object diagrams, communication diagrams. Process: first principles, the macro process: SDLC, the micro process: the analysis and design process. Pragmatics: management and planning, staffing, release management, reuse, quality assurance and metrics, documentation, tools, special topics, the benefits and risks of object-oriented development.

### **UNIT – IV**

**System Architecture** – Satellite-Based Navigation: inception, elaboration, construction, post-transition. Control System – Traffic Management: inception, elaboration, construction, post-transition, Web Application – Vacation Tracking System: inception, elaboration, construction, transition and post-transition,

### **TEXT-BOOK**

1. GRADY BOOCH, OBJECT-ORIENTED ANALYSIS AND DESIGN WITH APPLICATIONS

### **REFERENCE BOOKS**

1. ALI BAHRAMI, OBJECT ORIENTED SYSTEMS DEVELOPMENT
2. GRADY BOOCH, THE UNIFIED MODELING LANGUAGE USER GUIDE
3. BERD OESTEREICH, DEVELOPING SOFTWARE WITH UML – OOAD IN PRACTICE
4. SARNATH RAMNATH, BRAHMA DATHAN, OBJECT-ORIENTED ANALYSIS AND DESIGN
5. B. D. MCLAUGHLIN, GARY POLLICS, DAVID WEST, HEAD FIRST – OBJECT ORIENTED ANALYSIS & DESIGN , New Delhi.

## **MCA 404T (B): DISTRIBUTED SYSTEMS**

### **UNIT – I**

**Introduction:** definition of distributed system, goals, types of distributed systems. Architectures: architectural styles, system architectures, architectures versus middleware, self-management in distributed systems. Processes: threads, virtualization, clients, servers, code migration.

### **UNIT – II**

**Communication:** Remote Procedure Call, Message-Oriented Communication, Stream-Oriented Communication, Multicast Communication. Naming: names, identifiers, and addresses, flat naming, structured naming, attribute based naming. Synchronization: clock synchronization, logical clocks, mutual exclusion, global positioning of nodes, election algorithms.

### **UNIT – III**

**Consistency and Replication:** introduction, data-centric consistency models, client-centric consistency models, replica management, consistency protocols. Fault Tolerance: introduction, process resilience, reliable client-server communication, reliable group communication, distributed commit, recovery. Security: introduction, secure channels, access control, security management.

### **UNIT – IV**

**Distributed Object-Based Systems:** architecture, processes, communication, naming, synchronization, consistency and replication, fault tolerance, security. Distributed File Systems: architecture, process, communication, naming, synchronization, consistency and replication, fault tolerance, security. Distributed Web-based Systems: architecture, process, communication, naming, synchronization, consistency and replication, fault tolerance, security.

### **TEXT BOOK**

1. ANDREW S.TANENBAUM, MAARTEN VAN STEEN, DISTRIBUTED SYSTEMS – PRINCIPLES AND PARADIGMS (2E)

### **REFERENCE BOOKS**

1. SUKUMAR GHOSH, DISTRIBUTED SYSTEMS AN ALGORITHMIC APPROACH
2. JOEL M. CRICLOW, DISTRIBUTED SYSTEMS COMPUTING OVER NETWORKS
3. KAI HWANG, DISTRIBUTED AND CLOUD COMPUTING FROM PARALLEL PROCESSING TO INTERNET OF THINGS
4. AJAY D. KSHEMKALYANI, MUKESH SINGHAL, DISTRIBUTED COMPUTING PRINCIPLES, ALGORITHMS, AND SYSTEMS
5. GEORGE COULOURIS, JEAN DOLLIMORE, TIM KINDBERG, GORDON BLAIR, DISTRIBUTED SYSTEMS CONCEPTS AND DESIGN

## **MCA 405T(A): DESIGN AND ANALYSIS OF ALGORITHM**

### **UNIT – I**

**Introduction:** Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types. Fundamentals of the Analysis of Algorithm: The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive & Recursive Algorithms. Brute Force Search: Selection Sort, Bubble Sort, Sequential Search, Brute-Force String Matching, Exhaustive Search, Depth-First Search, Breadth-First Search.

### **UNIT – II**

**Decrease-&-Conquer:** Insertion Sort, Topological Sorting, Binary Search, Interpolation Search  
**Divide-and-Conquer:** Merge Sort, Quick Sort, Multiplication of Large Integers, Strassen's Matrix Multiplication, Transform-and-Conquer: Presorting, Balanced Search Trees, Heaps and Heap Sort, Problem Reduction. Space and Time Trade-Offs: Hashing, B-Trees.

### **UNIT – III**

**Dynamic Programming:** Knapsack Problem, Optimal Binary Search Trees, Warshall's and Floyd's Algorithms. Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes. Iterative Improvement: Simplex Method, Maximum-Flow Problem.

### **UNIT – IV**

**Limitations of Algorithm Power:** Lower-Bound Arguments, Decision Trees, P, NP, and NP-Complete Problems. Backtracking: n-Queens Problem, Hamiltonian Circuit Problem, Subset-Sum Problem, Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesman Problem, Approximation Algorithms for the Knapsack Problem.

### **TEXT BOOK**

1. ANANYEVITIN, INTRODUCTION TO THE DESIGN AND ANALYSIS OF ALGORITHMS (3E)

### **REFERENCE BOOKS**

1. RICHARD NEAPOLITAN, FOUNDATIONS OF ALGORITHMS
2. THOMAS H. CORMEN, INTRODUCTION TO ALGORITHMS
3. E.HOROWITZ, S. SAHNI, FUNDAMENTALS OF COMPUTER ALGORITHMS
4. A.V. AHO, J.V. HOPCROFT, J.D. ULLMANN, THE DESIGN AND ANALYSIS OF COMPUTER ALGORITHMS
5. DONALD E KNUTH, THE ART OF PROGRAMMING\_VOLUMES-1, 2, 3, 4



## **MCA 405T(B): SOFT COMPUTING**

### **UNIT – I**

AI Problems and Search: AI problems, Techniques, Problem Spaces and Search, Heuristic Search Techniques- Generate and Test, Hill Climbing, Best First Search Problem reduction, Constraint Satisfaction and Means End Analysis. Approaches to Knowledge Representation- Using Predicate Logic and Rules.

### **UNIT – II**

Supervised Learning Networks-perceptron, Back propagation algorithm-Classification Problem-Speech Processing Case study.

Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Maxnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization,

### **UNIT – III**

Introduction to Classical Sets ( crisp Sets)and Fuzzy Sets- operations and Fuzzy sets. Classical Relations –and Fuzzy Relations- Cardinality, Operations, Properties and composition. Tolerance and equivalence relations.Membership functions- Features, Fuzzification, membership value assignments, Defuzzification.

### **UNIT – IV**

Fuzzy Arithmetic and Fuzzy Measures, Fuzzy Rule Base and Approximate Reasoning Fuzzy Decision making ,Fuzzy Logic Control Systems. Genetic Algorithm- Introduction and basic operators and terminology. Applications: Optimization of TSP, Internet Search Technique

### **TEXT BOOKS**

1. Principles of Soft Computing- S N Sivanandam, S N Deepa, Wiley India, 2007
2. Soft Computing and Intelligent System Design –Fakhreddine O Karray, Clarence D Silva,. Pearson Edition, 2004.

### **REFERENCE BOOKS**

1. Artificial Intelligence and Soft Computing- Behavioural and Cognitive Modelling of the Human Brain- Amit Konar, CRC press, Taylor and Francis Group.
2. Artificial Intelligence – Elaine Rich and Kevin Knight, TMH, 1991, rp2008.
3. Artificial Intelligence – Patric Henry Winston – Third Edition, Pearson Education.
4. A first course in Fuzzy Logic-Hung T Nguyen and Elbert A Walker, CRC. Press Taylor and Francis Group.

## **MCA 406P: WEB TECHNOLOGIES - LAB**

1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble: [www.amazon.com](http://www.amazon.com) The website should consist the following pages.
  1. Home page
  2. Registration and user Login
  3. User Profile Page
  4. Books catalog
  5. Shopping Cart
  6. Payment By credit card
  7. Order Conformation
2. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
3. Write a PHP Program to read the employee details.
4. Write a PHP program to prepare the student marks list.
5. Write a PHP program to generate the multiplication of two matrices.
6. Write a PHP Application to perform demonstrates the college website.
7. Write a PHP application to update Rows in a Table.
8. Develop a PHP application to make following Operations
  - i. Registration of Users.
  - ii. Insert the details of the Users.
  - iii. Modify the Details.
  - iv. Transaction Maintenance.
    - A) No of times Logged in
    - B) Time Spent on each login.
    - C) Restrict the user for three trials only.Delete the user if he spent more than 100 Hrs of transaction
9. Write a programme on connecting to database on type 1 driver?
10. Write a program to insert the record in the table?
11. Write a program to delete the record in the database?
12. Write a programme to update the record in the database?
13. Write a program to Display the records in the database?
14. Write a program to excute the mutiple DML statements in the database?
15. Write a programme to insert records in the MS Access?
16. Write the programme to accept the vaues from key board to insert the values in database?
17. Write the programme to accept the vaues from key board to update the values in database?
18. Write the program to connect the mysql and insert the values?
19. Write a programe to call a procedure?
20. Develop a voter application using servlets.
21. Develop a web applicaton using doGet() method?
22. Develop a web applicaton using doPost() method?
23. Develop a web applicaton to insert ,delete,update and display the records in database?

## **MCA 407P: DATA MINING - LAB**

1. Introduction about launching the Weka tool.
2. Introduction to Weka Explorer.
3. Introduction to the classification of Mining techniques.
4. Introduction to Attribute Relation File Format (ARFF).
5. To perform Preprocessing, Classification and Visualization techniques on Customer dataset.
6. To perform Preprocessing, Classification and Visualization techniques on Agriculture dataset.
7. To perform Preprocessing, Classification and Visualization techniques on Weather dataset.
8. To perform Clustering technique on Customer dataset.
9. To perform Clustering technique on Agriculture dataset.
10. To perform Clustering technique on Weather dataset.
11. To perform Association technique on Customer dataset.
12. To perform Association technique on Agriculture dataset.
13. To perform Association technique on Weather dataset.
14. To perform all the techniques on Friends dataset.
15. To perform all the techniques on Company dataset.

## **MCA 408P: NETWORK SECURITY - LAB**

1. Write a program to generate cipher text and recover the plaintext using
  - a) Caesar-Cipher text algorithm
  - b) Product Cipher
2. Write a program to generate cipher text and recover the plaintext using
  - a) Play fair cipher
  - b) Hill cipher
3. Write a program to generate random numbers using pseudo random number generation algorithm
4. Write a program to implement Poly-Alphabetic Cipher
5. Write a program to implement Transposition Cipher and Rail fence Technique
6. Write a program to implement DES Algorithm
7. Write a program to implement AES Algorithm
8. Write a program to implement RSA public key encryption algorithm
9. Write a program to implement Diffie- Helman Key Exchange Algorithm
10. Write a program to implement SHA-1 algorithm
11. Write a program to implement MD5 algorithm
12. Write a program to Implement the Signature Scheme Digital Signature Standard
13. Write a program to retrieve the data from the database and encrypt them using any encryption algorithm
14. Generation of public key and private keys
15. Write a program to write the contents in the file in encrypted manner and read them in decrypted manner using any algorithm
16. Write a program to implement FEM (Fast Exponentiation Method) for find congruential values used in RSA algorithms. Ex:  $577 \bmod 45$ .

### **Note:**

- Recommended to use the C/C++/Java (without using collection framework) language on Linux systems.

## **MCA 501T: PROGRAMMING IN C#**

### **UNIT – I**

**Introduction:** architecture and components of .NET framework 4.5, managed code, intermediate language, meta data and JIT compiler, common language runtime, automatic memory management, private and shared assemblies, exploring Visual Studio .NET IDE.

**Introduction to C#:** identifiers, keywords, data types, variables, constants, operators, precedence, associativity, type conversion, decision and loop statements, generics, enumerations, namespaces. **Object Oriented Programming:** encapsulation, inheritance, polymorphism, abstraction, interfaces.

### **UNIT – II**

**Introduction to Windows Programming:** creating windows forms, windows controls, menus and dialogue boxes, overview of xml. Window programming vs. Window presentation foundation, main features of WPF 4.5, WPF 4.5architecture, WPF 4.5class hierarchy, types of WPF applications, WPF4.5designer interface, Using XAML in WPF 4.5 applications, WPF properties, WPF events, working with dialog boxes in WPF application, compiling and running WPF 4.5 applications, WPF 4.5controls, resources, styles, templates, commands.

### **UNIT – III**

Introduction to SQL, architecture of ADO.NET, creating a connection to a database, OLEDB database, using OLEDB adapter for excel file, ODBC data source, ADO.NET commands, data adapters, creating data view, data reader, stored procedures.

**Introduction to ASP.NET:** ASP.NET life cycle, exploring ASP.NET 4.5 web application, creating a sample ASP.NET 4.5 website, application structure and state, global. asax application file, web forms – standard controls, validation controls, master pages, web services.

### **UNIT – IV**

**Introduction to ASP.NET MVC:** ASP.NET MVC project templates, understanding the structure of an ASP.NET MVC project, naming conventions, creating views, defining controllers, defining a data model, execution flow of MVC. Deploying Websites and Services, deploying to internet information server, preparing a web application using IIS manager, web deploying IIS.

### **TEXT-BOOK**

1. KOSENT LEARNING SOLUTIONS INC., .NET 4.5 PROGRAMMING – BLACK BOOK (DREAMTECH)

### **REFERENCE BOOKS**

1. JOSEPH ALBAHARI, BEN ALBAHARI, C# 6.0 IN A NUTSHELL
2. CHRISTIAN NAGEL, PROFESSIONAL C# 6 AND .NET CORE 1.0
3. ANDREW TROELSEN, PHILIP JAPIKSE, C# 6.0 AND THE .NET 4.6
4. PAUL DEITEL, HARVEY DEITEL, VISUAL C# 2012 – HOW TO PROGRAM
5. BILL SEMPFF, CHUCK SPHAR, STEPHEN R DAVIS, C# 5.0 ALL-IN-ONE FOR DUMMIES
6. MARK J PRICE, C# 6 AND .NET CORE 1.0 – MODERN CROSS-PLATFORM DEVELOPMENT
7. BENJAMIN PERKINS, JACOB VIBE HAMMER, JON D. REID, BEGINNING C# 6 PROGRAMMING WITH VISUAL STUDIO 2015

## **MCA 502T: BIG DATA ANALYTICS**

### **UNIT – I**

**Overview of Big Data:** What is Big Data? Evolution of Big Data, Structuring Big Data, Elements of Big Data, Big Data Analytics. Exploring the Use of Big Data in Business Context: Use of Big Data in Social Networking, Use of Big Data in Preventing Fraudulent Activities, Use of Big Data in Detecting Fraudulent Activities in Insurance Sector, Use of Big Data in Retail Industry. Introducing Technologies for Handling Big Data: Distributed and Parallel Computing for Big Data, Introducing Hadoop. Understanding Hadoop Ecosystem: Hadoop Ecosystem, HDFS, MapReduce, Hadoop YARN, HBase, Hive, Pig and Pig Latin, Sqoop, ZooKeeper, Flume, Oozie.

### **UNIT – II**

**Understanding MapReduce Fundamentals and HBase:** The MapReduce Framework, Techniques to Optimize MapReduce Jobs, Role of HBase in Big Data Processing. Exploring the Big Data Stack, Virtualization and Big Data, Virtualization Approaches. Storing Data in Databases and Data Warehouses: RDBMS and Big Data, Non-Relational Database, Integrating Big Data with Traditional Data Warehouses, Big Data Analysis and Data Warehouse, Changing Deployment Models in Big Data Era. Processing Your Data with MapReduce: Developing Simple MapReduce Application, Points to Consider while Designing MapReduce. Customizing MapReduce Execution: Controlling MapReduce Execution with InputFormat, Reading Data with Custom RecordReader, Organizing Output Data with OutputFormats, Customizing Data with RecordWriter, Optimizing MapReduce Execution with Combiner, Implementing a MapReduce Program for Sorting Text Data.

### **UNIT – III**

**Understanding Hadoop YARN Architecture:** Introduction YARN, Advantages of YARN, YARN Architecture, Working of YARN. Exploring Hive: Introducing Hive, Getting Started with Hive, Hive Services, Data Types in Hive, Built-In Functions in Hive, Hive DDL, Data Manipulation in Hive, Data Retrieval Queries, Using JOINS in Hive. Analyzing Data with Pig: Introducing Pig, Running Pig, Getting Started with Pig Latin, Working with Operators in Pig, Working with Functions in Pig, Debugging Pig, Error Handling in Pig.

### **UNIT – IV**

**Using Oozie:** Introducing Oozie, Installing and Configuring Oozie, Understanding the Oozie Workflow, Simple Application. NoSQL Data Management: Introduction to NoSQL, Types of NoSQL Data Models, Schema-Less Databases, Materialized Views, Distributed Models, Sharding, MapReduce Partitioning and Combining, Composing MapReduce Calculations. Understanding Analytics and Big Data: Comparing Reporting and Analysis, Types of Analytics, Developing an Analytic Team. Analytical Approaches and Tools to Analyze Data: Analytical Approaches, History of Analytical Tools, Introducing Analytical Tools, Comparing Various Analytical Tools.

### **TEXT-BOOK**

1. DT EDITORIAL SERVICES, BIG DATA – BLACK BOOK (DREAMTECH)

### **REFERENCE BOOKS**

1. RADHA S, M. VIJAYALAKSHMI, BIG DATA ANALYTICS
2. ARSHDEEP B AND VIJAY M, BIG DATA SCIENCE & ANALYTICS – A HANDS-ON APPROACH.
3. FRANK OHLHORST, BIG DATA FUNDAMENTALS – CONCEPTS, DRIVERS, TECHNIQUES
4. KUAN-CHING LI, H JIANG, L T YANG, A CUZZOCREA, BIG DATA ALGORITHMS, ANALYSIS AND APPLICATIONS
5. TOM WHITE, HADOOP: THE DEFINITIVE GUIDE
6. SHIVA ACHARI, HADOOP ESSENTIALS

## **MCA 503T: MOBILE APPLICATION DEVELOPMENT**

### **UNIT – I**

**Introduction to Android Operating System:** Android OS design and Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools

**Android application components** – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes

**Android Application Lifecycle** – Activities, Activity lifecycle, activity states, monitoring state changes

### **UNIT - II**

**Android User Interface:** Measurements – Device and pixel density independent measuring units Layouts – Linear, Relative, Grid and Table Layouts

**User Interface (UI) Components** – Editable and non editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers

**Event Handling** – Handling clicks or changes of various UI components

**Fragments** – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

### **UNIT - III**

**Intents and Broadcasts:** Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS

**Broadcast Receivers** – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity

**Notifications** – Creating and Displaying notifications, Displaying Toasts

### **UNIT - IV**

**Persistent Storage:** Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference

**Advanced Topics: Alarms** – Creating and using alarms

**Using Internet Resources** – Connecting to internet resource, using download manager

**Location Based Services** – Finding Current Location and showing location on the Map, updating location

### **TEXT BOOKS**

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox) , 2012
2. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013

### **REFERENCES**

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

## **MCA 504T(A): AUTOMATA THEORY**

### **UNIT – I**

**Fundamentals** – alphabets, strings, languages, problems, graphs, trees, Finite State Systems, definitions, Finite Automaton model, acceptance of strings, and languages, Deterministic finite automaton and Nondeterministic finite automaton, transition diagrams, transition tables, proliferation trees and language recognizers, equivalence of DFA's and NFA's.

Finite Automata with  $\epsilon$ -moves, significance, acceptance of languages,  $\epsilon$ -closure, Equivalence of NFA's with and without  $\epsilon$ -moves, Minimization of finite automata, Two-way finite automata, Finite Automata with output– Moore and Meelay machines.

### **UNIT – II**

**Regular Languages:** regular sets, regular expressions, identity rules, constructing finite automata for a given regular expressions, conversion of finite automata to regular expressions. Pumping lemma of regular sets and its applications, closure properties of regular sets.

**Grammar Formalism:** Regular grammars–right linear and left linear grammars, equivalence between regular linear grammar and finite automata, inter conversion, Context free grammar, derivation trees, sentential forms, right most and leftmost derivation of strings, ambiguity.

### **UNIT – III**

**Context Free Grammars:** Simplification of Context Free Grammars, Chomsky normal form, Greiback normal form, Pumping lemma for context free languages and its applications, closure of properties of CFL (proofs omitted).

**Push Down Automata:** PDA definition, model, acceptance of CFL, acceptance by final state and acceptance by empty state and its equivalence. Equivalence of PDA's and CFL's, inter-conversion. (Proofs not required).

### **Unit – IV**

**Membership Algorithm (CYK Algorithm) for Context Free Grammars.**

**Turing Machine:** TM definition, model, design of TM, computable functions, unrestricted grammars, recursively enumerable languages. Church's hypothesis, counter machine, types of Turing machines (proofs omitted). Linear bounded automata and Context sensitive language.

**Computability Theory:** Chomsky hierarchy of languages, Introduction to DCFL, DPDA, LR(0) grammar, decidability and undecidable problems. Definitions of P and NP problems, NP complete and NP hard problems.

### **TEXT-BOOK**

1. J. E. HOPCROFT, J. D. ULLMAN, INTRODUCTION TO AUTOMATA THEORY, LANGUAGES, AND COMPUTATION

### **REFERENCE BOOKS**

1. JOHN C. MARTIN, INTRODUCTION TO LANGUAGES AND THE THEORY OF COMPUTATION
2. MISHRA, CHANDRASHEKARAN, THEORY OF COMPUTER SCIENCE
3. PERTER LINZ, AN INTRODUCTION TO FORMAL LANGUAGES AND AUTOMATA
4. ZVIKOHAV, NIRAJ K JHA, SWITCHING AND FINITE AUTOMATA THEORY

## **MCA 504T (B): MOBILE COMPUTING**

### **UNIT – I**

**Introduction to Mobile Computing:** applications, a simplified reference model, Wireless Transmission: frequencies of radio transmission, signals, antennas, signal propagation, multiplexing, modulation, spread spectrum, cellular system. Media Access Control: motivation for a specialized MAC, SDMA, FDMA, TDMA, CDMA, and Comparisons.

### **UNIT – II**

GSM, DECT, Wireless LAN: Infrared vs. radio transmission, Infrastructure and ad-hoc networks, IEEE 802.11, HPERLAN, Bluetooth.

### **UNIT – III**

**Mobile Network Layer:** mobile IP, dynamic host configuration protocol, ad-hoc networks. Mobile Transport Layer: Traditional TCP, classical TCP improvements, TCP over 2.5/3G wireless networks.

### **UNIT – IV**

File Systems, World Wide Web, Wireless Application Protocol (WAP) and WAP 2.0.

### **TEXT BOOK**

1. JOCHEN H. SCHILLER, MOBILE COMMUNICATIONS(2E)

### **REFERENCE BOOKS**

1. RAJ KAMAL, MOBILE COMPUTING
2. ASOKE K. TALUKDER, ROOPA R. YAVAGAL, MOBILE COMPUTING
3. MAZLIZA OTHMAN, PRINCIPLES OF MOBILE COMPUTING AND COMMUNICATIONS
4. PRASANT KUMAR PATTNAIK, RAJIB MALL, FUNDAMENTALS OF MOBILE COMPUTING
5. IVAN STOJMENOVIC, HANDBOOK OF WIRELESS NETWORKS AND MOBILE COMPUTING
6. DAVID TANIAR, MOBILE COMPUTING CONCEPTS, METHODOLOGIES, TOOLS, AND APPLICATIONS



## **MCA 505T(A): INFORMATION RETRIEVAL SYSTEMS**

### **UNIT – I**

**Introduction:** Motivation, Basic Concepts, Past-Present and Future, the Retrieval Process  
**Modeling :** Introduction, A Taxonomy of Information retrieval Models, Retrieval: Ad hoc and Filtering, A Formal Characteristics of IR Models, Classic Information Retrieval, Alternative Ste Theory Models, Alternative Probabilistic Models, Structured Text Retrieval Models, Model for Browsing

### **UNIT - II**

**Retrieval Evaluation** Introduction, retrieval Performance Evaluation, Reference Collections  
**Query languages** Introduction, Keyword-Based Querying, Pattern Matching, Structural Queries, Query Protocols  
**Query Operations** Introduction, User Relevance Feedback, Automatic Local Analysis, Automatic global Analysis  
**Text Operations** Introduction, Document Preprocessing, Document Clustering, Text Compression, Comparing text Compression Techniques

### **UNIT - III**

**Indexing and Searching** Introduction, Inverted Files, Other Indices for Text, Boolean queries, Sequential Searching, pattern Matching, Structural Queries, Compression  
**Searching the Web** Introduction, Challenges, Characterizing the Web, Search Engines, Browsing, Metasearches, Finding the Needle in the Haystack, Searching using Hyperlinks

### **UNIT - IV**

**User Interfaces and Visualization** Introduction, human-Computer Interaction, The Information Access Process, Starting Points, Query Specification, Context, User Relevance Judgments, Interface Support for the Search Process

### **TEXT BOOKS:**

1. Modern Information Retrival by Yates and Neto Pearson Education.

### **REFERENCES:**

1. Kowalski, Gerald, Mark T Maybury: Information Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 1997.
2. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
3. Information Storage & Retieval By Robert Korfhage – John Wiley & Sons.

## **MCA 505T(B): INTERNET OF THINGS**

### **UNIT – I**

**Introduction to Internet of Things:** Introduction, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, IoT Levels & Deployment Templates, Domain Specific IoTs: Home Automation, Cities, Environment, Energy, Retail, Agriculture, Health & Lifestyle.

### **UNIT – II**

**IoT and M2M:** Introduction to M2M, Difference between IoT and M2M, SDN and NFV for IoT. IoT System Management with NETCONF-YANG: Need for IoT Systems Management, SNMP, Network Operator requirements, NETCONF, YANG, IoT Systems Management with NETCONF-YANG. IoT Platforms Design Methodology: Introduction, IoT Design Methodology, Case Study on IoT system for weather Monitoring. Motivation for Using Python. Python Packages for IoT.

### **UNIT – III**

**IoT Physical Devices & Endpoints:** What is an IoT Device, Exemplary Device: Raspberry Pi, About the Board, Linux on Raspberry Pi, Raspberry Pi Interfaces, programming Raspberry Pi with Python, Other IoT Devices. IoT Physical Servers & Cloud Offerings: Introduction to Cloud Storage Models & Communication APIs, WAMP-AutoBahn for IoT, Xively Cloud for IoT, Python Web Application Framework-Django, Designing a RESTful Web API, Amazon Web Services for IoT, SkyNet IoT Messaging Platform.

### **UNIT – IV**

**Case Studies of IoT Design:** Home Automation, Cities, Environment, Agriculture, Productivity Applications. Introduction to Data Analytics for IoT, Apache Hadoop, YARN, Oozie, Spark, Storm, Health Monitoring Case study. An IoT Tool: chef, Chef Case Studies.

### **TEXT BOOK**

1. ARSHDEEP BAHGA, VIJAY MADISETTI, INTERNET OF THINGS – A HANDS ON APPROACH

### **REFERENCE BOOKS**

1. GRAHAM MEIKLE, MERCEDES BUNZ, THE INTERNET OF THINGS
2. RAJKUMAR BUYYA, AMIR VAHID DASTJERDI, INTERNET OF THINGS
3. ADRIAN MCEWEN, HAKIM CASSIMALLY, DESIGNING THE INTERNET OF THINGS
4. OLIVIER H, DAVID B, OMAR E, THE INTERNET OF THINGS: KEY APPLICATIONS AND PROTOCOLS
5. JEAN PHILIPPE V, ADAM DUNKEL, INTERCONNECTING SMART OBJECTS WITH IP: THE NEXT INTERNET
6. DANIEL MINOLI JOHN, BUILDING THE INTERNET OF THINGS WITH IPV6 AND MIPV6 – THE EVOLVING WORLD OF M2M COMMUNICATIONS
7. OVIDIU VERMESAN, PETER FRIESS, INTERNET OF THINGS CONVERGING TECHNOLOGIES FOR SMART ENVIRONMENTS AND INTEGRATED ECOSYSTEM.

## **MCA 506P: PROGRAMMING IN C# - LAB**

1. Using Windows Forms Application, collect the user details like First Name, Middle Name, Last Name, Address, Gender, User Photo, Course name, Course Timing with Submit and Clear buttons. Display the User Details in another form.
2. Create a Windows Forms Application containing different genre of movies and two buttons to display the genre liked by user and favorite genre selected by user.
3. Create a Windows Forms Application to demonstrate any 3 mouse and keyboard events each.
4. Create a Windows Forms Application to load an image into a picture box using file open dialogue box.
5. Create a WPF Application for auto complete text box with suitable windows icon.
6. Create a WPF Application to validate user data like name (cannot be more than 10 characters), age (must be a number), phone number (xxxx-xxxx-xx).
7. Create a WPF Application for a bank customer. If user name and password is correct then display a welcome screen with options "balance enquiry", "deposit", "withdraw" and "transfer money". Display Successful or failure messages accordingly.
8. Create a WPF calendar control with the following features.
  - Change the calendar background and foreground
  - Select a date
  - Blackout your vacation dates
  - Change the first day of week and highlight today's date
  - Select 2nd, 4th Saturdays and Sundays from the calendar
  - Display a selected date, month and year in separate textboxes
9. Develop a database application to store the details of an employee using ADO.NET using windows Forms.
10. Develop a database application using ADO.NET to insert, modify, update and delete operations on and display the result of the following queries.
  - Find all clerks who earn between 1000 and 2000
  - List of employees who are working in dept. 20 or getting salary is more than 3000
  - Only names of employees, which have TH or LL in their names
  - Display salary, dept. and names of employees whose name contains the letter 's' and ends with 's'
11. Create a web form to collect user details like Name (required field validator), password (Range Validator), confirm password (compare validator), email (regular expression validator), country name and phone number (range validator) with suitable validation controls. Display validation summary. If validation is successful display the user details.
12. Create a web application using master page for which displays course list on left pane. On selecting the course its content.
13. Create a web application using master page with a title "DATA CENTER". In which left pane contains "Home" and "Databases". Display the table data selected from "Databases" in a GridView control with add, edit, delete records and Page size options and also DetailsView control.
14. a. Create a web service application to call web service to calculate compound interest for the given data.  
b. Create a database web service to display the list of employees in the decreasing order of their Job and salary.
15. a. Create a simple MVC application to display a message "MVC is Amazing...!".  
b. Create a simple MVC application which accepts user name and password with suitable validation controls. If the details provided are correct then say Hello to the user.
16. Create movie database using MVC with the following specifications
  - List a set of movie database records
  - Create a new movie database record
  - Edit an existing movie database record

### **Note:**

- Recommended to use the C/C++/Java (without using collection framework) language on Linux systems.

## **MCA 507P: BIG DATA ANALYTICS - LAB**

1. Perform setting up and installing Hadoop in its three operating modes: stand alone, Pseudo distributed.
2. Perform some tasks by using web based tools of Hadoop system.
3. Implement the following file management tasks in Hadoop:
  - Adding file and directories
  - Creating file, Retrieving file and deleting files
4. Write a Map Reduce program for basic word count.
5. Write a Map Reduce program for sorting text data.
6. Write a Map Reduce program for analyzing student report.
7. Write a Map Reduce program for mining weather data.
8. Installing and running Hive, practice some Hive commands.
9. Using Hive; create, insert, update, alter, delete, and drop the tables
10. Using Hive; query the data from the data base tables.
11. Using Hive; create views, use functions, create indexes for the data base tables.
12. Installing and running Pig, practice some Pig commands.
13. Write Pig Latin scripts using eval functions to analyze your data.
14. Write Pig Latin scripts using math functions to analyze your data.
15. Write Pig Latin scripts using string functions to analyze your data.
16. Write simple scripts to understand the using NoSQL in Hadoop systems.

### **Note:**

- Recommended to install Hadoop on Linux systems.
- Recommended to use the Java/Python/Rlanguage on Linux systems.

## **MCA 808P: MOBILE APPLICATION DEVELOPMENT - LAB**

The student is expected to be able to do the following problems, though not limited.

1. Create an Android application that shows Hello + name of the user and run it on an emulator.  
(b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.
3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a "Back" button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use fragment transactions and Rotation event listener.
4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.
5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.

6. Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.
7. Create a user registration application that stores the user details in a database table.
8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.
9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.
10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.
11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.
12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.
13. Create an application that shows the given URL (from a text field) in a browser.  
Develop an application that shows the current location's latitude and longitude continuously as the device is moving (tracking).
14. Create an application that shows the current location on Google maps.

**Note:**

Android Application Development with MIT App Inventor: For the first one week, the student is advised to go through the App Inventor from MIT which gives insight into the various properties of each component.

The student should pay attention to the properties of each components, which are used later in Android programming. Following are useful links:

1. <http://ai2.appinventor.mit.edu>
2. [https://drive.google.com/file/d/0B8rTtW\\_91YclTWF4czdBMEpZcWs/view](https://drive.google.com/file/d/0B8rTtW_91YclTWF4czdBMEpZcWs/view)