BANGALORE UNIVERSITY

Regulations of Master of Computer applications (MCA) Course

1 **TITLE OF THE COURSE:** The course shall be called MCA – Master of Computer Applications.

2 **DURATION OF THE COURSE**: The course of study shall be three years.

3 **ELIGIBILITY FOR ADMISSION**: A candidate with any degree of a minimum of 3 years duration (10+2+3) of Bangalore university or of any other University equivalent there in to with a minimum of 50% of marks in the aggregate of all subjects including languages, if any, provided further , that the candidate has studied mathematics / Computer science /Business mathematics / Statistics / Computer applications / Electronics as a subject at PUC level or equivalent HSC (XII Standard) or at Degree level is eligible for admission to MCA Course. Relaxation to SC/ST, Group I be extended as per University norms.

4 **ATTENDANCE(for Regular course)**: In each Semester a candidate should be considered to have successfully undergone the prescribed Course of study if the candidate has attended at least 75% of the classes in each subject (Theory , Lab & Practical).

5 **SCHEME OF EXAMINATION(regular)**:

A The Internal Assessment marks should be decided for each of the theory subjects by conducting 2 tests , each of 60 minutes duration, spread over the span of a Semester . A seminar should also be given by the student in the third year and the same to be assessed and evaluated for internal assessment along with the two tests.

B The Internal Assessment marks in Practical course is based on the performance in the Laboratory. The Internal Assessment marks for Project work of a candidate is based on the dissertation Seminar.

6 **ELIGIBILITY TO GO TO THE HIGHER SEMESTER**:

A A Candidate is allowed to carry over all the previous uncleared (failed) theory papers and Practicals to subsequent semesters from the first to sixth semester.

B The maximum period for completion of the course shall be six years from the date of admission.

7 **MINIMUM FOR PASS AND DECLARATION OF RESULTS**

A For a pass in a semester, a candidate shall secure a minimum of 40% of the marks prescribed for a subject in the University Examination (Theory, Practical, Project work) and 50% of the marks in the aggregate inclusive of the Internal Assessment marks obtained in all subjects put together.

B The candidates who do not satisfy 7(a) shall be deemed to have failed and have to take exams in the subjects in which he has secured less than 40% at the University examination.

C Provision is made for rejection of results of all the subjects of a Semester only once, if the candidate decides to reappear for all the subjects of that semester. Such rejection should be made within 30 days of announcement of result, by making a written application, through the Head of the Institution. If such rejection is in respect of the results of all the subjects of one semester and earn fresh Internal marks as well.

D The results of any semester will be declared as pass or fail as the case may be in accordance with regulation 7(a).

E To be eligible for the award of the MCA degree, a candidate shall have completed the scheme of training and passed in all subjects prescribed for the Course

F Further to regulation 7(a), the classification followed by the University for all PG courses shall be made applicable for the declaration of results of each Semester.

8 **CLASSIFICATION OF RESULT FOR THE MCA COURSE AND DECLARATION OF RANKS**:

Further to regulations 7(a) and 7(f), the names of all successful candidates securing First Class with Distinction and First Class in the First attempt shall be arranged in the order of Merit and only first FIVE Ranks shall be declared.

9 A candidate shall complete examinations of all Semesters of MCA Course within - **SIX** years from the date of admission

SCHEME OF STUDY AND EXAMINATION FOR MASTER of COMPUTER APPLICATIONS (MCA)

I SEMESTER

Subject code	Subject Name	Lecture & Tutorials Hrs/week	Lab Hrs/ week	Internal Assess-ment	Exam marks (Maximum)	Total
1 MCA 1	Discrete Mathematics	4		20	80	100
1 MCA 2	Financial Accounting	4		20	80	100
1 MCA 3	Concepts of Computing & Problem Solving	4		20	80	100
1 MCA 4	Computer Organization	4		20	80	100
1 MCA 5	Assembly Language Programming 8086	4		20	80	100
1 MCA 6	Problem Solving Lab	---	6	20	80	100
1 MCA 7	Assembly Language Programming Lab	---	6	20	80	100
					Total	700

Duration of Exam: 3 Hours (Theory and practical)
Total no. of hours per semester in each paper: 52 hrs/semester (both theory and practicals)

II SEMESTER

Subject code	Subject Name	Lecture & Tutorials Hrs/week	Lab Hrs/ week	Internal Assess-ment	Exam marks (Maximum)	Total
2 MCA 1	Statistical Analysis	4		20	80	100
2 MCA 2	Data Structures	4		20	80	100
2 MCA 3	Operating System & UNIX	4		20	80	100
2 MCA 4	Database Management	4		20	80	100
2 MCA 5	Advanced programming in JAVA	4		20	80	100
2 MCA 6	Data Structures Lab	---	6	20	80	100
2 MCA 7	JAVA Lab	---	6	20	80	100
					Total	700

Duration of Exam: 3 Hours (Theory and practical)
Total no. of hours per semester in each paper: 52 hrs/semester (both theory and practicals)

III SEMESTER

Subject code	Subject Name	Lecture & Tutorials Hrs/week	Lab Hrs/ week	Internal Assess-ment	Exam marks (Maximum)	Total
3 MCA 1	Fundamentals of Algorithms	4		20	80	100
3 MCA 2	System Software	4		20	80	100
3E3A	Elective I	4		20	80	100
3 MCA 4	Object Oriented Analysis and Design	4		20	80	100
3 MCA 5	Theory of Computation	4		20	80	100
3 MCA 6	Soft Skills Practices	---	6	20	80	100
3 MCA 7	Database Lab	---	6	20	80	100
					Total	700

Duration of Exam: 3 Hours (Theory and practical)
Total no. of hours per semester in each paper: 52 hrs/semester (both theory and practicals)

IV SEMESTER

Subject code	Subject Name	Lecture & Tutorials Hrs/week	Lab Hrs/ week	Internal Assess-ment	Exam marks (Maximum)	Total
4 MCA 1	Software Engineering	4		20	80	100
4 MCA 2	Computer Networks	4		20	80	100
4 MCA 3	Computer Graphics	4		20	80	100
4 E4A	Elective II	4		20	80	100
4 E5A	Elective III	4		20	80	100
4 MCA 6	Algorithm Lab	---	6	20	80	100
4 MCA 7	Graphics Lab	---	6	20	80	100
					Total	700

Duration of Exam: 3 Hours (Theory and practical)
Total no. of hours per semester in each paper: 52 hrs/semester (both theory and practicals)

V SEMESTER

Subject code	Subject Name	Lecture & Tutorials Hrs/week	Lab Hrs/ week	Internal Assess-ment	Exam marks (Maximum)	Total
5 MCA 1	Human Resource Management	4		20	80	100
5 MCA 2	Information & Network Security	4		20	80	100
5 MCA 3	Internet Technology	4		20	80	100
5 E5A	Elective IV	4		20	80	100
5 E5B	Elective V	4		20	80	100
5 MCA 6	Internet Programming & UNIX Lab	---	6	20	80	100
5 MCA 7	Software Engineering Lab	---	6	20	80	100
					Total	700

Duration of Exam: 3 Hours (Theory and practical)
Total no. of hours per semester in each paper: 52 hrs/semester
(both theory and practicals)

VI SEMESTER

Individual Project Work and Viva Voce

Duration of the project is six months. During the project work the students are to interact with the Internal guides / External guides.

The evaluation pattern is:
IA – 150
Project demo + Viva - 350
Total marks - 500

Scheme for the examination:

Part A: 4 questions (out of 6) with 8 marks each
Part B: 4 questions (out of 6) with 12 marks each

1MCA1: Concepts of Computing and Problem Solving

UNIT I:

Algorithms and Flowcharts **8 Hours**

The meaning of algorithms, Flowcharts and their need, Writing algorithms and drawing flowcharts for simple exercises like finding largest of three numbers, roots of given quadratic equation, the biggest and smallest of given set of numbers and such other problems

Constants, Variables and Data Types

Character set, C tokens, keywords & identifiers, structure of C program, executing a C program. Constants, variables, data types, declaration of variables, declaration of storage classes, assigning values to variables defining symbolic constants, declaring a variable as constant, declaring a variable as volatile, overflow and underflow of data.

UNIT II:

Operators and Expressions **14 Hours**

Arithmetic operators, relational operators, logical operators, assignment operator, increment and decrement operator, conditional operator, bitwise operators, comma operator, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, type conversions in expressions, operator precedence and associativity, mathematical functions

Managing Input and Output Operations

The *scanf() & printf()* functions for input and output operations, reading a character, writing a character, (the *getchar() & putchar()* functions) , the address operator(&), formatted input and output using format specifiers, Writing simple complete C programs.

Control Statements

Decision making with *if* statement, simple if statement, the *if..else* statement, nesting of *if..else* statements, the *else..if* ladder, the *switch* statement, the **?:** operator, the *goto* statement, the break statement, programming examples

Loop Control Structures

The while statement, the do..while statement, the for statement, nested loops, jumps in loops, the continue statement, programming examples

UNIT III:

Arrays **8 Hours**

The meaning of an array, one dimensional and two dimensional arrays, declaration and initialization of arrays, reading , writing and manipulation of above types of arrays, multidimensional arrays, dynamic arrays, programming examples.

Character Arrays and Strings

Declaring and initialing string variables, reading string from terminal, writing string to screen, arithmetic operations on characters, putting strings together, comparison of two strings, string handling functions, table of strings, other features of strings, programming examples.

UNIT IV:

User Defined Functions **14 Hours**

Need for user defined functions, a multi function program, elements of User defined functions, defining functions, return values and their types, function calls, function declaration, category of functions, no arguments and no return values, arguments but no return values, arguments with return values, no

arguments with return value, functions that return multiple values, nesting of functions, recursion, passing arrays to functions, passing string to functions, programming examples.

Structures and Unions
Defining a structure, declaring structure variables, accessing structure members, structure initialization, copying and comparing structure variables, operations on individual members, array of structures, structures within structures, structures and functions, Unions, size of structures, bit fields, programming examples

Pointers
Understanding pointers, accessing the address space of a variable, declaring and initialization pointer variables, accessing a variable through its pointer, chain of pointers, pointer expressions, pointers and arrays, pointer and character strings, array of pointers, pointer as function arguments, functions returning pointers, pointers to functions, pointers and structures, programming examples

UNIT V:
File Management in C **08 Hours**
Defining and opening a file, closing a file, input/output operations on files, error handling during I/O operations, random access files, command line arguments, programming examples.

Dynamic Memory Allocation
Dynamic memory allocation, allocating a block of memory: m*alloc*, allocating multiple blocks of memory: *calloc*, releasing the used space: *Free*, altering the size of a block: r*ealloc*, programming examples

The Preprocessor
Introduction, macro substitution, files inclusion, compiler control directives, ANSI additions, programming exercises.

Reference Books:

1. Balagurusamy: Programming in ANSI C, 4th Edition, Tata McGraw Hill, 2008.
2. V Rajaraman: Computer Programming in C, PHI, 2000.
3. Behrouz A Forouzan and Richard F Gilberg: Structured Programming Approach C, 2nd Edition, Thomson, 2005.
4. M G Venkateshmurthy: Programming Techniques through C,Pearson Education, 2005.
5. Ivor Horton: Beginning C from Novice to Professional, 4th Edition, Springer, 2005.
6. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Education, 2002.

1MCA2: Discrete Mathematics

UNIT I: Set Theory **6 Hours**

Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, A First Word on Probability, Countable and Uncountable Sets

UNIT II: Fundamentals of Logic **13 Hours**

Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference; The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems

UNIT III: Properties of the Integers **7 Hours**

Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions

UNIT IV: Relations and Functions **14 Hours**

Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions – Stirling Numbers of the Second Kind, Special Functions, The Pigeon-hole Principle, Function Composition and Inverse Functions; Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions

UNIT V: Groups, Coding Theory and Rings **12 Hours**

Definitions, Examples, and Elementary Properties, Homomorphisms, Isomorphisms, and Cyclic Groups, Cosets, and Lagrange's Theorem,Elements of Coding Theory, The Hamming Metric, The Parity Check, and Generator Matrices

Group Codes, Rings and Modular Arithmetic
Decoding with Coset Leaders, Hamming Matrices

The Ring Structure – Definition and Examples, Ring Properties and Substructures, The Integers Modulo n

Reference Books:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education, 2004.
2. Kenneth H. Rosen:, Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.
3. Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
4. D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications, Thomson, 2004**.**
5. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

1MCA3: Computer Organization.

UNIT I

Number Systems and Logic Gates **10 Hours**

Counting in Decimal and Binary, Place Value, Binary to Decimal Conversion, Decimal to Binary Conversion, Hexadecimal Numbers, Octal Numbers, Bits, Bytes, Nibbles, and Word Size, The AND Gate, The OR gate, The inverter and Buffer, The NAND gate, the NOR Gate, The exclusive OR gate, The Exclusive NOR Gates, The NAND Gate as an universal Gate, Gates with More than two inputs, Using Inverters to convert gates.

UNIT II

Combining Logic Gates and Arithmetic Circuits **10 Hours**

Constructing Circuits from Boolean Expression, Drawing a circuit from a Maxterm Boolean Expression, Truth Tables and Boolean Expressions, Sample Problem Simplifying BooleanExpression, Karnaugh Maps, Karnaugh Maps with three variables, Karnaugh Maps with four variables, more Karnaugh Maps, using Demorgan's Theorem, Binary Addition, Half Adders, Full Adders, , Binary Subtraction, Parallel Subtractor and adders, Binary Multiplication, Binary Multipliers, Addition and Subtraction, r complement, (r-1) complement.

UNIT III

Sequential circuit design **10 Hours**

Latches , SR Flip Flops, concept of edge triggering, D- flip flop ,JK- flip flop, Master slave flip flop, T- flipflop, Registers, shift Registers, asynchronous and synchronous counters, Mod 10 – counter.

UNIT III

Basic Structure of Computer & Machine Instructions **12 Hours**

Computer Types, Functional Units, Basic Operational Concepts, Bus structures, Performance, Memory Location and Addresses, Memory Operations, Instructions & Instruction cycle, Addressing Modes, Assembly Language, Basic Input/Output Operations

UNIT IV: Input/Output Organization & Memory Systems **10 Hours**

Accessing I/O Devices, Interrupts, Direct Memory Accesses, System bus, Interface Circuits, hard disk, floppy disk, CD-ROM. Some Basics concepts, Semiconductors RAM Memories, Read-Only Memories, Cache Memories, Virtual Memories

Reference Books:

1. Tokheim: Digital Electronics Principles and Applications, McGraw Hill, 6th Edition, 2004.
2. Carl Hamacher, Z Varnesic and S Zaky: Computer Organization, 5th Edition, McGraw Hill, 2002.
3. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson, 2008.
4. Morris Mano, and Charles R Kime: Logic and Computer Design Fundamentals, 2nd Edition, Pearson Education, 2001.

1MCA4: Assembly Language Programming 8086.

UNIT I: 15hours

Introduction to Microprocessor: Introduction, Applications, Basic block diagram, speed, word size, memory capacity, classification of Microprocessors (mention of different microprocessors)

Microprocessor 8085: Features, Architecture–Block diagram, Internal registers, Register pairs, Flags, Stack pointer, Program counter, Pin description of 8085,Types of Buses, Multiplexed Address and Data bus, Generation of Control signals.

Interfacing I/O devices:Basic Interfacing Concepts, Memory-Mapped I/O,I/O mapped I/O

8085 Instructions: Operation code, Operand & Mnemonics. Instruction set of 8085, Instruction classification, Addressing modes, Instruction format. Data transfer Instructions, Arithmetic instructions, Increment & Decrement instructions, Logical instructions, Branch instructions and Machine control instructions.

Stack operations and Timing diagrams: Stack operations, Subroutine calls and return operations. Timing diagrams-Instruction cycle, Machine cycle, T- states, Delay loops Time delay-Numerical examples. Simple Programming examples.

Unit II: **10 hours**

8086Architecture and programming: 8086 Architecture and programming model, registers, flags, memory segmentation, pin description, odd & even bank of memory, interfacing of memory RAM and EPROM.

Addressing modes: Immediate addressing, register addressing, memory addressing, indexed addressing with displacement, I/O port addressing.

Unit III 15 hours

8086 Instructions: Instruction template for 8086 instructions, code generation using template.

Data Transfer Instruction: Move data to register/memory from register/memory/immediate data, data transfer between a segment register and register/memory, PUSH and POP, exchange, data transfer with I/O ports.

Data Conversion instructions: XLAT, LEA, LDS, LES, LAHF and SAHF instructions.

Arithmetic Instructions: Add, subtract, negate, compare, CBW, CWD, multiply and divide instructions.

Logical Instructions: AND, OR, EX-OR, Test, NOT, ROTATE and shift instructions.

Process Control Instructions: Instructions to set/reset flags, halt, wait, lock, prefix and escape to co-processor instructions.

String Instructions: CMPS, MOVS, LODS, STOS, and SCAS instructions.

Branch Instructions: JMP, conditional jump, LOOP, LOOPE, LOOPNE, JCXZ, CALL, RET.

Unit IV **8 hours**

Interrupts of 8086: Hardware interrupt, software interrupt and exception, priority of interrupts

Assembly language programming: Assembly language programming examples, subroutines and macros, examples.

Hardware features of 8086: Bus buffering, latching, timing diagrams, wait state, MIN/MAX modes of operation.

UNIT V **4 hours**

Advanced Microprocessor Features: Segmentation, Paging, Real and Virtual Mode Execution, Protection Mechanism, Task Management, Concluding comment.

Reference books:

1. Ramesh S. Gaonkar, "Microprocessor Architecture, programming, and Applications", 2nd Edition, New age International.
2. Microprocessors and Interfacing: Douglas V.Hall, Second Edition, TMH.
3. Advanced Microprocessors and IBM -PC Assembly Language Programming: Dr.K.Udayakumar, B.S.Umashankar, TMH
4. Advanced Microprocessors and Peripherals, Architecture, Programming and Interfacing : Ray Bhurchandi, Tata McGraw Hill
5. 8088/8086 Processors Programming, Interfacing, Software, Hardware and applications: Walter A.Triebel and Avatar Singh, PHI.
6. Daniel Tabak, "Advanced Microprocessors", 2nd Edition, McGraw-Hill.

1MCA5: ACCOUNTING AND FINANCIAL MANGEMENT

UNIT I **15 Hrs**

Accounting: Principles, concepts and conventions, double entry system of accounting, Introduction to basic books of accounts of sole proprietary concern, closing of books of accounts and preparation of trial balance.

Final Accounts: Trading, Profit and Loss accounts and Balance Sheet of sole proprietary concern (Without adjustments).

UNIT II **15 Hrs**

Financial Management: Meaning, scope and role, A brief study of functional areas of financial management. Introduction to Various FM Tools: Ratio Analysis, Fund flow statement & Cash flow statement. Introduction to Cost Accounting: Nature, Importance & Basic Principles. Brief Introduction to methods of Costing & Elements of Cost, Unit Costing.

UNIT III **5 Hrs**

What is Tally? Using Tally Software: Introduction and Installation, General, Number symbols, accts/inv info menu, voucher entry, invoice/orders entry and printing, security issue.

UNIT IV **10 Hrs**

Working in Tally: Ledger Accounts, writing voucher, voucher entry, different types of voucher, correcting sundry debtors and sundry creditor's accounts, Trial balance, Accounts books, Cash book, Bank books, Ledger Accounts, Group Summary, Sales Register and Purchase Register, journal Register, Statement of Accounts & Balance sheet.

UNIT V **7 Hrs**

Output reports, basic features of displaying reports, printing reports, other printing options, display account books and statements, viewing cash/bank books, configure balance sheet, columnar balance sheet, show fortnightly balance sheet, integrate accounts with inventory, display profit and loss account.

Reference Books :

1. Ramachandran, "Financial Accounting for Managers", Tata McGraw Hill - 2005
2. I.M. Pandey,: Financial Management, Vikas Publications, 2003
3. P.H. Bassett, "Computerised Accounting", BPB. 2003
4. Neeraj Sharma "Computerized Accounting & Business Systems", Kalyani Publishers. 2004
5. Jain and Narang, "Principles of Accounting".-2003
6. P.V. Kulkarni, "Financial Management", Himalaya Publishing House., 2003
7. Sharma, Gupta & Bhalla, "Management Accounting". 2004
8. Jain and Narang, "Cost Accounting"., 2004
9. Katyal, "Cost Accounting"., 2003
10. Charlotte Eudy McConn, "Business Computer Systems: Design, Programming & Maintenance", (PHI), 2004.
11. Kellock. J : Elements of Accounting, Heinemann., 2003
12. Rockely.L.E : Finance for the Non-accountant, 2nd Ed., Basic Books., 2003
13. Levy and sarnat : Principles of Financial Management, Prentice –Hall International. 2004

1MCA6: Concepts of Computing and Problem Solving Lab

SECTION A

1. Write a program to print whether the number entered is even or odd use conditional operators.
2. Write a program to convert hexadecimal to decimal numbers
3. Write a program to display list of C program files and directories.
4. Write a program to ensure that the difference between any two digit number and its reverse is always a multiple of nine.
5. Write a program to display number of days in calendar format of an entered month of current year.
6. Write a program to display the numbers in increasing and decreasing order using infinite loop.
7. Write a program to accept a number and find the sum of its individual digits repeatedly till the result is a single digit.
8. Write a program to enter integer number and find the largest and smallest digit of the number.
9. Write a program to read three digits +ve integer number 'n' and generate possible permutations of number using their digits.
10. Accept a text upto 50 words and perform following actions
    a) Count total vowels, constants, spaces, sentences and words with spaces.
    b) Program should erase more than one space between two successive words.
11. Write a program to enter names of cities and display all the entered names alphabetically.
12. Write a program to enter some text and display the text in reverse order(Eg. I am happy as Happy am I).

SECTION B

13. Write a program to calculate the result of the following with recursive calls of function.
    X = 1! + 2! + 3!----------+ n!.
14. Write a program to use macros as an array and pointer.
15. Write a program to display the attributes of a file using dos interrupt.
16. Write a program to delete a file using dos interrupt.
17. Create user defined data type equivalent to int. Declare three variables of its type. Perform arithmetic operations using these variables.
18. Write a program to reboot the system. Use following data with int 86() function.
    a) Interrupt 0x19. b) Input Void(nothing)
19. Write a program to read a C program file and count the following in the complete program.
    a) Total number of statements b) Total number of included files
    c) Total number of brackets.
20. Write a program to display C Program files in current directory. The user should select on of the files. Convert the file contents in Capital and Display the same on the screen.
21. Write a program to delete the given file from the disk.
22. Write a program to read the contents of three files and find the largest file.
23. Write a program to interchange the contents of two files.
24. Write a program to change mouse cursor.

1MCA 7: Assembly Language Programming Lab

(All programs to be written using 8086 assembly language)

1. Addition, subtraction, multiplication and division of 8-bit, 16-bit binary and decimal numbers.
2. Addition and subtraction of two 32-bit binary and decimal numbers.
3. Average of N- 8-bit/16-bit binary and decimal numbers.
4. To generate the Fibonacci series up to the given limit (both binary and decimal)
5. To find Minimum and maximum out of N numbers
6. To sort given N numbers in ascending/descending order
7. To Find the GCD of 2 integer numbers (both binary and decimal)
8. To calculate factorial of a given number using recursion technique.
9. To generate and print prime numbers up to a limit N (both binary and decimal).
10. To generate and print perfect dividing numbers up to a limit N (both binary and decimal).
11. a) Conversion of array of Binary code to Gray code.
    b) Conversion of array of Gray code to Binary code
12. a) To find the Sum and difference of two matrices of order MxN and PxQ (both binary and decimal)
    b) To find the transpose of given MxN matrix
13. Search for an element using binary search in an array of an 8-bit signed numbers.
14. Reverse of an array of numbers,
15. Reverse of an byte and word

2MCA1: Data Structures

Unit I: Introduction: Algorithm, Characteristics of algorithm, Algorithm analysis, Pseudo code, The Abstract Data Type, A Model for an Abstract Data Type, Time and Space Complexity, worst case, best case and average case complexity, The Big O notation. Definition of data structure and Classification of data structures. (**8 Hours)**

Unit II: Introduction to Linear List, Creation, Traversing, Insertion, deletion, searching an element in Singly Linked List, creation; Creation, Traversing, Insertion, deletion, searching an element doubly Linked List; Creation, Traversing, Insertion, deletion, searching an element Circular Linked List; Introduction to Header Linked List, Polynomial addition using linked list. (**10 Hours)**

Unit III: Basic Stack Operations, Table and Linked implementation of stack, Stack Applications - Procedure call, Recursion, Conversion of Infix expression into Postfix expression, and Evaluation of Postfix expression; Queue Operations, Linear Queue and Circular Queues, Table and Linked Implementation of Queues, Queue Applications - process scheduling algorithms. (**10 Hours)**

Unit IV: Graph terminology, Representation of Graphs in memory – adjacency matrix, and path matrix representation, Warshall's algorithm, Linked representation of Graphs, Binary Tree, Binary Trees creation, insertion and delete operation on binary tree, Binary Tree Traversal algorithms, Expression Trees, Binary Search Trees, creation, insertion, deletion, searching and traversing in Binary Search Tree, Multiway Trees: B-Tree – Creation, Insertion and deletion in 2-3-4 Tree, B+ Tree. (**10 Hours)**

Unit V: Linear Searching, Binary Search Algorithms, Hashing, Hashed List Searches, and Collision Resolution, General Sort concepts: Internal and external Sorting, Bubble Sort, Insertion Sort, Selection Sort, Exchange Sort and heap sort algorithms. (**10 Hours)**

References:

1. Robert L. Kruse, Bruce P. Leung, Clovis L.Tondo, "*Data Structures and Program Design in C*" (2nd Edition). Prenctice Hall India, 2001.
2. Mark Allen Weiss, "*Data Structures and Algorithm Analysis in C*" (2nd Edition). Addison-Wesley, 1996.
3. Richard F.Gilberg, Behrouz A. Forouzan, "*Data Structures: A Pseudo code Approach with C*". Thomson Asia Pvt. Ltd, 2002

2MCA2: Statistical Techniques

UNIT 1: **12 hours**

Sample spaces - events - Axiomatic approach to probability - conditional probability - Independent events - Baye's formula - Random Variables - Continuous and Discrete random variables - distribution function of a random variables - Characteristic of distributions - Expectation, variance - coefficient of variation, moment generation function - Chebyshev's inequality

UNIT 2: **10 hours**

Bivariate distribution - conditional and marginal distributions - Discrete distributions - discrete uniform, Binomial poison and geometric Distributions - Continuous distributions - Uniform, Normal, Exponential and Gamma distributions.

UNIT 3: **8 hours**

Correlation coefficient - Rank correlation coefficient of determination - Linear Regression - Method of Least squares - Fitting of the curve of the form $ax + b$, $ax^2 + bx + c$, ab^x and ax^b - multiple and partial correlation (3 - variables only).

UNIT 4: **15 hours**

Concept of sampling – Methods of sampling - simple random sampling - Systematic sampling and stratified random sampling (descriptions only) - concepts of sampling distributions and standard error - point estimation (concepts only) - Interval Estimation of mean and proportion. Tests of Hypotheses - Critical Region - two types of Errors - Level of significance - power of the test - Large sample tests for mean and proportion - Exact tests based on Normal, t, F and Chi-square distributions.

UNIT 5 : **7 hours**

Basic principles of experimentation - Analysis of variance - one way and two way classifications - computing randomized design - Randomized Block design - Time series Analysis - Measurement of Trend and Seasonal variations.

Reference Books:

1. Mood, A.M., Graybill, F. and Boes, 1974, Introduction to Mathematical Statistics, McGraw-Hill.
2. Trivedi, K.S, 1994, Probability and Statistics with Reliability, Queuing and Computer Science Applications. Prentice Hall India, New Delhi.
3. Arnold O. Allen, 1978, Probability, Statistics and Queuing Theory with Computer Science Application.
4. Bajpai, A.C. Calus, I.M. Fairley, J.A., 1979, Statistical Methods for Engineers and Scientists. John Wiley & Sons.
5. Doughlas, C.,Montagomery, Lynwood,A. & Johnson, 1976, Forecasting and Time Series Analysis, Tata McGraw-Hill, New Delhi.
6. Baisnab, A.P. and Manoranjan Jas, 1993, Elements of Probability and Statistics, Tata McGraw-Hill, New Delhi.

7. Kossack, C.F. and Hensschkec, C.I., Introduction to Statistics and Computer Programming, Tata McGraw-Hill, New Delhi.

2MCA3: OPERATING SYSTEMS

UNIT I:

Introduction to Operating Systems, Process Management 13 Hours

What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments. Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot.

Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling.

UNIT II:

Process Synchronization & Dead Locks 13 Hours

Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

UNIT III:

Memory Management 7 Hours

Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

UNIT IV:

File System, Implementation of File System 13 Hours

File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management

UNIT V

Secondary Storage Structures, Protection and case study 12 Hours

Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems.

Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication.

Reference Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 7th edition, Wiley India, 2006.

2. D.M Dhamdhere: Operating systems - A concept based Approach, 2nd Edition, Tata McGraw- Hill, 2002.
3. P.C.P. Bhatt: Operating Systems, 2nd Edition, PHI, 2006.
4. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 2005.

2MCA4: Database Management Systems

UNIT I: Introduction **10 Hours**

Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems.

UNIT II: Entity-Relationship Model **10 Hours**

Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two.

UNIT III: Relational Model and Relational Algebra **8 hours**

Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations : JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping.

UNIT IV: **12 Hours**

SQL Data Definition and Data Types; Specifying basic constraints in SQL; Schema change statements in SQL; Basic queries in SQL; More complex SQL Queries. Insert, Delete and Update statements in SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL / PSM.

UNIT V: Database Design & Transaction Management **12 Hours**

Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form.

The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery and Database security.

Reference Books:

1. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Pearson Education, 2007.

2. Silberschatz, Korth and Sudharshan: Data base System Concepts, 5th Edition, Mc-GrawHill, 2006

3. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003.

4. C.J. Date, A. Kannan, S. Swamynatham: A Introduction to Database Systems, 8th Edition, Pearson education, 2006.

2MCA5: Advanced Programming (JAVA)

UNIT – I **12 Hrs**

Introduction to Java: Importance and features of Java, Keywords, constants, variables and Data Types, Operators and Expressions, Decision Making, Branching and Looping: if..else, switch,?: operator, while, do, for statements, labeled loops, jump statements: break, continue return. Introducing classes, objects and methods: defining a class, adding variables and methods, creating objects, constructors, class inheritance.

Arrays and String: Creating an array, one and two dimensional arrays, string array and methods, Classes: String and String Buffer classes, Wrapper classes: Basics types, using super, Multilevel hierarchy abstract and final classes, Object class, Packages and interfaces, Access protection, Extending Interfaces, packages.

UNIT – II **10 Hrs**

Exception Handling: Fundamentals exception types, uncaught exceptions, throw, throw, final, built in exception, creating your own exceptions, Multithreaded Programming: Fundamentals, Java thread model: priorities, synchronization, messaging, thread classes, Runnable interface, inter thread Communication, suspending, resuming and stopping threads.

UNIT –III **10 Hrs**

Input/Output Programming: Basics, Streams, Byte and Character Stream, predefined streams, Reading and writing from console and files. Using Standard Java Packages (lang, util, io, net). Networking: Basics, networking classes and interfaces, using java.net package, doing TCP/IP and Data-gram Programming

UNIT – IV **12 Hrs**

Event Handling: Different Mechanism, the Delegation Event Model, Event Classes, Event Listener Interfaces, Adapter and Inner Classes, Working with windows, Graphics and Text, using AWT controls, Layout managers and menus, handling Image, animation, sound and video, Java Applet. Beans: Introduction to Java Beans and Swings, Servlets

UNIT-V **8 Hrs**

Database Connectivity, JDBC architecture, Establishing connectivity and working with connection interface, Working with statements , Creating and executing SQL statements, Working with ResultSet

Reference Books:

1. Patrick Naughton and Herbertz Schildt, "Java-2 The Complete Reference", TMH, 1999.
2. Muthu C., Programming with Java (Second Edition), TMH
3.Balagurusamy E., Programming with Java, TMH

2MCA6: DATA STRUCTURES LAB

1. Write a menu driven program to implement linear and binary search also find the location of its first occurrence
2. Write a menu driven program to sort the array in ascending/descending order using a) Quick sort b) Merge sort
3. Write a menu driven program to create a linked list and to perform insert and delete operations
4. Write a program to add two polynomials using a linked list
5. Write a menu driven program to perform insert and delete operations in a circular linked list
6. Write a menu driven program to perform operations on a stack (linked list implementation)
7. Write a menu driven recursive program to a) find factorial of a given number b) generate first N terms of a fibonacci sequence c) GCD of three numbers
8. Write a program to solve the problem of towers of hanoi with 3 pegs and N discs
9. Write a menu driven program to perform operations on a circular queue (linked list implementation)
10. Write a menu driven program to a) find the length of a string b) concatenate two strings c) to extract a substring from a given string d) finding and replacing a string by another string in a text (Use pointers and user-defined functions)
11. Write a program to convert the given infix expression into its postfix form
12. Write a program to evaluate the postfix expression with a set of values
13. Write a menu driven program to a create binary tree and to perform insert and delete operations
14. Write a menu driven program to create a binary search tree and to perform inorder, preorder and postorder traversals
15. Write a program to sort N elements in ascending order using heap sort
16. Write a program to obtain the path matrix of the given graph

2MCA7: JAVA PROGRAMMING LAB

1. Programs using constructor and destructor
2. Creation of classes and use of different types of functions
3. Count the number of objects created for a class using static member function
4. Write programs on interfaces
5. Write programs on packages
6. Write programs using function overloading
7. Programs using inheritance
8. Programs using IO streams
9. Programs using files
10. Write a program using exception handling mechanism
11. Programs using AWT
12. Programs on swing
13. Programs using JDBC