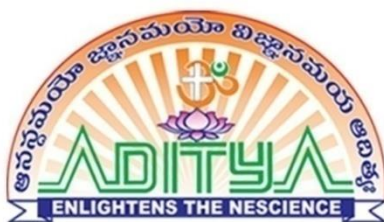# PROGRAM STRUCTURE AND SYLLABUS

MASTER OF COMPUTER APPLICATIONS

For
MCA THREE YEAR DEGREE PROGRAM
(Applicable for the batches admitted from 2017-18)

ADITYA ENGINEERING COLLEGE

An Autonomous Institution
Approved by AICTE, Affiliated to JNTUK & Accredited by NBA, NAAC with 'A' Grade
Recognized by UGC under the sections 2(f) and 12(B) of UGC act 1956
Aditya Nagar, ADB Road, SURAMPALEM - 533 437

VISION & MISSION OF THE COLLEGE

VISION

To induce higher planes of learning by imparting technical education with International standards, Applied research, Creative ability and Value based instruction to emerge as a premier institute.

MISSION

Achieving academic excellence by providing globally acceptable technical education by

forecasting technology through

- Innovative research & development
- Industry institute interaction
- Empowered manpower

PROGRAM STRUCTURE

I SEMESTER

Course Code	Name of the Course	Lecture (L)	Tutorial (T)	Practice (P)	Credits (C)
173MC1T01	C Programming & Data Structures	3	1	---	3
173MC1T02	Computer Organization	3	1	---	3
173MC1T03	Discrete Mathematical Structures & Graph Theory	3	1	---	3
173MC1T04	Statistics With R Programming	3	1	---	3
173MC1T05	Accounting & Financial Management	3	1	---	3
173MC1L01	English Language Communication Skills Lab	---	---	3	2
173MC1L02	C Programming Lab	---	---	3	2
173MC1L03	Statistics With R Programming Lab	---	---	3	2
	TOTAL	**15**	**5**	**9**	**21**

II SEMESTER

Course Code	Name of the Course	Lecture (L)	Tutorial (T)	Practice (P)	Credits (C)
173MC2T06	OOPS Through Java	3	1	---	3
173MC2T07	Operating Systems	3	1	---	3
173MC2T08	Software Engineering	3	1	---	3
173MC2T09	Optimization Techniques	3	1	---	3
173MC2T10	Computer Graphics	3	1	---	3
173MC2L04	OOPS Through Java Lab	---	---	3	2
173MC2L05	Data Structures Lab	---	---	3	2
173MC2L06	Operating System & Computer Graphics Lab	---	---	3	2
	TOTAL	**15**	**5**	**9**	**21**

III SEMESTER

Course Code	Name of the Course	Lecture (L)	Tutorial (T)	Practice (P)	Credits (C)
173MC3T11	Database Management Systems	3	1	---	3
173MC3T12	Computer Networks	3	1	---	3
173MC3T13	Unix Programming	3	1	---	3
173MC3T14	Management Information System	3	1	---	3
173MC3T15	Design & Analysis of Algorithms	3	1	---	3
173MC3L07	Database Management Systems Lab	---	---	3	2
173MC3L08	Unix Programming Lab	---	---	3	2
173MC3L09	Computer Networks Lab	---	---	3	2
	TOTAL	15	5	9	21

IV SEMESTER

Course Code	Name of the Course	Lecture (L)	Tutorial (T)	Practice (P)	Credits (C)
173MC4T16	Object Oriented Analysis & Design	3	1	---	3
173MC4T17	Advanced Java & Web Technologies	3	1	---	3
173MC4T18	Data Warehousing & Mining	3	1	---	3
Elective – I					
173MC4E01	Mobile Computing				
173MC4E02	Human Computer Interaction	3	1	---	3
173MC4E03	Cloud Computing				
Elective – II					
173MC4E04	Software Project Management				
173MC4E05	Artificial Intelligence	3	1	---	3
173MC4E06	Embedded Systems				
173MC4L10	Advanced Java & Web Technologies Lab	---	---	3	2
173MC4L11	Data Warehousing & Mining Lab	---	---	3	2
173MC4L12	Object Oriented Analysis & Design Lab	---	---	3	2
	TOTAL	15	5	9	21

V SEMESTER

Course Code	Name of the Course	Lecture (L)	Tutorial (T)	Practice (P)	Credits (C)
173MC5T19	Big Data Analytics	3	1	---	3
173MC5T20	Network Programming	3	1	---	3
173MC5T21	Python Programming	3	1	---	3
Elective – III					
173MC5E07	Cyber Security				
173MC5E08	Computer Forensics	3	1	---	3
173MC5E09	E – Commerce				
Elective – IV					
173MC5E10	Internet of Things				
173MC5E11	Multimedia Application Development	3	1	---	3
173MC5E12	Software Testing Methodologies				
173MC5L13	Big Data Analytics Lab	---	---	3	2
173MC5L14	Network Programming Lab	---	---	3	2
173MC5L15	Python Programming Lab	---	---	3	2
	TOTAL	**15**	**5**	**9**	**21**

VI SEMESTER

Course Code	Name of the Course	Lecture (L)	Tutorial (T)	Practice (P)	Credits (C)
173MC6R01	Seminar	---	---	---	2
173MC6P01	Major Project	---	---	---	19
	TOTAL	---	---	---	21

C PROGRAMMING & DATA STRUCTURES

I Semester   **L   T   P   C**
Course Code: 173MC1T01   **3   1   0   3**

UNIT-I:
Introduction to Computers, HW and SW concepts, Algorithm, pseudo code, flowchart, program development steps, Introduction to various IDE's and their use in C program development, structure of C program, A Simple C program, identifiers, basic data types and sizes, Constants, variables, arithmetic, relational and logical operators, increment and decrement operators, conditional operator, bit-wise operators, assignment operators, expressions, type conversions, conditional expressions, precedence and order of evaluation. Control structures such as if, go to, labels, and switch statements.

UNIT-II:
Loops- while, do-while and for statements, break, continue, Arrays -concepts, declaration, definition, accessing elements, storing elements, Strings and string manipulations, 1- D arrays other than strings, 2-Dcharacter arrays – 2-D arrays other than character arrays – Multidimensional arrays.

UNIT-III:
Functions: basics, parameter passing, storage classes- extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions, header files, C preprocessor. Passing 1-D arrays, 2-D arrays, and functions. Pointers: concepts, initialization of pointer variables, pointers and Function arguments, passing by address –dangling memory, Character pointers and functions, pointers to pointers, pointers and multidimensional arrays, dynamic memory managements functions, command line arguments.

UNIT-IV:
Derived types: structures- declaration, definition and initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self referential structures, unions, typed of, bit-fields, Input and output – concept of a file, text files and binary files, Formatted I/o, file I/o operations Data Structures: Introduction to Data Structures – Time Complexity –Space Complexity – Pattern matching – naive method – Robin Karp Algorithm - Searching – Linear and binary search methods, sorting –Bubble sort, selection sort, Insertion sort, Quick sort, merge sort.

UNIT-V:
Single linked lists, doubly linked lists, circular list, representing stacks and queues in C using arrays and linked lists, infix to post fix conversion, postfix expression evaluation. Trees- Binary tress, terminology, representation, traversals, Graphs - terminology, representation, graph versals (dfs & bfs) –Warshalls – Dijkstra – Kruskal – Prims Algorithms

TEXT BOOKS:
1. C and Data Structures: A snapshot oriented treatise using live engineering examples, N B Venkateswarlu, E. V Prasad, S Chand & Co.
2. Computer science, A structured programming approach using C, B.A. Forouzan and R.F.Gilberg, Third edition, Thomson.

REFERENCE BOOKS:
1. Fundamentals of Data Structures in C, Horowitz, Sahni, Anderson- Freed, 2nd ed, universities Press, 2008.
2. Classic Data Structures, Samanta, 2nd ed, PHI, 2009.
3. The C Programming Language, B.W. Kernighan, Dennis M.Ritchie, PHI/ Pearson.
4. C Programming with problem solving, J.A. Jones & K. Harrow,   Dreamtech Press
5. Data Structures Using C , A.S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/ Pearson.
6. Programming in C , Stephen G. Kochan, III Edition, Pearson.
7. Data Structures and Program Design in C, R.Kruse,, Tondo, Leung, Shashi M, 2nd Edition, Pearson.
8. Data Structures and Algorithms, Aho, Hopcroft, Ullman, Pearson, 2006
9. C and Data Structures, Ashok N.Kamthane, Pearson.
10. C Programming and Data Structures, E Balaguruswamy, TMH, 2008.

****

COMPUTER ORGANIZATION

I Semester   **L   T   P   C**
Course Code: 173MC1T02   **3   1   0   3**

UNIT-I:
Basic Structure Of Computers: Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development.

UNIT-II:
Machine Instruction and Programs: Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types, Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equation. Component of Instructions: Logic Instructions, shift and Rotate Instructions

UNIT-III:
Type of Instructions: Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations

UNIT-IV:
INPUT/OUTPUT ORGANIZATION: Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB)

UNIT-V:
The MEMORY SYSTEMS: Basic memory circuits, Memory System Consideration, Read- Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, Cache Memories: Mapping Functions, INTERLEAVING Secondary Storage: Magnetic Hard Disks, Optical Disks, Processing Unit: Fundamental Concepts: Register Transfers, Performing An Arithmetic Or Logic Operation, Fetching A Word From Memory, Execution of Complete Instruction, Hardwired Control, Micro programmed Control: Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next –Address Field

TEXT BOOKS:
1. Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill.
2. Computer Architecture and Organization , John P. Hayes ,3rd Edition, McGraw Hill.

REFERENCE BOOKS:
1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI
2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson
3. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int.Edition.

****

DISCRETE MATHEMATICAL STRUCTURES & GRAPH THEORY

I Semester   **L   T   P   C**
Course Code: 173MC1T03   **3   1   0   3**

UNIT-I:
Mathematical Logic: Statements and notations, Connectives, Well formed formulas, Truth tables, tautology, equivalence implication, Normal forms, Theory of inference for the statement calculus, Rules of inference, Consistency of premises and indirect method of proof, Automatic Theorem Proving Predicate calculus: Predicates, statement functions, variables and quantifiers, predicate formulas, free & bound variables, universe of discourse, inference theory of predicate calculus

UNIT-II:
Set theory & Relations: Introduction, Relations and ordering, Properties of binary Relations, Equivalence, Compatibility Relations, Partial ordering, Hasse diagram. Functions: composition of functions, Inverse Function, Recursive Functions, Lattice and its Properties, Pigeon hole Principles and its application. Algebraic structures: Algebraic systems, Examples and general properties, Semi groups and monoids, groups, sub groups, Definitions, Examples, mom orphism, Isomorphism and related problems.

UNIT-III:
Elementary Combinatorics: Basis of counting, Enumeration of Combinations & Permutations, Enumerating of Combinations & Permutations with repetitions and constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, principles of Inclusion – Exclusion.

UNIT-IV:
Recurrence Relations: Generating Function of Sequences, Calculating Coefficient of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, The method of Characteristic roots, Solution of Inhomogeneous Recurrence Relation.

UNIT-V:
Graph Theory: Representation of Graph, Spanning Trees, BFS, DFS, Kruskals Algorithm, Binary trees, Planar Graphs, Graph Theory and Applications, Basic Concepts, Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers

TEXT BOOKS
1. Discrete Mathematical Structures with Applications to computer science J.P Tremblery, R.Manohar, TMH
2. Discrete Mathematical for computer Scientists & Mathematicians " J.L. Molt, A.Kandel, T.P.Baker, PHI

REFERENCE BOOKS:
1. Elements of Discrete Mathematics, C L Liu, D P Mohanpatra,TMH
2. Discrete Mathematics, Schaum's Outlines, Lipschutz, Lipson, TMH.
3. Discrete Mathematical Structures, Kolman, Busby, Ross, 6th ed., PHI, 2009
4. Discrete Mathematics, Johnsonbaugh, 6th ed., Pearson, 2005
5. Discrete Mathematics, Malik, Sen, 6th ed., Cengage Learning, 2004
6. Discrete Mathematics for computer science, Bogart, Stein and Drysdale, Springer, 2005

****

STATISTICS WITH R PROGRAMMING

I Semester	L	T	P	C
Course Code: 173MC1T04	3	1	0	3

UNIT-I:
Introduction, How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes.

UNIT-II:
R Programming Structures, Control Statements, Loops, - Looping Over Non vector Sets,- If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quick sort Implementation- Extended Example: A Binary Search Tree.

UNIT-III:
Doing Math and Simulation in R, Math Function, Extended   Example Calculating Probability- Cumulative Sums and Products-Minima and Maxima- Calculus, Functions Fir Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Extended Example: Finding Stationary Distribution of Markov Chains, Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files,

UNIT-IV:
Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function – Customizing Graphs, Saving Graphs to Files.

UNIT-V:
Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions Other Distribution, Basic Statistics,   Correlation and Covariance, T-Tests,-ANOVA. Linear Models, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Logistic Regression, - Poisson Regression- other Generalized Linear Models-Survival Analysis, Nonlinear Models, Spines- Decision- Random Forests

TEXT BOOKS:
1. The Art of R Programming, Norman Matloff, Cengage Learning
2. R for Everyone, Lander, Pearson

REFERENCE BOOKS:
1. R Cookbook, Paul Teetor, Oreilly.
2. R in Action, Rob Kabacoff, Manning

***

ACCOUNTING & FINANCIAL MANAGEMENT

I Semester   **L   T   P   C**
Course Code: 173MC1T05   **3   1   0   3**

UNIT-I:
Accounting Generally Accepted Accounting Principles (GAAP) & Accounting standards, Characteristics and limitations of single entry system, double entry system of accounting, introduction of basis books of accounts, ledgers. Preparation of trail balance – Final accounts – company final accounts – Users of   Accounting Information, Role of Accountant in modern Organizations.

UNIT-II:
Financial Management – meaning and scope, role, objectives of time value of money – over vitalization – under capitalization – profit maximization – wealth maximization – EPS maximization. Ration Analysis - advantages - limitations - Fund flow analysis – meaning, importance, preparation and interpretation of Funds flow and cash flow statements – statements of changes in working capital.

UNIT–III:
Costing – nature and importance and basic principles. Elements of cost – Absorption costing Vs. Marginal costing – Financial accounting Vs. cost Accounting Vs. management accounting. Marginal costing and Break – even Analysis: nature, scope and importance– practical applications of marginal costing, limitation and importance of cost – volume, profit analysis, short run decisions.

UNIT-IV:
Standard costing and budgeting : nature, scope and computation and analysis – materials variance, labor variance and sales variane – cash budget, sales - budget – flexible Budgets, master budgets.

UNIT-V:
Introduction to computerized accounting system: coding logic and codes, master files, transaction files, introduction documents used for data collection, processing of different files and Outputs obtained.

TEXT BOOKS:
1. Accounting for Managers, P. Vijaya Kumar, and Himalaya Publications.
2. Accounting for Management. Vijaya Kumar.TMH.
3. Financial Accounting, S.N Maheswari and S.K. Maheswari, Vikas.
4. Financial Accounting, A. Mukherjee and M. Heneef, TMH.

REFERENCE BOOKS:
1. Basic Financial Accounting for Management, Ambaresh Gupta,   Pearson.
2. Accounts And Finance for Non accounts, Chatterjee, D.K.Himalaya.
3. Financial Analysis and Accounting, P. Premchand Babu and M.   Madam Mohan, Himalaya.
4. Essential of Financial Accounting, Ashish, K and Ballacharya, PHI.
5. Guide to Financial Management, John Tannent, Viva.

****

ENGLISH LANGUAGE COMMUNICATION SKILLS LAB

	L	T	P	C
I Semester				
Course Code: 173MC1L01	**0**	**0**	**3**	**2**

Objectives: The language lab focuses computer-aided multi-media instruction and language acquisition to achieve the following targets:

To expose the students to a variety of self-instructional, learner-friendly modes of language learning.

To help the students cultivate the habit of reading passages from the computer monitor, thus providing them with the required facility to face computer-based competitive exams such GRE, TOEFL, GMAT etc.

To enable them to learn better pronunciation through stress on word accent, intonation, and rhythm

To train them to use language effectively to face interviews, group discussions, public speaking.

To initiate them into greater use of the computer is resume preparation, report writing, format- making etc.

However, depending upon the available of infrastructure and budget, the above targets can also be achieved by procuring the minimum required equipment suggested for the establishment of conventional lab the details of which are given below. The lab should cater to the needs of the students to build up their confidence to help them develop leadership qualities through through their communicative competence.

ENGLISH LANGUAGE LABORATORY PRACTICE

Module	TOPICS/SUB-TOPICS	LAB SESSION
1.	INTRODUCTION TOPHONETICS -vowels, -Consonants, -Diphthongs INTRODUCTION TO STRESS & INTONATION -Articulation, -Respiration, -phonation	3
2.	Group Discussions Facing Interviews	4
3.	Situational / Dialogue / Role Play Resume Preparation	2
4.	Public Speaking, Debate	2
5.	GRE, TOEFL, GMAT, MODELS, e-CORRESPONDENCE	3

Introduction to phonetics. 2. Introduction to Vowels and Consonants and associated phonetic symbols. 3. Introduction to Accent, Intonation and Rhythm. 4. Situational Dialogues/Role Play. 5. Debate 6. Public Speaking. 7. Group Discussions 8. Facing Interviews 9. Resume preparation 10. e – correspondence

Suggested Software for Lab classes:

Cambridge Advanced Learner's Dictionary with exercise

The Rosetta Stone English Library

Clarity Pronunciation Power

Mastering English in Vocabulary, Grammar, Spellings, Composition

Dorling Kindersley series of grammar, Punctuation, Composition etc.

Oxford Advanced Learner's Compass, 7th Edition

Language in Use, Foundation Books Pvt Ltd

Learning to Speak English – 4 CDs Microsoft Encarta

Murphy's English Grammar, Cambridge

Time series of IQ Test, Brain-teasers, Aptitude Test etc.

English in Mind, Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge

REFERENCE BOOKS:
1. The Human Touch: personal Skills for Professional Success – by Debra Paul.
2. The Definitive Book of body Language – by Allan Pease, Barbara Pease.
3. How to Face Interviews – by Clive Fletcher.
4. The 7 Habits of Highly Effective People – by StephenCovery.
5. The Google Resume: How to Prepare of a Career and Land a Job at Apple, Microsoft.
6. Good English –by G.H Vallns
7. Better English – G.H Vallins
8. Best English – G.H. Vallins
9. How to Talk to Anyone: 92 little tricks for big success in Relationships by Leli Lowndes.
10. The leader in you - by Dale Carnagie
11. 250 Job Interview Questions You'II most likely Be Asked – by Peter veluki, Peter Verki.
12. Contemporary English Grammer, structures and Composition - by David Green.

C PROGRAMMING LAB

I Semester	L	T	P	C
Course Code: 173MC1L02	0	0	3	2

OBJECTIVES:

1. To learn/strengthen a programming language like C, To learn problem solving techniques
2. To introduce the student to simple linear and non linear data structures such as lists, stacks, queues, etc.,

Recommended Systems/Software Requirements:

1. Intel based desktop PC, ANSI C Compiler with Supporting Editors, IDE's such as Turbo C, Bloodshed C

EXERCISE I:

a. Write a C program to find the sum of individual digits of a positive integer.
b. A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and Subsequent terms are found by adding the preceding two terms in the sequence. Write a C Program to generate the first n terms of the sequence.
c. Write a C program to generate all the prime numbers between 1 and n, where n is a value Supplied by the user.
d. Write a program which checks a given integer is Fibonacci number or not.

EXERCISE II:

a. Write a C program to calculate the following Sum: Sum=1-x2/2! +x4/4!-x6/6!+x8/8!-x10/10!
b. Write a C program to find the roots of a quadratic equation.
c. Write a C program to implement Newton Raphson method for a quadratic equation
d. Write a C program to implement Newton Raphsonmethod for a general purpose algebraic equation

EXERCISE III:

a. Write C programs that use both recursive and non-recursive functions i) To find the factorial of a given integer.
b. To find the GCD (greatest common divisor) of two given integers.
c. To solve Towers of Hanoi problem.
d. Write program to calculate probability of head/tail by generating random numbers using random () function.

EXERCISE IV:

a. The total distance travelled by vehicle in 't' seconds is given by distance = ut+1/2at2 where 'u' and 'a' are the initial velocity (m/ sec.) and acceleration (m/sec2). Write C program to find the distance travelled at regular intervals of time given the values of 'u' and 'a'. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of 'u' and 'a'.

b. Write a C program, which takes two integer operands and one operator form the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement)

EXERCISE V:

a. Write a C program to find both the larges and smallest number in a list of integers. b) Write a C program that uses functions to perform the following:

(i). Addition of Two Matrices

(ii). Multiplication of Two Matrices

(iii). Checking symmetricity of a square matrix.

(iv). Calculating transpose of a matrix in- place manner.

EXERCISE VI:

a. Write a C program that uses functions to perform the following operations: i) To insert a sub-string in to given main string from a given position. ii) To delete n Characters from a given position in a given string.

b. Write a C program to determine if the given string is a palindrome or not

EXERCISE VII:

a. Write a C program that displays the position/ index in the string S where the string T begins, or –1 if S doesn't contain T.

b. Write a C program to count the lines, words and characters in a given text.

EXERCISE VIII:

a. Write a C program to generate Pascal's triangle.

b. Write a C program to construct a pyramid of numbers.

EXERCISE IX:

Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression: 1+x2+x3+………….+xn For example: if n is 3 and x is 5, then the program computes 1+5+25+125. Print x, n, the sum Perform error checking. For example, the formula does not make sense for negative exponents – if n is less than 0. Have your program print an error message if n<0, then go back and read in the next pair of numbers of without computing the sum. Are any values of x also illegal? If so, test for them too.

EXERCISE X:
   a. 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100.Write a C program to find the 2's complement of a binary number.
   b. Write a C program to convert a Roman numeral to its decimal equivalent.

EXERCISE XI:

Write a C program that uses functions to perform the following operations using Structure:
   (i). Reading a complex number ii) Writing a complex number iii)Addition of two complex numbers
   (ii).Multiplication of two complex numbers

EXERCISE XII:
   a. Write a C program which copies one file to another.
   b. Write a C program to reverse the first n characters in a file.

   (Note: The file name and n are specified on the command line.)

****

STATISTICS WITH R PROGRAMMING LAB

I Semester   **L   T   P   C**
Course Code: 173MC1L03   **0   0   3   2**

1. Write a program to illustrate basic Arithmetic in R
2. Write a program to illustrate variable assignment in R
3. Write a program to illustrate data types in R
4. Write a program to illustrate creating and naming a vector in R
5. Write a program to illustrate create a matrix and naming matrix in R
6. Write a program to illustrate Add column and Add a Row in Matrix in R
7. Write a program to illustrate Selection of elements in Matrix in R
8. Write a program to illustrate Performing Arithmetic of Matrices
9. Write a program to illustrate Factors in R
10. Case study of why you need use a Factor in R
11. Write a program to illustrate Ordered Factors in R
12. Write a program to illustrate Data frame selection of elements in a data frame
13. Write a program to illustrate Sorting a data frame
14. Write a program to illustrate List? Why would you need a List?
15. Write a program to illustrate Adding more elements into a List
16. Write a program to illustrate if-else-else if in R
17. Write a program to illustrate While and For loops in R
18. Write a program to illustrate Compare and Matrices and Compare mVectors
19. Write a program to illustrate Logical & and Logical | operators in R
20. Write a program to illustrate Functions in Quick sort implementation in R
21. Write a program to illustrate Function inside function in R
22. Write a program to illustrate to create graphs and usage of plot() function in R
23. Write a program to illustrate Customising and Saving to Graphs in R
24. Write a program to illustrate some built in Mathematical Functions

****

OOPS THROUGH JAVA

II Semester	L	T	P	C
Course Code: 173MC2T06	3	1	0	3

UNIT–I:
Basics of Object Oriented Programming(OOP): Need for OO paradigm , A way of viewing world- Agents, responsibility, messages, methods, classes and instances, class hierarchies(Inheritance), method binding, overriding and exceptions, summary of oop concepts, coping with complexity, abstraction mechanisms
Java Basics: Data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and costing, simple java program, classes and objects- concepts of classes, objects, constructors methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling.

UNIT–II:
Inheritance: Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, sombination, benefits of inheritance costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism, abstract classes.
Packages and Interfaces: Defining, Creating and Accessing a package, Understanding CLASSPATH, Importing packages, differences between classes and interfaces, defining an interface, Implementing interface, applying interfaces variables in interface and extending interfaces.

UNIT–III:
Exception handling and Multithreading: Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, usage of try, catch, throws and finally, built in exceptions, creating own exception sub classes. Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

UNIT–IV:
Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy , user-interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, list panes- scroll pane, dialogs, menu bar, graphics, layout manager- layout manager types- boarder, grid, flow, card and grid bag.

UNIT-V:
Applets: Concepts of Applets, differences between applets and applications, lifecycle of an applet, types of applets, creating applets, passing parameters to applets.
Swings: Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons-The JButton class, Check boxes, Radio Buttons, Combo boxes, Tabbed panes, Scroll panes, Trees and Tables.

TEXT BOOKS:

1. Java-The complete reference,7/e, Herbert schildt, TMH.

2. JAVA: How to program, 8/e, Dietal , Dietal,PHI.

3. Introduction of programming with JAVA,S.Dean,TMH.

4. Introduction to Java programming, 6/e, Y.Daniel Liang, Pearson.

REFERENCE BOOKS:

1. Core Java 2, Vol 1(Vol 2) Fundamentals(Advanced), 7/e, Cay.S.Horstmann,Gary Cornell, Pearson.

2. Big Java2,3/e, Cay.S. Horstmann,Wiley.

3. Object Oriented Programming through Java, P.Radha Krishna, University Press.

4. JAVA& Object Orientation an Introduction, 2/e, John Hunt, Springer.

5. Introduction to JAVA Programming, 7/e, Y. Daniel Liang, Pearson. , TMH.

****

OPERATING SYSTEMS

II Semester   **L   T   P   C**
Course Code: 173MC2T07   **3   1   0   3**

UNIT-I
Introduction: Computer –system organization, Computer- system Architecture, Operating- system Structure, Operating-system Operations, Process Management, Memory Management, Storage Management, Protection and Security, Distributed Systems, Special-purpose systems, Computing Environments , Operating-system Structure:, Operating-system Services, User , Operating-system Interface, System calls, System programs, Operating-system Design and Implementation, , Operating-system structure, Virtual Machine.

UNIT-II
Process Management:
Processes: Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication, Examples of IPC Systems, Communication in Client-Server systems
Threads: Overview, Multithreading Models, Thread Libraries, Java Threads, Threading Issues, OS Examples
CPU Scheduling: Basic concepts, Scheduling Criteria, Scheduling Algorithms, Multiple- Processor Scheduling, Thread Scheduling, Operating system Examples Process Synchronization: Background, The Critical- section problem, Petersons solution, Synchronization Hardware, Semaphores, Classic problems of Synchronization, Monitors, Atomic Transactions.

UNIT–III:
Memory management: Main memory: Swapping, Contiguous memory Allocation, Paging, Structure of the Page table, Segmentation Virtual memory: Background, Demand paging, copy- on-Write, Page Replacement, Allocation of frames, Thrashing, Memory-Mapped Files.

UNIT–IV:
File-system Interface: Concept, Access Methods, Directory structure, File system Mounting, File sharing, Protection.
File-system Implementation: File-system Structure, Implementation, Directory Implementation, Allocation Methods, Free- Space Management, Efficiency and Performance, Recovery, Log-Structured File systems, NFS Mass –storage Structure: Overview, Disk Structure, Disk Attachment, Disk Scheduling, Disk and swap-space Management, RAID Structure, Stable- Storage Implementation, Tertiary-Storage Structure I/O systems: Overview, I/O Hardware, Application I/O Interface, Kernel I/O Subsystem, Transforming I/O requests to Hardware Operations, STREAMS, Performance.

UNIT-V

Deadlocks: System model, Deadlock Characterization, Methods for handling Deadlocks, Deadlock Prevention, Deadlock avoidance Deadlock Detection and Recovery form Deadlock. 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, Language-Based Protection
Security: The Security Problem, Program Threads, System and Network

Threats, Cryptography as a security tool, User Authentication, Implementing security Defenses, Firewalling to protect systems and Networks.

TEXT BOOKS:

1. Operating system concepts, 7/e, Abraham Siliberschatz, Galvin, John Wiley &sons , Inc.

REFERENCES BOOKS:

1. Operating systems, 6/E, William stallings, PHI/Pearson.
2. Operating systems 3/e, Dietal, Dietal, Pearson.
3. Operating systems, 2/e, Dhamdhere, TMH.
4. An introduction to Operating systems, Concepts and practice, Pramod Chandra P. Bhat, PHI
5. Operating systems, Elmasri, Carrick, Levine, TMH.
6. Operating systems, 3/e ,Nutt, Chaki, Neogy Pearson.
7. Operating systems, Brian L. Stuart, Cengage.
8. Operating systems, Haldar, Aravind, Pearson.
9. Operating systems, PAL Choudhury, PHI.
10. Operating systems: design and Implementation, 3/e, Tanenbaum, Woodhull.

****

SOFTWARE ENGINEERING

II Semester   **L   T   P   C**
Course Code: 173MC2T08   **3   1   0   3**

UNIT- I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths. (Text Book 3)
The software problem: Cost, schedule and quality, Scale and change.

UNIT–II:

Software Process: Process and project, component software process, Software development process models : Waterfall model, prototyping, iterative development, relational unified process, time boxing model, Extreme programming and agile process, using process models in a project. Project management process.

UNIT–III

Software requirement analysis and specification: Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, other approaches for analysis, validation.
Planning a software project: Effort estimation, project schedule and staffing, quality planning, risk management planning, project monitoring plan, detailed scheduling.

UNIT–IV

Software Architecture: Role of software architecture, architecture views, components and connector view, architecture styles for C & C view, documenting architecture design, evaluating architectures.
Design: Design concepts, function-oriented design, object oriented design, detailed design, verification, metrics.

UNIT–V

Coding and Unit testing: Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing, code inspection, metrics.
Testing: Testing concepts, testing process, black-box testing, white-box testing, metrics.

TEXT BOOKS:

1. A Concise introduction to software engineering (undergraduate topics in computer science), Pankaj Jalote, Springer International Edition.
2. Software Engineering, APrecise approach, Pankaj Jalote, Wiley
3. Software Engineering, 3/e, & 7e Roger S.Pressman, TMH

REFERENCE BOOKS:

1. Software Engineering, 8/e, Sommerville, Pearson.
2. Software Engineering principles and practice, W S Jawadekar, TMH
3. Software Engineering concepts, R Fairley, TMH

****

OPTIMIZATION TECHNIQUES

II Semester   **L   T   P   C**
Course Code: 173MC2T09   **3   1   0   3**

UNIT–I:
Development: Definition, Characteristics and Phrases, scientific method. Types of models, general methods for solving, operations research modes.
Allocation: introduction linear programming formulation, graphical solution, simplex methods, artificial variable technique, duality principle.

UNIT-II:
Transportation problem: Formulation, optimal solution, unbalanced transportation, assignment problem: formulation, optimal solution, variations problem, degeneracy i.e. non square MXN) matrix, restrictions sequencing: Introduction, optimal solution for processing each of n jobs through three machines, travelling salesman problem(i.e.) shortest acyclic route models.

UNIT-III:
Replacement: Introduction, replacement of items that deteriorate when money value is not counted and counted, and replacement of items that fail completely (i.e.) group replacements.
Waiting lines: Introduction , single channel, poisson arrivals, exponential service time infinite population and unrestricted queue.

UNIT-VI:
Inventory: Introduction, single item, deterministic models, production is instantaneous or at a constant rate , shortages are allowed or not allowed and with drawls from stock is continuous, purchase inventory model with one price break ,shortages are not allowed , instantaneous production demand production or purchase cost is relevant, stochastic models, simple problems.

UNIT–V:
Theory of Games: Introduction, minmax (maximum), criterion and optimal strategy solution of games with saddle points, rectangular without saddle points. Dynamic programming: Introduction, Bellman's Principle of optimality, solutions for simple problems.
Project Management: PERT and CPM , difference between PERT and CPM, PERT/CPM network components and precedence relations, Time Estimates for activities.

TEXT BOOKS:
1. Operations Research, S.D.Sharma, Ramnath, & Kedarnath co, Meerut.
2. Operations Research, An introduction , 8/e, Taha, Pearson.

REFERENCE BOOKS:
1. Operations Research, P.K.Gupta, D.S. Hira, S.Chand.
2. Operations Research, R.D.Asrhedkar, R.V.Kulkarni.
3. Operations Research, Problems & sollutons, 3/e, JKSharma, Macmillan.
4. Operations Research, 8/e, Hillier, Liberman, TMH.
5. Operations Research, 2/e, Panneerselvam.

COMPUTER GRAPHICS

II Semester	**L**	**T**	**P**	**C**
Course Code: 173MC2T10	**3**	**1**	**0**	**3**

UNIT-I:
Introduction: Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices.
Output primitives : Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms.

UNIT-II:
2-D geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems. (p.nos 204-227 of text book-1).
2-D viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen- Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm

UNIT-III:
3-D object representation: Polygon surfaces, quadric surfaces, spline representation, Hermit curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon rendering methods.
3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations.

UNIT-IV:
3-D viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping
Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods

UNIT-V:
Computer animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications.

TEXT BOOKS:
1. Computer Graphics *C version*, Donald Hearn, M.Pauline Baker, Pearson
2. Computer Graphics Principles & practice, 2/e, Foley, VanDam, Feiner, Hughes, Pearson

REFERENCE BOOKS:
1. Computer Graphics, Donald Hearn and M.Pauline Baker, 2/E,PHI
2. Computer Graphics, Zhigand xiang, Roy Plastock, Schaum's outlines, 2/E, TMH
3. Procedural elements for Computer Graphics, David F Rogers, 2/e, TMH
4. Principles of Interactive Computer Graphics, Neuman , Sproul, TMH.
5. Principles of ComputerGraphics, Shalini Govil, Pai, 2005, Springer.

***

OOPS THROUGH JAVA LAB

II Semester   **L   T   P   C**
Course Code: 173MC2L04   **0   0   3   2**

- Use JDK 1.5 or above on any platform e.g. Windows or Unix.
- Student is expected to complete any 16 programs.

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1.every subsequent value is the sum of the 2 values preceding it. Write A Java Program (WAJP) that uses both recursive and non recursive functions to print the nth value of the Fibonacci sequence.
2. WAJP to demonstrate wrapper classes and to fix the precision.

3. WAJP that prompts the user for an integer and then prints out all the prime numbers upto that Integer.
4. WAJP that checks whether a given string is a palindrome or not. Ex. MALAYALAM

   is a palindrome.

5. WAJP for sorting a given list of names in ascending order.

6. WAJP to check the compatibility for multiplication, if compatible multiply two matrices and find its transpose.
7. WAJP that illustrates how runtime polymorphism is achieved.

8. WAJP to create and demonstrate packages.

9. WAJP, using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.
10. WAJP that reads on file name form the user then displays information about whether the file exists, whether the file is readable/writable, the type of file and the length of the file in bytes and display the content of the using FileInputStream class.
11. WAJP that displays the number of characters, lines and words in a text/text file.

12. Write an Applet that displays the content of a file.

13. WAJP that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +-*?% operations. Add a text field to display the result.
14. WAJP for handling mouse events.

15. WAJP demonstrating the life cycle of a thread.

16. WAJP that correctly implements Producer-Consumer problem using the concept of

   Inter Thread Communication.

17. WAJP that lets users create Pie charts. Design your own user interface(with Swings & AWT).

18. WAJP that allows user to draw lines, rectangles and ovals.

19. WAJP that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex: The data

    send form the client is the radius of a circle and the result produced by the server is the area of the circle.

20. WAJP to generate a set of random numbers between two numbers x1 and x2, and x1>0.

21. WAJP to create an abstract class named shape, that contains an empty method named number Of Sides(). Provide three classes named Trapezoid, Triangle and Hexagon, such that each one of the classes contains only the method number Of Sides(), that contains the number of sides in the given geometrical figure.

22. WAJP to implement a Queue, using user defined Exception Handling (also make use of throw, throws).

23. WAJP that creates 3 threads by extending Thread class. First thread displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2 seconds and the third displays"Welcome" every 3 seconds. (Repeat the same by implementing Runnable).

24. Create an inheritance hierarchy of Rodent, Mouse, Gerbil, Hamsteretc. In the base class provide methods that are common to all Rodents and override these in the derived classes to perform different behaviours, depending on the specific type of Rodent. Create an array of Rodent, fill it with different specific types of Rodents and call your base class methods.

DATA STRUCTURES LAB

II Semester   **L   T   P   C**
Course Code: 173MC2L05   **0   0   3   2**

Exercise 1:
Write recursive program which computes the nth Fibonacci number, for appropriate values of n. Analyze behavior of the program Obtain the frequency count of the statement for various values of n.

Exercise 2:
Write recursive program for the following
a) Write recursive and non recursive C program for calculation of Factorial of an integer
b) Write recursive and non recursive C program for calculation of GCD (n, m)
c) Write recursive and non recursive C program for Towers of Hanoi: N disks are to be transferred from peg S to peg D with Peg I as the intermediate peg.

Exercise 3:
a) Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.
b) Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.
c) Write C program that use both recursive and non recursive functions to perform Fibonacci search for a Key value in a given list.

Exercise 4:
a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order
b) Write C program that implement Quick sort, to sort a given list of integers in ascending order
c) Write C program that implement Insertion sort, to sort a given list of integers in ascending order

Exercise 5:
a) Write C program that implement heap sort, to sort a given list of integers in ascending order
b) Write C program that implement radix sort, to sort a given list of integers in ascending order
c) Write C program that implement merge sort, to sort a given list of integers in ascending order

Exercise 6:
a) Write C program that implement stack (its operations) using arrays
b) Write C program that implement stack (its operations) using Linked list

Exercise 7:
a) Write a C program that uses Stack operations to Convert infix expression into postfix expression
a) Write C program that implement Queue (its operations) using arrays.
b) Write C program that implement Queue (its operations) using linked lists

Exercise 8:
a) Write a C program that uses functions to create a singly linked list
b) Write a C program that uses functions to perform insertion operation on a singly linked list
c) Write a C program that uses functions to perform deletion operation on a singly linked list

Exercise 9:
a) Adding two large integers which are represented in linked list fashion.
b) Write a C program to reverse elements of a single linked list.
c) Write a C program to store a polynomial expression in memory using linked list
d) Write a C program to representation the given Sparse matrix using arrays.
e) Write a C program to representation the given Sparse matrix using   linked list

Exercise10:
a) Write a C program to Create a Binary Tree of integers
b) Write a recursive C program for traversing a binary tree in preorder, inorder and postorder.
c) Write a non recursive C program for traversing a binary tree in preorder, inorder and postorder.
d) Program to check balance property of a tree.

Exercise 11:
a) Write a C program to Create a BST
b) Write a C program to insert a node into a BST.
c) Write a C program to delete a node from a BST.

****

OPERATING SYSTEM & COMPUTER GRAPHICS LAB

II Semester   **L   T   P   C**
Course Code: 173MC2L06   **0   0   3   2**

1. Implementation of DDA Line Algorithm

2. Implementation of Bresenham's Line Algorithm

3. Implementation of Midpoint Circle Algorithm

4. Implementation of Midpoint Ellipse Algorithm

5. Implementation of Two Dimensional Transformations

6. Implementation of Two Dimensional Composite Transformations

7. Simulate the Following cpu Scheduling Algorithms
   A) Round Robin B) Sjf C) Fcfs D) Priority

8. Multiprogramming-Memory Management- Implementation Of Fork (), Wait (), Exec () And Exit ()

9. Simulate The Following

       a. Multiprogramming with A Fixed Number Of Tasks (Mft)
       b. Multiprogramming with A Variable Number Of Tasks (Mvt)

12. Simulate Bankers Algorithm for Dead Lock Avoidance

13. Simulate Bankers Algorithm for Dead Lock Prevention.

14. Simulate The Following Page Replacement Algorithms.

   A) Fifo B) Lru C) Lfu

15. Simulate the Following File Allocation Strategies

       A) Sequenced   B) Indexed   C) Linked

***

DATA BASE MANAGEMENT SYSTEMS

		L	**T**	**P**	**C**
III Semester					
Course Code: 173MC3T11		**3**	**1**	**0**	**3**

UNIT-I:
Database System Applications, Purpose of Database Systems, View of Data – Data Abstraction, Instances and Schemas, Data Models – the ER Model, Relational Model, Other Models – Database Languages –DDL, DML, Database Access from Applications Programs, Transaction Management, Data Storage and Querying, Database Architecture, Database Users and Administrators, History of Data base Systems. Introduction to Data base design, ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets ,Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises. Relational Model: Introduction to the Relational Model – Integrity Constraints Over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views – Destroying /altering Tables and Views.

UNIT- II:
Relational Algebra and Calculus: Relational Algebra – Selection and Projection, Set operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus – Tuple relational Calculus – Domain relational calculus – Expressive Power of Algebra and calculus. Form of Basic SQL Query – Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set – Comparison Operators, Aggregate Operators, NULL values – Comparison using Null values – Logical connectives – AND, OR and NOT – Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers and Active Data bases.

UNIT- III:
Introduction to Schema Refinement – Problems Caused by redundancy, Decompositions – Problem related to decomposition, Functional Dependencies - Reasoning about FDS, Normal Forms – FIRST, SECOND, THIRD Normal forms – BCNF –Properties of Decompositions- Loss less join Decomposition, Dependency preserving Decomposition, Schema Refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form, Join Dependencies, FIFTH Normal form, Inclusion Dependencies.

UNIT- IV:
Overview of Transaction Management: The ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions – Lock Based Concurrency Control, Deadlocks – Performance of Locking – Transaction Support in SQL. Concurrency Control: Serializability, and recoverability – Introduction to Lock Management – Lock Conversions, Dealing with Dead Locks, Specialized Locking Techniques – Concurrency Control without Locking. Crash recovery: Introduction to Crash recovery, Introduction to ARIES, the Log, Other Recovery related Structures, the Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media recovery

UNIT-V
Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index data Structures – Hash Based Indexing, Tree based Indexing, Comparison of File Organizations. Storing data: Disks and Files: -The Memory Hierarchy – Redundant Arrays of Independent disks. Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM) B+ Trees: A Dynamic Index Structure, Search, Insert, Delete. Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendible vs. Linear Hashing.

TEXT BOOKS:
1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, TMH, 3rd Edition,2003.
2. Data base System Concepts, A.Silberschatz, H.F. Korth, S.Sudarshan,McGraw hill, VI edition,2006.
3. Fundamentals of Database Systems 5th edition., Ramez Elmasri, Shamkant. Navathe, Pearson Education,2008.

REFERENCE BOOKS:
1. Database Management System Oracle SQL and PL/SQL,P.K.Das Gupta, PHI.
2. Database System Concepts, Peter Rob & Carlos Coronel, Cengage Learning,2008.
3. Database Systems, A Practical approach to Design Implementation and Management Fourth edition, Thomas Connolly, Carolyn Begg, Pearson education.
4. Database Principles, Programming, and Performance, P.O'Neil, E.O'Neil, 2nd ed., ELSEVIER
5. Fundamentals of Relational Database Management systems, S.Sumathi, S.Esakkirajan, Springer.
6. Introduction to Database Management, M.L.Gillenson and others, Wiley Student Edition.
7. Database Development and Management, Lee Chao, Auerbach publications, Taylor & Francis Group.
8. Introduction to Database Systems, C.J.Date, Pearson Education.

COMPUTER NETWORKS

		L	T	P	C
III Semester		**3**	**1**	**0**	**3**
Course Code: 173MC3T12					

UNIT-I:
Network Hardware reference model: Transmission media, Narrowband ISDN, Broad band ISDN, ATM.
The data Link layer: Design Issues, Error detection and correction,
Elementary Data Link Protocols, Sliding window protocols : Data link layer in HDLC, Internet and ATM.

UNIT-II:
Channel allocation methods: TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols – IEEE standard BO2 for LANS – Ethernet, Token Bus, Token ring, Bridges.
Network layer Routing Algorithms: Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, Broadcast routing, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Choke packets and Load shedding.

UNIT-III:
Internet Working: Tunneling, internetworking, Fragmentation, network layer in the internet – IP protocols, IP address, Subnets, Internet control protocols, DSPF, BOP, Internet multicasting, Mobile IP. Network layer in the ATM Networks – cell formats, connection setup, routing and switching, service categories, and quality of service, ATM LANs.

UNIT-IV:
The Transport Layer: Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, end to end protocols: UDP, reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call – BLAST, CHAN, SELECT, DCE.

UNIT-V:
Application Layer: Network Security, Cryptographic Algorithms: DES, RSA. Security Mechanisms: Authentication Protocols, Firewalls. Name service (DNS) Domains Hierarchy, Name servers. Traditional Applications: SMTP, MIME, World Wide Web: HTTP, Network Management: SNMP.

TEXT BOOKS:
1. Computer Networks and rew, Tanenbaum, 4/e, Pearson
2. Data and computer communications, stallings, 8/e, PHI

REFERENCE BOOKS
1. Data communications and networking Forouzan, 4/e, TMH
2. Computer Networks – A System Approach, Peterson, Bruce Davie,2/e ,Harcourt Asia
3. Compute communications and networking technologies, Gallo, Hancock, Cengage
4. An Engineering approach to compute networking, Kesha, Pearson
5. Communication networks, 2/e, Leon-Garcia, TMH
6. Computer networks, Anuranjan Misra, ACME Learning

UNIX PROGRAMMING

III Semester   **L   T   P   C**
Course Code: 173MC3T13   **3   1   0   3**

UNIT-I
Review of Unix Utilities and Shell Programming: -File handling utilities, security by file permissions, process utilities, disk utilities, networking commands, backup utilities, text processing utilities, Working with the Bourne shell-, What is a shell, shell responsibilities, pipes and input redirection, output redirection, here documents, the shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shell script examples.

UNIT-II
Unix Files: Unix file structure, directories, files and devices, System calls, library functions, low level file access, usage of open, create, read, write, close, lseek, stat, fstat, octl, umask, dup, dup2. The standard I/O (fopen, fclose, fflush, fseek, fgetc, getc, getchar, fputc, putc, putchar, fgets, gets), formatted I/O, stream errors, streams and file descriptors, file and directory maintenance (chmod, chown, unlink, link, symlink, mkdir, rmdir, chdir, getcwd), Directory handling system calls (opendir, readdir, closedir, rewinddir, seekdir, telldir)

UNIT-III
Unix Process: Threads and Signals: What is process, process structure, starting new process, waiting for a process, zombie process, process control, process identifiers, system call interface for process management, -fork, vfork, exit, wait, waitpid, exec, system, Threads, -Thread creation, waiting for a thread to terminate, thread synchronization, condition variables, cancelling a thread, threads vs. processes, Signals-, Signal functions, unreliable signals, interrupted system calls, kill and raise functions, alarm, pause functions, abort, sleep functions.

UNIT-IV
Data Management: Management Memory (simple memory allocation, freeing memory) file and record locking (creating lock files, locking regions, use of read/ write locking, competing locks, other commands, deadlocks). Interprocess Communication: Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes, FIFOs, streams and messages, namespaces, introduction to three types of IPC (system-V)- message queues, semaphores and shared memory
Message Queues-: IPC, permission issues, Access permission modes, message structure, working message queues, Unix system-V messages, Unix kernel support for messages, Unix APIs for messages, client/server example.

UNIT-V
Semaphores: -Unix system-V semaphores, Unix kernel support for semaphores, Unix APIs for semaphores, file locking with semaphores. Shared Memory: -Unix system-V shared memory, working with a shared memory segment, Unix kernel support for shared memory, Unix APIs for shared memory, semaphore and shared memory example.
Sockets: Berkeley sockets, socket system calls for connection oriented protocol and connectionless protocol, example- client/server program,
advanced socket system calls, socket options.

TEXT BOOKS:
1 Unix and shell Programming, N B Venkateswarlu, Reem
2. Unix Concepts and Applications, 3/e, Sumitabha Das, TMH

REFERENCE BOOKS:
1. Unix and shell Programming, Sumitabha Das, TMH
2. A Beginner's Guide to Unix, N.P.Gopalan, B.Sivaselva, PHI
3. Unix Shell Programming, Stephen G.Kochan, Patrick Wood, 3/e, Pearson
4. Unix Programming, Kumar Saurabh, Wiley, India
5. Unix Shell Programming, Lowell Jay Arthus & Ted Burns,3/e, Gal Gotia
6. Nix Concepts and Applications, Das, 4/e, TMH

MANAGEMENT INFORMATION SYSTEM

III Semester   **L   T   P   C**
Course Code: 173MC3T14   **3   1   0   3**

UNIT-I:
Management Information Systems: A Framework: Importance of MIS, MIS: A Definition nature and Scope of MIS, **Structure and Classification of MIS:** Structure of MIS, MIS Classification

Information and System Concepts: Information: A Definition, Types of Information, Dimensions of Information, System: A Definition, Kinds of Systems, System Related Concepts, Elements of a System, Human as an Information Processing System

Information Systems for Competitive Advantage: Introduction, Changing concepts of Information System, Competitive Advantage, Information systems Strategies for Dealing with competitive Force, Porter's Value Chain Model, Strategic Information Systems (SIS)

UNIT–II:
BUSINESSAPPLICATIONSOFIS
e – Commerce: Introduction, e – Commerce **ERP Systems:** Introduction, Enterprise Information Systems

Decision – Support Systems: Decision – Making: A Concept, Simon's Model of Decision – Making Types of Decisions, Methods for Choosing Among Alternatives, Decision – Making and MIS, Decision Support Systems – Why?, Decision Support Systems: A framework, Characteristics and Capabilities of DSS

Business Intelligence and knowledge Management System: Business Intelligence, Knowledge Management System

UNIT–III:
Information System Planning: Information System Planning: WHY? Planning Terminology Information System Planning, The Nolan Stage Model, The Four – Stage Model of is planning Selecting A Methodology, Information Resources Management (IRM), Organization Structure and Location of MIS

System Acquisition: Acquisition of Information Systems, Acquisition of Hardware and Software

UNIT– IV:
System Implementation: IMPLEMENTATION PROCESS, Organisational Change
Evaluation &Maintenance of IS: Evaluation of MIS, System Maintenance
IS Security and Control: IS Security Threats, Protecting Information System, IS Security Technology The Disaster Recovery Plan

UNIT–V:
BUILDINGOFIS
System Development Approaches: System Development Stages, System Development Approaches
System Analysis and Design: SYSTEM ANALYSIS - Introduction, Requirement Determination, Strategies for Requirement Determination, Structured Analysis Tools
SYSTEMS DESIGN: Design Objectives, Conceptual Design, Design Methods, Detailed System Design

TEXTBOOKS:
4. Management Information System, Managerial Perspecives, D P Goyal, 3 ed, McMillan Publications

REFERENCE BOOKS:
1. Laudon K.C. Ladon J.P. Bradston, "Management Information Systems – Managing the digital firm", Person Education, 2004.
2. Jeffrey A.Hoffer, Joey F.George, Joseph S.Valachich, Modern Systems Analysis and Design" Third Edition, Prentice Hall, 2002.

DESIGN & ANALYSIS OF ALGORITHMS

III Semester   **L   T   P   C**
Course Code: 173MC3T15   **3   1   0   3**

UNIT-I:
Introduction: Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space Complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and little oh notation, probabilistic analysis, Amortized analysis. Disjoint Sets- disjoint set operations, union and find algorithms, spanning trees, connected components and bi-connected components.

UNIT-II:
Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Stassen's matrix multiplication. Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

UNIT-III:
Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal Binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

UNIT-IV:
Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph Coloring Hamiltonian cycles.

UNIT-V:
Branch and Bound: General method, applications - Travelling sales person problem,0/1 knapsack Problem- LC Branch and Bound solution, FIFO Branch and Bound solution. NP-Hard and NP-Complete problems: Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem.

TEXTBOOKS:
1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. The Algorithm Design Manual, 2nd edition, Steven S. Skiena, Springer.
3. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, PHI Pvt. Ltd.

REFERENCEBOOKS:
1. Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA
2. Design and Analysis of Algorithms, Parag Himanshu Dave, Himansu BAlachandra Dave, Pearson Education.
3. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T. Lee, S.S.Tseng, R.C.Chang and T.Tsai, Mc GrawHill.
4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.

DATA BASE MANAGEMENT SYSTEMS LAB

		L	**T**	**P**	**C**
III Semester		0	0	3	2
Course Code: 173MC3L07					

1. Execute a single line and group functions for a table.

2. Execute DCL and TCL Commands.

3. Create and manipulate various DB objects for a table.

4. Create views, partitions and locks for a particular DB.

5. Write PL/SQL procedure for an application using exception handling.

6. Write PL/SQL procedure for an application using cursors.

7. Write a DBMS program to prepare reports for an application using functions.

8. Write a PL/SQL block for transaction operations of a typical application using triggers.

9. Write a PL/SQL block for transaction operations of a typical application using package.

10. Design and develop an application using any front end and back end tool (make use of ER diagram and DFD).

11. Create table for various relation

12. Implement the query in sql for a) insertion b) retrieval c) updation d) deletion

13. Creating Views

14. Writing Assertion

15. Writing Triggers

16. Implementing operation on relation using PL/SQL

17. Creating Forms

18. Generating Reports

Typical Applications – Banking, Electricity Billing, Library Operation, Pay roll, Insurance, Inventory etc.

UNIX PROGRAMMING LAB

III Semester   **L   T   P   C**
Course Code: 173MC3L08   **0   0   3   2**

1. Program using basic network commands
2. Program using system calls: create, open, read, write, close, stat, fstat, lseek .
   Program to implement inter process communication using pipes
4. Program to perform inter process cots : sniffer
5. Program using TCP sockets (Client and Server)
6. Program using UDP sockets (Client and Server)
7. Program using URL class to download web pages
8. Write a shell script for sorting, searching and insertion/deletion of elements in a list
9. Create two processes to run a for loop, which adds numbers 1 to n, say one
   process adds odd numbers and the other even
10. By creating required number of processors, simulate a communication
    between them as below:
11. Create a file that is shared among some users, write a program that finds whether
    a specific user has created read and write operations on the file
12. Create a shared lock and exclusive lock among some number of processes; say 1 to
    10 on any data of 100 elements. For example, process 5 wants a shared lock on
    elements 5 to 50 or process 8 wants exclusive lock on elements 32 to 45. Create
    access violations on the locks and show what occurs, then.
13. Write a program demonstrating semaphore operation on a shared file for
    reading but not writing
14. Create a distributed key among some processes which exchange messages of the
    form (m, Ti, I) for resource sharing, where m=request, reply, release, Ti=time
    stamp and I=process id
15. Write a program demonstrating mutual exclusion principle
16. Write a program which reads a source file name and destination file name using
    command line arguments and then converts into specified format (i.e. either
    from lower case to upper case or upper case to lower case or inverse of each)
17. Write a program which takes a set of filenames along with the command line and
    print them based on their size in bytes either ascending or descending order
18. Write a program which takes directory name along the command line and
    displays names of the files which are having more than one link
19. Write a program to demonstrate the use of temporary files
20. Write a program to demonstrate the use of exec family functions
21. Write a program to display the good morning, good afternoon, good evening and good
    night depending on the users log on time 22. Write a program to demonstrate the
    working of simple signal handler that catches either of the two user defined signals and
    prints the signal number
23. Write a program to demonstrate the locking mechanism while accessing the shared files
24. Write a shell script containing a function mycd() using which, it is possible to
    shuttle between directories
25. Write a shell script which works similar to the wc command. This script can receive the
    ption -l, -w, -c to indicate whether number of lines/ words/characters
26. Write a program to print prime numbers between x and y Write a shell script which
    deletes all lines containing the word
27. Write a shell script which deletes all lines containing the word "UNIX" in the files
    supplied as arguments to this shell script

28. Write a shell script which displays a list of all files in the current directory to which you have read, write and execute permissions
29. Write a menu-driven program which has the following options:
30. Write a shell script for renaming each file in the directory such that it will have the current shell's PID as an extension. The shell script should ensure that the directories do not get renamed
31. Write a program which demonstrates the shared memory functions

COMPUTER NETWORKS LAB

III Semester	**L**	**T**	**P**	**C**
Course Code: 173MC3L09	**0**	**0**	**3**	**2**

PART – A

1. Implement the data link layer farming methods such as character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
3. Implement Dijkstra's algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm
5. Take an example subnet of hosts. Obtain broadcast tree for it.

PART – B

1. Implement the following forms of IPC.
   a) Pipes   b) FIFO
2. Implement file transfer using Message Queue form of IPC
3. Write a program to create an integer variable using shared memory concept and increment the variable
4. Simultaneously by two processes. Use semaphores to avoid race conditions
5. Design TCP iterative Client and server application to reverse the given input sentence
6. Design TCP iterative Client and server application to reverse the given input sentence
7. Design TCP client and server application to transfer file
8. Design a TCP concurrent server to convert a given text into upper case using multiplexing system call "select"
9. Design a TCP concurrent server to echo given set of sentences using poll functions
10. Design UDP Client and server application to reverse the given input sentence
11. Design UDP Client server to transfer a file
12. Design using poll client server application to multiplex TCP and UDP requests for converting a given text into upper case.
13. Design a RPC application to add and subtract a given pair of integers

OBJECT ORIENTED ANALYSIS & DESIGN

IV Semester   **L   T   P   C**
Course Code: 173MC4T16   **3   1   0   3**

UNIT- I:
Introduction to UML: The meaning of Object-Orientation, object identity, encapsulation, information hiding, polymorphism, genericity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.

UNIT-II:
Basic structural Modeling: Classes, relationships, common mechanisms, diagrams, Advanced structural modeling: advanced relationships, interfaces, types & roles, packages, instances.
Class & object diagrams: Terms, concepts, examples, modeling techniques, class & Object diagrams.

UNIT-III:
Collaboration diagrams: Terms, Concepts, depicting a message, polymorphism in collaboration diagrams, iterated messages, use of self in messages.
Sequence diagrams: Terms, concepts, differences between collaboration and sequence diagrams, depicting synchronous messages with/without priority call back mechanism broadcast message.

UNIT- IV:
Behavioral Modeling: Interactions, use cases, use case diagrams, activity diagrams.
Advanced Behavioral Modeling: Events and signals, state machines, processes & threads, time and space, state chart diagrams.

UNIT–V:
Architectural Modeling: Terms, concepts, examples, modeling techniques for component diagrams and deployment diagrams.

TEXT BOOKS:
1. The Unified Modeling Language User Guide, Grady Booch, Rambaugh, Ivar Jacobson, PEA
2. Fundamentals of Object Oriented Design in UML, Meilir Page- Jones, Addison Wesley

REFERENCE BOOKS:
1. Head First Object Oriented Analysis &Design, Mclaughlin, SPD OReilly,2006
2. Object oriented Analysis& Design Using UML, Mahesh ,PHI
3. The Unified Modeling Language Reference Manual, 2/e, Rambaugh, GradyBooch,etc., PEA
4. Object Oriented Analysis & Design, Satzinger, Jackson, Thomson
5 Object Oriented Analysis Design & implementation, Dathan.,Ramnath, University Press
6. Object Oriented Analysis & Design, John Deacon, PEA
7. Fundamentals of Object Oriented Analysis and Design in UML, M Pages-Jones, PEA
8. Object-Oriented Design with UML, Barclay, Savage, Elsevier, 2008

ADVANCED JAVA & WEB TECHNOLOGIES

IV Semester   **L   T   P   C**
Course Code: 173MC4T17   **3   1   0   3**

UNIT-I:
Review of HTML4: Common tags, HTML Tables and formatting internal linking, Complex HTML forms. Introduction to Scripting Languages: Java Scripts, Control structures, functions, arrays & objects, DHTML, CSS, event model, filters & transitions.

UNIT-II:
Review of Applets, Class, Event Handling, AWT Programming:
Introduction to Swing: Japplet, Handling Swing Controls like Icons, Buttons, Text Boxes, Combo Boxes, Tabbed Pains, Scroll Pains, Trees, Tables, Differences between AWT Controls & Swing Controls, Developing a Home page using Applets & Swing.

UNIT-III:
Java Beans: Introduction to Java Beans, Advantages of Java Beans, BDK, Introspection, Using Bound properties, Bean Info Interface, Constrained properties, Persistence, Customizers, Java Beans API.
Introduction to Serve lets: Lifecycle of a Servlet, JSDK, The Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization Parameters, The javax. Servlet. HTTP package, Handling, Http Request & responses, Using Cookies, Session Tracking, Security Issues.

UNIT-IV:
Introduction to JSP: The Problem with Serve lets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC.
Setting Up the JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat.
JSP Application Development: Generating Dynamic Content, Using Scripting Elements, Implicit JSP Objects, Conditional Processing – Displaying Values, Using an Expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Passing Control and Data Between Pages – Sharing Session and Application Data Memory Usage Considerations.

UNIT-V:
Database Access: Database Programming using JDBC, Studying Javax.sql.* package. Accessing a Database from a JSP Page, Application – Specific Database Actions Deploying JAVA Beans in a JSP Page.

TEXT BOOKS:
1. Internet and World Wide Web: How to program,6/e, Dietel, Dietel, Pearson.
2. The Complete Reference Java2, 8/e, Patrick Naughton, Herbert Schildt, TMH.
3. Java Server Faces, Hans Bergstan, O'reilly.

REFERENCE BOOKS:
1. Web Programming, building internet applications, 2/e, Chris Bates, Wiley Dreamtech
2. Programming World Wide Web, Sebesta, PEA
3. Web Tehnologies, 2/e, Godbole, kahate, TMH
4. An Introduction to web Design , Programming ,Wang, Thomson

DATA WAREHOUSING & MINING

IV Semester   **L   T   P   C**
Course Code: 173MC4T18   **3   1   0   3**

UNIT-1:
Introduction to Data mining, types of Data, Data Quality, Data Processing, Measures of Similarity and Dissimilarity, Exploring Data: Data Set, Summary Statistics, Visualization, OLAP and multi dimensional data analysis.

UNIT-II:
Classification: Basic Concepts, Decision Trees and model evaluation: General approach for solving a classification problem, Decision Tree induction, Model over fitting: due to presence of noise, due to lack of representation samples, Evaluating the performance of classifier. Nearest Neighborhood classifier, Bayesian Classifier, Support vector Machines: Linear SVM, Separable and Non Separable case.

UNIT-III:
Association Analysis: Problem Definition, Frequent Item-set generation, rule generation, compact representation of frequent item sets, FP-Growth Algorithms. Handling Categorical, Continuous attributes, Concept hierarchy, Sequential, Sub graph patterns

UNIT-IV:
Clustering: Over view, K-means, Agglomerative Hierarchical clustering, DBSCAN, Cluster evaluation: overview, Unsupervised Cluster Evaluation using cohesion and separation, using proximity matrix, Scalable Clustering algorithm

UNIT-V:
Web data mining: Introduction, Web terminology and characteristics, Web content mining, Web usage mining, web structure mining,
Search Engines: Characteristics, Functionality, Architecture, Ranking of WebPages, Enterprise search

TEXT BOOKS:
1. Introduction to Data Mining: Pang-Ning tan, Michael Steinbach, Vipin kumar, Addision - Wesley.
2. Introduction to Data Mining with Case Studies: GK Gupta; Prentice Hall.

REFERENCE BOOKS:
1. Data Mining: Introductory and Advanced Topics, Margaret H Dunham, Pearson, 2008.
2. Fundamentals of data warehouses, 2/e, Jarke, Lenzerini, Vassiliou, Vassiliadis, Springer.
3. Data Mining Theory and Practice, Soman, Diwakar, Ajay, PHI, 2006.
4. Data Mining, Concepts and Techniques, 2/e, Jiawei Han, Micheline Kamber, Elsevier, 2006.

MOBILE COMPUTING
(ELECTIVE - 1)

	L	T	P	C
IV Semester				
Course Code: 173MC4E01	**3**	**1**	**0**	**3**

UNIT-I:
Mobile Communications: An Overview- Mobile Communication-guided transmission, unguided transmission- signal propagation frequencies, antennae, modulation, modulation methods and standards for voice-oriented data communication standards, modulation methods and standards for data and voice communication, mobile computing- novel applications and limitations, mobile computing architecture, mobile system networks. Mobile devices and systems: Cellular networks and frequency reuse, Mobile smart phones, Smart mobiles and systems, handheld pocket computers, Handheld devices, Smart systems, Limitations of mobile devices

UNIT-II:
GSM and other 2G Architectures: GSM-services and system architecture, Radio interfaces of GSM, Protocols of GSM, Localization, Call handling, GPRS system architecture. Wireless medium access control, CDMA, 3G,and4G
 Communication: Modulation, Multiplexing, Controlling the medium access, Spread spectrum, Coding methods, IMT-20003G wireless communication standards, WCDMA 3G communication standards, CDMA 3G communication standards, Broadband wireless access, 4Gnetworks.

UNIT-III:
Mobile IP Network layer: IP and Mobile IP network layers: OSI layer functions, TCP/IP and Internet protocol, Mobile internet protocol; Packet delivery and Handover Management; Location Management: Agent Discovery; Mobile TCP Introduction to Mobile Adhoc network: fixed infrastructure architecture, MANET infrastructure architecture; MANET: properties, spectrum, applications; Security in Ad-hoc network; Wireless sensor networks; sensor network applications.

UNIT-IV:
Synchronization: Synchronization in mobile computing systems, Usage models for Synchronization in mobile application, Domain-dependant specific rules for data synchronization, Personal information manager, synchronization and conflict resolution strategies, synchronizer; Mobile agent: mobile agent design, aglets; Application Server

UNIT-V:
Mobile Wireless Short Range Networks and Mobile Internet: Wireless networking and wireless LAN, Wireless LAN (WLAN) architecture, IEEE 802.11protocol layers, Wireless application protocol (WAP)-WAP1.1 architecture, wireless datagram protocol (WDP), Wireless Transport Layer Security (WTLS), wireless transaction and session layers, wireless application environment.

TEXTBOOK:
1. RAJ KAMAL, "Mobile Computing," second edition, Oxford.
2. ASOKE K TALUKDER, HASANAHMED, ROOPA R YAVAGAL,"Mobile Computing, Technology Applications and Service Creation" Second Edition, Mc Graw Hill.
3. UWE Hansmann, Lother Merk, Martin S. Nocklous, Thomas Stober, "Principles of Mobile Computing," Second Edition, Springer

HUMAN COMPUTER INTERACTION
(ELECTIVE I)

IV Semester   **L   T   P   C**
Course Code: 173MC4E02   **3   1   0   3**

UNIT-I:
Introduction: Importance of user Interface, definition, importance of good design. Benefits of good design. A brief history of Screen design
The graphical user interface: Popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user –interface popularity, characteristics- Principles of user interface.

UNIT-II:
Design process: Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds,
Understanding business junctions.

UNIT–III:
Screen Designing : Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow, Visually pleasing composition, amount of information, focus and emphasis, presentation information simply and meaningfully, information retrieval on web, statistical graphics, Technological consideration in interface design.

UNIT-IV:
Windows: Windows new and Navigation schemes selection of window, selection of devices based and screen based controls.
Components: Components text and messages, Icons and increases, Multimedia, colors, uses problems, choosing colors.

UNIT-V:
Software tools : Specification methods, interface, Building Tools.
Interaction Devices: Keyboard and function keys, pointing devices, speech recognition digitization and generation, image and video displays, drivers.

TEXT BOOKS :
1. Human Computer Interaction. 3/e, Alan Dix, Janet Finlay, Goryd, Abowd, Russell Beal, PEA, 2004.
2. The Essential guide to user interface design,2/e, Wilbert O Galitz, Wiley Dreama Tech.

REFERENCE BOOKS:
1. Designing the user interface. 4/e, Ben Shneidermann, PEA.
2. User Interface Design, Soren Lauesen , PEA.
3. Interaction Design PRECE, ROGERS, SHARPS, Wiley .
4. Human Computer, Interaction Dan R.Olsan, Cengage ,2010.

CLOUD COMPUTING
(ELECTIVE - 1)

IV Semester   L   T   P   C
Course Code: 173MC4E03   3   1   0   3

UNIT I:
Introduction: Network centric computing, Network centric content, peer-to –peer systems, cloud computing delivery models and services, Ethical issues, Vulnerabilities, Major challenges for cloud computing
Parallel and Distributed Systems: introduction, architecture, distributed systems, communication protocols, logical clocks, message delivery rules, concurrency, model concurrency with Petri Nets.

UNIT II:
Cloud Infrastructure: At Amazon, The Google Perspective, Microsoft Windows Azure, Open Source Software Platforms, Cloud storage diversity, Inter cloud, energy use and ecological impact, responsibility sharing, user experience, Software licensing
Cloud Computing : Applications and Paradigms: Challenges for cloud, existing cloud applications and new opportunities, architectural styles, workflows, The Zookeeper, The Map Reduce Program model, HPC on cloud, biological research

UNIT III:
Cloud Resource virtualization: Virtualization, layering and virtualization, virtual machine monitors, virtual machines, virtualization- full and para, performance and security isolation, hardware support for virtualization, Case Study: Xen, vBlades
Cloud Resource Management and Scheduling: Policies and Mechanisms, Applications of control theory to task scheduling, Stability of a two-level resource allocation architecture, feed back control based on dynamic thresholds, coordination, resource bundling, scheduling algorithms, fair queuing, start time fair queuing, cloud scheduling subject to deadlines, Scheduling Map Reduce applications, Resource management and dynamic application scaling

UNIT IV:
Storage Systems: Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system. Apache Hadoop, Big Table, Megastore (text book 1), Amazon Simple Storage Service(S3) (Text book 2)
Cloud Security: Cloud security risks, security – atop concern for cloud users, privacy and privacy impact assessment, trust, OS security, Virtual machine security, Security risks

UNIT V:
Cloud Application Development: Amazon Web Services : EC2 – instances, connecting clients, security rules, launching, usage of S3 in Java, Installing Simple Notification Service on Ubuntu 10.04, Installing Hadoop on Eclipse, Cloud based simulation of a Distributed trust algorithm, Cloud service for adaptive data streaming (Text Book 1)
Google: Google App Engine, Google Web Toolkit (Text Book 2)
Microsoft: Azure Services Platform, Windows live, Exchange Online, Share Point Services, Microsoft Dynamics CRM (Text Book 2)

TEXT BOOKS:
1. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier
2. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH

REFERENCE BOOK:
1. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammarai selvi, TMH

SOFTWARE PROJECT MANAGEMENT
(ELECTIVE-II)

IV Semester   **L   T   P   C**
Course Code: 173MC4E04   **3   1   0   3**

UNIT-I:
Conventional Software Management: The waterfall model, conventional software Management performance.
Evolution of Software Economics: Software Economics, pragmatic software cost estimation.
Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.
The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT-II:
Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.
Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT- III:
Model based software architectures: A Management perspective and technical perspective.
Work Flows of the process: Software process workflows, Iteration workflows.
Checkpoints of the process: Major mile stones, Minor Milestones, Periodic status assessments.
Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT- IV:
Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.
Process Automation: Automation Building blocks, The Project Environment.
Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

UNIT-V:
Tailoring the Process: Process discriminates.
Future Software Project Management: Modern Project Profiles, Next generation Software economics, modern process transitions.

TEXTBOOKS:
1. Software Project Management, Walker Royce, PEA, 2005.

REFERENCE BOOKS:
1. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH
2. Software Project Management, Joel Henry, PEA
3. Software Project Management in practice, Pankaj Jalote, PEA, 2005,
4. Effective Software Project Management, Robert K.Wysocki, Wiley,2006
5. Project Management in IT, Kathy Schwalbe, Cengage
6. Quality Software Project Management, Futrell,Donald F. Shafer, Donald I. Shafer, PEA

ARTIFICIAL INTELLIGENCE
(ELECTIVE-II)

IV Semester   **L   T   P   C**
Course Code: 173MC4E05   **3   1   0   3**

UNIT-I:
Introduction to artificial intelligence: Introduction , history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of ai languages, current trends in AI
Problem solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative-deepening a*, constraint satisfaction

UNIT-II:
Problem reduction and game playing: Introduction, problem reduction, game playing, alpha-beta pruning, two-player perfect information games
Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

UNIT-III:
Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames
advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, cyc theory, case grammars, semantic web
Expert system and applications: Introduction phases in building expert
systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

UNIT-IV:
Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory
Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

UNIT-V:
Machine learning paradigms: Introduction, machine learning systems, supervised and unsupervised learnings, inductive learning, deductive learning, clustering, support vector machines, case based reasoning and learning
Artificial neural networks: Introduction, artificial networks, single layer feed forward networks, multi layered forward networks, design issues of artificial neural networks

TEXT BOOKS:
1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nd ed, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rd ed, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

REFERENCE BOOKS:
1. Artificial intelligence, structures and Strategies for Complex problem solving, -George F Lugar, 5th ed, PEA
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

EMBEDDED SYSTEMS
(ELECTIVE-II)

IV Semester	**L**	**T**	**P**	**C**
Course Code: 173MC4E06	**3**	**1**	**0**	**3**

UNIT-I:
Introduction to Embedded systems: What is an embedded system Vs. General computing system, history, classification, major application areas, and purpose of embedded systems. Core of embedded system, memory, sensors and actuators, communication interface, embedded firmware, other system components, PCB and passive components.

UNIT-II:
8—bit microcontrollers architecture: Characteristics, quality attributes application specific, domain specific, embedded systems. Factors to be considered in selecting a controller, 8051 architecture, memory organization, registers, oscillator unit, ports, source current, sinking current, design examples.

UNIT-III:
RTOS and Scheduling, Operating basics, types, RTOS, tasks, process and threads, multiprocessing and multitasking, types of multitasking, non preemptive, preemptive scheduling.

UNIT-IV:
Task communication of RTOS, Shared memory, pipes, memory mapped objects, message passing, message queue, mailbox, signaling, RPC and sockets, task communication/synchronization issues, racing, deadlock, live lock, the dining philosopher's problem.

UNIT-V:
The producer-consumer problem, Reader writers problem, Priority Inversion, Priority ceiling, Task Synchronization techniques, busy waiting, sleep and wakery, semaphore, mutex, critical section objects, events, device, device drivers, how to clause an RTOS, Integration and testing of embedded hardware and fire ware.
Simulators, emulators, Debuggers, Embedded Product Development life cycle (EDLC), Trends in embedded Industry, Introduction to ARM family of processor.

TEXT BOOK:
1. Introduction to embedded systems Shibu. K.V, TMH, 2009.

REFERENCE BOOKS:
1. Ayala & Gadre: The 8051 Microcontroller & Embedded Systems using Assembly and CENGAGE
2. Embedded Systems, Rajkamal, TMH, 2009.
3. Embedded Software Primer, David Simon, Pearson.
4. The 8051 Microcontroller and Embedded Systems, Mazidi, Mazidi, Pearson,.

ADVANCED JAVA & WEB TECHNOLOGIES LAB

IV Semester   **L   T   P   C**
Course Code: 173MC4L10   **0   0   3   2**

Week-1:

Design the following static web pages required for an online book store web site.

1) **HOMEPAGE:**

The static home page must contain three **frames**.

Top frame:

Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame:

At least four links for navigation, which will display the catalogue of respective links.

For e.g.: When you click the link **"CSE"** the catalogue for **CSE** Books should be displayed in the Right frame.

Right frame:

The pages to the links in the left frame must be loaded here. Initially this page contains description of the web site.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Description of the Web Site			
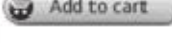
2)**LOGINPAGE:**This page looks like below

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Login : Password: [ ] [Submit] [Reset]			

3) CATOLOGUEPAGE:

The catalogue page should contain the details of all the books available in the web site in a table. The details should contain the following:

1. Snap shot of Cover Page. 2. Author Name. 3. Publisher. 4. Price. 5. Add to cart button.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL		Book : XML Bible Author : Winston Publication : Wiely	$ 40.5	Add to cart
		Book : AI Author : S.Russel Publication : Princeton hall	$ 63	Add to cart
		Book : Java 2 Author : Watson Publication : BPB publications	$ 35.5	Add to cart
		Book : HTML in 24 hours Author : Sam Peter Publication : Sam publication	$ 50	Add to cart

Note: Week 2 contains the remaining pages and their description.

Week-2:
4) CART PAGE: The cart page contains the details about the books which are added to the cart.The cart page should look like this:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Book name	Price	Quantity	Amount
	Java 2	$35.5	2	$70
	XML bible	$40.5	1	$40.5
		Total amount -		$130.5

5)REGISTRATIONPAGE:
Create a "*registration form* "with the following fields
1) Name (Text field)   2) Password (password field)
3) E-mail id (text field)   4) Phone number (text field)
5) Sex (radio button)   6)Date of birth (3 select boxes)
7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
8) Address (text area)

WEEK3:
VALIDATION:
Write *JavaScript* to validate the following fields of the above registration page.
1. Name (Name should contains alphabets and the length should not be less than 6 characters).
2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the   standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only). Note : You can also validate the login page with these parameters. Use PHP to connect with the database to store the above details.

Week-4:
Design a web page using **CSS (C**ascading **S**tyle
Sheets**)** which includes the following:

      1) Use different font, styles: In the style definition you
      define how each selector should work (font, color etc.).
      Then, in the body of your pages, you refer to these
      selectors to activate the styles.

      For example:

```
<HTML>
<HEAD>
<style type="text/css">
B.headline {color:red; font-size:22px; font-family:arial; text-decoration:underline}
</style>

</HEAD>
```

```
<BODY>
<b>This is normal bold</b><br>
Selector {cursor:value}

For example:

<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink{cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>

<b class="headline">This is headline style bold</b>
</BODY>

</HTML>
```

2) Set a background image for both the page and single
   elements on the page.

```
BODY {background-image:url(myimage.gif);}
```

You can define the background image for the page like this:
3) Control the repetition of the image with the background-repeat property. As
background- repeat: repeat Tiles the image until the entire page is
filled, just like an ordinary background image in plain HTML.
4) Define styles for links as
A:link
A:visitedA:active A:hover Example:

    Aditya Engineering College (A)   61

```
<style type="text/css">
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline; color: red;}
</style>
```
5) Work with layers:
For example:
LAYER 1 ON TOP:
```
<div style="position: relative; font-size:50px; z-index:2;">LAYER 1</div>
<div style="position: relative; top:-50; left:5; color: red; font-size:80px; zindex:
1">LAYER 2</div>
```
LAYER 2 ON TOP:
```
<div style="position: relative; font-size:50px; z-index:3;">LAYER 1</div>
<div style="position: relative; top:-50; left:5; color:red; font-size:80px; zindex:
4">LAYER 2</div>
```
6) Add a customized cursor:
Selector {cursor:value}

For example:

```
<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink{cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
```

Week-5:
Write an XML file which will display the Book information which
includes the following:
1) Title of the book   2) Author Name
3) ISBN number   4) Publisher name
5) Edition   6) Price
Write a Document Type Definition (DTD) to validate the above XML file.
Display the XML file as follows.
The contents should be displayed in a table. The header of the table should be in color GREY.
And the Author names column should be displayed in one color and should be capitalized and in

bold. Use your own colors for remaining columns. Use XML schemas XSL and CSS for the above purpose.

Note: Give at least for 4 books. It should be valid syntactically.

Hint: You can use some xml editors like XML-spy

Week-6:
VISUALBEANS:

Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the "property window".

Week-7:

1) Install TOMCAT web server and APACHE.

While installation assign port number 4040 to TOMCAT and 8080 to

APACHE. Make sure that these ports are available i.e., no other process is using this port.

2) Access the above developed static web pages for books web site,

using these servers by putting the web pages developed in week-1 and week-2 in the document root. Access the pages by using the urls : http://localhost:4040/rama/books.html (for tomcat) http://localhost:8080/books.html (for Apache)

Week-8:
User Authentication:

Assume four users user1, user2, user3 and user4 having the passwords pwd1,pwd2,pwd3 and pwd4 respectively. Write a serve let for doing the following.

1. Create a Cookie and add these four user id's and passwords to this

Cookie.

2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies. If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display " You are not an authenticated user ". Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters () method.

Week-9:

Install a database (Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number (these should hold the data from the registration form). Practice 'JDBC' connectivity.

Write a java program/servlet/JSP to connect to that database and

extract data from the tables and display them. Experiment with various

SQL queries. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Week-10:

Write a JSP which does the following job:

Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

Week-11:

Create tables in the database which contain the details of items (books in our case like Book name, Price, Quantity, Amount)) of each category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

Week-12:

HTTP is a stateless protocol. Session is required to maintain the state. The user may add some items to cart from the catalog page. He can check the cart page for the selected items. He may visit the catalogue again and select some more items. Here our interest is the selected items should be added to the old cart rather than a new cart. Multiple users can do the same thing at a time(i.e., from different systems in the LAN using the ip-address instead of local host). This can be achieved through the use of sessions. Every user will have his own session which will be created after his successful login to the website. When the user logs out his session should get invalidated (by using the method session. Invalidate ()). Modify your catalogue and cart JSP pages to achieve the above mentioned functionality using sessions.

DATA WAREHOUSING AND MINING LAB

IV Semester   **L   T   P   C**
Course Code: 173MC4L11   **0   0   3   2**

1. Demonstration of preprocessing on dataset student. arff

2. Demonstration of preprocessing on dataset labor. arff

3. Demonstration of Association rule process on dataset contact lenses. arff using apriority algorithm

4. Demonstration of Association rule process on dataset test. arff using apriority algorithm

5. Demonstration of classification rule process on dataset student. arff using j48 algorithm

6. Demonstration of classification rule process on dataset employee .arff using j48 algorithm

7. Demonstration of classification rule process on dataset employee. arff using id3 algorithm

8. Demonstration of classification rule process on dataset employee. arff using naïve bayes algorithm

9. Demonstration of clustering rule process on dataset iris. arff using simple k-means

10. Demonstration of clustering rule process on dataset student. arff using simple k-means

OBJECT ORIENTED ANALYSIS & DESIGN LAB

IV Semester   **L   T   P   C**
Course Code: 173MC4L12   **0   0   3   2**

OBJECTIVES:
- Construct UML diagrams for static view and dynamic view of the system.
- Generate creational patterns by applicable patterns for given context.
- Create refined model for given Scenario using structural patterns.
- Construct behavioral patterns for given applications.

Week 1:
Familiarization with Rational Rose or Umbrella
For each case study: Week 2, 3 & 4:
For each case study:
a) Identify and analyze events
b) Identify Use cases
c) Develop event table
d) Identify & analyze domain classes
e) Represent use cases and a domain class diagram using Rational Rose
f) Develop CRUD matrix to represent relationships between use cases and problem domain classes

Week 5 & 6:
For each case study:
a) Develop Use case diagrams
b) Develop elaborate Use case descriptions & scenarios
c) Develop prototypes (without functionality)
d) Develop system sequence diagrams

Week 7, 8, 9 & 10:
For each case study:
a) Develop high-level sequence diagrams for each use case
b) Identify MVC classes / objects for each use case
c) Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects
d) Develop detailed design class model (use GRASP patterns for responsibility assignment)
e) Develop three-layer package diagrams for each case study

Week 11 & 12:
For each case study:
a) Develop Use case Packages
b) Develop component diagrams
c) Identify relationships between use cases and represent them
d) Refine domain class model by showing all the associations among classes

Week 13 onwards: For each case study:
a) Develop sample diagrams for other UML diagrams - state chart diagrams, activity diagrams and Deployment diagrams

OUTCOMES:
- Understand the Case studies and design the Model.
- Understand how design patterns solve design problems.
- Develop design solutions using creational patterns.
- Construct design solutions by using structural and behavioral patterns

BIG DATA ANALYTICS

V Semester	**L**	**T**	**P**	**C**
Course Code:173MC5T19	**3**	**1**	**0**	**3**

UNIT-I:
Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization

UNIT-II:
Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Name node, Data node, Secondary Name node, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT-III:
Writing Map Reduce Programs: A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Partitioner

UNIT-IV
Hadoop I/O: The Writable Interface, Writable Comparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, Bytes Writable, Null Writable, Object Writable and Generic Writable, Writable collections, Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators

UNIT-V
Pig: Hadoop Programming Made Easier Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin Applying Structure to Hadoop Data with
Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data

TEXTBOOKS:
1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss

REFERENCE BOOKS:
1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop Map Reduce Cookbook, Srinath Perera, Thilina Gunarathne

Software Links:
1. Hadoop: http://hadoop.apache.org/
2. Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3. Piglatin: http://pig.apache.org/docs/r0.7.0/tutorial.html

NETWORK PROGRAMMING

V Semester   **L   T   P   C**
Course Code:173MC5T20   **3   1   0   3**

UNIT-I:
Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

UNIT-II:
TCP client server: Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

UNIT-III:
Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function.
I/O Multiplexing and socket options: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server, getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and TCP socket options.

UNIT-IV:
Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP.
Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

UNIT-V:
IPC : Introduction, File and record locking, Pipes, FIFOs streams and messages, Name spaces, system IPC, Message queues, Semaphores.
Remote Login: Terminal line disciplines, Pseudo- Terminals, Terminal modes, Control Terminals, rlogin Overview, RPC Transparency Issues.

Text Book:
1. UNIX Network Programming, Vol. I, SocketsAPI, 2nd Edition. - W.Richard Stevens, Pearson Edn. Asia.
2. UNIX Network Programming, 1st Edition, - W.Richard Stevens. PHI.

References Books:
1. UNIX Systems Programming using C++ T CHAN, PHI.
2. UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education
3. Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson   Education

PYTHON PROGRAMMING

V Semester   **L   T   P   C**
Course Code: 173MC5T21   **3   1   0   3**

UNIT – I:

Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

UNIT – II:

Types, Operators and Expressions: Types - Integers, Strings, Booleans; Operators-Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass

UNIT – III:

Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

UNIT – IV:

Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions(Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

Modules: Creating modules, import statement, from. Import statement, name spacing,
Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages

UNIT – V:

Object Oriented Programming OOP in Python: Classes, 'self variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding,
Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions
Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics
Testing: Why testing is required ?, Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

TEXTBOOKS:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Learning Python, Mark Lutz, Orielly

REFERENCE BOOKS:

1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W.Chun, Pearson.
3. Introduction to Python, Kenneth A. Lambert, Cengage

CYBER SECURITY
(ELECTIVE – III)

V Semester   **L   T   P   C**
Course Code: 173MC5E07   **3   1   0   3**

UNIT- I: Introduction to Cybercrime:

Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals? , Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens

UNIT -II: Cyber offenses:

How Criminals Plan Them – Introduction, How Criminals Plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector Cloud Computing.

UNIT -III: Cybercrime Mobile and Wireless Devices:

Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops.

UNIT -IV: Tools and Methods Used in Cybercrime:

Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Phishing and Identity Theft: Introduction,   Phishing,   Identity   Theft   (ID   Theft)

UNIT -V: Cybercrimes and Cyber security:

Why Do We Need Cyber laws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Information Security Planning and Governance, Information Security Policy Standards, Practices, The information Security Blueprint, Security education, Training and awareness program, Continuing Strategies.

TEXTBOOKS:

1. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Nina Godbole, Sunit Belapure, Wiley.
2. Principles of Information Security, Micheal E.Whitman and Herbert J.Mattord, Cengage Learning.

References Books :

1. Information Security, Mark Rhodes, Ousley, MGH.

COMPUTER FORENSICS
(ELECTIVE–III)

V Semester   **L   T   P   C**
Course Code: 173MC5E08   **3   1   0   3**

UNIT-I:
Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations, Taking A Systematic Approach, Procedure for Corporate High-Tech Investigations, Understanding Data Recovery Workstations and Software,
Investor's Office and Laboratory: Understanding Forensics Lab Certification Requirements, Determining the Physical Requirements for a Computer Forensics Lab, Selecting a Basic Forensic Workstation

UNIT-II
Data Acquisition: Understanding Storage Formats for Digital Evidence, Determining the Best Acquisition Method, Contingency Planning for Image Acquisitions, Using Acquisition Tools, Validating Data Acquisition, Performing RAID Data Acquisition, Using Remote Network Acquisition Tools, Using Other Forensics Acquisition Tools
Processing Crime and Incident Scenes: Identifying Digital Evidence, Collecting the Evidence in Private-Sector Incident Scenes, Processing law Enforcement Crime Scenes, Preparing for a Search, Securing a Computer Incident or Crime Scene, Sizing Digital evidence at the Scene, Storing Digital evidence, obtaining a Digital Hash.

UNIT-III
Current Computer Forensics Tools: Evaluating Computer Forensics Toll Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software
Computer Forensics Analysis and Validation: Determining What Data to Collect and Analyze, Validating Forensic Data, Addressing Data-Hiding Techniques, Performing Remote Acquisition

UNIT-IV
Recovering Graphics and Network Forensics: Recognizing a Graphics File, Understanding Data Compression, Locating and Recovering Graphics Files, Understanding Copyright Issues with Graphics, Network Forensic, Developing Standard Procedure for Network Forensics, Using Network Tools, Examining Hiney Project

UNIT-V
E-mail Investigations Cell Phone and Mobile Device Forensics: Exploring the Role of E-mail in Investigations, Exploring the Role of Client and Server in E-mail, Investigating E-mail Crimes and Violations, Understanding E-mail Servers, Using Specialized E-mail Forensics Tools, Understanding Mobile Device Forensics, Understanding Acquisition Procedure for Cell Phones and Mobile Devoices

Textbook:
1. Nelson, Phillips Enfinger, Steuart, " Computer Forensics and Investigations, Cengage Learning.

E-COMMERCE (ELECTIVE–III)

V Semester   **L   T   P   C**
Course Code: 173MC5E09   **3   1   0   3**

UNIT-I
Electronic Commerce, Frame work, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications. Consumer Oriented Electronic commerce, Mercantile Process models.

UNIT-II
Electronic payment systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems.

UNIT-III
Inter Organizational Commerce - EDI, EDI Implementation, Value added networks. Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management.

UNIT- IV
Corporate Digital Library - Document Library, digital Document types, corporate Data Warehouses. Advertising and Marketing, Information based marketing, Advertising on Internet, on-line marketing process, market research.

UNIT-V
Consumer Search and Resource Discovery, Information search and Retrieval, Commerce Catalogues, Information Filtering. Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processings, Desktop video conferencing.

Textbook:
1. Frontiers of Electronic Commerce, Kalakata, Whinston, PEA,2006.

REFERENCE BOOKS:
1. E-Commerce Fundamentals and Applications Hendry Chan, Raymond Lee, Dillon, Chang, John Wiley.
2. E-Commerce, A Managerial Perspective, Turban E, Lee J , King, Chung H.M.,PEA,2001.
3. E-Commerce An Indian Perspective , 3/e, P.T. Joseph, PHI,2009.
4. E-Commerce, S.Jaiswal , Galgotia.
5. Electronic Commerce , Gary P.Schneider, Thomson.

INTERNET OF THINGS
(ELECTIVE–IV)

		L	**T**	**P**	**C**
V Semester					
Course Code: 173MC5E10		**3**	**1**	**0**	**3**

UNIT- I:

The Internet of Things: An Overview of Internet of things, Internet of Things Technology, behind Io Ts Sources of the Io Ts, M2M Communication, Examples OF Io Ts, Design Principles For Connected Devices Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

UNIT -II:

Business Models for Business Processes in the Internet of Things ,IoT/M2M systems LAYERS AND designs standardizations ,Modified OSI Stack for the IoT/M2M Systems , ETSI M2M domains and High-level capabilities ,Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability

UNIT- III:

Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

UNIT -IV:

Data Acquiring, Organizing and Analytics in IoT/M2M, Applications/Services/Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in the Internet Of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

UNIT- V:

Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology, Sensing the World.

TEXTBOOKS:
1. Internet of Things: Architecture, Design Principles And Applications, Rajkamal, McGraw Hill Higher Education
2. Internet of Things, A.Bahgya and V.Madisetti, Univesity Press, 2015

REFERENCE BOOKS:
1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley
2. Getting Started with the Internet of Things CunoPfister, Oreilly.

MULTIMEDIA APPLICATION DEVELOPMENT
(ELECTIVE–IV)

V Semester   **L   T   P   C**
Course Code: 173MC5E11   **3   1   0   3**

UNIT- I
Fundamental concepts in Text and Image:
Multimedia and hypermedia, world wide web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color in image and video: color science, color models in images, color models in video.

UNIT- II
Fundamental Concepts in Video and Digital Audio:
Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization and transmission of audio.

UNIT-III
Action Script I: Action Script Features, Object-Oriented Action Script, Data types and Type Checking, Classes, Authoring an Action Script Class.
Action Script II: Inheritance, Authoring an Action Script 2.0 Subclass, Interfaces, Packages, Exceptions.
Application Development: An OOP Application Frame work, Using Components with Action Script Movie Clip Subclasses.

UNIT- IV
Multimedia Data Compression: Lossless compression algorithm: Run-Length Coding, Variable Length Coding, Dictionary Based Coding, Arithmetic Coding, Lossless Image Compression, Lossy compression algorithm: Quantization, Transform Coding, Wavelet-Based Coding, Embedded Zero tree of Wavelet Coefficients Set Partitioning in Hierarchical Trees (SPIHT).
Basic Video Compression Techniques: Introduction to video compression, video compression based on motion compensation, search for motion vectors, MPEG, Basic Audio Compression Techniques.

UNIT-V
Multimedia Networks: Basics of Multimedia Networks, Multimedia Network Communications and
Applications: Quality of Multimedia Data Transmission, Multimedia over IP, Multimedia over ATM etworks, Transport of MPEG- 4, Media-on- Demand (MOD).

TEXTBOOKS:
1. Fundamentals of Multimedia, Ze-Nian Li , Mark S. Drew, PHI/ PEA.
2. Multimedia Systems, Parag Havaldar, Gerard Medioni, cengage, 2009.
3. Essentials Action Script 3.0, Colin Moock, SPD O, Reilly, 2007.

REFERENCE BOOKS:
1. Multimedia Applications, Steinmetz, Nahrstedt, Springer.
2. Digital Multimedia, Nigel Chapman, Jenny Chapman, Wiley- Dreamtech.
3. Multimedia & Communications Technology, Steve Heath, Elsevier .
4. Multimedia Technology & Applications, David Hilman , Galgotia.
5. Multimedia Technologies, Banerji, Mohan Ghosh, MGH.

SOFTWARE TESTING METHODOLOGIES
(ELECTIVE IV)

V Semester   **L   T   P   C**
Course Code:173MC5E12   **3   1   0   3**

UNIT-I
Introduction:- Purpose of testing, Dichotomies, model for testing, consequences of bugs, Taxonomy of bugs Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT-II
Transaction Flow Testing:-transaction flows, transaction flow testing techniques. Dataflow
Testing: - Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT-III
Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains and Interfaces testing, domain and interface testing, domains and testability.

UNIT-IV
Paths, Path products and Regular expressions:- path products & path _expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing:- overview, decision tables, path expressions, kv charts, specifications.

UNIT-V
State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips.Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools.

TEXTBOOKS:
1.   Software testing techniques - Boris Beizer, International Thomson computer press, second edition.
2.   Software Testing- Yogesh Singh, CAMBRIDGE

REFERENCE BOOKS:
1.   Introduction to Software Testing, Paul Amman, Jeff Offutt, CAMBRIDGE
2.   Effective Software testing, 50 Specific ways to improve your testing, Elfriede Dustin, PEA

BIG DATA ANALYTICS LAB

		L	T	P	C
V Semester

Course Code:173MC5L13   L   T   P   C
                                  0   0   3   2

Week 1,2:

1. Implement the following Data structures in Java a)Linked Lists b) Stacks c) Queues d) Set e) Map

Week 3, 4:

2. (i)   Perform setting up and Installing Hadoop in its three operating modes: Standalone, Pseudo distributed, Fully distributed
    (ii) Use web based tools to monitor your Hadoop setup.

Week 5:

3. Implement the following file management tasks in Hadoop:

• Adding files and directories
• Retrieving files
• Deleting files

Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.

Week 6:

4. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

Week 7:

5. Write a Map Reduce program that mines weather data.

Weather sensors collecting data every hour at many locations across the globe gather a large volume of log data, which is a good candidate for analysis with MapReduce, since it is semi structured and record-oriented.

Week 8:

6. Implement Matrix Multiplication with Hadoop Map Reduce

Week 9,10:

7. Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.

Week 11, 12:

8. Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes

NETWORK PROGRAMMING LAB

V Semester   **L   T   P   C**
Course Code: 173MC5L14   **0   0   3   2**

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 that accepts any number of arguments and prints them in the reverse order.
7. Write a shell script that determines the period for which a specified user is working on the system.
8. Write a shell script to list all of the directory files in a directory.
9. Write an interactive file-handling shell program- Let it offer the user the choice of copying, removing or linking files. Once the user has made a choice, have the program ask him for the necessary information such as the file name, new name and so on.
10. Write a shell script to find factorial of a given integer.
11. Write a shell script to find the G.C.D. of two integers.
12. Write a shell script to generate a multiplication table.
13. Write a shell script that copies multiple files to a directory.
14. Write a shell script that counts the number of lines and words present in a given file.

PYTHON PROGRAMMING LAB

	L	T	P	C
V Semester				
Course Code: 173MC5L15	**0**	**0**	**3**	**2**

Exercise 1 - Basics
a) Running instructions in Interactive interpreter and a Python Script
b) Write a program to purposefully raise Indentation Error and Correct it

Exercise 2 - Operations
a) Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)
b) Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

Exercise - 3 Control Flow
    a) Write a Program for checking whether the given number is a even number or not.
    b) Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . . . , 1/10
    c) Write a program using a for loop that loops over a sequence. What is sequence?
    d) Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Exercise 4 - Control Flow - Continued
    a) Find the sum of all the primes below two million. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be: 1,   2, 3, 5, 8, 13, 21, 34, 55, 89, ...
    b) By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Exercise - 5 - DS
    a) Write a program to count the numbers of characters in the string and store them in a Dictionary data structure.
    b) Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Exercise - 6 DS - Continued
    a) Write a program combine lists that combines these lists into a dictionary.
    b) Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

Exercise - 7 Files
    a) Write a program to print each line of a file in reverse order.
    b) Write a program to compute the number of characters, words and lines in a file.

Exercise - 8 Functions

a)   Write a function ball_collide that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

   Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius If   (distance between two   balls centers) <= (sum of their radii) then (they are colliding)

b)   Find mean, median, mode for the given set of numbers in a list.

Exercise - 9 Functions - Continued

a)   Write a function nearly equal to test whether two strings are nearly equal. Two strings a and b are   nearly equal when a can be generated by a single mutation on b.

b)   Write a function dups to find all duplicates in the list.

c)   Write a function unique to find all the unique elements of a list.

Exercise - 10 - Functions - Problem Solving

a) Write a function cumulative product to compute cumulative product of a list of numbers.

b) Write a function reverse to reverse a list. Without using the reverse function.

c) Write function to compute gcd, lcm of two numbers. Each function shouldn't   exceed   one line.

Exercise 11 - Multi-D Lists

a)   Write a program that defines a matrix and prints

b)   Write a program to perform addition of two square matrices

c)   Write a program to perform

d)   Multiplication of two square matrices

Exercise - 14 GUI, Graphics

1.   Write a GUI for an Expression Calculator using tk

2.   Write a program to implement the following figures using turtle
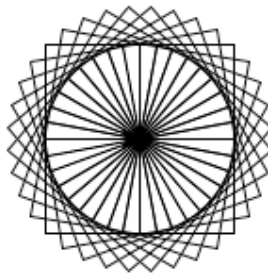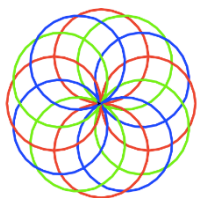
Exercise - 15 - Testing

a)   Write a test-case to check the function even _numbers which return True on   passing a list of all even numbers

b)   Write a test-case to check the function reverse _string which returns the reversed   string

Exercise - 16 - Advanced

a)   Build any one classical data structure.

b)   Write a program to solve knapsack problem.

SEMINAR

VI Semester   **L   T   P   C**
Course Code: 173MC6R01   **0   0   0   2**

****

MAJOR PROJECT

VI Semester   **L   T   P   C**
Course Code:173MC6P01   **0   0   0   19**

****