

**(Common to CE, CSE, IT, Chemical, PE, PCE,  
Civil Branches)**

**UNIT I Random variables and Distributions:**

Introduction- Random variables- Distribution function- Discrete distributions ( Review of Binomial and Poisson distributions)- Continuous distributions: Normal, Normal approximation to Binomial distribution, Gamma and Weibull distributions

Subject Category

ABET Learning Objectives a b e k

ABET internal assessments 1 2 6

JNTUK External Evaluation A B E

**UNIT II Moments and Generating functions:**

Introduction-Mathematical expectation and properties - Moment generating function - Moments of standard distributions ( Binomial, Poisson and Normal distributions) – Properties

Subject Category

ABET Learning Objectives a e

ABET internal assessments 1 2 6

JNTUK External Evaluation A B E

**UNIT III Sampling Theory:**

Introduction - Population and samples- Sampling distribution of mean for large and small samples (with known and unknown variance) - Proportion sums and differences of means -Sampling distribution of variance -Point and interval estimators for means and proportions

Subject Category

ABET Learning Objectives a e k

ABET internal assessments 1 2 6

JNTUK External Evaluation A B E

**UNIT IV Tests of Hypothesis:**

Introduction - Type I and Type II errors - Maximum error - One tail, two-tail tests- Tests concerning one mean and proportion, two means- Proportions and their differences using Z-test, Student's t-test - F-test and Chi -square test - ANOVA for one-way and two-way classified data

Subject Category

ABET Learning Objectives a b d e h k

ABET internal assessments 1 2 6 7 10

JNTUK External Evaluation A B D E F

**UNIT V Curve fitting and Correlation:**

Introduction - Fitting a straight line –Second degree curve-exponential curve-power curve by method of least squares.

Simple Correlation and Regression - Rank correlation - Multiple regression

Subject Category

ABET Learning Objectives a d e h k

ABET internal assessments 1 2 6 10  
JNTUK External Evaluation A B E

### UNIT VI Statistical Quality Control Methods:

Introduction - Methods for preparing control charts – Problems using  $\bar{x}$ -bar, p, R charts and attribute charts

Subject Category

ABET Learning Objectives a e k

ABET internal assessments 1 2 6

JNTUK External Evaluation A B E F

### Books:

1. Probability and Statistics for Engineers: Miller and John E. Freund, Prentice Hall of India
2. Probability and Statistics for Engineers and Scientists: Ronald E. Walpole, Sharon L. Mayers and Keying Ye: Pearson
3. Probability, Statistics and Random Processes, Murugesan, Anuradha Publishers, Chennai:

Subject Category	ABET Learning Objectives	ABET Internal Assessments	JNTUK External Evaluation	Remarks
Theory Design Analysis Algorithms Drawing Others	a) Apply knowledge of math, science, & engineering b) Design & conduct experiments, analyze & interpret data c) Design a system/process to meet desired needs within economic, social, political, ethical, health/safety, manufacturability, & sustainability constraints d) Function on multidisciplinary teams e) Identify, formulate, & solve engineering problems f) Understand professional & ethical responsibilities g) Communicate effectively h) Understand impact of engineering solutions in global, economic, environmental, & societal context i) Recognize need for & be able to engage in lifelong learning j) Know contemporary issues k) Use techniques, skills, modern tools for engineering practices	1. Objective tests 2. Essay questions tests 3. Peer tutoring based 4. Simulation based 5. Design oriented 6. Problem based 7. Experiential (project based) based 8. Lab work or field work based 9. Presentation based 10. Case Studies based 11. Role-play based 12. Portfolio based	A. Questions should have: B. Definitions, Principle of operation or philosophy of concept. C. Mathematical treatment, derivations, analysis, synthesis, numerical problems with inference. D. Design oriented problems E. Trouble shooting type of questions F. Applications related questions G. Brain storming questions	

## II Year – II SEMESTER

T	P
3+1	0

### JAVA PROGRAMMING

**Objective: Implementing programs for user interface and application development using core java principles**

#### UNIT I:

**Objective: Focus on object oriented concepts and java program structure and its installation**

##### Introduction to OOP

Introduction, Need of Object Oriented Programming, Principles of Object Oriented Languages, Procedural languages Vs OOP, Applications of OOP, History of JAVA, Java Virtual Machine, Java Features, Program structures, Installation of JDK1.6

#### UNIT II:

**Objective: Comprehension of java programming constructs, control structures in Java**

##### Programming Constructs

Variables , Primitive Datatypes, Identifiers- Naming Conventions, Keywords, Literals, Operators-Binary,Unary and ternary, Expressions, Precedence rules and Associativity, Primitive TypeConversion and Casting, Flow of control-Branching,Conditional, loops.,

**Classes and Objects-** classes, Objects, Creating Objects, Methods, constructors-Constructor overloading, cleaning up unused objects-Garbage collector, Class variable and Methods-Static keyword, this keyword, Arrays, Command line arguments

#### UNIT III:

**Objective: Implementing Object oriented constructs such as various class hierarchies, interfaces and exception handling**

**Inheritance:** Types of Inheritance, Deriving classes using extends keyword, Method overloading, super keyword, final keyword, Abstract class

**Interfaces, Packages and Enumeration:** Interface-Extending interface, Interface Vs Abstract classes, Packages-Creating packages , using Packages, Access protection, java.lang package

**Exceptions & Assertions** - Introduction, Exception handling techniques-try...catch, throw, throws, finally block, user

defined exception, Exception Encapsulation and Enrichment, Assertions

#### **UNIT IV:**

##### **Objective: Understanding of Thread concepts and I/O in Java**

**MultiThreading** : java.lang.Thread, The main Thread, Creation of new threads, Thread priority, Multithreading- Using isAlive() and join(), Synchronization, suspending and Resuming threads, Communication between Threads

**Input/Output:** reading and writing data, java.io package

#### **UNIT V:**

##### **Objective: Being able to build dynamic user interfaces using applets and Event handling in java**

**Applets-** Applet class, Applet structure, An Example Applet Program, Applet Life Cycle, paint(),update() and repaint()

**Event Handling** -Introduction, Event Delegation Model, java.awt.event Description,Sources of Events, Event Listeners, Adapter classes, Inner classes

#### **UNIT VI:**

##### **Objective: Understanding of various components of Java AWT and Swings and writing code snippets using them**

###### **Abstract Window Toolkit**

Why AWT?, java.awt package, Components and Containers, Button, Label, Checkbox, Radio buttons, List boxes, Choice boxes, Text field and Text area, container classes, Layouts, Menu, Scroll bar

###### **Swing:**

Introduction , JFrame, JApplet, JPanel, Components in swings, Layout Managers, JList and JScroll Pane, Split Pane, JTabbedPane, Dialog Box

Pluggable Look and Feel

#### **Text Books:**

1. The Complete Refernce Java, 8ed, Herbert Schildt, TMH
2. Programming in JAVA, Sachin Malhotra, Saurabh choudhary, Oxford.
3. JAVA for Beginners, 4e, Joyce Farrell, Ankit R. Bhavsar, Cengage Learning.
4. Object oriented programming with JAVA, Essentials and Applications, Raj Kumar Bhuyya, Selvi, Chu TMH

5. Introduction to Java programming, 7<sup>th</sup> ed, Y Daniel Liang, Pearson

**Reference Books:**

1. JAVA Programming, K.Rajkumar.Pearson
2. Core JAVA, Black Book, Nageswara Rao, Wiley, Dream Tech
3. Core JAVA for Beginners, Rashmi Kanta Das, Vikas.
4. Object Oriented Programming Through Java, P. Radha Krishna, Universities Press

## II Year – II SEMESTER

T	P
3+1	0

### ADVANCED DATA STRUCTURES

(Note: C++ and Java implementation is not included in the syllabus)

**Objectives:** Exposed to hashing approaches, variants of trees, heaps, queues, implementation of graph algorithms, analysis of sorting algorithms with respect to bounds and file organizations and operations

#### Unit I :

**Objectives:** Comprehensive understanding of dictionaries, hashing mechanism which supports faster retrieval and skip lists

**Dictionaries :** Sets, Dictionaries, Hash Tables, Open Hashing, Closed Hashing (Rehashing Methods), Hashing Functions( Division Method, Multiplication Method, Universal Hashing), Skip Lists, Analysis of Skip Lists. (Reference 1)

#### Unit II :

**Objectives:** Illustration of Balanced trees and their operations

AVL Trees: Maximum Height of AVL Tree, Insertions and Deletions. 2-3 Trees : Insertion, Deletion.

#### Unit III :

**Objectives:** Concentration on heaps, queues and their operations

#### Priority Queues :

Binary Heaps : Implementation of Insert and Delete min, Creating Heap.

Binomial Queues : Binomial Queue Operations, Binomial Amortized Analysis, Lazy Binomial Queues

#### Unit IV :

**Objectives:** Detailed knowledge of nonlinear data structures and various algorithms using them

**Graph algorithms :** Minimum-Cost Spanning Trees- Prim's Algorithm, Kruskal's Algorithm Shortest Path Algorithms: Dijkstra's Algorithm, All Pairs Shortest Paths Problem: Floyd's Algorithm, Warshall's Algorithm,

#### Unit V :

**Objectives:** Analysis of complexities for various sorting techniques along with their lower bounds

**Sorting Methods :** Order Statistics: Lower Bound on Complexity for Sorting Methods: Lower Bound on Worst Case Complexity, Lower Bound on Average Case Complexity, Heap Sort, Quick Sort, Radix Sorting, Merge Sort.

## **Unit VI :**

**Objectives:** Illustration of tries which share some properties of table look up, various issues related to the design of file structures

**Pattern matching and Tries :** Pattern matching algorithms- the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm

Tries: Definitions and concepts of digital search tree, Binary trie, Patricia , Multi-way trie

**File Structures:** Fundamental File Processing Operations- opening files, closing files, Reading and Writing file contents, Special characters in files.

Fundamental File Structure Concepts- Field and record organization, Managing fixed-length, fixed-field buffers.

( Reference 5)

### **Text Books :**

1. Data Structures, A Pseudocode Approach, Richard F Gilberg, Behrouz A Forouzan, Cengage.
2. Fundamentals of DATA STRUCTURES in C: 2<sup>nd</sup> ed, , Horowitz , Sahani, Anderson-freed, Universities Press
3. Data structures and Algorithm Analysis in C, 2<sup>nd</sup> edition, Mark Allen Weiss, Pearson

### **Reference Books:**

1. Web : <http://lcm.csa.iisc.ernet.in/dsa/dsa.html>
2. [http://utubersity.com/?page\\_id=878](http://utubersity.com/?page_id=878)
3. <http://freevideolectures.com/Course/2519/C-Programming-and-Data-Structures>
4. <http://freevideolectures.com/Course/2279/Data-Structures-And-Algorithms>
5. File Structures :An Object oriented approach with C++, 3<sup>rd</sup> ed, Michel J Folk, Greg Riccardi, Bill Zoellick
6. C and Data Structures: A Snap Shot oriented Treatise with Live examples from Science and Engineering, NB Venkateswarlu & EV Prasad, S Chand, 2010.

## **II Year – II SEMESTER**

<b>T</b>	<b>P</b>
<b>3+1</b>	<b>0</b>

### **COMPUTER ORGANIZATION**

**Objectives:** Comprehensive knowledge of computer system including the analysis and design of components of the system

#### **UNIT I :**

**Objectives:** Gives a view of computer system from user's perspective, representation of data

**BASIC STRUCTURE OF COMPUTERS :** Computer Types, Functional unit, Basic Operational concepts, Bus structures,

Data Representation: Data types, Complements, Fixed Point Representation. Floating – Point Representation. Other Binary Codes, Error Detection codes.

#### **UNIT II :**

**Objectives:** Understanding RTL, Micro operations, ALU, Organization of stored program computer, types of instructions and design of basic components of the system

**REGISTER TRANSFER LANGUAGE AND MICROOPERATIONS:** Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

**BASIC COMPUTER ORGANIZATION AND DESIGN :** Instruction codes, Computer Register Computer instructions, Timing and control, Instruction cycle, Memory – Reference Instructions. Input – Output and Interrupt, Design of basic computer, Design of Accumulator Logic.

#### **UNIT III :**

**Objectives:** Illustration of data paths and control flow for sequencing in CPUs, Microprogramming of control unit of CPU

**CENTRAL PROCESSING UNIT :** General Register Organization, STACK organization. Instruction formats. Addressing modes. DATA Transfer and manipulation. Program control. Reduced Instruction set computer.

**MICRO PROGRAMMED CONTROL :** Control memory, Address sequencing, micro program example, design of control unit

#### **UNIT IV :**

**Objectives:** Illustration of algorithms for basic arithmetic operations using binary and decimal representation

**COMPUTER ARITHMETIC :** Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating –



point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

#### **UNIT V :**

**Objectives: Description of different parameters of a memory system, organization and mapping of various types of memories**

**THE MEMORY SYSTEM :** Memory Hierarchy, Main memory, Auxiliary memory, Associative Memory, Cache Memory, Virtual Memory.

#### **UNIT-VI**

**Objectives: Describes the means of interaction devices with CPU, their characteristics, modes and introduction multiprocessors.**

**INPUT-OUTPUT ORGANIZATION :** Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct memory Access.

**MULTI PROCESSORS :** Introduction, Characteristics or Multiprocessors, Interconnection Structures, Inter processor Arbitration.

#### **TEXT BOOKS :**

1. Computer System Architecture, M.Moris Mano, 3<sup>rd</sup> Edition, Pearson/PHI
2. Computer Organization , Carl Hamacher, Zvonks Vranesic, SafeaZaky, 5<sup>th</sup> Edition, McGraw Hill.
3. Computer Architecture a quantitative approach, John L. Hennessy and David A. Patterson, Fourth Edition Elsevier

#### **REFERENCES :**

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.

## II Year – II SEMESTER

T	P
3+1	0

### LANGUAGE PROCESSORS (50%FLAT+50%CD)

**Objectives:** *Describes how a programming language works, how input is converted into output from the machine hardware level and various phases of compiler*

#### UNIT I:

**Objectives:** *Delineation of various components of formal languages and grammars, regular expressions and equivalence of finite automata and regular expressions.*

#### Formal Language and Regular Expressions:

Languages, operations on languages, regular expressions (re), languages associated with (re), operations on (re), Identity rules for (re), Finite Automata: DFA, NFA, Conversion of regular expression to NFA, NFA to DFA. Applications of Finite Automata to lexical analysis

#### UNIT II:

**Objectives:** *Illustration of grammars and their role in compilers and various parsing techniques*

#### Context Free grammars and parsing:

Context free Grammars, Leftmost Derivations, Rightmost Derivations, Parse Trees, Ambiguity Grammars, Top-Down Parsing, Recursive Descent Parsers: LL(1) Parsers.

Rightmost Parsers: Shift Reduce Parser, LR (0) Parser, SLR (1) Parser, LR (1) & LALR (1) Parsers, Ambiguous Grammars

#### UNIT III:

**Objectives:** *Description of Syntax trees, its variants, language classifications*

#### Syntax Directed Translation:

Definitions, construction of Syntax Trees, S-attributed and L-attributed grammars, Intermediate code generation, abstract syntax tree, translation of simple statements and control flow statements.

#### Semantic Analysis:

Semantic Errors, Chomsky hierarchy of languages and recognizers, Type checking, type conversions, equivalence of type expressions.

#### UNIT IV:

**Objectives:** *Focus on various storage allocation schemes*

**Storage Organization:**

Storage language Issues, Storage Allocation, Storage Allocation Strategies, Scope, Access to Nonlocal Names, Parameter Passing, Dynamics Storage Allocation Techniques.

**UNIT V:**

*Objectives: Enforces various schemes for optimizing code*

**Code Optimization:**

Issues in the design of code optimization, Principal sources of optimization, optimization of basic blocks, Loop optimization, peephole optimization

**UNIT VI:**

*Objectives: Describes the role of code generator and its design issues*

**Code Generation:**

Issues in the design of code Generation, Machine Dependent Code Generation, object code forms, Register allocation and assignment, DAG representation of basic Blocks, Generating code from DAGs.

**Text Books:**

1. A Text Book on Automata Theory, Nasir S.F.B, P.K. Srimani, Cambridge university Press
2. Introduction to Automata Theory, Formal languages and computation, Shamalendu kandar,

3. Compilers Principles, Techniques and Tools, Aho, Ullman, Ravi Sethi, PEA
4. Introduction to theory of computation, 2<sup>nd</sup> ed, Michel sipser, CENGAGE
5. Principles of Compiler Design, A.V. Aho . J.D.Ullman; PEA

**Reference Books:**

1. Theory of Computer Science, Automata languages and computation , 2/e, Mishra, Chandra Shekaran, PHI

2. Theory of Computation , aproblem solving approach, kavi  
Mahesh, Wiley

## II Year – II SEMESTER

T P  
- 3

### ADVANCED DATA STRUCTURES LAB

1. To implement functions of Dictionary using Hashing (division method, Multiplication method, Universal hashing)
2. To perform various operations i.e, insertions and deletions on AVL trees
3. To perform various operations i.e., insertions and deletions on 2-3 trees.
4. To implement operations on binary heap.
5. To implement operations on graphs
  - i) vertex insertion
  - ii) Vertex deletion
  - iii) finding vertex
  - iv) Edge addition and deletion
6. To implement Depth First Search for a graph non recursively.
7. To implement Breadth First Search for a graph non recursively.
8. To implement Prim's algorithm to generate a min-cost spanning tree.
9. To implement Krushkal's algorithm to generate a min-cost spanning tree.
10. To implement Dijkstra's algorithm to find shortest path in the graph.
11. To implement pattern matching using Boyer-Moore algorithm.
12. To implement Knuth-Morris-Pratt algorithm for pattern matching.

**JAVA PROGRAMMING LAB**

1. Write a JAVA program to display default value of all primitive data types of JAVA
2. Write a JAVA program that displays the roots of a quadratic equation  $ax^2+bx+c=0$ . Calculate the discriminant D and basing on the value of D, describe the nature of roots.
3. Write a JAVA program to display the Fibonacci sequence
4. Write a JAVA program give example for command line arguments.
5. Write a JAVA program to sort given list of numbers.
6. Write a JAVA program to search for an element in a given list of elements (linear search).
7. Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
8. Write a JAVA program to determine the addition of two matrices.
9. Write a JAVA program to determine multiplication of two matrices.
10. Write a JAVA program to sort an array of strings
11. Write a JAVA program to check whether given string is palindrome or not.
12. Write a JAVA program for the following
  - 1. Example for call by value.
  2. Example for call by reference.
13. Write a JAVA program to give the example for 'this' operator. And also use the 'this' keyword as return statement.
14. Write a JAVA program to demonstrate static variables, methods, and blocks.
15. Write a JAVA program to give the example for 'super' keyword.
16. Write a JAVA program that illustrates simple inheritance.
17. Write a JAVA program that illustrates multi-level inheritance
18. Write a JAVA program demonstrating the difference between method overloading and method overriding.
19. Write a JAVA program demonstrating the difference between method overloading and constructor overloading.

20. Write a JAVA program that describes exception handling mechanism.
21. Write a JAVA program for example of try and catch block. In this check whether the given array size is negative or not.
22. Write a JAVA program to illustrate sub class exception precedence over base class.
23. Write a JAVA program for creation of user defined exception.
24. Write a JAVA program to illustrate creation of threads using runnable class.(start method start each of the newly created thread. Inside the run method there is sleep() for suspend the thread for 500 milliseconds).
25. Write a JAVA program to create a class MyThread in this class a constructor, call the base class constructor, using super and starts the thread. The run method of the class starts after this. It can be observed that both main thread and created child thread are executed concurrently
26. Write a JAVA program illustrating multiple inheritance using interfaces.
27. Write a JAVA program to create a package named pl, and implement this package in ex1 class.
28. Write a JAVA program to create a package named mypack and import it in circle class.
29. Write a JAVA program to give a simple example for abstract class.
30. Write a JAVA program that describes the life cycle of an applet.
  - Write a JAVA program to create a dialogbox and menu.
  - Write a JAVA program to create a grid layout control.
31. Write a JAVA program to create a border layout control.
32. Write a JAVA program to create a padding layout control.
33. Write a JAVA program to create a simple calculator.
34. Write a JAVA program that displays the x and y position of the cursor movement using Mouse.
35. Write a JAVA program that displays number of characters, lines and words in a text file.

## FREE OPEN SOURCE SOFTWARE (FOSS) LAB

### Objectives:

- To teach students various unix utilities and shell scripting

### Programs:

#### 1.

##### Session-1

- a)Log into the system
- b)Use vi editor to create a file called myfile.txt which contains some text.
- c)correct typing errors during creation.
- d)Save the file
- e)logout of the system

##### Session-2

- a)Log into the system
- b)open the file created in session 1
- c)Add some text
- d)Change some text
- e)Delete some text
- f)Save the Changes
- g)Logout of the system

#### 2.

- a)Log into the system
- b)Use the cat command to create a file containing the following data. Call it mytable use tabs to separate the fields.

1425	Ravi	15.65
4320	Ramu	26.27
6830	Sita	36.15
1450	Raju	21.86

- c)Use the cat command to display the file, mytable.
- d)Use the vi command to correct any errors in the file, mytable.
- e)Use the sort command to sort the file mytable according to the first field. Call the sorted file my table (same name)
- f)Print the file mytable
- g)Use the cut and paste commands to swap fields 2 and 3 of mytable. Call it my table (same name)
- h)Print the new file, mytable
- i)Logout of the system.

#### 3.

- 1) a)Login to the system



b) Use the appropriate command to determine your login shell

c) Use the `/etc/passwd` file to verify the result of step b.

d) Use the `who` command and redirect the result to a file called `myfile1`. Use the `more` command to see the contents of `myfile1`.

e) Use the `date` and `who` commands in sequence (in one line) such that the output of `date` will display on the screen and the output of `who` will be redirected to a file called `myfile2`. Use the `more` command to check the contents of `myfile2`.

2) a) Write a `sed` command that deletes the first character in each line in a

file.

b) Write a `sed` command that deletes the character before the last character in each line in a file.

c) Write a `sed` command that swaps the first and second words in each line in a file.

4. a) Pipe your `/etc/passwd` file to `awk`, and print out the home directory of each user.

b) Develop an interactive `grep` script that asks for a word and a file name and then tells how many lines contain that word.

c) Repeat

d) Part using `awk`

5. a) Write a shell script that takes a command –line argument and reports on whether it is directory, a file, or something else.

b) Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory.

c) Write a shell script that determines the period for which a specified user is working on the system.

6. a) 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.

b) Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

7. a) Write a shell script that computes the gross salary of an employee according to the following rules:

i) If basic salary is  $< 1500$  then  $HRA = 10\%$  of the basic and  $DA = 90\%$  of the basic.

ii) If basic salary is  $\geq 1500$  then HRA = Rs500 and DA = 98% of the basic

The basic salary is entered interactively through the key board.

b) Write a shell script that accepts two integers as its arguments and computes the value of first number raised to the power of the second number.

**8.** a) Write an interactive file-handling shell program. Let it offer the user the choice of copying, removing, renaming, or linking files. Once the user has made a choice, have the program ask the user for the necessary information, such as the file name, new name and so on.

b) Write shell script that takes a login name as command – line argument and reports when that person logs in

c) Write a shell script which receives two file names as arguments. It should check whether the two file contents are same or not. If they are same then second file should be deleted.

**9.** a) 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.

b) Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file.

c) Write a shell script to perform the following string operations:

i) To extract a sub-string from a given string.

ii) To find the length of a given string.

**10.** Write a C program that takes one or more file or directory names as command line input and reports the following information on the file:

i) File type      ii) Number of links      iii) Read, write and execute permissions  
iv) Time of last access

(Note : Use stat/fstat system calls)

**11.** Write C programs that simulate the following unix commands:

a) mv      b) cp      (Use system calls)

**12.** Write a C program that simulates ls Command (Use system calls / directory API)

13. Do the following Shell programs also

- 1) Write a shell script to check whether a particular user has logged in or not. If he has logged in, also check whether he has eligibility to receive a message or not
- 2) Write a shell script to accept the name of the file from standard input and perform the following tests on it
  - a) File executable   b) File readable       c) File writable
  - d) Both readable & writable
- 3) Write a shell script which will display the username and terminal name who login recently in to the unix system
- 4) Write a shell script to find no. of files in a directory
- 5) Write a shell script to check whether a given number is perfect or not
- 6) Write a menu driven shell script to copy, edit, rename and delete a file
- 7) Write a shell script for concatenation of two strings
- 3) Write a shell script which will display Fibonacci series up to a given number of argument
- 9) Write a shell script to accept student number, name, marks in 5 subjects. Find total, average and grade. Display the result of student and store in a file called stu.dat

Rules: avg $\geq$ 80 then grade A

Avg $<$ 80&&Avg $\geq$ 70 then grade B

Avg $<$ 70&&Avg $\geq$ 60 then grade C

Avg $<$ 60&&Avg $\geq$ 50 then grade D

Avg $<$ 50&&Avg $\geq$ 40 then grade E

Else grade F

- 10) Write a shell script to accept empno,empname,basic. Find DA,HRA,TA,PF using following rules. Display empno, empname, basic, DA,HRA,PF,TA,GROSS SAL and NETSAL. Also store all details in a file called emp.dat

Rules: HRA is 18% of basic if basic  $>$  5000  
otherwise 550

DA is 35% of basic

PF is 13% of basic

IT is 14% of basic

TA is 10% of basic

- 11) Write a shell script to demonstrate break and continue statements
- 12) Write a shell script to satisfy the following menu options
  - a. Display current directory path
  - b. Display today's date
  - c. Display users who are connected to the unix system
  - d. Quit
- 13) Write a shell script to delete all files whose size is zero bytes from current directory

- 14) Write a shell script to display string palindrome from given arguments
- 15) Write a shell script which will display Armstrong numbers from given arguments
- 16) Write a shell script to display reverse numbers from given argument list
- 17) Write a shell script to display factorial value from given argument list
- 18) Write a shell script which will find maximum file size in the given argument list
- 19) Write a shell script which will greet you "Good Morning", "Good Afternoon", "Good Evening" and "Good Night" according to current time
- 20) Write a shell script to sort the elements in a array using bubble sort technique
- 21) Write a shell script to find largest element in a array
- 22) Write an awk program to print sum, avg of students marks list
- 23) Write an awk program to display students pass/fail report
- 24) Write an awk program to count the no. of vowels in a given file
- 25) Write an awk program which will find maximum word and its length in the given input File
- 26) Write a shell script to generate the mathematical tables.
- 27) Write a shell script to sort elements of given array by using selection sort.
- 28) Write a shell script to search given number using binary search.
- 29) Write a shell script to find number of vowels, consonants, numbers, white spaces and special characters in a given string.
- 30) Write a shell script to lock the terminal.