

CS-31 PROGRAMMING IN C++

Maximum Marks: 50

Time: 3 hours

External: 40

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- to understand the limitation of predefined data types and the idea of defining new data types with functionality.
- to understand the new way of programming called object oriented programming.
- to understand various terms like class, object constructor, destructor, friend function, Inheritance, polymorphism and exception handling.

Learning Outcomes: At the end of this course, the student will be able to:

- To understand the limitation of predefined data types and the idea of defining new data types with functionality.
- To understand the new way of programming called object oriented programming.
- To understand various terms like class, object, constructor, destructor, friend function, Inheritance, polymorphism and exception handling.

UNIT-I

Introduction to Programming in C++: Object-Oriented Features of C++, comparing structure and Class, Data members and Member functions, concept of private and public section , Data Hiding & Encapsulation, Static Data Members and Member Functions, Inline Functions, Friend Functions, idea of Namespace, Comparing C with C++.

UNIT-II

Constructors & Destructors: default, parameterized and copy constructor Overloading: unary and binary operator overloading (through member function and friend function), function overloading
I/O operations in C++: concept of streams, input stream and output stream, cin and cout objects, formatted and unformatted I/O operations using manipulators and functions.

UNIT-III

Type Conversion: implicit and explicit conversion, type conversion in classes: basic to class, class to basic and class to class. Polymorphism: compile-time and run time Polymorphism

Inheritance: Types of Derivations, Forms of Inheritance, Roles of Constructors and Destructors in Inheritance.

UNIT-IV

Genericity in C++: Template Function, Template Class

Exception Handling: try, throw and catch constructs, rethrowing an exception, catch all Handlers.

Text Books:

- Herbert Schildt, C++, The Complete Reference, Tata McGraw-Hill
- Robert Lafore, Object Oriented Programming in C++, PHI
- Stephen Prata, C++ Primer Plus, Galgotia

Reference Books:

- Bjarne Stroustrup, The C++ Programming Language, Pearson.
- Balaguruswami, E., Object Oriented Programming In C++, Tata McGraw-Hill

CS-32 DATA STRUCTURE

Maximum Marks : 50

External:40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of six (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- The course is intended to get the students familiar with various types of data structure and different techniques to implement the data structures and their real-life applications.

Learning Outcomes: At the end of this course, the student will be able to:

- define Abstract data types, algorithms, complexity of algorithms, linear data structures, non-linear data structures, Searching, Sorting, Stack, Queues, Linked Lists, Trees, and Graphs.
- give Original examples of data structures and its types
- explain Sorting techniques, Searching methods, various operations on Stacks, Queues and Linked Lists. Traversal techniques in Trees.
- calculate (complexity of algorithm). Use array, stack, queue, linked list, tree, graph, linear search, binary search, bubble sort, selection sort,
- differentiate Data Structures, searching techniques, sorting techniques, analyze: time and space complexity.

UNIT-I

Data structure and Algorithm Preliminaries: Definitions, The concept of Data Structure, Basic Terminology: Elementary Data Organization, Data Structure Operations, Algorithms: Algorithmic Notations, Control Structures, Complexity of Algorithms-Big 'O' Notation, Time-Space Tradeoff.

Arrays: Introduction, Linear Arrays, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Concatenating Two Arrays, Insertion into a Linear Array, Deletion from a Linear Array, Largest/Smallest Element from an Array, Linear Search, Binary Search. Introduction to Multidimensional Arrays, Address Calculation of Elements of Arrays, Multiplication of Two Matrices.

UNIT-II

Sorting: Bubble Sort, Selection Sort.

Strings: Introduction to Strings, Basic Terminology, Storing Strings, String Operations,

Linked list: Introduction to Linked List, Representation of Linked List in Memory, Traversing a Linked List, Searching a Linked List, Insertion into a Linked List, Deletion from a Linked List, Header Linked Lists,

Two way Linked List (Doubly Linked List): Introduction, Inserting a node into Two way Linked List, Deleting a node from Two way Linked List, Introduction to Circular Linked List.

UNIT-III

Stack: Introduction to Stacks, Array Representation of Stacks, Operations on stack,: PUSH and POP, Representation of Stack as Linked List, Polish Notations and Reverse Polish Notation, Evaluation of Postfix Expressions, Transforming Infix Expressions into Postfix Expressions, Transforming Infix Expressions into Prefix Expressions, Introduction to Recursion.

Graphs: Introduction, Graph Theory Terminology, Sequential Representation of Graphs: Adjacency Matrix: Path Matrix , Linked Representation of a Graph, Shortest Path Algorithms.

UNIT-IV

Queues: Introduction to Queues, Operations on the Queues: Enqueue and Dequeue, Circular Queue, Double Ended Queue(DEQUE), Representation of a Queue as an Array, Representation of a Queue as Linked List,

Trees – Introduction, Basic Terminology, Binary Tree, Tree Representations using Array & Linked List, Binary Trees Traversing by Recursive procedures: Preorder In-order, & Post-order Traversal (NLR, LNR and LRN),

Introduction to Binary Search Tree (BST), Insertion and Deletion in BST (only illustrations)

Text Books:

1. Seymour Lipschutz, “Data Structures”, Tata McGraw- Hill Publishing Company Limited, Schaum’s Outlines, New Delhi.
2. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, Pearson Education, 2002.

Reference Books:

1. Trembley, J.P. And Sorenson P.G., “An Introduction to Data Structures With Applications”, Mcgrraw- Hill International Student Edition, New York.
2. Tanenbaum A.M., Langsam Y, Augenstien M.J., Data Structures using C & C++, Prentice Hall of India, 2002.
3. SartajSahni, Data structures, Algorithms and Applications in C++, University Press (India) Pvt.Ltd, 2e, Universities Press Orient Longman Pvt. Ltd.

CS-41 DATABASE MANAGEMENT SYSTEMS

Maximum Marks: 50

Time: 3 hours

External: 40

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- to understand the fundamental concepts, historical perspectives, current trends, structures, operations and functions of different components of Databases.
- to understand the types of integrity constraints in a relational database system and the concepts of SQL to create and access the database.
- to understand basic concepts of ER model and database design using normalization process.
- to understand transaction processing.

Learning Outcomes: At the end of this course, the student will be able to:

- Describe the basic concepts of database systems.
- Explain various data models and database system architectures.
- Write queries to access database using SQL.
- Design a database using normalization theory and explain the concepts of transaction processing.

UNIT-I

Basic Terminology: Traditional File Based Systems- File Based Approach-Limitations of File Based Approach, Database Approach: Characteristics of Database Approach, Database Management System (DBMS), Components of DBMS Environment, DBMS Functions and Components, Advantages and Disadvantages of DBMS.

Roles in the Database Environment: Database Administrator, Database Designers, Applications Developers and End Users.

Unit-II

Database System Architecture: Three Levels of ANSI/SPARC Architecture, Schemas and Instances, Data Independence: Logical and Physical Data Independence.

Classification of Database Management System: Centralized and Client Server architecture to DBMS.

Introduction to Data Models: Entity-Relationship Model, Entity Types, Entity Sets, Attributes Relationship Types, Relationship Instances and ER Diagrams, Network Model, Hierarchical Model, Relational Model.

Unit-III

Relational Model: Relational Model Terminology, Relational Data Structure, Database Relations, Properties of Relations, Keys, Domains, Integrity Constraints over Relations, Base Tables and Views. Relational Algebra & its various Operators (with respective SQL commands), Tuple and Domain Calculus

Unit-IV

Functional Dependency & Normalization: Data Redundancy and Update Anomalies. Functional Dependencies: Full Functional Dependencies and Transitive Functional Dependencies, Decomposition and Normal Forms (1NF, 2NF, 3NF & BCNF).

Text Books:

1. Database System Concepts by A. Silberschatz, H.F. Korth and S. Sudarshan, 3rd edition, 1997, McGraw-Hill, International Edition.
2. Introduction to Database Management system by Bipin Desai, 1991, Galgotia Pub.

Reference Books:

1. Fundamentals of Database Systems by R. Elmasri and S.B. Navathe, 3rd edition, 2000, Addison-Wesley, Low Priced Edition.
2. An Introduction to Database Systems by C.J. Date, 7th edition, Addison-Wesley, Low Priced Edition, 2000.
3. Database Management and Design by G.W. Hansen and J.V. Hansen, 2nd edition, 1999, Prentice-Hall of India, Eastern Economy Edition.
4. Database Management Systems by A.K. Majumdar and P. Bhattacharyya, 5th edition, 1999, Tata McGraw-Hill Publishing.
5. A Guide to the SQL Standard, Date, C. and Darwen, H. 3rd edition, Reading, MA: 1994, Addison-Wesley.

CS-42 SOFTWARE ENGINEERING

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Maximum Marks: 50

Time: 3 hours

External: 40

Internal: 10

Course Objectives:

- to study fundamental concepts in software engineering, SDLC, Software requirements specification, formal requirements specification and verification.
- to study the basic techniques for improving quality of software.
- to understand the fundamental principles of Software Project management & will also have a good knowledge of responsibilities of project manager and how to handle these.
- to understand the basic knowledge of Estimation model.

Learning Outcomes: At the end of this course, the student will be able to:

- Ability to analyze and specify software requirements.
- Ability to apply software engineering principles and techniques to develop large-scale software systems.
- Ability to plan and work effectively in a team.

Unit-I

Software and software engineering, Software characteristics, software crisis, Software Life Cycles, software engineering paradigms.

Unit-II

Software requirement analysis - structured analysis, object oriented analysis, software requirement specification, Tools for requirement analysis, Software cost estimation, Project scheduling, Personnel Planning, team structure.

Unit-III

Design and implementation of software - software design fundamentals, Structured design methodology and Object Oriented design, design verification, monitoring and control, coding.

Software Reliability - metric and specification, fault avoidance and tolerance, exception handling, defensive programming.

Unit-IV

Testing - Testing fundamentals, white box and black box testing, software testing strategies: unit testing, integration testing, validation testing, system testing, debugging.

Software maintenance - maintenance characteristics, maintainability, maintenance tasks, maintenance side effects.

Software configuration management: Project monitoring, Risk Management.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall.
2. Software Engineering, a book by Aggarwal K.K, Singh Yogesh, New Age International

Reference Books:

1. Pressman S. Roger, Software Engineering, Tata McGraw-Hill.
2. Jalote Pankaj, An integrated Approach to software, Engineering, Narosa Publishing House.

CS-51 COMPUTER NETWORKS

Maximum Marks: 50

External: 40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks (i.e. 8 marks)

Course Objectives:

- the basic concepts of Networking.
- It will also make the students familiar with the working of latest network technologies and applications.

Learning Outcomes: At the end of this course, the student will be able to:

- to gain the knowledge of Networking models, different media for transmission, addressing types and their difference, routing protocols.
- to gain knowledge of layered structure
- to working of different network technologies used in today's world.

UNIT-I

Introduction to Data Transmission: Representing Data as Analog Signals and Digital Signals, Data Encoding, Transmission Media: Guided and Wireless, Bandwidth, Capacity, Data Rate and Baud Rate, framing and errors, Asynchronous and Synchronous Communication.

Local Asynchronous Communication: Introduction, LAN Topologies and their Media Access Control (MAC) protocols.

UNIT-II

Long Distance Communication: Concept of Carrier Waves, Baseband, Broadband, Modulation, Demodulation, Switching (Packet Switching and Circuit Switching) and Multiplexing (FDM, TDM).

Reference Models: OSI and TCP/IP Models, various layers of Reference Models, communication between layers of Reference Models, various Protocols.

UNIT-III

Data Link Layer: Flow Control (Stop and Wait, Sliding Window), Error Detection (Parity Check and CRC), Error Correction (Stop and Wait, Go-back-N, Selective Reject)

UNIT-IV

Network Layer: Addressing Schemes (Logical Addressing and Address Mapping), Routing Techniques (Flooding, Shortest Path, Distance Vector); Congestion Control in Packet Switching Networks.

Network Security: Issues, Introduction to Cryptography; Symmetric –Keys and Public-Keys, Authentication and Authorization, Firewall.

Text Books:

1. William Stallings, Data and Computer Communications, PHI.
2. Behrouz A. Forouzan, Data Communications and Networking, McGraw Hill.
3. Douglas E. Comer and M. S. Narayanan, Computer Networks and Internets, Pearson.

Reference Books:

1. Andrew S. Tanenbaum, Computer Networks, Pearson.

CS-52(i) PYTHON PROGRAMMING

Maximum Marks: 50

External: 40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- to learn basics of Python programming
- to do Object Oriented Programming using Python
- Files Handling in Python
- GUI Programming and Databases operations in Python
- Network Programming in Python

Learning Outcomes: At the end of this course, the student will be able to:

- describe the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python
- express different Decision Making statements and Functions
- interpret Object oriented programming in Python
- understand and summarize different File handling operations
- design and develop Client Server network applications using Python

UNIT-I

Introduction Installing and Working With Python, Tokens, Operators, Data Types, Sequence Types, Mapping Types: Dictionaries, Tuples. Control Structures: Conditional Branching, Looping, Custom Functions, Strings.

UNIT-II

Sequences, iteration and recursion. Modules And Packages Modules And Packages, Overview Of Python Standard Library.

UNIT-III

OOPS Concepts and Classes In Python, Constructor, deconstructor, inheritance, polymorphism, Exception handling: catching and raising exceptions, custom exceptions,

UNIT-IV

Database programming. Numpy basics: Introduction, data types, arrays: indexing, slicing, shape, iteration, join, split, search, sort, filter.

Text Books:

- Mark Summerfield, "Programming in python
- A Complete Introduction to Python Programming".

Reference Books:

- Mark Lutz, "Learning Python", O Reilly, 4th Edition, 2009.
- Brian K. Jones, "Python Cookbook".
- Alex Martelli, "Python in a nutshell".

CS-52(ii) JAVA PROGRAMMING

Maximum Marks: 50

External: 40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- to understand the basic concepts and fundamentals of platform independent object oriented language.
- to demonstrate skills in writing programs using exception handling techniques and multithreading.
- to understand streams and efficient user interface design techniques.

Learning Outcomes: At the end of this course, the student will be able to:

- use the syntax and semantics of java programming language and basic concepts of OOP.
- develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.
- apply the concepts of Multithreading and Exception handling to develop efficient and error free codes.
- design event driven GUI and web related applications which mimic the real word scenarios.

Unit-I

JAVA and the Internet; The Java Programming Language and its characteristics, Java run-time environment; Java compiler; Java developers kit; running Java applications and Java applets.

Unit-II

JAVA Programming: Elements of Java: Data types, scalar data types, operators & expressions, control structures. Class, objects & methods, constructors, finalizer, visibility controls, array, string & Vectors, Inheritance, interfaces, packages multithreading, applet programming.

Unit-III

Exception handling-defining and throwing exceptions, creating your own exceptions..

Input/Output: streams, byte and character stream, the class Print stream, data streams String Tokenizes class, stream tokenizes.

Unit-IV

Delegation event model: AWT classes, AWT controls Layout Managers & meanness.

Text Books:

1. Compiler Networks & Internet, 2nd edition - Dongles E.Comet, Addison Wesley, 2000.
2. Programming the Internet with Java, revised edition- Darrel Ince & Adam Freeman, Addison Wesley, 2000.

Reference Books:

1. Programming with Java –E.Balaguruswami 2nd Edition, TNH – 1998.
2. The complete reference Java 2, 4th edition – Herbert Schildt (TMH, 2001).
3. A Programmer’s guide to Java certification, Mug Hal K.A., Rasmussen R.W., Addison Wesley, 2000.

CS-61 ARTIFICIAL INTELLIGENCE

Maximum Marks: 50

External: 40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- to study about intelligent agent and search methods.
- to study the concept of expert systems.
- to study about representing knowledge.
- to construct plan and methods for generating knowledge.

Learning Outcomes: At the end of this course, the student will be able to:

- understand what the AI is.
- apply search and knowledge representation techniques to solve AI problems.
- have ability to identify the solution of AI problems.

Unit-I

Introduction: background and history, overview of AI applications areas.

The predicate calculus: syntax and semantic for propositional logic and FOPL, clausal form, inference rules, resolution and unification.

Knowledge representation: network representation, associative network & conceptual graphs, structured representation, frames & scripts.

Unit-II

Search strategies: strategies for state space search, data-driven and goal driven search,

Search algorithms: uninformed search (depth-first, breadth-first, depth-first with iterative deepening) and informed search (hill climbing, best first, A* algorithm, mini-max etc.), computational complexity, properties of search algorithms, admissibility, monotonicity, optimality, dominance.

Unit-III

Production system: types of production system-commutative and non-commutative production systems, decomposable and non-decomposable production systems, control of search in production systems.

Rule-based expert systems: architecture, development, managing uncertainty in expert systems, Bayesian probability theory, Stanford certainty factor algebra, nonmonotonic logic and reasoning with beliefs, Fuzzy logic, Dempster/Shaffer and other approaches to uncertainty.

Unit-IV

Knowledge acquisition: types of learning, learning by automata, intelligent editors, learning by induction.

Genetic algorithms: problem representation, encoding schemes, operators: selection, crossover, mutation, replacement.

Text Books:

1. George F. Luger, William A. Stubblefield, Artificial Intelligence, The Benjamin/Cummings Publishing Company, Inc.
2. Dan W. Patterson, Introduction to Artificial Intelligence and Expert system, PHI.

Reference Books

1. Eugene Charniak, Drew McDermott, Introduction to Artificial Intelligence" Addison Wesley.
2. Wils J. Nilsson, Principles of Artificial Intelligence, Narosa Publishing house.
3. Jackson Peter, Introduction to Expert systems, 3e, Addison Wesley, 2000.

CS-62(i) ANDROID SOFTWARE DEVELOPMENT

Maximum Marks: 50

External: 40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- Covers introductory mobile application development for the Android Operating System using XML and Java.
- Includes developing simple applications that could run on Android phones and tablets. Covers Android application development phases, terminologies, application design, and coding.

Learning Outcomes: At the end of this course, the student will be able to:

- Install and configure Android application development tools.
- Design and develop user Interfaces for the Android platform.
- Save state information across important operating system events.
- Apply Java programming concepts to Android application development.

Unit-I

Android Application Development: Android Application Development: Getting started with Android, Mastering Android Development tools: Using Android Documentation, Debugging Applications with DDMS, Working with Android Emulator.

Unit-II

Building Android Applications: Designing typical Android Application, Using the Application Context, Working with Activities, Working with intents, Dialogs, Fragments, Logging application information.

Unit-III

Managing Application Resources: Working with Simple Resource values, Draw Table Resources, Layouts, Files; Configuring the Android Manifest file and basic application Settings.

Unit-IV

Development of Application: Registering activities, designating the launch activity, Managing Application permissions, designing an application framework.

Text Books:

1. Burton Michael, Android App Development for Dummies, Wiley, 2015.
2. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley India (Wrox), 2013.

Reference Books:

1. Ian F. Darwin, Android Cookbook Problems and Solutions for Android Developers, 2e, O'Reilly, 2017.

CS-62(ii) INTRODCUTION TO LINUX

Maximum Marks: 50

External: 40

Time: 3 hours

Internal: 10

Note: Examiner will be required to set Nine Questions in all. First Question will be compulsory, consisting of eight (objective type/short-answer type) questions covering the entire syllabus. In addition to that eight more questions will be set, two questions from each Unit. A candidate will be required to answer five questions in all, selecting one question from each unit in addition to compulsory Question No. 1. All questions will carry equal marks.

Course Objectives:

- to learn to develop software for Linux/UNIX systems.
- to learn the C language and get experience programming in C.
- to learn the important Linux/UNIX library functions and system calls.
- to understand the inner workings of UNIX-like operating systems.

Learning Outcomes: At the end of this course, the student will be able to:

- understanding the basic set of commands and utilities in Linux/UNIX systems.
- prepare students to develop software in and for Linux/UNIX environments.
- include basic operating system concepts, effective command line usage, shell programming.
- C language, programming development tools, system programming, network programming, and GUI programming.

Unit-I

Unix Operating System, Knowing Your Machine, Linux and GNU, The Unix Architecture, Unix/Linux features, Command Usage, General Purpose Utilities- cal, date, echo, printf, bc, script, Email basics, passwd, who, uname, tty, stty, file system. Linux Startup , Accounts, accessing Linux – starting and shutting processes, logging in and logging out.

Unit-II

Handling Ordinary Files - cat, cp, rm, mv, more, lp, file, wc, od, cmp, comm, diff, gzip and gunzip, tar, zip and unzip etc., tee

Basic File Attributes- ls, file permissions, Directory Permissions, vi editor, The process, More file Attributes.

Unit-III

Simple Filters- pr, head, tail, cut, paste, sort, uniq, tr etc.

Filters using Regular Expressions- grep and sed.

Unit-IV

awk- An advanced filter- Simple filtering, splitting into fields, variables and expressions etc

perl- The master Manipulator- perl preliminaries, chop function, string handling functions, split, join, for each.

Process Control- pipe, signal, kill process.

Text Books:

1. Sumitabha dass, Your Unix – The Ultimate Guide, TMH.
2. John Goerzen, Linux Programming Bible, IDG Books, New Delhi.

Reference Books:

1. Aho, Hopcroft and Ullman, the Design and Analysis of Computer Algorithms, Addison Wesley.
2. Yaswant Kanetkar, “Unix Shell Programming”, BPB Publication.