Savitribai Phule Pune University
S.Y.B.Sc. (Computer Science)
Computer Science Paper - I
Course Code: CS 231
Title : Data Structures and Algorithms – I

Teaching Scheme	No. of Credits	Examination Scheme
3 Lectures / week (50 mins duration)	2	IE : 15 marks UE: 35 marks

Prerequisites:
Basic knowledge of algorithms and problem solving
Knowledge of C Programming Language

Course Objectives
1. To learn the systematic way of solving problem
2. To understand the different methods of organizing large amount of data
3. To efficiently implement the different data structures
4. To efficiently implement solutions for specific problems
5. To apply linear data structures.

Course Outcomes: On completion of the course, student will be able to
1. To use well-organized data structures in solving various problems.
2. To differentiate the usage of various structures in problem solution.
3. Implementing algorithms to solve problems using appropriate data structures.

Course Contents

Chapter 1	**Introduction to Data Structures and Algorithm Analysis**	**4 lectures**

1.1 Introduction
- 1.1.1 Need of Data Structure
- 1.1.2 Definitions - Data and information, Data type, Data object, ADT, Data Structure
- 1.1.3 Types of Data Structures

1.2 Algorithm analysis
1.2.1 Space and time complexity, Graphical understanding of the relation between different functions of n, examples of linear loop, logarithmic,quadratic loop etc.
1.2.2 Best, Worst, Average case analysis, Asymptotic notations (Big O, Omega Ω, Theta θ), Problems on time complexity calculation.

Chapter 2	**Array as a Data Structure**	**10 lectures**

2.1 ADT of array, Operations
2.2Array applications - Searching
2.2.1 Sequential search, variations - Sentinel search, Probability search, ordered list search
- 2.2.2 Binary Search
- 2.2.3 Comparison of searching methods

2.3 Sorting Terminology- Internal, External, Stable, In-place Sorting
- 2.3.1 Comparison Based Sorting - Lower bound on comparison based sorting, Methods- Bubble Sort, Insertion Sort, Selection Sort, Algorithm design strategies - Divide and Conquer strategy, Merge Sort, Quick Sort, complexity analysis of sorting methods.

2.3.2 Non Comparison Based Sorting: Counting Sort, Radix Sort, complexity analysis.
2.3.3 Comparison of sorting methods

Chapter 3	**Linked List**	**10 lectures**

3.1 List as a Data Structure, differences with array.
3.2 Dynamic implementation of Linked List, internal and external pointers
3.3 Types of Linked List – Singly, Doubly, Circular
3.4 Operations on Linked List - create, traverse, insert, delete, search, sort, reverse, concatenate, merge, time complexity of operations.
3.5 Applications of Linked List – polynomial representation, Addition of two polynomials
3.6 Generalized linked list – concept, representation, multiple-variable polynomial representation using generalized list.

Chapter 4	**Stack**	**6 lectures**

4.1 Introduction
4.2 Operations – init(), push(), pop(), isEmpty(), isFull(), peek(), time complexity of operations.
4.3 Implementation- Static and Dynamic with comparison
4.4 Applications of stack
4.4.1 Function call and recursion, String reversal, palindrome checking
4.4.2 Expression types - infix, prefix and postfix, expression conversion and evaluation (implementation of infix to postfix, evaluation of postfix)
4.4.3Backtracking strategy - 4 queens problem (implementation using stack)

Chapter 5	**Queue**	**6 lectures**

5.1 Introduction
5.2 Operations - init(), enqueue(), dequeue(), isEmpty(), isFull(), peek(),time complexity of operations, differences with stack.
5.3 Implementation - Static and Dynamic with comparison
5.4 Types of Queue - Linear Queue, Circular Queue, Priority Queue, Double Ended Queue (with implementation)
5.5 Applications – CPU Scheduling in multiprogramming environment, Round robin algorithm

Reference Books:

1. Classic Data Structures-D. Samanta, Prentice Hall India Pvt. Ltd.
2. Fundamentals of Data Structures in C- Ellis Horowitz, SartajSahni,Susan Anderson-Freed, 2nd Edition, Universities Press.
3. Data Structures using C and C++-YedidyahLangsam, Moshe J. Augenstein, Aaron M. Tenenbaum, Pearson Education
4. Data Structures: A Pseudo code approach with C, Richard Gilberg ,Behrouz A. Forouzan, Cengage Learning.
5. Introduction to Data Structures in C-Ashok Kamthane, Pearson Education
6. Algorithms and Data Structures, Niklaus Wirth, Pearson Education

Savitribai Phule Pune University
S.Y.B.Sc. (Computer Science)
Computer Science Paper - I
Course Code: CS 241
Title : DATA STRUCTURES AND ALGORITHMS-II

Teaching Scheme	No. of Credits	Examination Scheme
3 Lectures / week (50 mins. duration)	02	IE : 15 marks UE: 35 marks

Prerequisites :

- Knowledge of C Programming Language
- Basic knowledge of algorithms
- Basic knowledge of linear data structures

Course Objectives

- To learn the systematic way of solving problems
- To design algorithms
- To understand the different methods of organizing large amount of data
- To efficiently implement the non-linear data structures

Course Outcomes: On completion of this course students will be able to

- Implementation of different data structures efficiently
- Usage of well-organized data structures to handle large amount of data
- Usage of appropriate data structures for problem solving

Course Contents

Chapter 1	**Tree**	**10 lectures**

1.1 Concept and Terminologies
1.2 Types of Binary trees - Binary tree, skewed tree, strictly binary tree, full binary tree, complete binary tree, expression tree, binary search tree, Heap
1.3 Representation – Static and Dynamic
1.4 Implementation and Operations on Binary Search Tree - Create, Insert, Delete, Search, Tree traversals– preorder, inorder, postorder (recursive implementation), Level-order traversal using queue, Counting leaf, non-leaf and total nodes, Copy, Mirror.
1.5 Applications of trees
1.5.1 Heap sort, implementation
1.5.2 Introduction to Greedy strategy, Huffman encoding (implementation using priority queue)

Chapter 2	**Efficient Search Trees**	**8 lectures**

2.1 Terminology: Balanced trees - AVL Trees, Red Black tree, splay tree, Lexical search tree -Trie
2.2 AVL Tree- concept and rotations
2.3 Red Black trees - concept, insertion and deletion.
2.4 Multi-way search tree - B and B+ tree - Insertion, Deletion

Chapter 3	**Graph**	**12 lectures**

3.1 Concept and terminologies
3.2 Graph Representation –Adjacency matrix, Adjacency list, Inverse Adjacency list, Adjacency multilist
3.3 Graph Traversals – Breadth First Search and Depth First Search (with implementation)
3.4 Applications of graph

3.4.1 Topological sorting
3.4.2 Use of Greedy Strategy in Minimal Spanning Trees (Prims and Kruskals algorithm)
3.4.3 Single source shortest path - Dijkstra's algorithm
3.4.4 Dynamic programming strategy, All pairs shortest path - Floyd Warshall algorithm
3.4.5 Use of graphs in social networks

Chapter 4	**Hash Table**	**6 lectures**

4.1 Concept of hashing
4.2 Terminologies – Hash table,Hash function, Bucket, Hash address, collision, synonym, overflow etc.
4.3 Properties of good hash function
4.4 Hash functions : division function, MID square , folding methods
4.5 Collision resolution techniques
4.5.1 Open Addressing - Linear probing, quadratic probing, rehashing
4.5.2 Chaining - Coalesced , separate chaining

Reference Books:

1. Fundamentals of Data Structures in C- Ellis Horowitz, SartajSahni,Susan Anderson-Freed, 2nd Edition, Universities Press.
2. Data Structures using C and C++-YedidyahLangsam, Moshe J. Augenstein, Aaron M. Tenenbaum, Pearson Education
3. Data Structures: A Pseudo code approach with C, Richard Gilberg ,Behrouz A. Forouzan, Cengage Learning.
4. Introduction to Data Structures in C-Ashok Kamthane, Pearson Education
5. Algorithms and Data Structures, Niklaus Wirth, Pearson Education
6. Introduction to Algorithms—Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein--MIT Press
7. Fundamentals of Computer Algorithms-- Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, Universities Press
8. The Algorithm Design Manual - Steven S Skiena, Springer