

## **SEMESTER-II Data Structures**

### **Core-III**

#### **Course Objectives:**

- To understand different ways of organizing data in computer's memory.
- To learn different operations on data structures.
- To explore different applications of data structures.

#### **Learning Outcomes:**

Upon completion of this course, students will be able to:

1. Learn about data structures and the use of array
2. Create linked lists and perform insertion/deletion operations on them
3. Represent Stack and Queue in the memory and learn their applications
4. Learn the use of various non-linear data structures and their applications

#### **Unit-I:**

Introduction to Data Structures: Definition, Concepts, Classification of Data Structures.

Array: Introduction, One-Dimensional Array, Memory representation, Operations: Traversing, Searching, Insertion, Deletion, Merge. Two-Dimensional Array & Memory Representation, Multidimensional Array. Linear Search versus Binary Search, Sorting: Selection Sort, Bubble Sort.

#### **Unit-II:**

Linked Lists: Definition, Single Linked List, Memory representation, Operations: Traversing, Searching, Insertion, Deletion and Merge. Double Linked List, Operations: Insertions, Deletion.

Circular, Double Circular Linked list, Operations: Traversing, Insertion. Applications of Linked List, Sparse Matrix and Polynomial representations.

#### **Unit-III:**

Stack: Definition, Representation: Array and Linked List representations, Operations: PUSH, POP, STATUS. Applications: Evaluation of Arithmetic Expressions: Notations, Infix to Postfix Conversion, Evaluation of Postfix expression. Recursion (Factorial and Fibonacci), Tower of Hanoi.

Queues: Definition, Representation: Array and Linked List representations, Operations: Enqueue, Dequeue. Structures of Queue: Circular, Deque and Priority Queue. Applications of Queue

**Unit-IV:**

Trees: Definition, Terminologies, Binary Tree: Properties, Representations (Linear and Linked List representations). Operations: Traversal (Inorder, Preorder, Postorder), Search. Introduction to Binary Search Tree, AVL tree, M-Way Search Tree. Applications of Trees. Graph: Definition, Terminologies, Representations (Set, Linked List, Matrix), Operations: Traversal (BFS, DFS). Applications of Graphs.

**Text book:**

Classic Data Structure, D. Samanta, PHI, 2/ed.

**Reference Books:**

- ✓ *Ellis Horowitz, Sartaj Sahni, "Fundamentals of Data Structures", Galgotia Pubs.*
- ✓ *Sastry C.V., Nayak R, Ch. Rajaramesh, Data Structure & Algorithms, I. K. International*
- ✓ *Publishing House Pvt. Ltd, New Delhi.*

**Core III - Lab: Data Structures**

Write a C Program for the followings

1. To search an element and print the total occurrences in the array.
2. To insert and delete elements into/from appropriate position in an array.
3. To perform Binary Search.
4. To perform Bubble sort.
5. To perform Selection sort.
6. To implement linear linked list and perform operations such as traverse, search, insert, delete, and reversing the list.
7. To implement circular linked list and perform operations such as node insert and delete.
8. To implement double linked list and perform operations such as node insert and delete.
9. To represent a Sparse Matrix using linked list.
10. Polynomial representation using linked list.
11. Array and Linked list implementations of Stack and perform operations such as push, pop and status.
12. Linked list implementation of Queue and perform operations such as enqueue and dequeue.
13. Linked list implementation of Circular Queue.

14. To implement a Binary Search Tree.
15. To perform tree traversal operations.
16. To implement adjacency matrix for a given graph.
17. To perform BFS and DFS traversal.