

DATA STRUCTURES LAB

Course Code: 19CS1102

L	T	P	C
1	0	3	2.5

Prerequisites: Computer Programming

Course outcomes: At the end of the course, a student will be able to

CO1: Develop programs using recursive functions.

CO2: Implement stacks and queues.

CO3: Develop Programs for searching, sorting and hashing techniques.

CO4: Implement different types of trees.

CO5: Apply concepts of graphs.

List of Programs:

1. Write a program that uses recursive function to:

i) Compute factorial of a given number ii) Solve the towers of Hanoi problem iii) GCD

2. Write a program to implement the following search algorithms:

i) Linear Search ii) Binary Search iii) Fibonacci Search.

3. Write a program to implement the following sorting algorithms

i) Bubble Sort ii) Insertion Sort iii) Quick Sort iv) Merge Sort.

4. Write a program to implement different types of hash functions.

5. Write a program that implements the following data structures using arrays:

i) Stack ii) Queue.

6. Write a program to implement the following Stack applications

i) Factorial ii) Evaluations of postfix expression iii) number conversion.

7. Write a program to implement the following types of queues

i) Priority Queue ii) Circular Queue.

8. Write a program to implement the following types of Lists

i) Singly linked list ii) Doubly linked list.

9. Write a program to implement binary tree using arrays and to perform binary tree traversals

i) inorder ii) postorder iii) preorder.

10. Write a program to perform the following operations using linked lists:
 - i) Insert an element into a binary search tree.
 - ii) Delete an element from a binary search tree.
 - iii) Search for a key element in a binary search tree.
11. Write a program to perform the following operations using linked lists:
 - i) Insert an element into an AVL tree. ii) Delete an element from an AVL tree.
12. Write a program for the implementation of BFS and DFS for a given graph.

Additional Programs/Beyond Syllabi:

1. Write a program to implement double stack.
2. Write a program to implement Dijkstra's algorithm for the single source shortest path problem.
3. Write a program to reverse a linked list.
4. Write a program to demonstrate working of generic linked list.
5. Write a program to check whether an expression consists of balanced parenthesis or not using stack.
6. Write a program to interchange two adjacent nodes in a circular linked list.
7. Write a program to implement Topological sorting technique.

References:

1. Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, *Fundamentals of Data structures in C++*, 2nd edition, University Press (India) Pvt.Ltd, 2008.
2. G A V PAI, *Data Structures and Algorithms, Concepts, Techniques and Applications*, Volume-1, 1st Edition, Tata McGraw-Hill, 2008.
3. Richard F. Gilberg & Behrouz A. Forouzan, *Data Structures, A Pseudo code Approach with C*, 2nd Edition, Cengage Learning India Edition, 2007.