ADVANCED DATA STRUCTURES LAB

Course Code: 13IT2109 **L P C**
0 3 2

Pre requisites:

1. Computer Programming through C.
2. Data Structures.

Course Educational Objectives:

The main objective is to deliver the programming techniques which are advanced for solving the problems regarding memory allocation, utilization and also object oriented features. Student can get the grip on:

1. Advanced programming
2. Solving the problems regarding large data structures like stack, queue, trees, graphs
3. The Top-Down Splay Trees, Red-Black Trees, Deterministic Skip lists, AA-Trees, Treaps, *k*-d Trees.
4. Dynamic equivalence problem.
5. Binomial Queues, Skew Heaps, Fibonacci Heaps.

Course Outcomes:

At the end of the course **s**tudent will be able to

1. Get knowledge on how to develop algorithms, operations on queues and stacks.
2. Understand different searching, different sorting methods and graphs, B-Trees, Splay Trees.

Implement the following using C/C++/Java

1. Write a program to perform the following operations on singly linked list.
   i) Creation ii) Insertion iii) Deletion iv) Traversal.
2. Write a program to perform the following operations on doubly linked list.
   i) Creation ii) Insertion iii) Deletion iv) Traversal in both ways
3. Write a program that implements stack (its operations) using
   i) Arrays ii) linked list
4. Write a programs that implements Queue (its operations) usin
   i) Arrays ii) linked list
5. Write C program that implements the Quick sort method to sort a given list of integers in ascending order.

6. Write C program that implement the Merge sort method to sort a given list of integers in ascending order.
7. Write C program that implement the SHELL sort method to sort a given list of integers in ascending order.
8. Write a program to perform the following: i) Creating a Binary Tree of integers
   ii) Traversing the above binary tree in preorder, inorder and postorder.
9. Write a C program to perform the following:
   i) Creating a AVL Tree of integers
   ii) Traversing the above binary tree in preorder, inorder and postorder.
10. Write a C program that uses functions to perform the following:
   i) Creating a SplayTree of integers
   ii) Traversing the above binary tree in preorder, inorder and postorder.
11. Write a C program to perform the following:
   i) Creating a B-Tree of integers
   ii) Traversing the above binary tree in preorder, inorder and postorder.
12. Write a program that implements Kruskals algorithm using a disjoint set data structure. The program takes as input a file (data.txt), in which each line either represents a vertex or an edge. For the edge lines, the first integer on that line representing the starting vertex, the second the ending vertex, and the third the weigh of the edge. Use this file to construct, line by line, the graph upon which Kruskal's algorithm will be run (do NOT hardcode this graph!).
13. Write a program to simulate various graph traversing algorithms.
14. Write a program to find the minimal spanning tree of a graph using the Prim's algorithm. The program should be able to read in the weight matrix of a graph and produce the minimal spanning tree Generate weight matrices (using a random number generator) with a large number of nodes and estimate the time complexity of the algorithm.

15. Write a program to find the closest pair of points using a divide and conquer strategy. Use the random number generator to generate a large number of points in a unit square as input to the algorithm. Test the correctness of the algorithm by using a brute force method.

16. Use dynamic programming to find the optimal binary search tree for a given set of numbers together with their probabilities. Remember that the numbers may be generated in any order, so, a presorting step is also required.

Text Books:

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C++*, 3rd Edition, Pearson Education, 2007.

References:

1. Sartaj Sahni, *Data Structures Algorithms and Applications in C++*, 2nd Edition, Universities Press, 2007.

2. Ellis Horowitz, Sartaj Sahni, Rajasekharan, *Fundamentals of Algorithms*, 2nd Edition, Universities Press, 2009.

3. Aho V Alfred, Hapcroft E John, Ullman D Jeffry, *Data Structures and Algorithms*, 1st Edition, Pearson Education, 2002.

4. Adam Drozdek,Thomson, *Data Structures and Algorithms in JAVA*, 3rd Edition, Cengage Learning , 2008.

5. Horowitz, Sahni, Mehta, *Fundamentals of Data Structures in C++*, 2nd Edition, Universities Press , 2007.

Web references:

www.nptel.iitm.ac.in/video.php?subjectid=106102064