Design and Analysis of Algorithms Lab

Detailed Contents:

All the problems have to be implemented either writing C programs or writing C++ programs.
Elementary Problems:

1. Using a stack of characters, convert an infix string to a postfix string.
2. implement polynomial addition using a single linked list
3. Implement insertion, deletion, searching of a BST, Also write a routine to draw the BST horizontally.
4. Implement binary search and linear search in a program
5. Implement heap sort using a max heap.
6. Implement DFS/ BFS routine in a connected graph
7. Implement Dijkstra’s shortest path algorithm using BFS
8. Greedy Algorithm (Any Two)
   i. Given a set of weights, form a Huffman tree from the weight and also find oot the code corresponding to each weight.
   ii. Take a weighted graph as an input, find out one MST using Kruskal/ prim’s algorithm
   iii. Given a set of weight and an upper bound M – Find out a solution to the Knapsack problem
9. Divide and Conquer Algorithm (any Two)
   i. Write a quick sort routine, run it for a different input sizes and calculate the time of running. Plot in graph paper input size verses time.
   ii. Implement two way merge sort and calculate the time of sorting
   iii. Implement Strasseem’s matrix multiplication algorithm for matrices whose order is a power of two.
10. Dynamic programming
    i. Given two sequences of character, find out their longest common subsequence using dynamic programming