**Data Structures and Algorithms Assignment Set – 3**

3.1 Write a C program to solve the n-queens problem. The value of n may be between 8 and 1000. We are interested in finding the first achieved solution only. (not all the solutions)

3.2 Define the ADTs for Stack and Queue.

a) Implement the ADTS using Array.

b) Implement the ADTs using Linked List. Use the data structures for Linked List already developed in Assignment II as header files.

c) Develop a program for converting a string containing an infix expression to a string containing the corresponding postfix expression. Both the strings are terminated by a sentinel ‘#’.

3.3 Develop a recursive program for generating all the permutations of the letters ‘a’ to ‘f’ with repetitions taking 3, 4, 5 and 6 characters.

3.4 Define an ADT for a threaded Binary Tree.

Implement the ADT including the three traversals.

3.5 Define an ADT for Binary Search Tree.

a) Implement the ADT.

b) Write functions for Print-preorder, Print-inorder, and Print-postorder, so that we can check the trees after insertion and deletion operations.

3.6 Define an ADT for Graph.

a) Implement Kruskal’s algorithm for finding the Minimun-Cost Spanning Tree of a Graph with positive edge-weights.

b) Implement Dijkstra’s algorithm for finding the shortest paths from a start node to all the other nodes in a graph with positive edge-weights.

c) Write a program to find out the Transitive Closure Matrix for a directed graph.

-----------------------X-------------------------