CSMI 11 Assignment Programs – Jan- May 2023

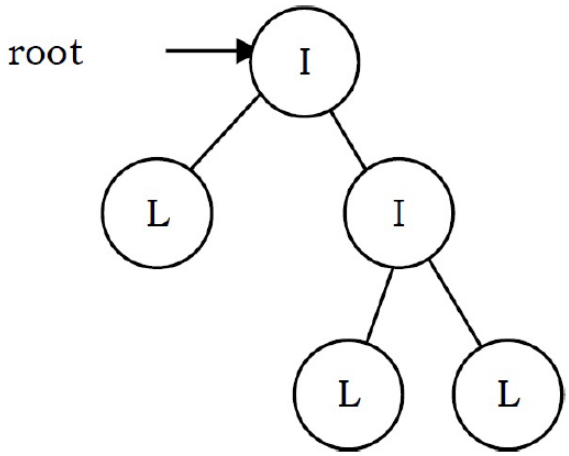
(All 15 programs must be implemented using either C or C++ or Java)

1. Implement Queue using two stacks. (Note: the functions such as Enqueue, Dequeue and Display need to be implemented.)
2. Implement circular descending priority queue. (Note: the functions such as Enqueue, Dequeue and Display need to be implemented.)
3. Given two unordered circular doubly linked lists, write a program for the printing common elements of them.
4. Given a singly linked list, write a program to find
   （i） the last element from the beginning whose $n\%k == 0$,
   （ii）the first from the end whose $n\%k == 0$,

where n is the number of elements in the list and k is an integer constant. For example, if $n = 19$ and $k = 3$ then (i) 18^{th} node should be returned. (ii) 16^{th} node should be returned.

5. Given a circular linked list with even and odd numbers, write a program to make changes to the list in such a way that all even numbers appear at the beginning.
6. Given a BST and two integers (minimum and maximum integers) as parameters, write a program to remove (prune) elements that are not within that range.
7. Give an algorithm for checking the existence of path with given sum. That means, given a sum, check whether there exists a path from root to any of the nodes.
8. Given a tree with a special property where leaves are represented with ‘L’ and internal node with ‘I’. Also, assume that each node has either 0 or 2 children. Given preorder traversal of this tree, write a program to construct the tree and display it in the tree format as shown below.

Example: Given preorder string => ILILL



9. Write a program for finding the maximum-weight spanning tree in a graph
10. Write a program to return the reverse of the directed graph (each edge from *v* to *w* is replaced by an edge from *w to v*).
11. Write a program to implement Warshall's algorithm on weighted as well as unweighted graphs.
12. Given an array A[] consisting of 0's, 1's and 2's, Write a program to sort this array *A*[] using Quick Sort.
13. Write a program for finding the *kth* smallest element in min-heap.
14. Implement TSP problem using Dynamic Programming approach.
15. Implement Strassen's Matrix multiplication using Divide and Conquer approach.