[CSL202]-Data Structures **2024-25-M**

LAB-V

Date: **Aug 29, 2024**.

You need to upload your solutions of Q1 (Q2 is optional) to canvas portal before 05:35pm on Aug 29, 2024.

1. Write a program that builds a binary search tree (BST) and supports the following operations. You can assume that the key values are distinct and positive integers.
   (a) Find the minimum element
   (b) Find the maximum element
   (c) Preorder traversal
   (d) Postorder traversal
   (e) Inorder traversal
   (f) Insert an element
   (g) Delete an element
   (h) Find the successor of an element
   (i) Find the height of the BST.

2. Given a binary tree with integers as its keys. You need to test whether it is a correct Binary Search Tree (BST). Your input format is as follows: The first line contains the number of vertices n (vertices are numbered from 0 to $n-1$). The next n lines contain information about vertices $0, 1, \ldots, n-1$ in order. Each of these lines contains three integers $key(i)$, $left(i)$ and $right(i)$ where $key(i)$ is the key of the i-th vertex, $left(i)$ is the index of the left child of the i-th vertex, and $right(i)$ is the index of the right child of the i-th vertex. If i doesn't have left or right child (or both), the corresponding $left(i)$ or $right(i)$ (or both) will be equal to -1. Your output should be YES if the given binary tree is a binary search tree. Otherwise ouput NO.

   (Hint: You can first start with case where all keys are distinct. Then you can go for a more general case, where binary search tree may contain equal keys. For the general case, the definition of the binary search tree is as follows: for any node of the tree, if its key is x, then for any node in its left subtree its key must be strictly less than x, and for any node in its right subtree its key must be greater than or equal to x).

   Sample Input 1 :

   3
   2 1 2
   1 -1 -1
   3 -1 -1
   Output: YES

   Sample Input 2:

   3
   1 1 2
   2 -1 -1

3 -1 -1
Output: NO

Sample Input 3:

4
4 1 -1
2 2 3
1 -1 -1
5 -1 -1
Output: NO