Programming With Rana Waqas	**Assignment No. 4** **Semester: Spring 2022** **SECP2043 – Data Structures and Algorithm**	**Total Marks: 20** **Due Date: 19/06/2022** **Due Time: 11:59 PM**

Instructions

Please read the following instructions carefully before submitting the assignment solution:

It should be clear that your assignment will not get any credit/marks if:

- Assignment is submitted after the due date.
- Submitted assignment does not open or file is corrupt.
- Assignment is copied (From internet/students).

Recommended Tools

- CodeBlocks C++
- Visual Studio

Objectives:

To enable the students, to understand and practice the concepts of:

- Construction of binary search tree.
- Post-order traversal of binary search tree.
- Pre-order traversal of binary search tree.
- Store Section information at each node of tree.
- Search specific section strength
- Delete specific course

Assignment Submission Instructions

You have to submit only the **code (.cpp)** file on the assignments interface from your CMS account.

Assignment submitted in any other format will not be accepted and will be scaled with **zero marks**. No excuse will be accepted for submitting the solution file in any other format.

For any query related to the assignment, please contact waqasali@ucp.edu.pk.

Problem Statement:

Write a C++ program to fulfill the following requirements.

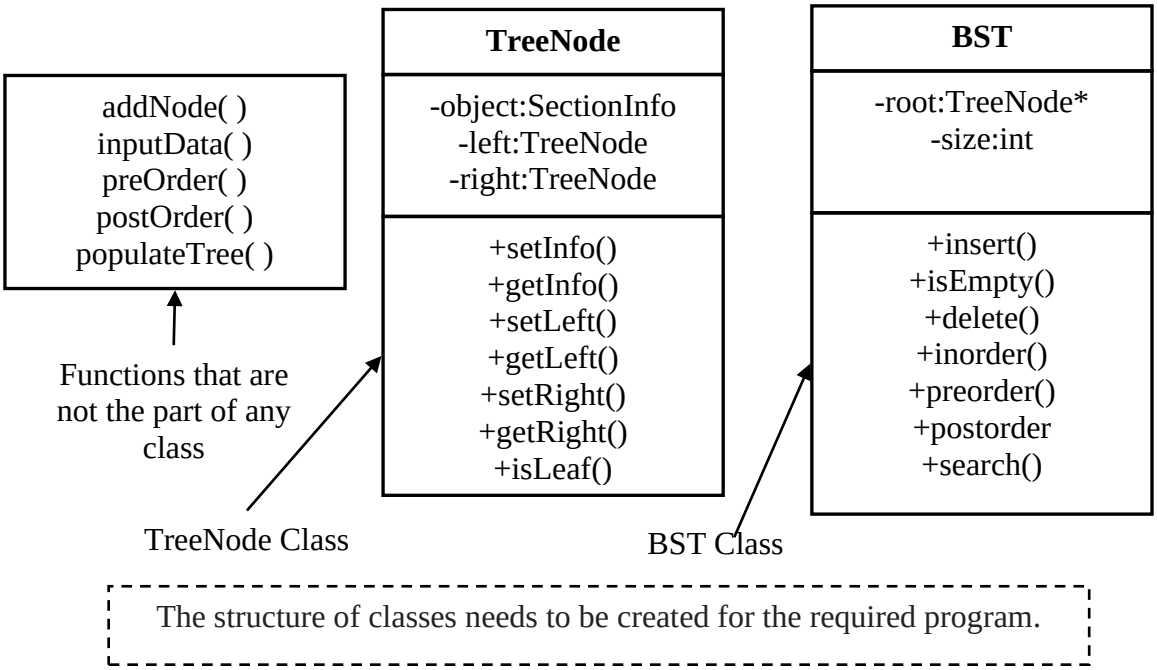
1. Build binary search tree (BST) from data taken by user through keyboard.
2. Traverse and display the BST using post-order traversal.
3. Traverse and display the BST using pre-order traversal.
4. Delete all elements from specific level
5. Search specific data from tree

Given Data/Array: 55, 78, 87, 25, 28, 56, 2, 99, 41, 13, 9, 1, 70, 16, 79

Steps to follow while developing the required program.

- Create a TreeNode class to construct nodes for binary search tree.
- Create Heap class to construct min-heap.
- Define non-member function populateTree( ) to create BST.
- Define non-member function add( ) to insert a node into BST.
- Define non-member function postOrder( ) for post-order traversal of BST.
- Define non-member function preOrder( ) for pre-order traversal of BST.
- Define non-member function inputData( ) to get data of BST from user.
- Instead of data entry one by one, your program should allow inserting whole data like shown in the sample video.

The structure of classes which you need to create in the required program is given below.



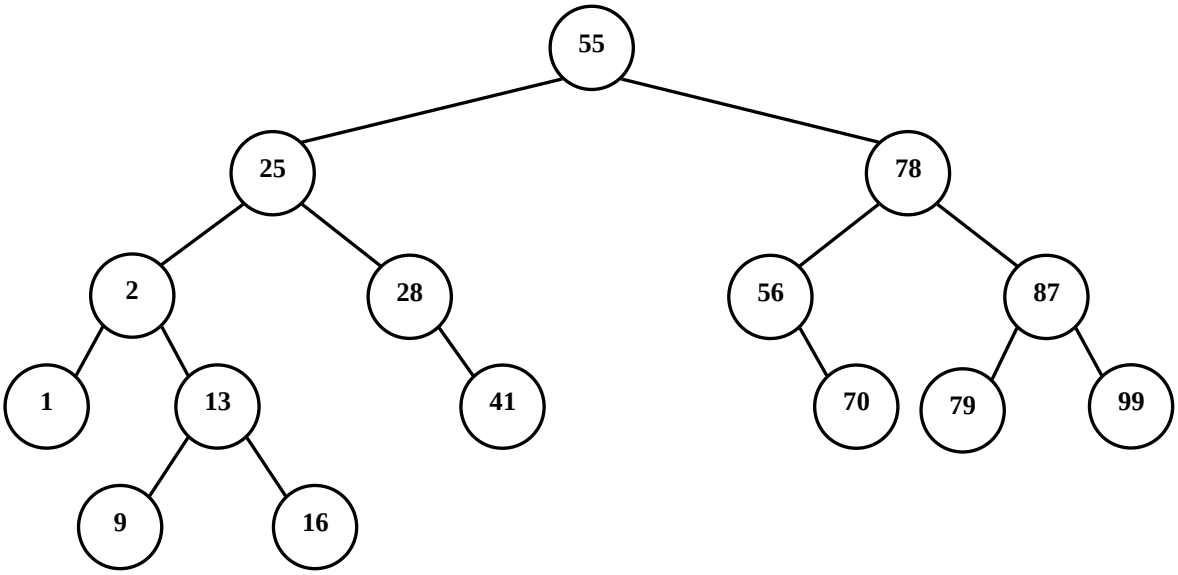
SectionInfo class will have following data members:

- Section name (e.g. N1, N2, C4 …)
- Course name (e.g. DSA, OOP, Machine Learning, …)
- Course instructor name (e.g. rana waqas ali)
- Section student strength (e.g. 45, 7, 59 …)

Graphical Solution

The graphical solution of required C++ program is given below. It will help you to compare and verify your program's output.

Binary Search Tree (BST) on the basis of just section strength, rest of the information is also present in every node: 55, 78, 87, 25, 28, 56, 2, 99, 41, 13, 9, 1, 70, 16, 79



Post-order Traversal of BST: 1, 9, 16, 13, 2, 41, 28, 25, 70, 56, 79, 99, 87, 78, 55

All other information of sections should be display with this traversal