**BRAC University**

**Department of Computer Science and Engineering**

**CSE 220: Data Structures**

**Task 1**

Implement a **MyBSTree** ADT that is a binary search tree ADT.

**Elements**

The elements in a binary search tree are of generic type TreeElement. Each TreeElement has an integer key that uniquely identifies the element. Besides this, TreeElements include additional data a right child and a left child (you are not allowed to use any other global references in the TreeElement class).

**Structure**

A BSTree ADT contains a root that is a reference to the root (a TreeElement) of the BSTree. For each tree element E in the Binary Search Tree, all the elements in E’s left sub-tree have keys that are less than E’s key and all the elements in E’s right sub-tree have keys that are greater than E’s key.

[You are not allowed to use any class variable other than root in the MyBSTreeADT.]

**Constructor**

**MyBSTree ( )**

Pre-condition:

None.

Post-condition:

This is the constructor of MyBSTree class. This constructor creates an empty Binary Search Tree.

**Methods in the Interface**

**void insert (TreeElement newElement)**

Pre-condition:

The newElement is not null.

Post-condition:

This method inserts newElement into a Binary Search Tree. If an element with the same key as newElement already exists in the tree, then it concludes the key already exists and do not insert the key in the Binary Search Tree.

**TreeElement retrieve (int searchKey)**

Pre-condition:

The tree is not empty.

Post-condition:

It searches the binary search tree for the TreeElement with key searchKey. If this TreeElement is found, then returns the TreeElement. Otherwise, returns null.

**TreeElement remove (int deleteKey)**

Pre-condition:

The tree is not empty.

Post-condition:

This method deletes the element with key deleteKey from a binary search tree, if it exists in the tree. Otherwise, returns null.

**void clear ( )**

Pre-condition:

None.

Post-condition:

This method removes all the elements from the binary search tree.

**boolean isEmpty ( )**

Pre-condition:

None.

Post-condition:

This method returns true if a binary search tree is empty. Otherwise, returns false.

**void showStructure ( )**

Precondition:

None.

Postcondition:

This method prints the keys of the Binary Search Tree in pre-order, in-order and post-order. If the list is empty, then outputs “Empty Tree”.