SOFTWARE TESTING

THE ROLE OF A TESTER IS TO BREAK THE SOFTWARE

Prepared For :
SIR MUKHTAIR ZAMIN

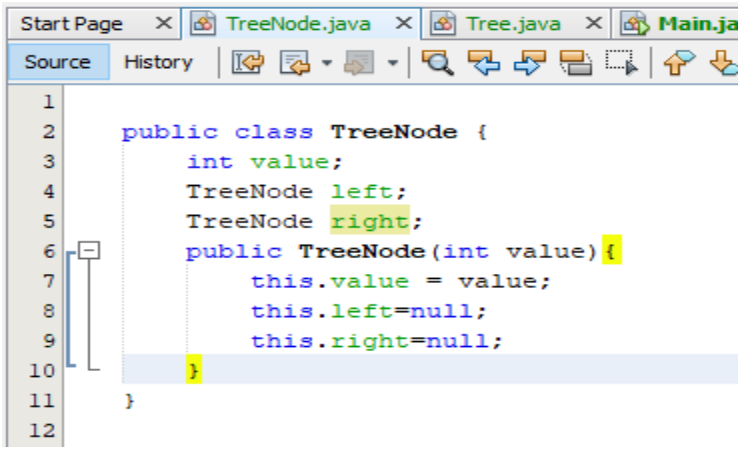
Prepared By :
BASIT IQBAL
FA21-BSE-050

Question:

Implement the Inorder Traversal of the Tree....?

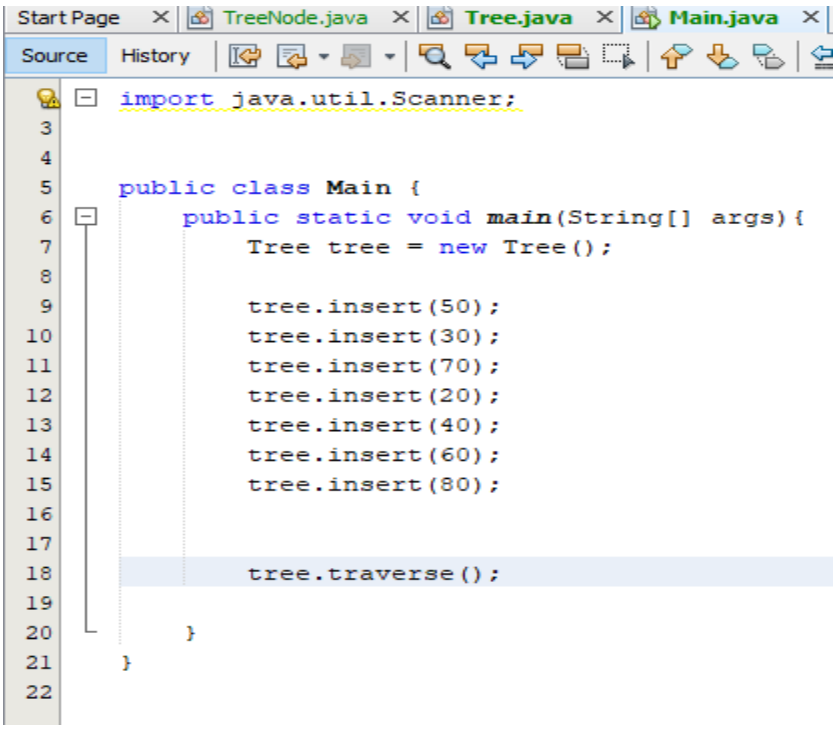
Answer:

TreeNode.java



```java
public class TreeNode {
    int value;
    TreeNode left;
    TreeNode right;
    public TreeNode(int value){
        this.value = value;
        this.left=null;
        this.right=null;
    }
}
```

Main.java



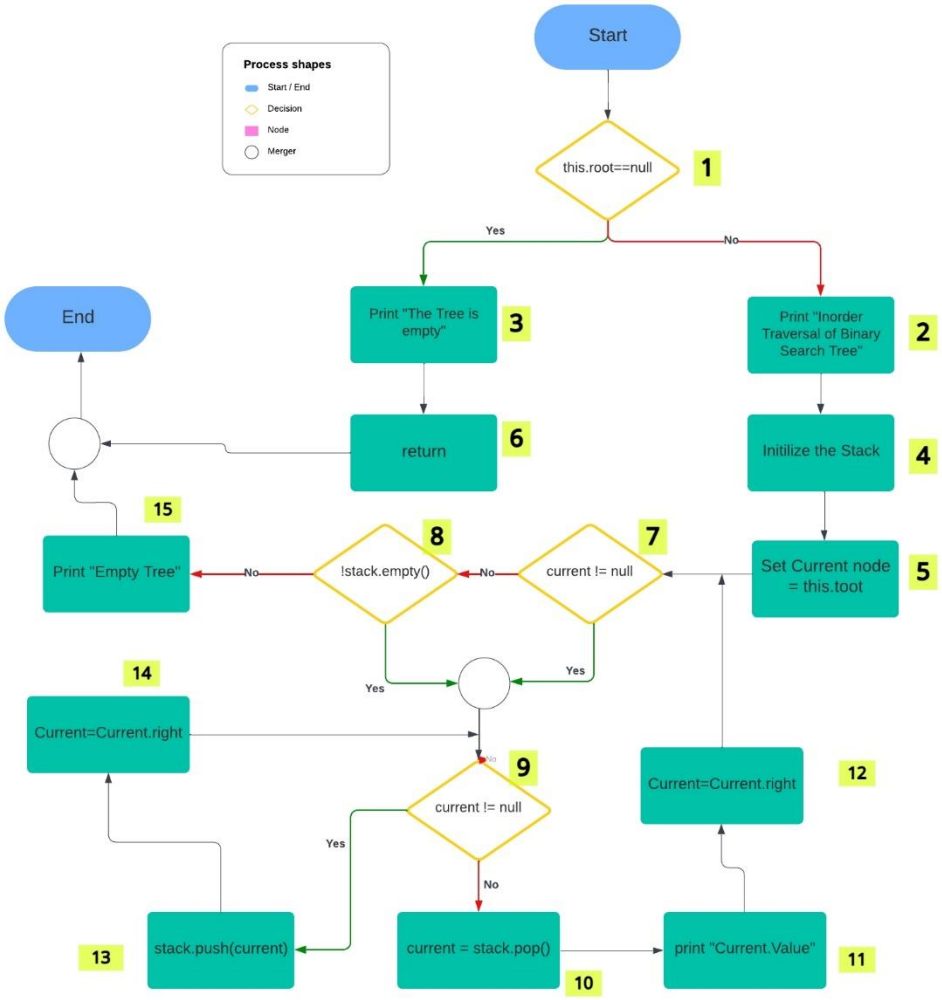
```java
import java.util.Scanner;


public class Main {
    public static void main(String[] args){
        Tree tree = new Tree();

        tree.insert(50);
        tree.insert(30);
        tree.insert(70);
        tree.insert(20);
        tree.insert(40);
        tree.insert(60);
        tree.insert(80);


        tree.traverse();

    }
}
```

Tree.java:

Source History

```
1
2  import java.util.Stack;
3
4  public class Tree {
5      public TreeNode root;
6
7      public Tree(){
8          this.root=null;
9      }
10     public void insert(int element) {
11         this.root = insertRecursive(this.root, element);
12     }
13     private TreeNode insertRecursive(TreeNode root, int element) {
14         if (root == null) {
15             return new TreeNode(element);
16         }
17         if (element < root.value) {
18             root.left = insertRecursive(root.left, element);
19         }
20         else {
21             root.right = insertRecursive(root.right, element);
22         }
23         return root;
24     }
25     public void traverseInOrderIterative() {
26         if (this.root == null) {
27             System.out.println("Tree is empty");
28             return;
29         }
30
31             System.out.println("Inorder traversal of the binary search tree (iterative):");
32             Stack<TreeNode> stack = new Stack<>();
33             TreeNode current = this.root;
34
35             while (current != null || !stack.isEmpty()) {
36                 while (current != null) {
37                     stack.push(current);
38                     current = current.left;
39                 }
40                 current = stack.pop();
41                 System.out.print(current.value + " ");
42                 current = current.right;
43             }
44
45             System.out.println();
46     }
47 }
48
```

Control Flow Graph:



Paths:

Path 1 → 1, 3, 6.

Path 2 → 1,2,4,5,7,9,13,14,9,10,11,12,7,8,15

Input Data Selection:

Paths	Input Data
Path 1	Tree = [] & root = null
Path 2	Tree = [1] & root = node (1)

Test Cases:

Test Case ID	Description	Input Data	Expected Output	Actual Output	Status
TC_001	The root of the tree is “null” (i.e the tree is empty)	Tree= [] Root = null	Display Msg: “Tree is Empty”	Display Msg: “Tree is empty”	Pass
TC_002	The tree has some elements in it. Now when this method will be called the elements will be printed.	Tree= [1] Root = node (1)	Display Msg: “Inorder traversal of the binary search tree (iterative):” and all element of Tree: 1 .	Display Msg: “Inorder traversal of the binary search tree (iterative):” and all the elements of the Tree: 1.	Pass