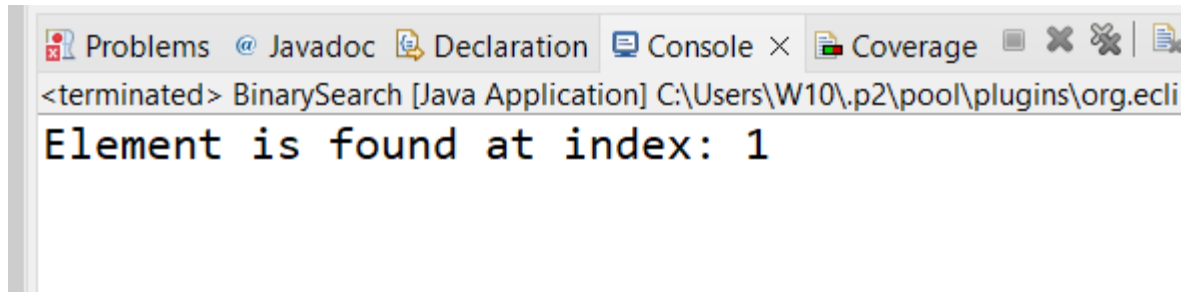Samiksha Shende
220960920078

LAB EXAM

DATA STRUCTURE AND ALGORITHMS

1. Write a Java program to

a. Perform binary search operation

```java
        package com.Binary;

public class BinarySearch {
    public static void binarySearch(int arr[], int first,
int last, int key){
            int mid = (first + last)/2;
            while( first <= last ){
                if ( arr[mid] < key ){
                   first = mid + 1;
                }else if ( arr[mid] == key ){
                   System.out.println("Element is found at
index: " + mid);
                    break;
                }else{
                    last = mid - 1;
                }
                mid = (first + last)/2;
            }
            if ( first > last ){
                System.out.println("Element is not found!");
            }
        }
        public static void main(String args[]){
                int arr[] = {111,122,253,56,236};
                int key = 122;
                int last=arr.length-1;
                binarySearch(arr,0,last,key);
        }
        }
```

Problems @ Javadoc Declaration Console × Coverage ■ ✖ ✖ ⬚

\<terminated\> BinarySearch [Java Application] C:\Users\W10\.p2\pool\plugins\org.ecli

Element is found at index: 1

b. Execute tree traversal in postorder

```java
package com.postorder;

public class Node {
    public int value;
    public Node left, right;

    public Node(int element)
    {
        value = element;
        left = right = null;
    }
}
```

```java
package com.postorder;

public class PostorderTraversal {
     Node root;

    PostorderTraversal() { root = null; }

    void traversePostorder(Node node)
    {
        if (node == null)
            return;
        traversePostorder(node.left);
```
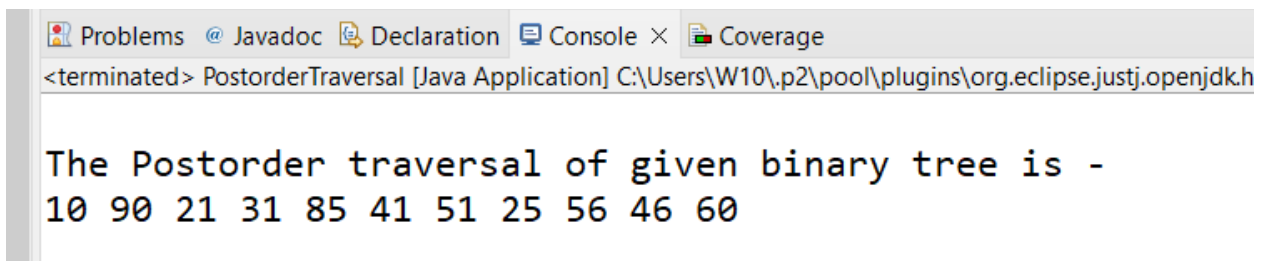
```java
            traversePostorder(node.right);
            System.out.print(node.value + " ");
    }

    void traversePostorder() { traversePostorder(root); }

    public static void main(String args[])
    {
        PostorderTraversal pt = new PostorderTraversal();
        pt.root = new Node(60);
        pt.root.left = new Node(85);
        pt.root.right = new Node(46);
        pt.root.left.left = new Node(21);
        pt.root.left.right = new Node(31);
        pt.root.left.left.left = new Node(10);
        pt.root.left.left.right = new Node(90);
        pt.root.right.left = new Node(41);
        pt.root.right.right = new Node(56);
        pt.root.right.right.left = new Node(51);
        pt.root.right.right.right = new Node(25);

        System.out.println();
        System.out.println("The Postorder traversal of given
binary tree is - ");
        pt.traversePostorder();
        System.out.println();
    }

}
```

Problems @ Javadoc Declaration Console × Coverage
<terminated> PostorderTraversal [Java Application] C:\Users\W10\.p2\pool\plugins\org.eclipse.justj.openjdk.h

```
The Postorder traversal of given binary tree is -
10 90 21 31 85 41 51 25 56 46 60
```

* * * * * * * * *