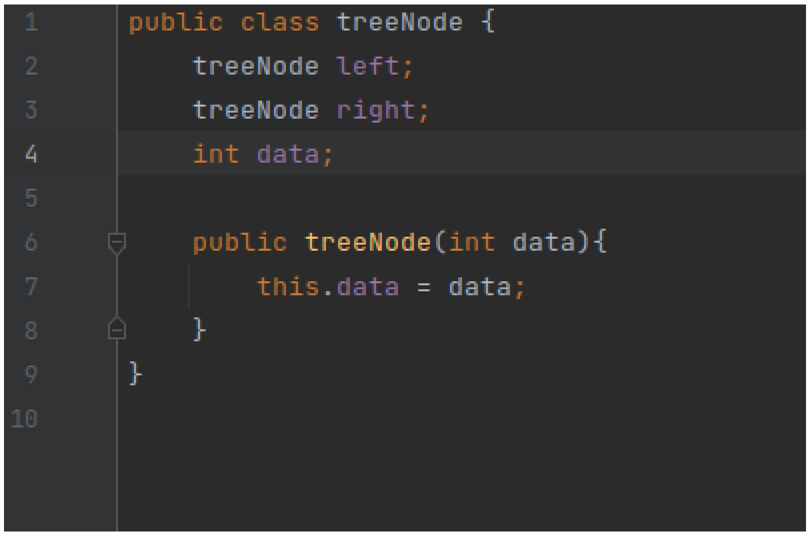
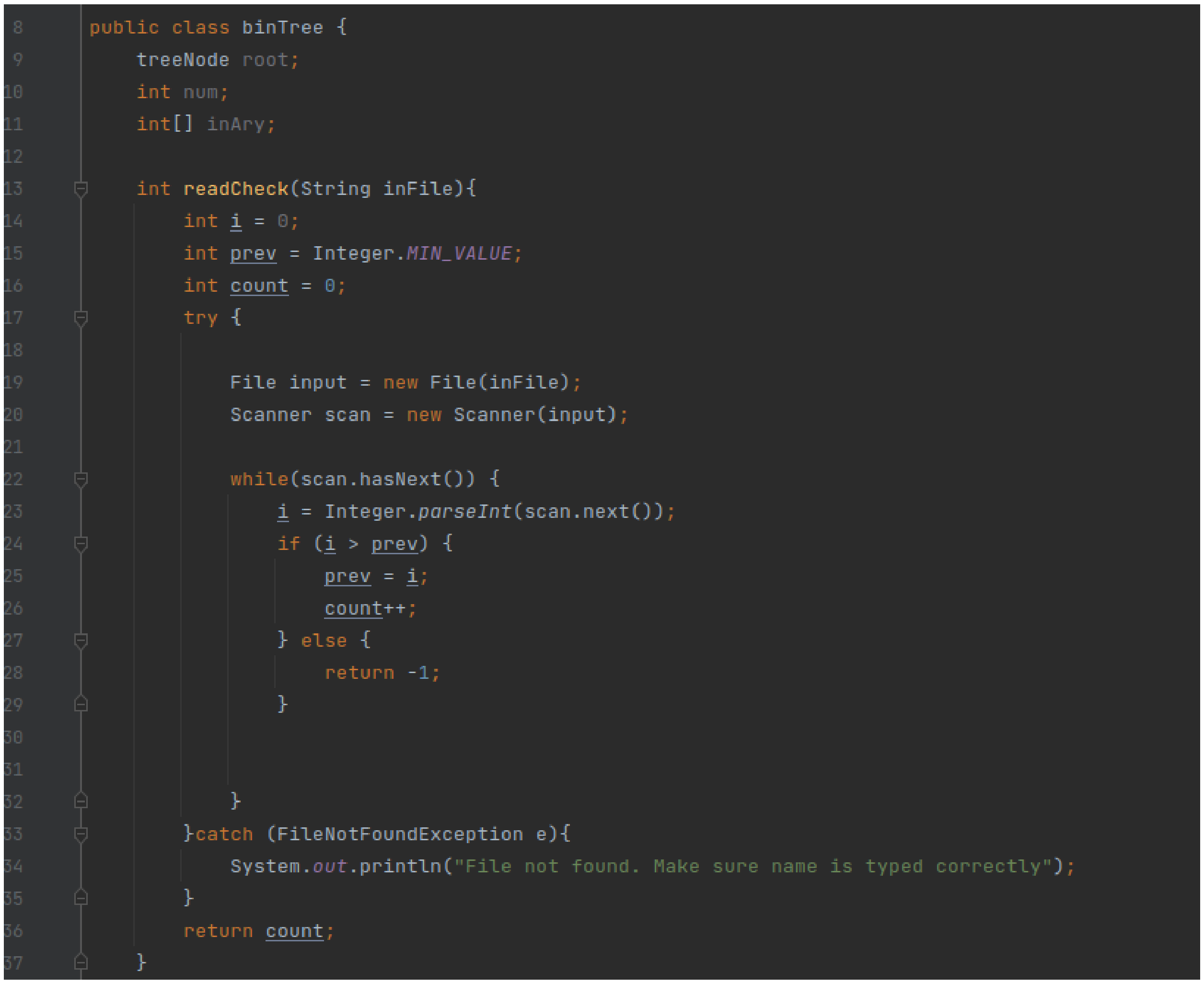
Mohammed Chowdhury
323.33
Q25

TREENODE CLASS:

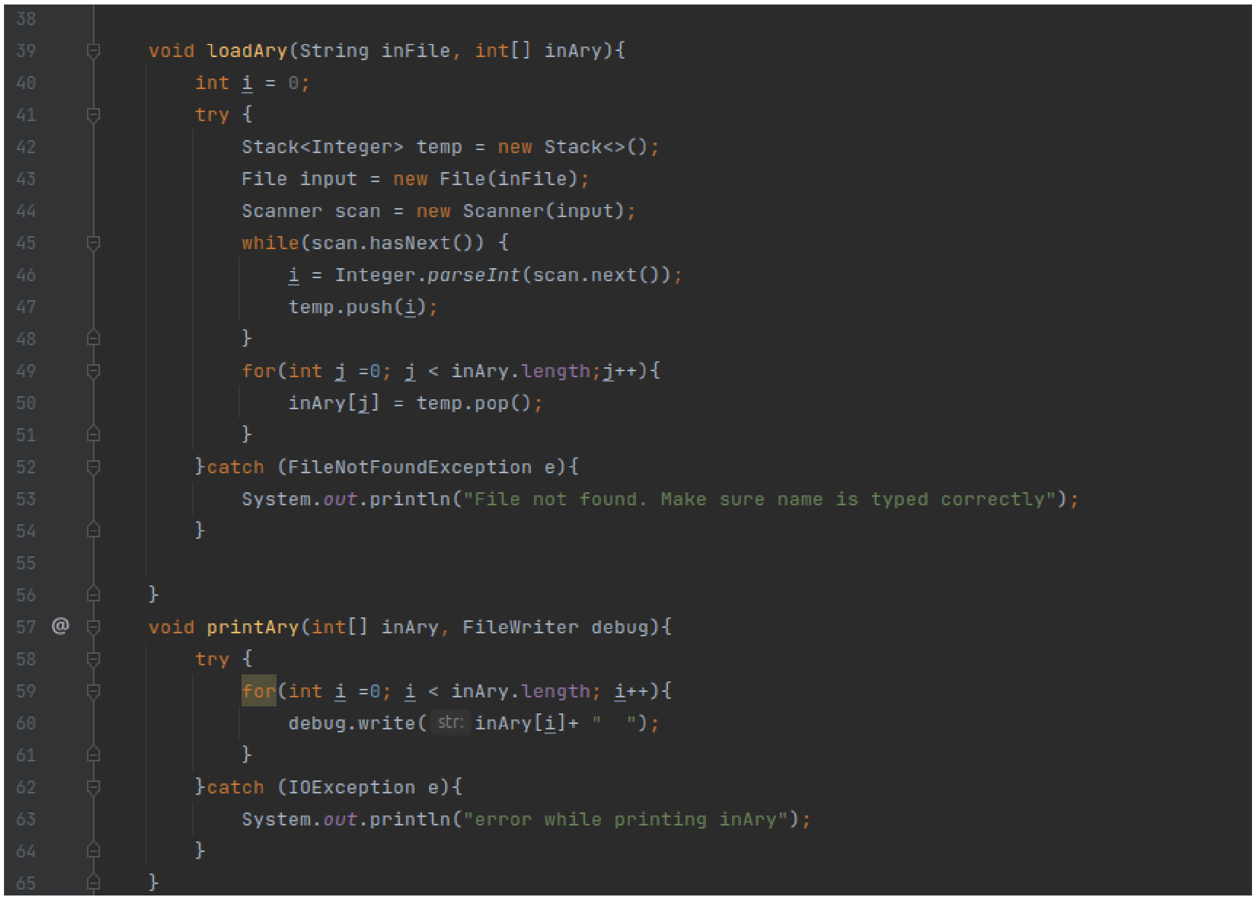


```java
public class treeNode {
    treeNode left;
    treeNode right;
    int data;

    public treeNode(int data){
        this.data = data;
    }
}
```

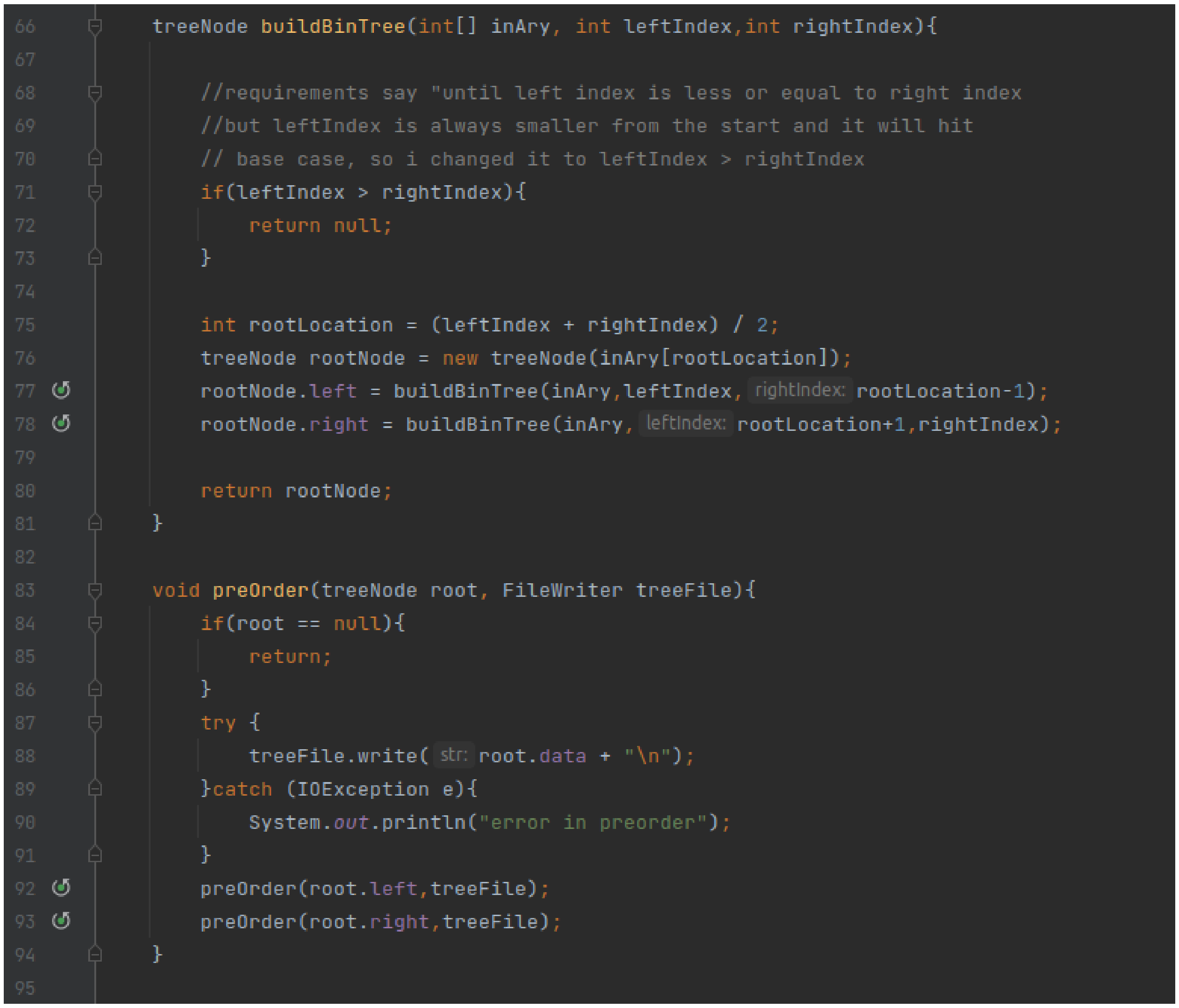
BinTree Class:



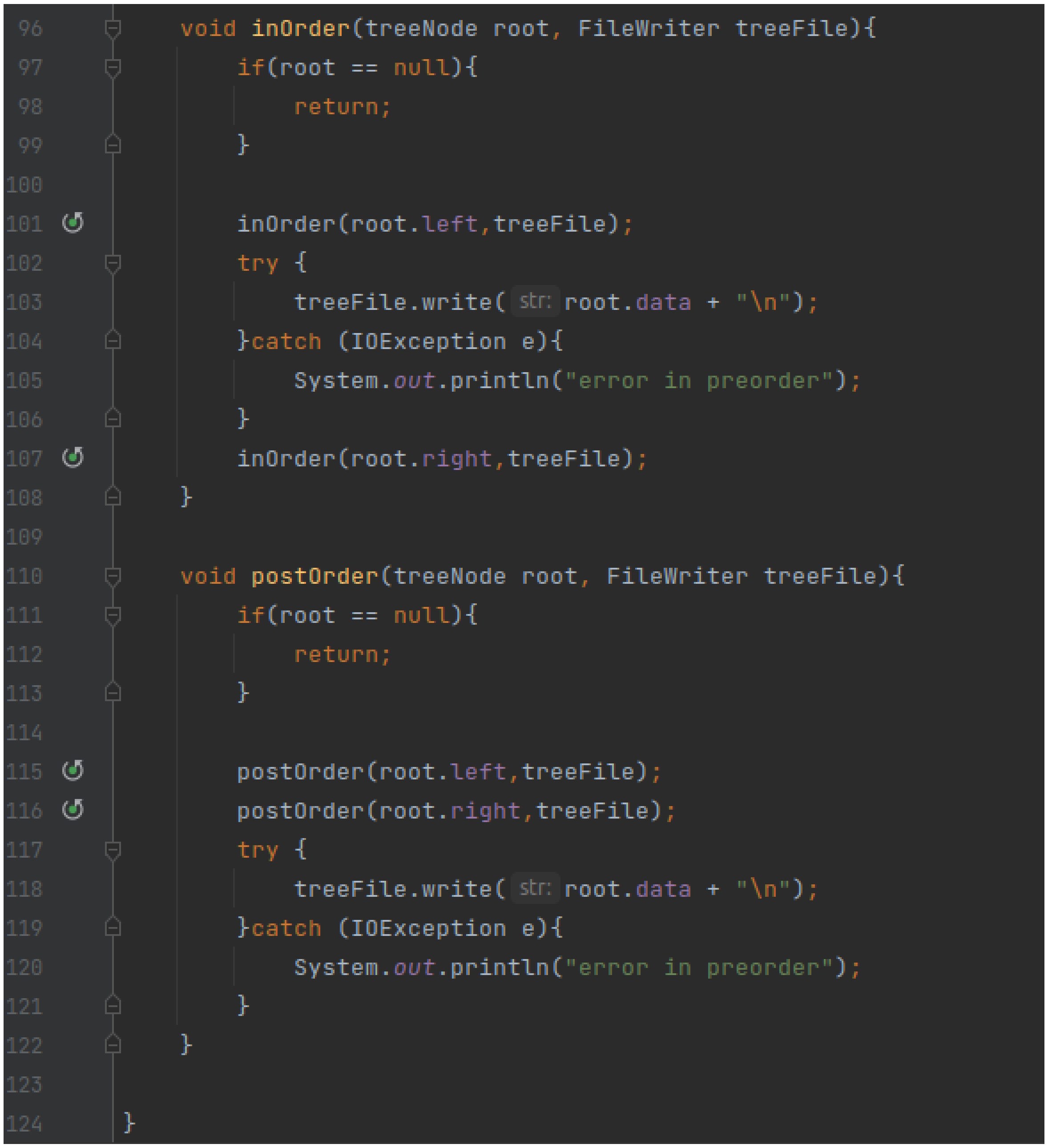
```java
public class binTree {
    treeNode root;
    int num;
    int[] inAry;

    int readCheck(String inFile){
        int i = 0;
        int prev = Integer.MIN_VALUE;
        int count = 0;
        try {

            File input = new File(inFile);
            Scanner scan = new Scanner(input);

            while(scan.hasNext()) {
                i = Integer.parseInt(scan.next());
                if (i > prev) {
                    prev = i;
                    count++;
                } else {
                    return -1;
                }


            }
        }catch (FileNotFoundException e){
            System.out.println("File not found. Make sure name is typed correctly");
        }
        return count;
    }
```



```java

void loadAry(String inFile, int[] inAry){
    int i = 0;
    try {
        Stack<Integer> temp = new Stack<>();
        File input = new File(inFile);
        Scanner scan = new Scanner(input);
        while(scan.hasNext()) {
            i = Integer.parseInt(scan.next());
            temp.push(i);
        }
        for(int j =0; j < inAry.length;j++){
            inAry[j] = temp.pop();
        }
    }catch (FileNotFoundException e){
        System.out.println("File not found. Make sure name is typed correctly");
    }

}
void printAry(int[] inAry, FileWriter debug){
    try {
        for(int i =0; i < inAry.length; i++){
            debug.write( str: inAry[i]+ "  ");
        }
    }catch (IOException e){
        System.out.println("error while printing inAry");
    }
}
```



```java
    treeNode buildBinTree(int[] inAry, int leftIndex,int rightIndex){

        //requirements say "until left index is less or equal to right index
        //but leftIndex is always smaller from the start and it will hit
        // base case, so i changed it to leftIndex > rightIndex
        if(leftIndex > rightIndex){
            return null;
        }

        int rootLocation = (leftIndex + rightIndex) / 2;
        treeNode rootNode = new treeNode(inAry[rootLocation]);
        rootNode.left = buildBinTree(inAry,leftIndex, rightIndex: rootLocation-1);
        rootNode.right = buildBinTree(inAry, leftIndex: rootLocation+1,rightIndex);

        return rootNode;
    }

    void preOrder(treeNode root, FileWriter treeFile){
        if(root == null){
            return;
        }
        try {
            treeFile.write( str: root.data + "\n");
        }catch (IOException e){
            System.out.println("error in preorder");
        }
        preOrder(root.left,treeFile);
        preOrder(root.right,treeFile);
    }

```



```
    void inOrder(treeNode root, FileWriter treeFile){
        if(root == null){
            return;
        }

        inOrder(root.left,treeFile);
        try {
            treeFile.write( str: root.data + "\n");
        }catch (IOException e){
            System.out.println("error in preorder");
        }
        inOrder(root.right,treeFile);
    }

    void postOrder(treeNode root, FileWriter treeFile){
        if(root == null){
            return;
        }

        postOrder(root.left,treeFile);
        postOrder(root.right,treeFile);
        try {
            treeFile.write( str: root.data + "\n");
        }catch (IOException e){
            System.out.println("error in preorder");
        }
    }

}
```

treeFile:

```
PREORDER:
56
81
91
94
100
95
93
92
87
90
89
85
83
70
76
80
78
74
72
65
69
67
60
63
58
32
44
50
54
52
48
46
38
42
40
36
34
20
26
30
28
24
```

```
22
14
18
16
10
12
8

INORDER:
100
95
94
93
92
91
90
89
87
85
83
81
80
78
76
74
72
70
69
67
65
63
60
58
56
54
52
50
48
46
44
42
40
38
36
34
```

```
32
30
28
26
24
22
20
18
16
14
12
10
8

POSTORDER:
95
100
92
93
94
89
90
83
85
87
91
78
80
72
74
76
67
69
63
58
60
65
70
81
52
54
46
48
50
40
```

```
42
34
36
38
44
28
30
22
24
26
16
18
12
8
10
14
20
32
56
```

DebugFile:

```
Input Size: 49
ARRAY:
100  95  94  93  92  91  90  89  87  85  83  81  80  78  76  74  72
70  69  67  65  63  60  58  56  54  52  50  48  46  44  42  40  38
36  34  32  30  28  26  24  22  20  18  16  14  12  10  8
```