Name – Yash Daga Roll Number – 20BCE7323

1) Write a program create Binary tree from Inorder and postorder sequence. Also define a method Max() which returns maximum element from Binary Tree.

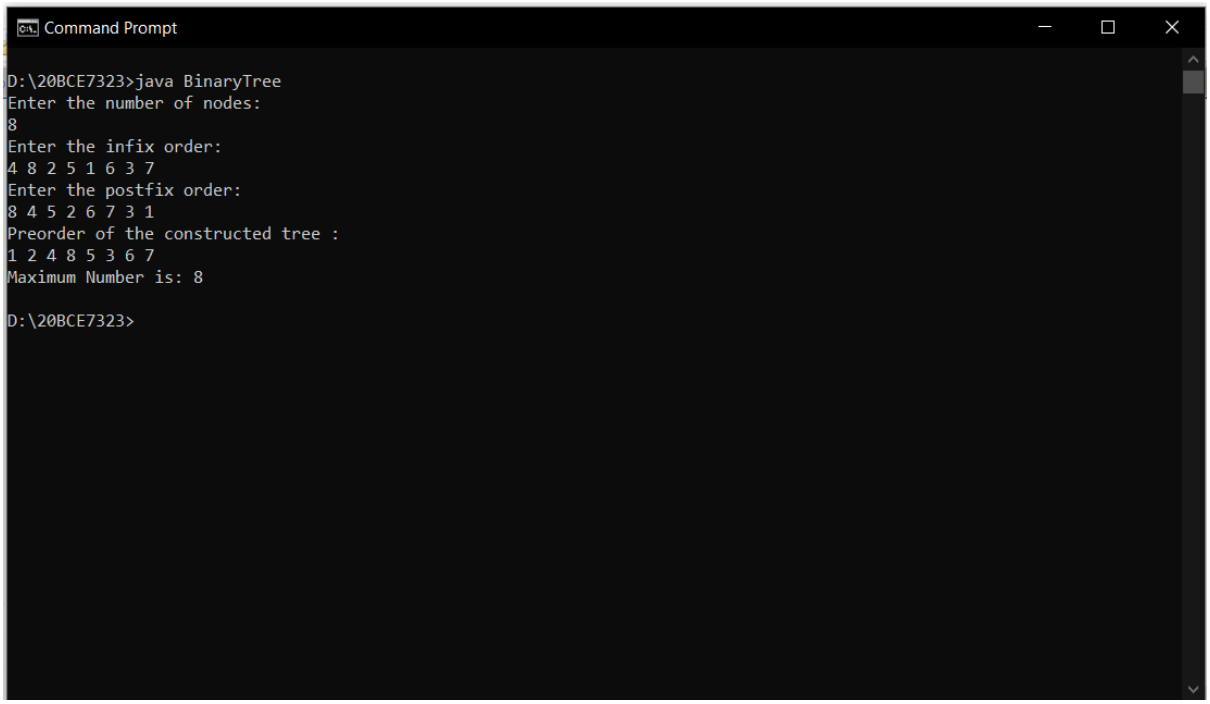
Code:

```
import java.util.*;

class Node {

    int data;

    Node left, right;


    public Node(int data)

    {

        this.data = data;

        left = right = null;

    }

}


class BinaryTree {

    Node buildUtil(int in[], int post[], int inStrt,

                   int inEnd, int postStrt, int postEnd)

    {

        if (inStrt > inEnd)

            return null;

        Node node = new Node(post[postEnd]);

        if (inStrt == inEnd)

            return node;

        int iIndex = search(in, inStrt, inEnd, node.data);
```

```
        node.left = buildUtil(
                in, post, inStrt, iIndex - 1, postStrt,
                postStrt - inStrt + iIndex - 1);
        node.right = buildUtil(in, post, iIndex + 1, inEnd,
                                        postEnd - inEnd + iIndex,
                                        postEnd - 1);

        return node;
}
int search(int arr[], int strt, int end, int value)
{
        int i;
        for (i = strt; i <= end; i++) {
                if (arr[i] == value)
                        break;
        }
        return i;
}

void preOrder(Node node)
{
        if (node == null)
                return;
        System.out.print(node.data + " ");
        preOrder(node.left);
        preOrder(node.right);
```

```
}

int max(Node node) {
        if (node == null)
                return -1;
        int maxLeft = max(node.left);
        int maxRight = max(node.right);
        return Math.max(node.data, Math.max(maxLeft, maxRight));
}

public static void main(String[] args)
{
        BinaryTree tree = new BinaryTree();
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of nodes: ");
        int n = sc.nextInt();
        int in[] = new int[n];
        int post[] = new int[n];
        System.out.println("Enter the infix order: ");
        for(int i = 0; i < n; i++) {
                in[i] = sc.nextInt();
        }
        System.out.println("Enter the postfix order: ");
        for(int i = 0; i < n; i++) {
                post[i] = sc.nextInt();
        }
```

```
        Node root
                = tree.buildUtil(in, post, 0, n - 1, 0, n - 1);
        System.out.println(
                "Preorder of the constructed tree : ");
        tree.preOrder(root);
        System.out.println("\nMaximum Number is: " + tree.max(root));
    }
}
```


Command Prompt
D:\20BCE7323>java BinaryTree
Enter the number of nodes:
8
Enter the infix order:
4 8 2 5 1 6 3 7
Enter the postfix order:
8 4 5 2 6 7 3 1
Preorder of the constructed tree :
1 2 4 8 5 3 6 7
Maximum Number is: 8
D:\20BCE7323>