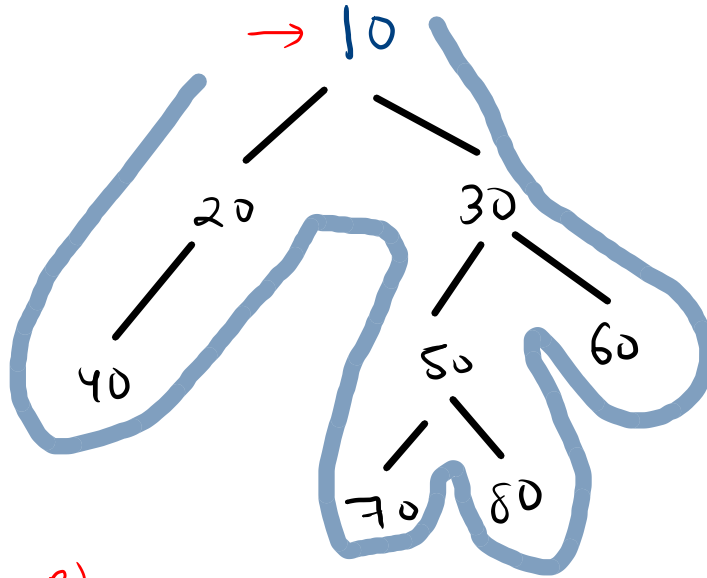


Traversal.

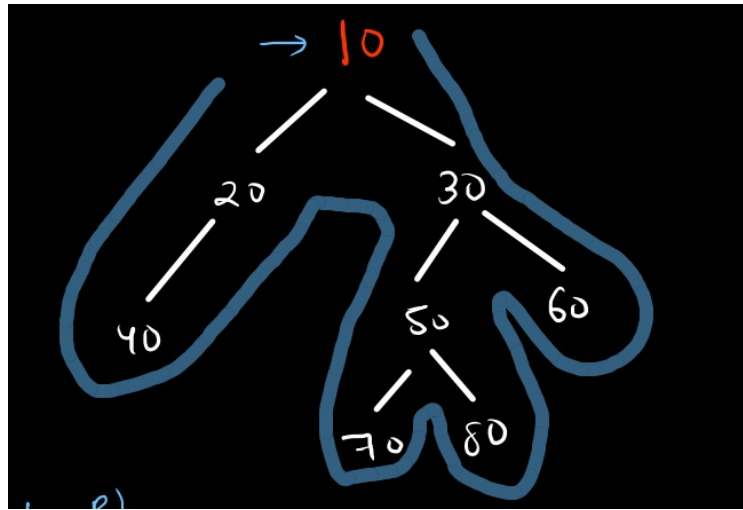


{ preorder
Inorder
Postorder

Preorder (Root L R)

10 20 40 30 50 70 80 60

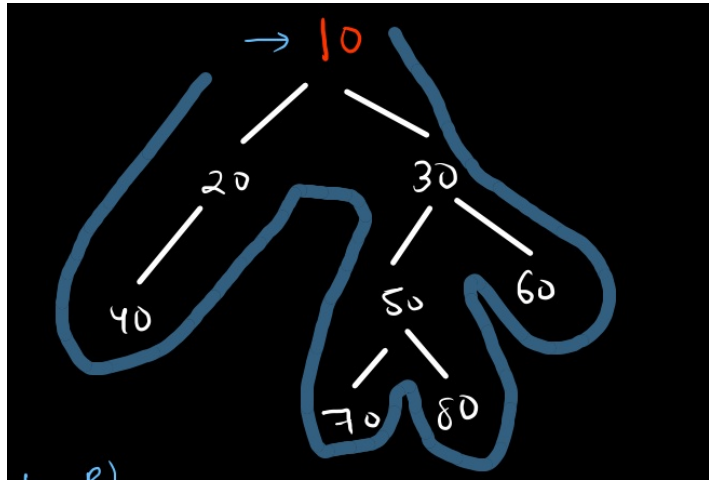
Inorder
(L Root R)



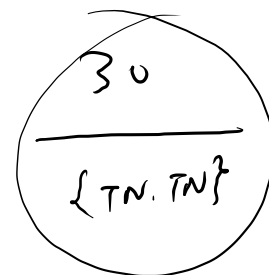
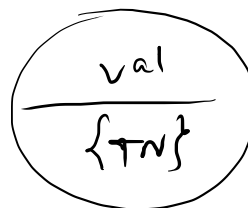
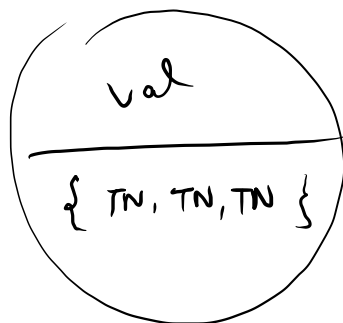
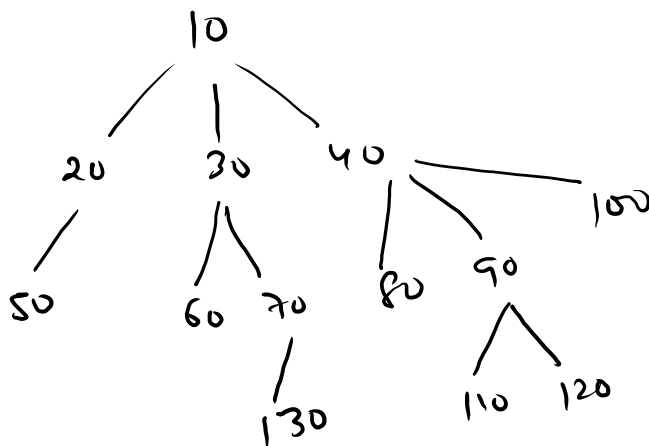
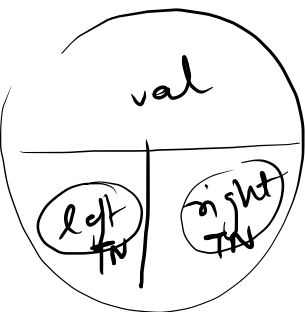
40 20 10 70 50 80 30 60

Postorder

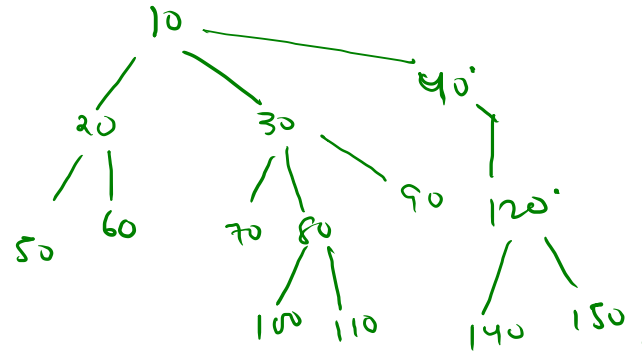
(L R Root)



40 20 70 80 (50) 60 30 10



10 20 50 60 30 70
80 100 110 90 40 120 140 150



arr{10, 20, 50, -1, 60, -1, -1, 30, 70, -1, 80, 100, -1, 110, -1, -1, 90, -1, -1, 40, 120, 140, -1, 150, -1, -1, -1, -1};

tmp = 10

```
public static TreeNode createTree(int [] A){
    Stack<TreeNode> st = new Stack<>();
    for(int i = 0; i < A.length-1; i++){
        if(A[i] == -1){
            TreeNode tmp = st.pop();
            st.peek().children.add(tmp);
        }
        else{
            st.push(new TreeNode(A[i]));
        }
    }
    return st.peek();
}
```

