

Traversals \rightarrow Pre / In / Post / Level Order

1) Same tree



For any recursive traversal in tree
the S.C. = $O(h)$ / $O(n)$

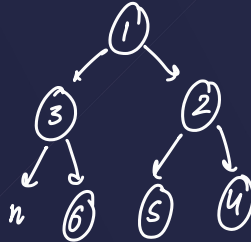
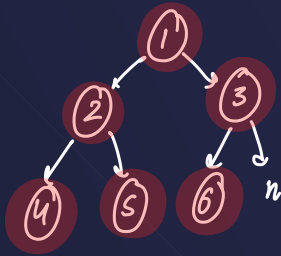
h = height of tree

$$T.C. = O(n)$$

$$S.C. = O(n) / O(\log n) \rightarrow \text{Balanced tree}$$

2) Invert Tree :

226

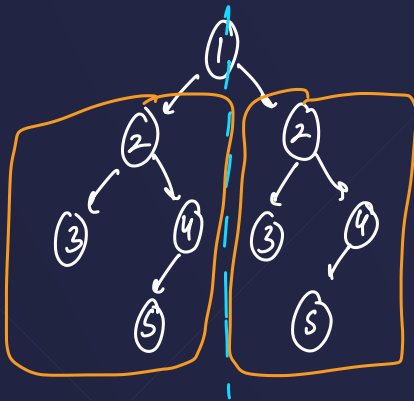


$$I.C. = O(n)$$

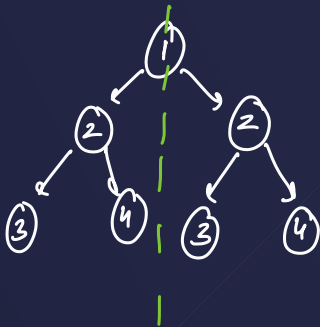
where n is height of tree

$$T.C = O(n)$$

COLLEGE
WALLAH



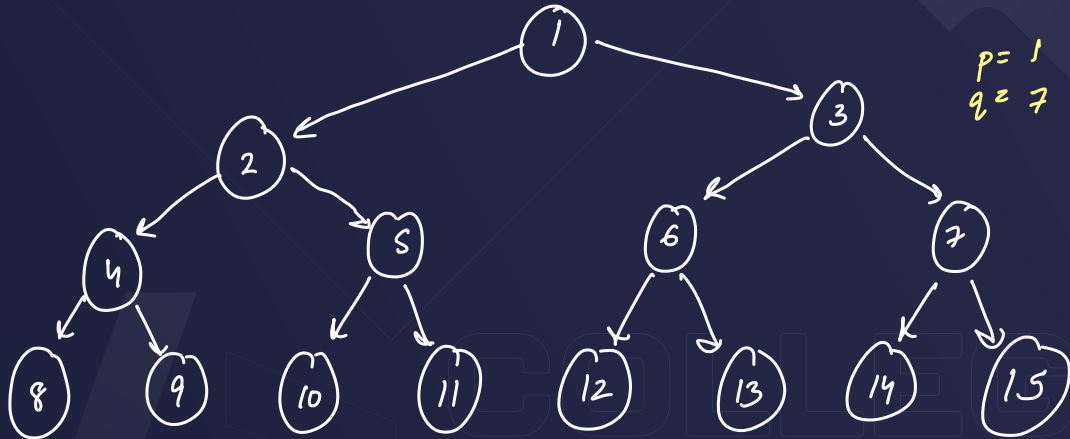
TC
SC



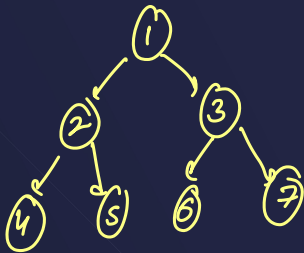
$\text{sym}(1) \rightarrow \text{sym}(2) \ \&\ \text{sym}(2)$ α

Lowest Common Ancestor (LCA) of Binary Tree

Diff Subtree



```
printPath(root, "")
```



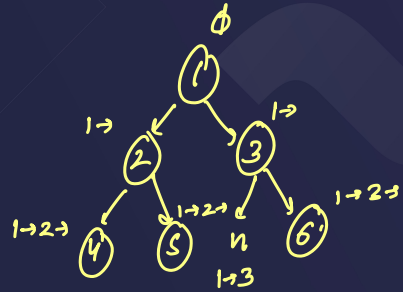
124, 125, 136, 137

```

public void helper(TreeNode root, String str, List<String> arr){
    if(root==null) return;
    if(root.left==null && root.right==null){
        arr.add(str+root.val);
        return;
    }
    helper(root.left, str+root.val+"->", arr);
    helper(root.right, str+root.val+"->", arr);
}

public List<String> binaryTreePaths(TreeNode root) {
    List<String> arr = new ArrayList<>();
    helper(root, "", arr);
    return arr;
}

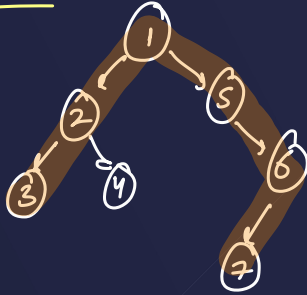
```

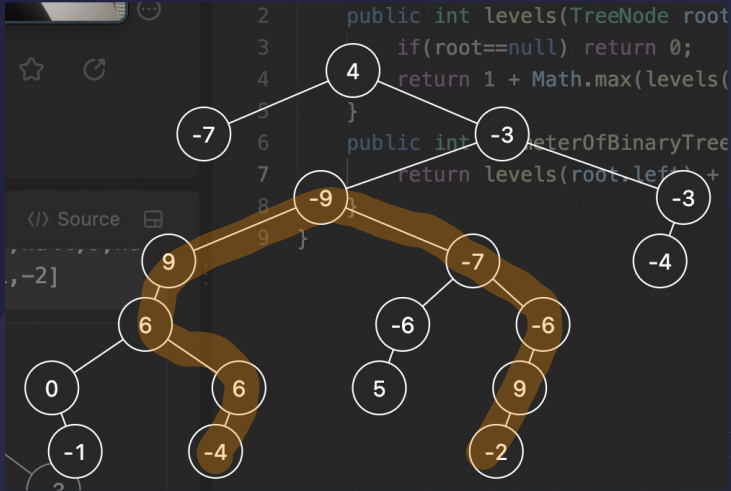


arr = { 1->2->4, 1->2->5, 1->3->6 }

Diameter of a tree

↓
max path
b/w any two
nodes





Is this visible ✓

Bad Writing is gone

COLLEGE
WALLAH