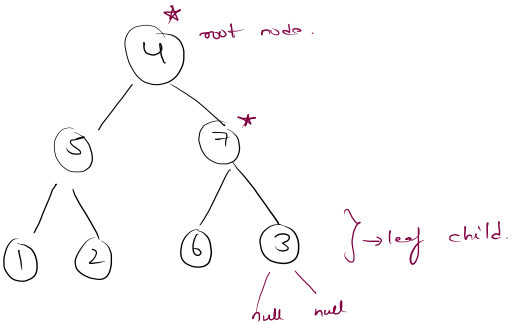
Revision.

TreeNode


3 types of traversals

1. Preorder
2. Inorder
3. Postorder

} printing.




Preorder → Root L R

Inorder → L Root R

Postorder → L R Root

✓

**Preorder**

process → Syso (node.data);
preorder (root.left);
preorder (root.right);

**Inorder**

Inorder (node.left);
Syso (node.data);
Inorder (node.right);

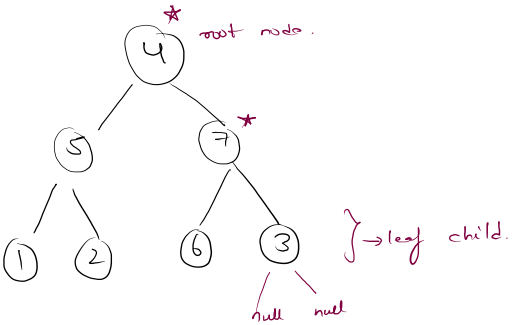
Revision.

TreeNode


3 types of traversals

1. Preorder
2. Inorder
3. Postorder

} printing.




Preorder → Root L R

Inorder → L Root R

Postorder → L R Root

✓

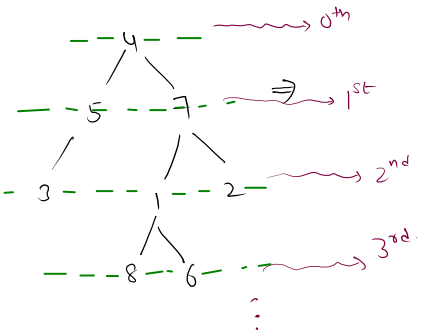
**Preorder**

process → Syso (node.data);
preorder (root.left);
preorder (root.right);

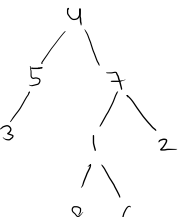
**Inorder**

Inorder (node.left);
Syso (node.data);
Inorder (node.right);

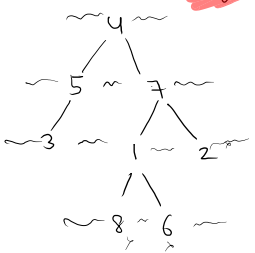
## Level-order Traversal (BFS) → Iterative Solution.



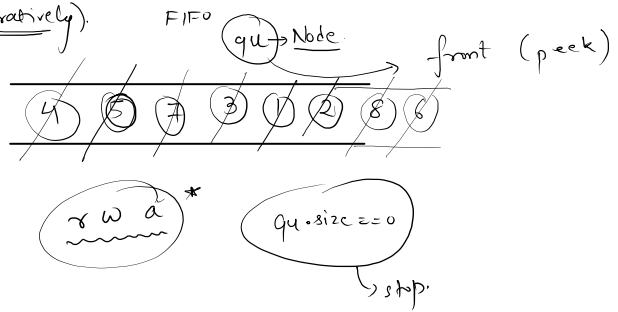

Level order { 4 5 7 3 1 2 8 6




Logic (Iteratively).




FIFO

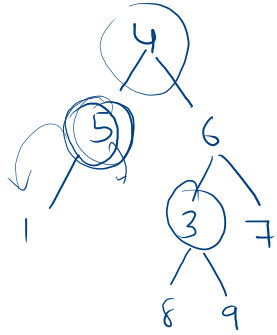



4 5 7 3 1 2 8 6

```java
public static void levelOrder(Node root) {

    Queue<Node> qu = new LinkedList();
    qu.add(root);     //seeding

    while(qu.size() != 0){      //if there is something in ur qu
        //rwa
        //remove
        Node rem = qu.remove();
        //work
        System.out.print(rem.data + " ");
        //add children
        if(rem.left != null)
            qu.add(rem.left);
        if(rem.right != null)
            qu.add(rem.right);
    }


}
```

rwa

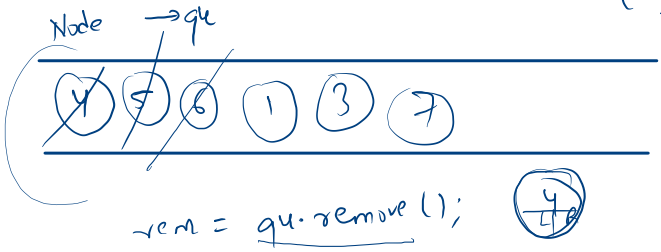
Node




4 5 6 1 3 7 8 9

4

4 5 6

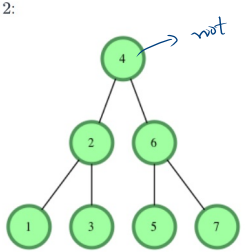


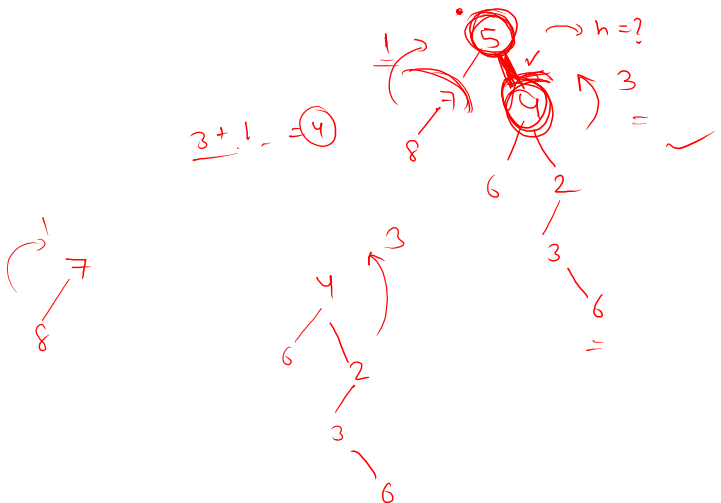

(s==0)

# Tree: Height of a Binary Tree

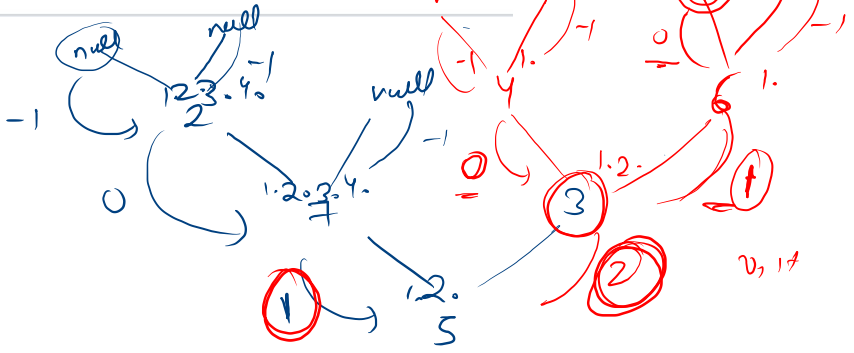
Problem | Submissions | Leaderboard | Discussions

The height of a binary tree is the number of edges between the tree's root and its furthest leaf. For example, the following binary tree is of height 2:



```java
public static int height(Node root) {
    if(root == null)
        return -1;

    int leftH = height(root.left);
    int rightH = height(root.right);

    return Math.max(leftH, rightH) + 1;

}
```

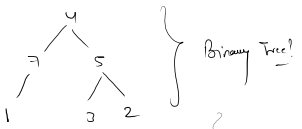



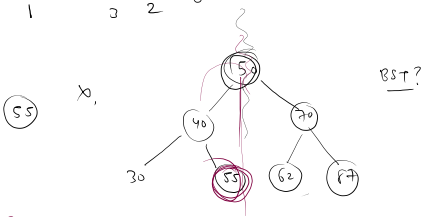

(1, 2) +1

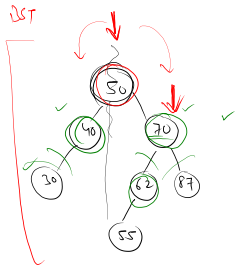
2 + 1 = 3.

# Binary Search Tree.

All BST are Binary Tree.




62

Binary Search.

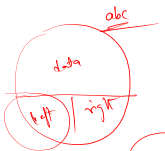
↳ sorted.

BST

left : max of left subtree will be less than root

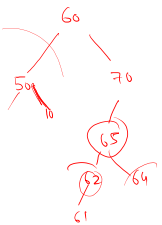
right : min of right subtree will be more than root

in order ? → sorted.

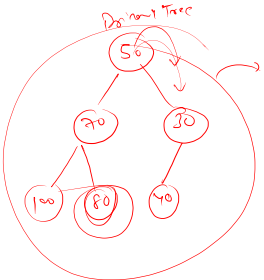



abc.right = 10

abc.left = 10




root.right =

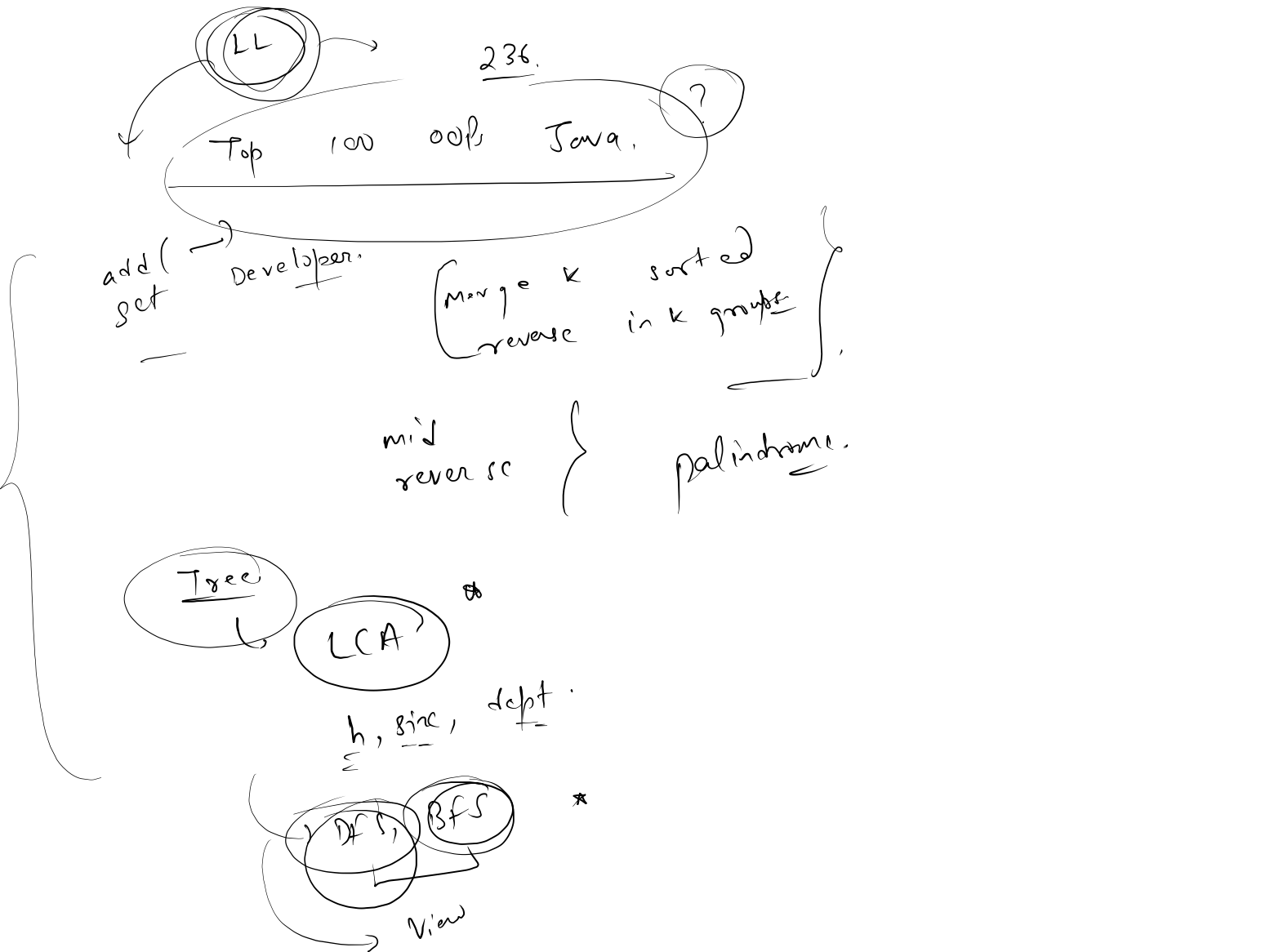



find = 80

n

O(n)

80


LL
236.
Top 100 OOPs Java.
?
add( Developer.
set
Merge k sorted
reverse in k groups.
mid
reverse
palindrome.
Tree
LCA
h, size, dept.
DFS
BFS
View