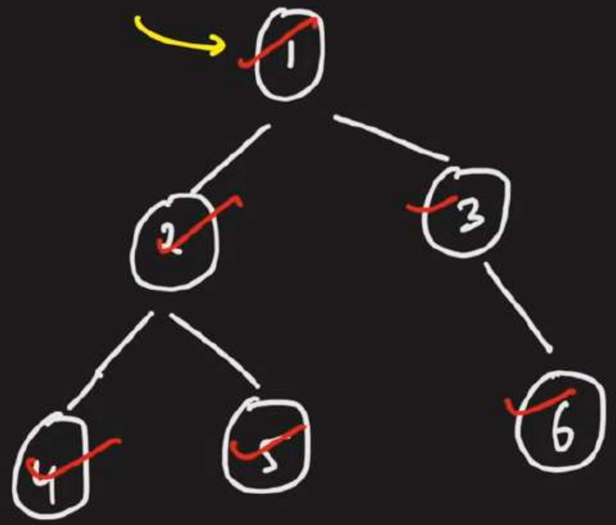
Zig-Zag Traversal

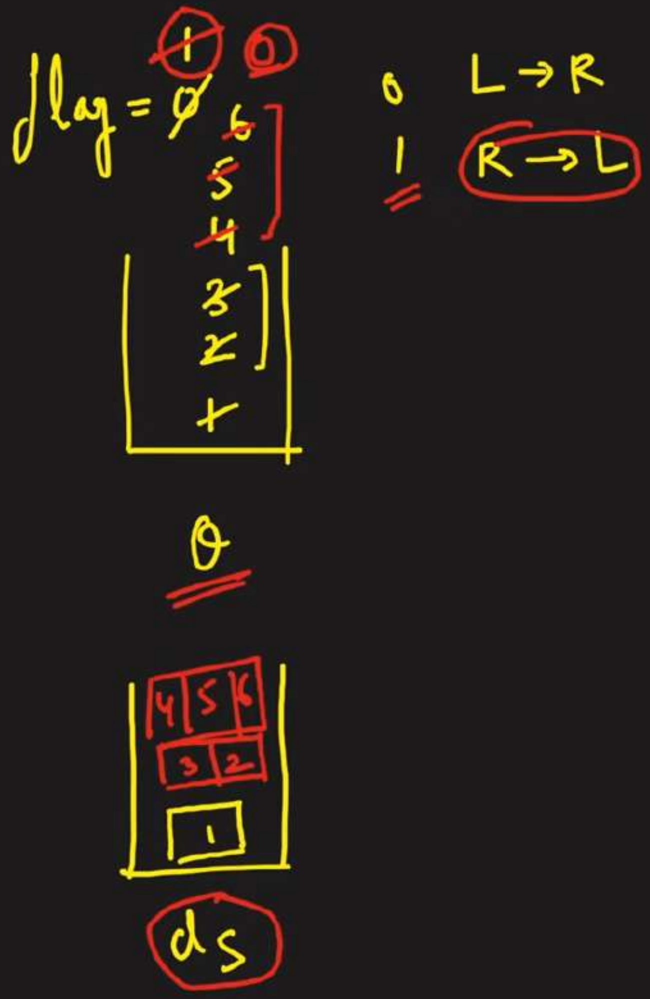


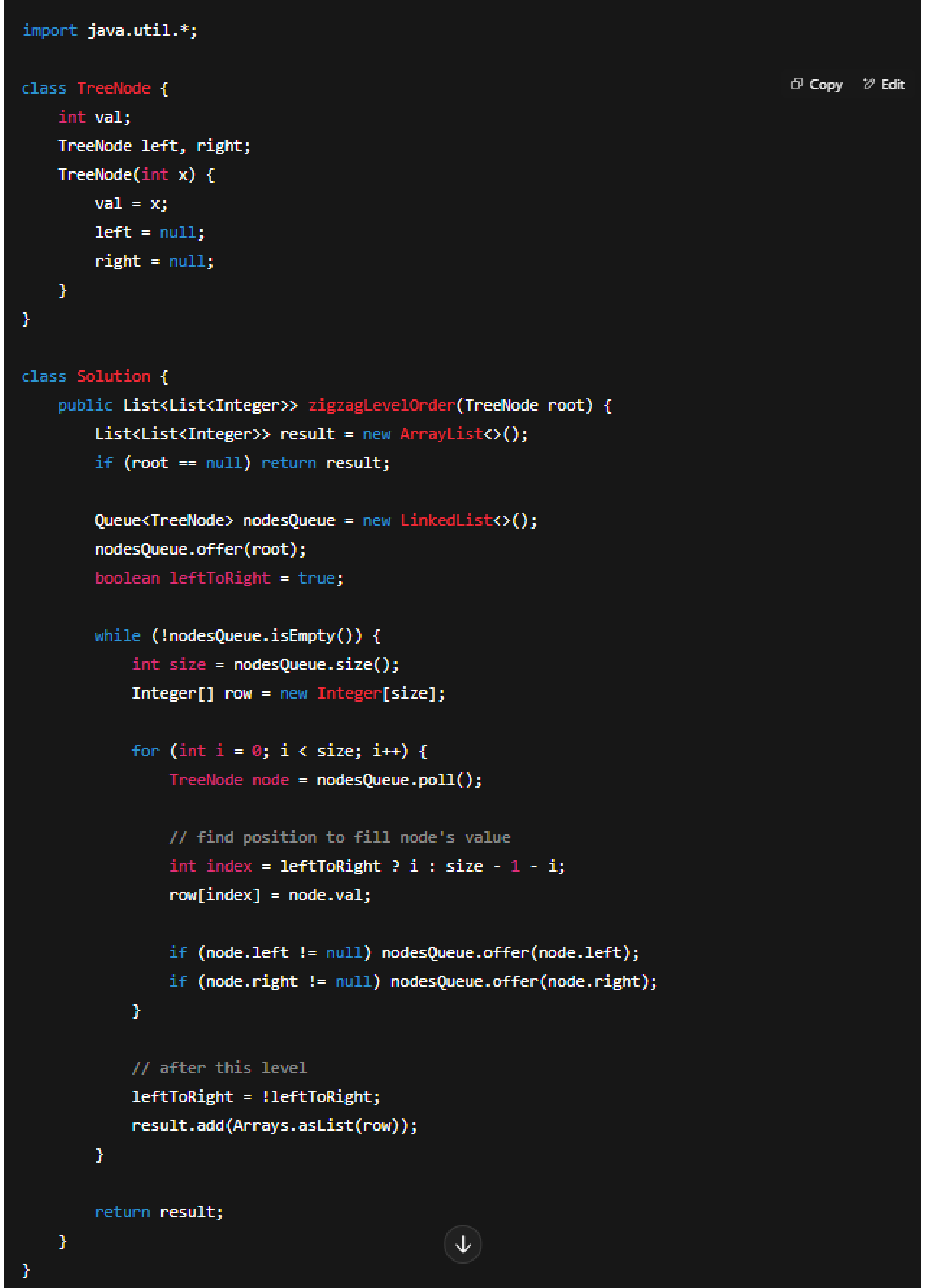
~~1~~

~~2~~ ~~3~~

~~4~~ ~~5~~ ~~6~~

flag → ~~φ~~ ~~1~~ 0





```java
import java.util.*;

class TreeNode {
    int val;
    TreeNode left, right;
    TreeNode(int x) {
        val = x;
        left = null;
        right = null;
    }
}

class Solution {
    public List<List<Integer>> zigzagLevelOrder(TreeNode root) {
        List<List<Integer>> result = new ArrayList<>();
        if (root == null) return result;

        Queue<TreeNode> nodesQueue = new LinkedList<>();
        nodesQueue.offer(root);
        boolean leftToRight = true;

        while (!nodesQueue.isEmpty()) {
            int size = nodesQueue.size();
            Integer[] row = new Integer[size];

            for (int i = 0; i < size; i++) {
                TreeNode node = nodesQueue.poll();

                // find position to fill node's value
                int index = leftToRight ? i : size - 1 - i;
                row[index] = node.val;

                if (node.left != null) nodesQueue.offer(node.left);
                if (node.right != null) nodesQueue.offer(node.right);
            }

            // after this level
            leftToRight = !leftToRight;
            result.add(Arrays.asList(row));
        }

        return result;
    }
}
```