Check if a given binary tree is completed. A complete binary tree is one in which every level of the binary tree is completely filled except possibly the last level. Furthermore, all nodes are as far left as possible.

**Examples**

        5

      /    \

    3        8

  /   \

1      4

is completed.

        5

      /    \

    3        8

  /   \        \

1      4        11

is not completed.

public class TreeNode {

public int key;

public TreeNode left;

public TreeNode right;

public TreeNode(int key) {

this.key = key;

}

}

public class Solution {

public boolean isCompleted(TreeNode root) {

// Write your solution here.

if (root == null) {

return true;

}

boolean flag = false;

Queue<TreeNode> q = new LinkedList<TreeNode>();

q.offer(root);

while (!q.isEmpty()) {

TreeNode node = q.poll();

if (node.left == null) {

flag = true;

} else if(flag) {

return false;

} else {

q.offer(node.left);

}

if (node.right == null) {

flag = true;

} else if (flag) {

return false;

} else {

q.offer(node.right);

}

}

return true;

}

}