Diameter of Binary Tree

Solution

Given a binary tree, you need to compute the length of the diameter of the tree. The diameter of a binary tree is the length of the **longest** path between any two nodes in a tree. This path may or may not pass through the root.

Example:
Given a binary tree

```
          1
         / \
        2   3
       / \
      4   5
```

Return **3**, which is the length of the path [4,2,1,3] or [5,2,1,3].

Note: The length of path between two nodes is represented by the number of edges between them.

Java

```java
/**
 * Definition for a binary tree node.
 * public class TreeNode {
 *     int val;
 *     TreeNode left;
 *     TreeNode right;
 *     TreeNode(int x) { val = x; }
 * }
 */
class Solution {

    public class MaxPath {
        int value = 0;
    }

    //int maxPath = 0;
    public int diameterOfBinaryTree(TreeNode root) {
        if(root == null) return 0;
        MaxPath maxPath = new MaxPath();
        maxDepth(root, maxPath);
        return maxPath.value;
        //return maxPath - 1;
    }

    public int maxDepth(TreeNode root, MaxPath maxPath) {

        if(root == null){
            return 0;
        }

        int leftDepth = maxDepth(root.left, maxPath);
        int rightDepth = maxDepth(root.right, maxPath);
        int pathLength = leftDepth + rightDepth;
        maxPath.value = Math.max(maxPath.value, pathLength);

        return Math.max(leftDepth, rightDepth) + 1;
    }
}
```

Custom Testcase (Contribute)

Run Code Submit