543. Diameter of Binary Tree

Solved

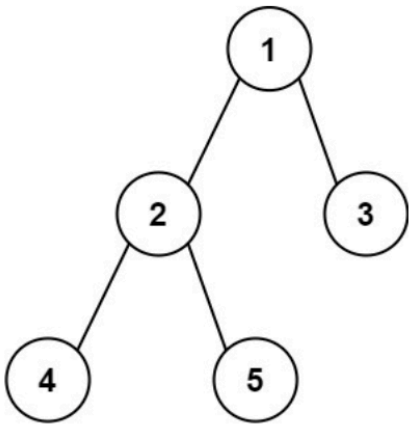
Easy | Topics | Companies

Given the `root` of a binary tree, return *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`.

The **length** of a path between two nodes is represented by the number of edges between them.

Example 1:




```
Input: root = [1,2,3,4,5]
Output: 3
Explanation: 3 is the length of the path [4,2,1,3] or [5,2,1,3].
```

```
/**
* Definition for a binary tree node.
* public class TreeNode {
*     public var val: Int
*     public var left: TreeNode?
*     public var right: TreeNode?
*     public init() { self.val = 0; self.left = nil; self.right =
nil; }
*     public init(_ val: Int) { self.val = val; self.left = nil;
self.right = nil; }
*     public init(_ val: Int, _ left: TreeNode?, _ right:
TreeNode?) {
*         self.val = val
*         self.left = left
*         self.right = right
*     }
* }
```

```swift
*/
class Solution {
    func diameterOfBinaryTree(_ root: TreeNode?) -> Int {
        var diameter = 0
        func dfs(_ root: TreeNode?) -> Int {
            guard let root = root else {
                return -1
            }

            if(root.left == nil && root.right == nil) {
                return 0
            }

            let left = 1 + dfs(root.left)
            let right = 1 + dfs(root.right)
            diameter = max(diameter,left+right)
            return max(left,right)
        }

        let _ = dfs(root)
        return diameter
    }
}
```