

Problem 687: Longest Univalue Path

Problem Information

Difficulty: Medium

Acceptance Rate: 43.18%

Paid Only: No

Tags: Tree, Depth-First Search, Binary Tree

Problem Description

Given the `root` of a binary tree, return _the length of the longest path, where each node in the path has the same value_. This path may or may not pass through the root.

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

****Example 1:****

****Input:**** root = [5,4,5,1,1,null,5] ****Output:**** 2 ****Explanation:**** The shown image shows that the longest path of the same value (i.e. 5).

****Example 2:****

****Input:**** root = [1,4,5,4,4,null,5] ****Output:**** 2 ****Explanation:**** The shown image shows that the longest path of the same value (i.e. 4).

****Constraints:****

* The number of nodes in the tree is in the range `[0, 104]` . * $-1000 \leq \text{Node.val} \leq 1000$ * The depth of the tree will not exceed `1000` .

Code Snippets

C++:

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *   int val;
 *   TreeNode *left;
 *   TreeNode *right;
 *   TreeNode() : val(0), left(nullptr), right(nullptr) {}
 *   TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}
 *   TreeNode(int x, TreeNode *left, TreeNode *right) : val(x), left(left),
right(right) {}
 * };
 */
class Solution {
public:
    int longestUnivaluePath(TreeNode* root) {

    }
};
```

Java:

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

```
}  
}
```

Python3:

```
# Definition for a binary tree node.  
# class TreeNode:  
#     def __init__(self, val=0, left=None, right=None):  
#         self.val = val  
#         self.left = left  
#         self.right = right  
class Solution:  
    def longestUnivaluePath(self, root: Optional[TreeNode]) -> int:
```