

[每日 LeetCode] 687. Longest Univalue Path

作者: [Hanseltu](#)

原文链接: <https://ld246.com/article/1559905013572>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

原文链接 [\[每日 LeetCode\] 687. Longest Univalue Path](#)

Description:

Given a binary tree, find 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 path between two nodes is represented by the number of edges between them.

Example 1:

Input:



Output: 2

Example 2:

Input:



Output: 2

Note: The given binary tree has not more than 10000 nodes. The height of the tree is not more than 1000.

思路：本题要求二叉树中相等结点的最长路径。考虑借助DFS，在进行递归的同时增加记录路径的变量，并及时更新最长路径。

C++代码

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

```
public:
    int longestUnivaluePath(TreeNode* root) {
        int res = 0;
        if (root)
            dfs(root, res);
        return res;
    }

    int dfs(TreeNode* root, int& res){
        int l = root->left ? dfs(root->left, res) : 0;
        int r = root->right ? dfs(root->right, res) : 0;

        int leftEdge = (root->left && root->left->val == root->val) ? l + 1 : 0;
        int rightEdge = (root->right && root->right->val == root->val) ? r + 1 : 0;
        res = max(res, leftEdge + rightEdge);
        return max(leftEdge, rightEdge);
    }
};
```

运行时间： 124ms

运行内存： 50M