

5398. Count Good Nodes in Binary Tree

[My Submissions \(/contest/biweekly-contest-26/problems/count-good-nodes-in-binary-tree/submissions/\)](/contest/biweekly-contest-26/problems/count-good-nodes-in-binary-tree/submissions/)

[Back to Contest \(/contest/biweekly-contest-26/\)](/contest/biweekly-contest-26/)

Given a binary tree `root`, a node X in the tree is named **good** if in the path from `root` to X there are no nodes with a value *greater than* X .

Return the number of **good** nodes in the binary tree.

User Accepted: 2413

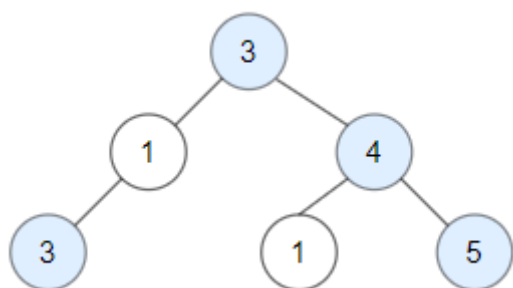
User Tried: 2565

Total Accepted: 2431

Total Submissions: 3188

Difficulty: **Medium**

Example 1:



Input: `root = [3,1,4,3,null,1,5]`

Output: 4

Explanation: Nodes in blue are **good**.

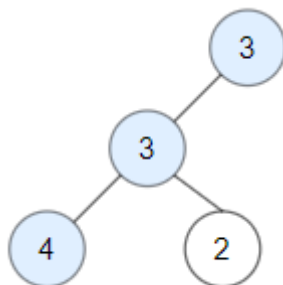
Root Node (3) is always a good node.

Node 4 -> (3,4) is the maximum value in the path starting from the root.

Node 5 -> (3,4,5) is the maximum value in the path

Node 3 -> (3,1,3) is the maximum value in the path.

Example 2:



Input: `root = [3,3,null,4,2]`

Output: 3

Explanation: Node 2 -> (3, 3, 2) is not good, because "3" is higher than it.

Example 3:

Input: root = [1]**Output:** 1**Explanation:** Root is considered as **good**.**Constraints:**

- The number of nodes in the binary tree is in the range $[1, 10^5]$.
- Each node's value is between $[-10^4, 10^4]$.

JavaScript ▼



```
1 ▾ /**
2  * Definition for a binary tree node.
3  * function TreeNode(val, left, right) {
4  *     this.val = (val===undefined ? 0 : val)
5  *     this.left = (left===undefined ? null : left)
6  *     this.right = (right===undefined ? null : right)
7  * }
8  */
9 ▾ /**
10  * @param {TreeNode} root
11  * @return {number}
12  */
13 ▾ var goodNodes = function(root) {
14
15 };
```

☐ Custom Testcase

Use Example Testcases

Run

Submit

Copyright © 2020 LeetCode

[Help Center \(/support/\)](/support/) | [Terms \(/terms/\)](/terms/) | [Privacy Policy \(/privacy/\)](/privacy/) [United States \(/region/\)](/region/)