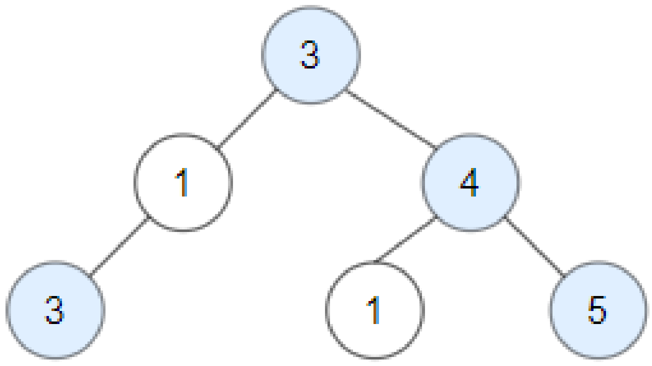
1448. Count Good Nodes in Binary Tree

Created	@November 27, 2022 9:26 AM
Difficulty	Medium
LC Url	https://leetcode.com/problems/count-good-nodes-in-binary-tree/
Importance	
Tag	NEET Tree
Video	

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.

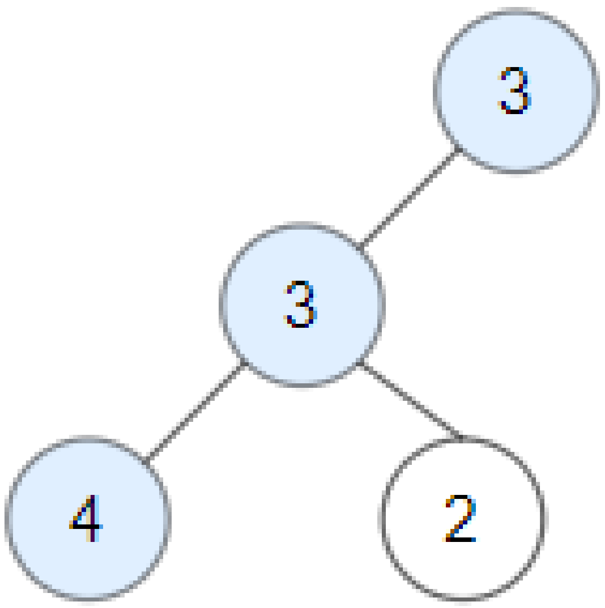
Return the number of **good** nodes in the binary tree.

Example 1:



```
Input: root = [3,1,4,3,null,1,5]
Output: 4
Explanation: Nodes in blue aregood.
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 asgood.
```

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]`.

Solution

前文 手把手刷二叉树总结篇 说过二叉树的递归分为「遍历」和「分解问题」两种思维模式，这道题需要用到「遍历」的思维，利用函数参数给子树传递信息。

函数参数 `pathMax` 记录从根节点到当前节点路径中的最大值，通过比较 `root.val` 和 `pathMax` 比较就可判断 `root` 节点是不是「好节点」。

```python
# 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:
    cnt = 0

    def goodNodes(self, root: TreeNode) -> int:
        self.traverse(root, root.val)
        return self.cnt

    def traverse(self, root, path_max):
        """
        二叉树遍历函数，pathMax 参数记录从根节点到当前节点路径中的最大值
        """
        if not root:
            return

        if path_max <= root.val:
            # 找到一个「好节点」
            self.cnt += 1

        # 更新路径上的最大值
        path_max = max(path_max, root.val)

        self.traverse(root.left, path_max)
        self.traverse(root.right, path_max)
```

Ref: labuladong