988. Smallest String Starting From Leaf

Solved

Medium | Topics | Companies

You are given the `root` of a binary tree where each node has a value in the range `[0, 25]` representing the letters `'a'` to `'z'`.

Return *the **lexicographically smallest** string that starts at a leaf of this tree and ends at the root*.

As a reminder, any shorter prefix of a string is **lexicographically smaller**.

- For example, `"ab"` is lexicographically smaller than `"aba"`.

A leaf of a node is a node that has no children.

Example 1:

```
class Solution {
    let abc = ["a", "b", "c", "d", "e", "f", "g", "h", "i", "j",
"k", "l", "m", "n", "o", "p", "q", "r", "s", "t", "u", "v", "w",
"x", "y", "z"]
    func smallestFromLeaf(_ root: TreeNode?) -> String {
        var minString = ""
        dfs(root, currentString: "", minString: &minString)
        return minString
    }
    func dfs(_ root: TreeNode?, currentString: String, minString:
inout String) {
        guard let root = root else {
            return
        }
        let currentString = abc[root.val] + currentString
        if root.left == nil && root.right == nil {
            if minString.count == 0 || currentString < minString {
                minString = currentString
            }
        } else {
            dfs(root.left, currentString: currentString,
minString: &minString)
            dfs(root.right, currentString: currentString,
minString: &minString)
        }
    }
}
```