

### 35. Check If a String Is a Valid Sequence from Root to Leaves Path in a Binary Tree

Given a binary tree where each path going from the root to any leaf form a valid

sequence, check if a given string is a valid sequence in such binary tree.

We get the given string from the concatenation of an array of integers arr and the

concatenation of all values of the nodes along a path results in a sequence in the given

binary tree.

#### **Code:**

```
class TreeNode:
    def __init__(self, val=0, left=None, right=None):
        self.val = val
        self.left = left
        self.right = right

def isValidSequence(root, arr):
    def dfs(node, index):
        if node is None:
            return False
        if index >= len(arr) or node.val != arr[index]:
            return False
        if node.left is None and node.right is None and index == len(arr) - 1:
            return True
        return dfs(node.left, index + 1) or dfs(node.right, index + 1)

    return dfs(root, 0)

root = TreeNode(0)
root.left = TreeNode(1)
root.right = TreeNode(0)
root.left.left = TreeNode(0)
root.left.right = TreeNode(1)
root.right.left = TreeNode(0)
root.left.left.right = TreeNode(1)
root.left.right.left = TreeNode(0)
root.left.right.right = TreeNode(0)

arr = [0, 1, 0, 1]
print(isValidSequence(root, arr))
```

### Output:

```
= RESTART: C:\User  
True  
|
```

### Time Complexity:

- $T(n) = O(n)$