

Description

Solution

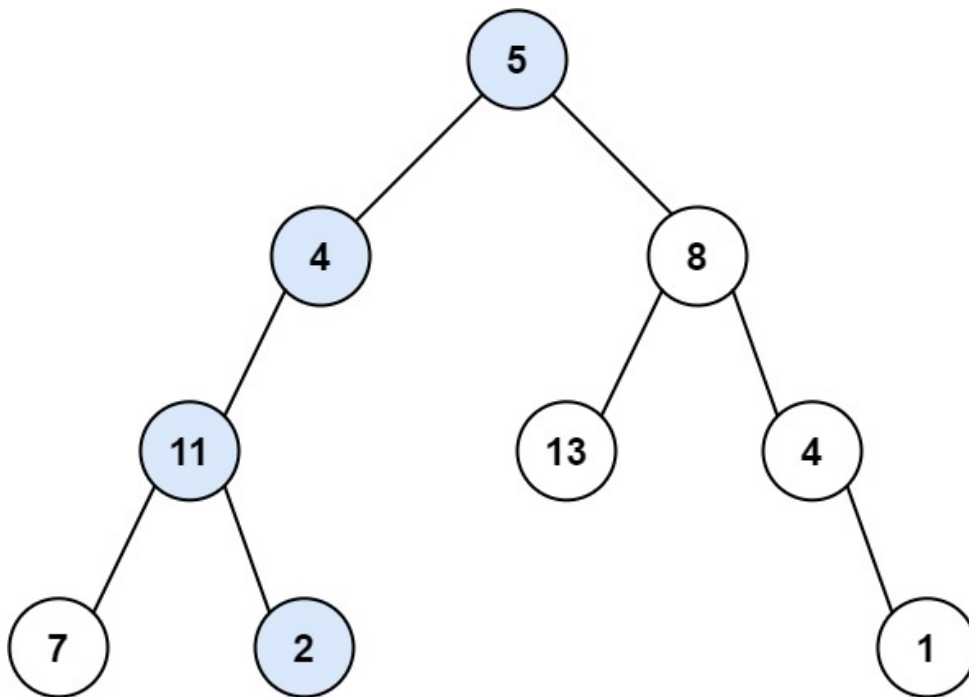
Discuss (999+)

Submissions

Go

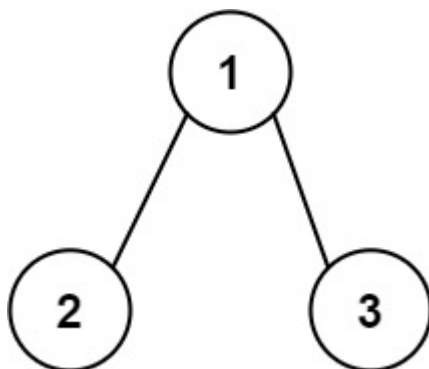
Given the `root` of a binary tree and an integer `targetSum`, return `true` if the tree has a **root-to-leaf** path such that adding up all the values along the path equals `targetSum`.

A **leaf** is a node with no children.

Example 1:

Input: `root = [5,4,8,11,null,13,4,7,2,null,null,null,1]`,
`targetSum = 22`

Output: `true`

Example 2:

Input: `root = [1,2,3]`, `targetSum = 5`

```

1  /**
2   * Definition
   * node.
3   * type TreeNode
4   * Val int
5   * Left *TreeNode
6   * Right *TreeNode
7   * }
8   */
9  func hasPathSum(root *TreeNode, targetSum int) bool {
10     if root == nil {
11         return false
12     }
13
14     if root.Left == nil && root.Right == nil {
15         return root.Val == targetSum
16     }
17
18     if root.Left != nil {
19         if hasPathSum(root.Left, targetSum - root.Val) {
20             return true
21         }
22     }
23
24     if root.Right != nil {
25         if hasPathSum(root.Right, targetSum - root.Val) {
26             return true
27         }
28     }
29
30     return false
31 }

```

Testcase Run Code Result

Accepted Runtime: 2 ms

Your input

[5,4,8,
22
..

Output

true
false

Expected

true
false

Problems

Pick One

< Prev

112/1998

Next >

[Use Example
Testcases](#)

?

Run Code