# 题目

给定一棵二叉树，其中每个节点都含有一个整数数值(该值或正或负)。设计一个算法，打印节点数值总和等于某个给定值的所有路径的数量。注意，路径不一定非得从二叉树的根节点或叶节点开始或结束，但是其方向必须向下(只能从父节点指向子节点方向)。

示例:

给定如下二叉树，以及目标和 sum = 22，

5

/ \

4 8

/ / \

11 13 4

/ \ / \

7 2 5 1

返回:

3

解释：和为22的路径有：[5,4,11,2], [5,8,4,5], [4,11,7]

提示：

节点总数<=10000

# 分析

class Solution {

public:

int res=0;

int pathSum(TreeNode\* root, int sum) {

if(!root)return 0;

dfs(root,sum);

pathSum(root->left,sum);

pathSum(root->right,sum);

return res;

}

void dfs(TreeNode\* root,int sum)

{

int cursum=0;

if(!root)return;

cursum+=root->val;

if(cursum==sum) res++;

dfs(root->left,sum-cursum);

dfs(root->right,sum-cursum);

}

};