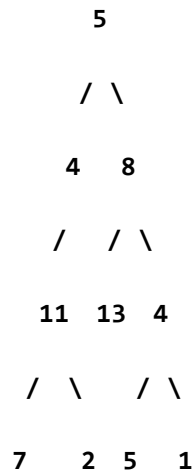


二叉树中和为某一值的路径

输入一棵二叉树和一个整数，打印出二叉树中节点值的和为输入整数的所有路径。从树的根节点开始往下一直到叶节点所经过的节点形成一条路径。

示例:

给定如下二叉树，以及目标和 `sum = 22`,



返回:

```
[
  [5,4,11,2],
  [5,8,4,5]
]
```

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode(int x) : val(x), left(NULL), right(NULL) {}
 * };
 */
class Solution {
private:
    vector<vector<int> > res;
    vector<int> temp;
public:
    vector<vector<int>> pathSum(TreeNode* root, int sum) {
        if(!root) return {};
    }
};
```

```
        recursion(root, sum);
        return res;
    }
    void recursion(TreeNode *root, int sum)
    {
        if(!root) return;
        temp.push_back(root -> val);
        sum -= root -> val;
        if(sum == 0 && !root -> left && !root -> right)
            res.push_back(temp);
        recursion(root -> left, sum); // 左
        recursion(root -> right, sum); // 右
        temp.pop_back(); // 函数退出之前先弹出当前节点
    }
};
```