

题目：输入一个整数数组，判断该数组是不是某二叉搜索树的后序遍历结果。如果是则返回 `true`，否则返回 `false`。假设输入的数组的任意两个

剑指 Offer——名企面试官精讲典型编程题（第2版）

数字都互不相同。例如，输入数组 {5, 7, 6, 9, 11, 10, 8}，则返回 `true`，因为这个整数序列是图 4.9 二叉搜索树的后序遍历结果。如果输入的数组是 {7, 4, 6, 5}，则由于没有哪棵二叉搜索树的后序遍历结果是这个序列，因此返回 `false`。

解

二叉搜索树的特点：中序遍历即 左中右 的顺序是递增的
后序遍历 是左右中

那么就先定位根节点是最后一个数，然后前面比根节点小的数都应该是左子树，左子树后面应该是比根节点大的右子树，如果不满足这种要求，就是 `false`

```
bool ispostOrderofBST(vector<int> sequence)
{
    if(sequence.empty())
        return false;
    int n=sequence.size();
    int root=sequence[n-1];
    vector<int> left,right;
    int i;
    for(i=0;i<n-1;i++)
    {
        if(sequence[i]>root)
            break;
        left.push_back(sequence[i]);
    }
    right.push_back(sequence[i]);
    return ispostOrderofBST(left) && ispostOrderofBST(right);
}
```

```
}
int j=i;
for(j;j<n-1;j++)
{
    if(sequence[j]<root)
        return false;
    right.push_back(sequence[j]);
}
bool l=true,r=true;
if(left.size()>0)
    l=ispostOrderofBST(left);
if(right.size()>0)
    r=ispostOrderofBST(right);
return l&& r;
}
```