

面试题33：二叉搜索树的后序遍历序列

题目描述

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的结果。如果是则输出Yes,否则输出No。假设输入的数组的任意两个数字都互不相同。

解题思路

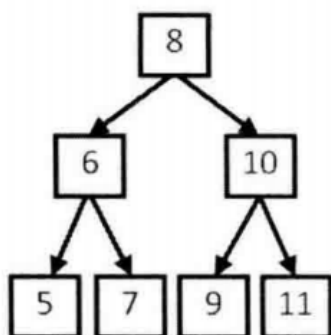


图 4.9 后序遍历序列{5,7,6,9,11,10,8}对应的二叉搜索树

在后序遍历得到的序列中，最后一个数字是树的根节点的值。数组中前面的数字可以分为两部分：第一部分是左子树节点的值，它们都比根节点的值小；第二部分是右子树节点的值，它们都比根节点的值大。

C++

```
1 class solution {  
2 public:
```

```

3      bool VerifySequenceOfBST(vector<int>
sequence) {
4          //非递归思想
5          int size = sequence.size(), i = 0;
6          if(size == 0) return false;
7          while(--size){
8              while(sequence[i++] <
sequence[size]);
9              while(sequence[i++] >
sequence[size]);
10             if(i < size) return false;
11             i = 0;
12         }
13         return true;
14     }
15 };

```

Java

```

1  public class Solution {
2      public boolean VerifySequenceOfBST(int []
sequence) {
3          if(sequence.length == 0)
4              return false;
5          if(sequence.length == 1)
6              return true;
7          return
verify(sequence, 0, sequence.length - 1);

```

```

8      }
9      //递归算法，后序遍历的序列最后一个根
10     public boolean verify(int[] a,int
start,int end){
11         //递归截至条件
12         if(start >= end)
13             return true;
14         //指针i，序列最右开始向左扫描，当找到i左边
部分小于根，i右边部分大于根
15         //并且左部分右部分同时满足以上条件，那么该
后序遍历可以构成二叉搜索树。
16         int i = end;
17         while(i >= 1 && a[i - 1] > a[end])
18             i--; //记录i
19         //假如现在i左边部分不满足a[i] < a[end]
则返回false
20         for(int j = i ; j >= start + 1;j--){
21             if(a[j - 1] > a[end])
22                 return false;
23         }
24         //当左右部分同时满足上述条件时
25         return verify(a,start,i-
1)&&verify(a,i,end-1);
26     }
27 }

```

Python 2.7.3

```
1  # -*- coding:utf-8 -*-
2  class Solution:
3  # Python 简单解法, 找到第一个比根节点大的整数的坐标, 左边肯定都比根小, 不需要再进一步判断
4  # 继续遍历坐标右边, 如果有比根小的数说明不是搜索树
5      def verifySequenceOfBST(self, sequence):
6          # write code here
7          if not sequence:
8              return False
9          root = sequence[-1]
10         index = 0
11         while sequence[index] < root:
12             index += 1
13         for i in sequence[index:]:
14             if i < root:
15                 return False
16         return True
```