# 33. 二叉搜索树的后序遍历序列

[NowCoder](https://www.nowcoder.com/practice/a861533d45854474ac791d90e447bafd?tpId=13&tqId=11176&tPage=1&rp=1&ru=/ta/coding-interviews&qru=/ta/coding-interviews/question-ranking&from=cyc_github)

## 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。假设输入的数组的任意两个数字都互不相同。

例如，下图是后序遍历序列 1,3,2 所对应的二叉搜索树。

## 解题思路

public boolean VerifySquenceOfBST(int[] sequence) {
 if (sequence == null || sequence.length == 0)
 return false;
 return verify(sequence, 0, sequence.length - 1);
}

private boolean verify(int[] sequence, int first, int last) {
 if (last - first <= 1)
 return true;
 int rootVal = sequence[last];
 int cutIndex = first;
 while (cutIndex < last && sequence[cutIndex] <= rootVal)
 cutIndex++;
 for (int i = cutIndex; i < last; i++)
 if (sequence[i] < rootVal)
 return false;
 return verify(sequence, first, cutIndex - 1) && verify(sequence, cutIndex, last - 1);
}