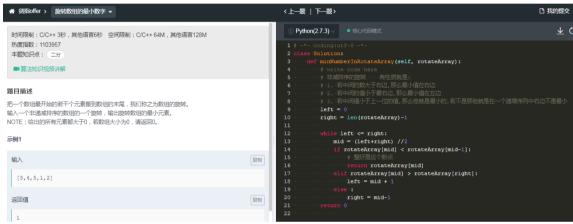
```
↑ 剑指offer > 二维数组中的查找 ▼
                                                             口相似的企业真题
                                                                              下—颗>
                                                                                 Python(2.7.3) > • 核心代码模式
 时间限制: C/C++ 1秒, 其他语言2秒 空间限制: C/C++ 64M, 其他语言128M
 热度指数: 2044457
 本颢知识点: 数组
 ■算法知识视频讲解
                                                                                     def Find(self, target, array):
题目描述
在一个二维数组中(每个一维数组的长度相同),每一行都按照从左到右递增的顺序排序,每一列都按照
从上到下递增的顺序排序。请完成一个函数,输入这样的一个二维数组和一个整数,判断数组中是否含有
                                                                                        j = len(array[0])-1
[1,2,8,9],
                                                                                        while i < len(array) and j >=0:
    if array[i][j] == target:
[2,4,9,12],
[4,7,10,13],
                                                                                           elif array[i][j] < target:</pre>
[6,8,11,15]
s
给定 target = 7,返回 true。
给定 target = 3,返回 false。
```


示例1

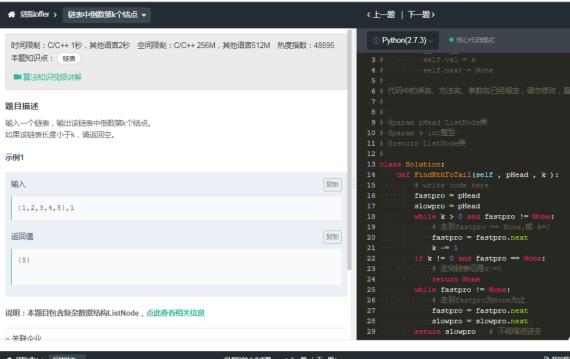
"We%20Are%20Happy"

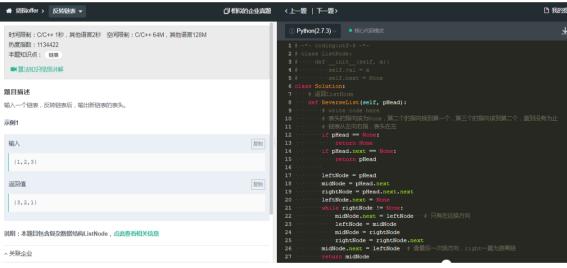
说明:本题目包含复杂数据结构ListNode,点此查看相关信息

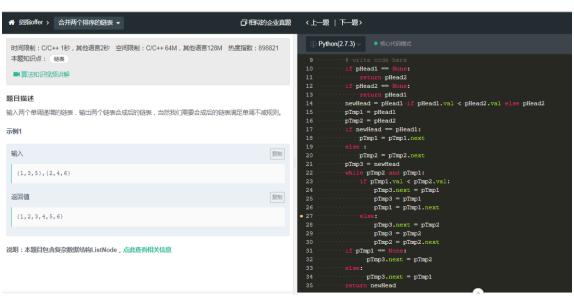


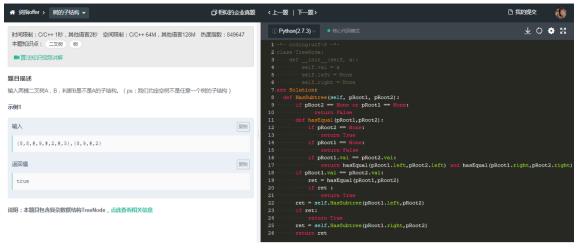






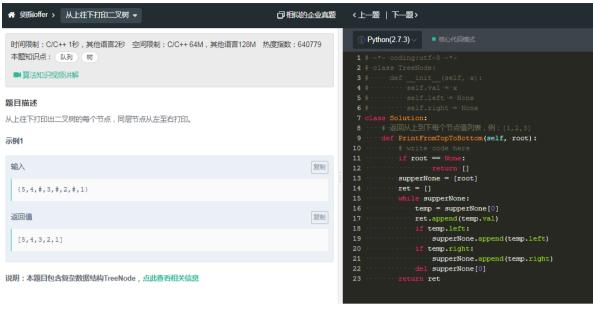






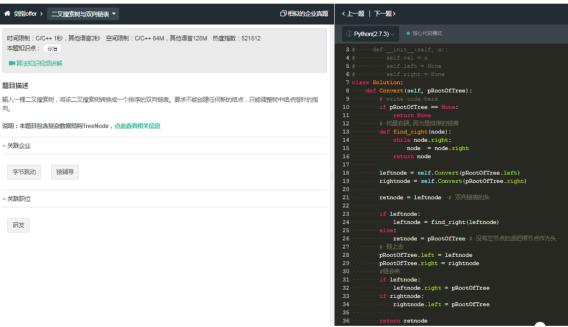


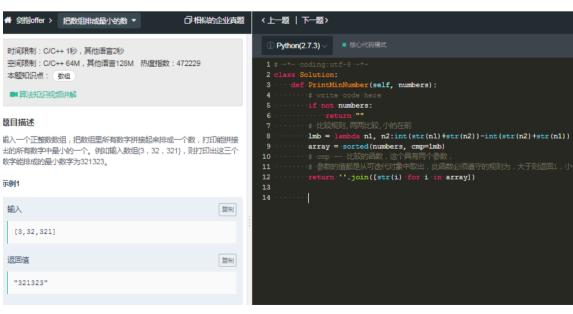


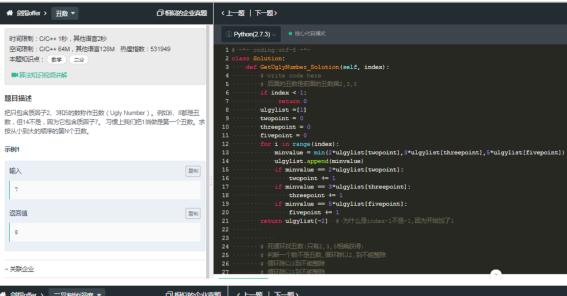


```
希 剣指offer > 二叉搜索树的后序遍历序列 ▼
                                                                                                                                                                                                                                                                                                                                   〈上—题 | 下—题〉
                                                                                                                                                                                                                                                             □相似的企业真题
                                                                                                                                                                                                                                                                                                                                                Python(2.7.3) ~ 核心代码模式
     时间限制: C/C++ 1秒, 其他语言2秒 空间限制: C/C++ 64M, 其他语言128M 热度指数: 776802
                                                                                                                                                                                                                                                                                                                                      1 # --*--coding:utf-8 --*-
2 class Solution:
3 ···def VerifySquenceOfBST(self, sequence):
4 ····· 左节点比根节点 小,右节点比根节点大
     本题知识点: 栈 树
       ■質決知识视频讲解
题月描述
                                                                                                                                                                                                                                                                                                                                                                               if sequence == []:
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。如果是则返回true,否则返回
                                                                                                                                                                                                                                                                                                                                                                            ·rootNum = sequence[-1] ··#·根节点
·del sequence[-1]
false。假设输入的数组的任意两个数字都互不相同。(ps:我们约定空树不是二叉搜素树)
                                                                                                                                                                                                                                                                                                                                                                            index=
示例1
                                                                                                                                                                                                                                                                                                                                                                                   procedure in the sequence of the sequence
                                                                                                                                                                                                                                                                                             复制
     输入
         [4,8,6,12,16,14,10]
     返回值
                                                                                                                                                                                                                                                                                             复制
                                                                                                                                                                                                                                                                                                                                                                           intercolor = self.VerifySquenceOfBST(sequence[:index])
if sequence[index:] == []:
    rightRet = True
else:
                                                                                                                                                                                                                                                                                                                                                                            if sequence[:index] == []:
....leftRet = True
.else:
         true
                                                                                                                                                                                                                                                                                                                                                                                            rightRet = self.VerifySquenceOfBST(sequence[index:])
```

```
希 剑指offer > 二叉树中和为某一值的路径 ▼
                                                            口相似的企业真题
                                                                         〈上—題 | 下—題〉
                                                                            〕 Python(2.7.3) ∨ ● 核心代码模式
时间限制: C/C++ 1秒, 其他语言2秒 空间限制: C/C++ 64M, 其他语言128M 热度指数: 702700
本题知识点: 材
                                                                              port copy
 ■算法知识视频讲解
                                                                          野日描述
                                                                                 if root = None
输入一颗二叉树的根节点和一个整数,按字典序打印出二叉树中结点值的和为输入整数的所有路径。路径定义为从树的根结点开
始往下一直到叶结点所经过的结点形成一条路径。
                                                                                  ret = []
示例1
                                                                                  supportNode = [root]
{10,5,12,4,7},22
返回值
                                                                  复制
[[10,5,7],[10,12]]
元(6)(2
输入
{10,5,12,4,7},15
返回值
                                                                  复制
```









```
# 到指offer > 和为S的连续正数序列 ▼ □相似的企业真颜 □相似的企业真颜 □ 时间限制:C/C++ 1秒,其他语言2秒 空间限制:C/C++ 64M,其他语言126M 热度指数:462455 本题知识点: 99 ▼ ■ 算法知识规须并解
```

题目描述

小明很喜欢数学,有一天他在做数学作业时,要求计算出9~16的和,他马上就写出了正确答案是100。但是他并不满足于此,他在想究竟有多少种连续的正数序列的和为100(至少包括两个数)。没多久,他就得到另一组连续正数和为100的序列:18,19,20,21,22。现在把问题交给你,你能不能也很快的找出所有和为S的连续正数序列? Good Luck!

返回值描述:

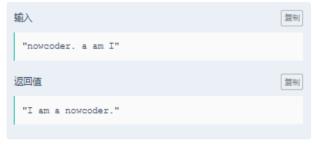
输出所有和为s的连续正数序列。序列内按照从小至大的顺序,序列间 按照开始数字从小到大的顺序

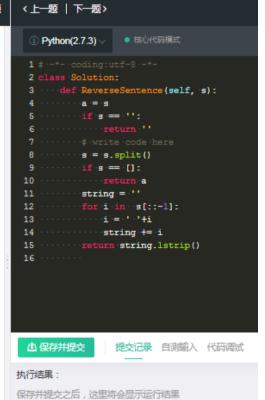


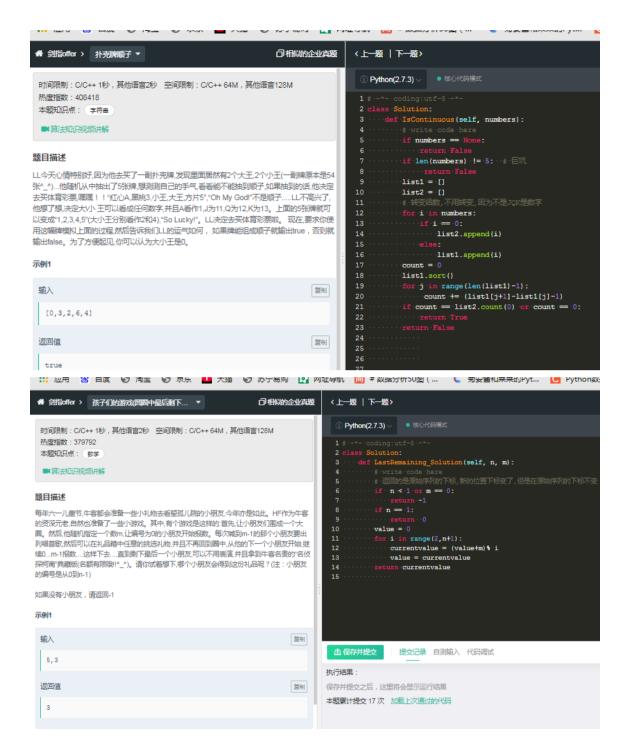


牛客最近来了一个新员工Fish,每天早晨总是会拿着一本英文杂志,写些句子在本子上。同事Cat对Fish写的内容颇感兴趣,有一天他向Fish借来翻看,但却读不懂它的意思。例如,"student. a am I"。后来才意识到,这家伙原来把句子单词的顺序翻转了,正确的句子应该是"I am a student."。Cat对——的翻转这些单词顺序可不在行,你能帮助他么?

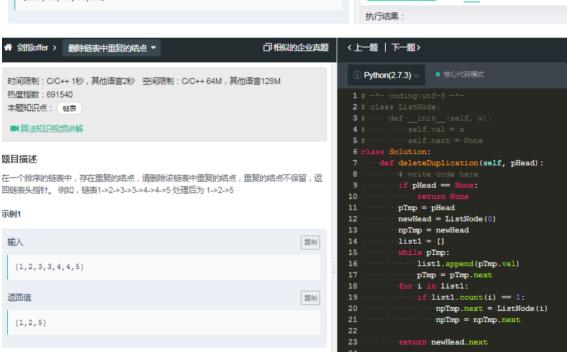
示例1

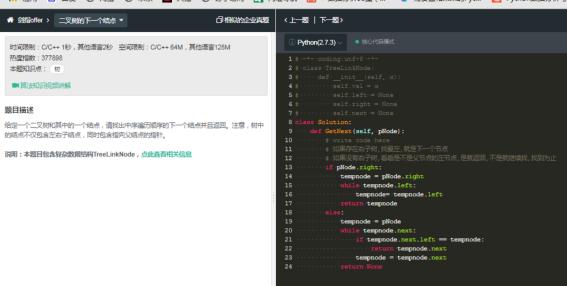




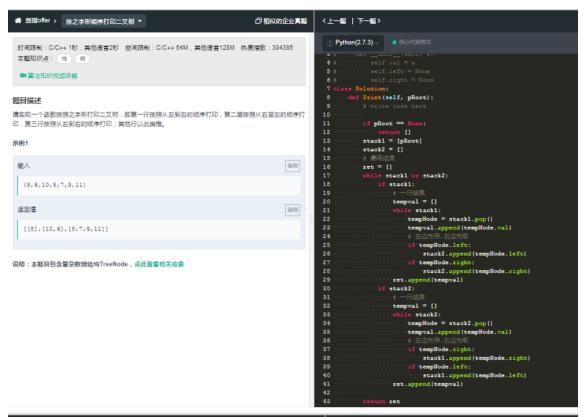


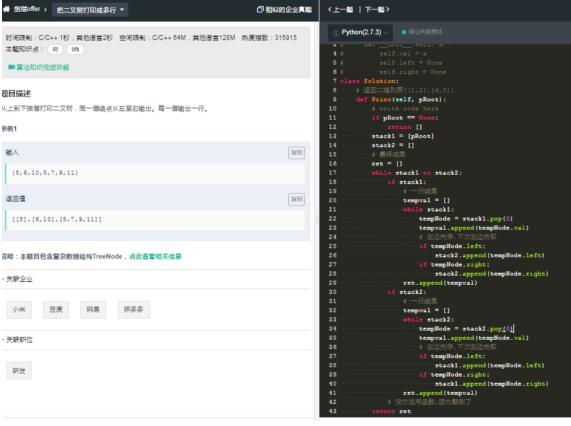






```
骨 剑指offer > 对称的二叉树 ▼
                                                       □相似的企业真题
                                                                         〈上—题 | 下—题〉
                                                                          ① Python(2.7.3) > 核心代码模式
 时间限制: C/C++ 1秒, 其他语言2秒 空间限制: C/C++ 64M, 其他语言128M
 执度指数:338721
 本题知识点: 材
 ■算法知识视频讲解
                                                                        请实现一个函数,用来判断一棵二叉树是不是对称的。注意,如果一个二叉树同此二叉树的镜
像是同样的, 定义其为对称的。
                                                                                   ...return True
.def .isMirror (left, right):
...if .left == None and right == None:
...return True
...elif .left == None or right == None:
示例1
 輸入
                                                               复制
  {8,6,6,5,7,7,5}
                                                                         17
18
19
20
                                                                                      · if left.val != right.val:
 返回值
                                                                                       -ret1 = isMirror(left.left,right.right)
                                                               复制
                                                                                      ret2 = isMirror(left.right,right.left)
                                                                                   return ret1 and ret2
return isMirror(pRoot.left,pRoot.right)
  true
```





```
骨 剑指offer > 序列化二叉树 ▼
                                                  ○相似的企业真题
                                                                  〈上 题 | 下 题>
                                                                   ① Python(2.7.3) > *** 核心代码模式
时间限制: C/C++ 1秒, 其他语言2秒 空间限制: C/C++ 64M, 其他语言128M
执度指数: 402848
本題知识点: 以列
                                                                          f Serialize(self, root):
 ■ 算法知识视频讲解
                                                                           retList = []
                                                                           def preOrder(root):
    if root == None
题目描述
                                                                                  retList.append('#')
请实现两个函数,分别用来序列化和反序列化二叉树
                                                                            retList.append(str(root.val))
    preOrder(root.left)
    preOrder(root.right)
二叉树的序列化是指:把一棵二叉树按照某种遍历方式的结果以某种格式保存为字符串,从而
使得内存中建立起来的二叉树可以持久保存。序列化可以基于先序、中序、后序、层序的二叉
树遍历方式来进行修改,序列化的结果是一个字符串,序列化时通过 某种符号表示空节点
(#),以!表示一个结点值的结束(value!)。
                                                                           preOrder(root)
                                                                            return ''.'.join(retList)
二叉树的反序列化是指:根据某种遍历顺序得到的序列化字符串结果str,重构二叉树。
                                                                        def Deserialize(self, s):
例如,我们可以把一个只有根节点为1的二叉树序列化为"1,",然后通过自己的函数来解析回这
                                                                            resList = s.split()
                                                                             ef -depreOrder():
---if-resList == -[]:
示例1
                                                                              rootval = resList.pop(0)
輸入
                                                         复制
                                                                               if rootval == '#':
  {8,6,10,5,7,9,11}
                                                                               node = TreeNode(int(rootval))
                                                                               ·leftNode = depreOrder()
                                                                               rightNode = depreOrder()
                                                         复制
                                                                               node.left = leftNode
                                                                               node.right = rightNode
  {8,6,10,5,7,9,11}
                                                                                   rn node
                                                                  38
39
                                                                           pRoot = depreOrder()
return pRoot
道明·太照日句今日为姚屏结约TraaNorla。 古山本美妇学后白
```

```
时间限制:C/C++ 1秒,其他语言2秒 空间限制:C/C++ 64M,其他语言128M
热度指数:496526
本题知识点: 材
■ 算法知识视频并解
```

题目描述

给定一棵二叉搜索树,请找出其中的第k小的TreeNode结点。

示例1

