

执行结果: 通过 [显示详情](#)

执行用时: 144 ms, 在所有 Python3 提交中击败了 85.38% 的用户

内存消耗: 19 MB, 在所有 Python3 提交中击败了 71.21% 的用户

炫耀一下:



[写题解, 分享我的解题思路](#)

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	144 ms	19 MB	Python3
1 分钟前	执行出错	N/A	N/A	Python3
2 分钟前	执行出错	N/A	N/A	Python3

1.

```
class MyHashSet:
```

```
    def __init__(self):
```

```
        """
```

```
        Initialize your data structure here.
```

```
        """
```

```
        self.col = set()
```

```
    def add(self, key: int) -> None:
```

```
        self.col.add(key)
```

```
    def remove(self, key: int) -> None:
```

```
        if self.contains(key):
```

```
            self.col.remove(key)
```

```
    def contains(self, key: int) -> bool:
```

```
        """
```

```
        Returns true if this set contains the specified element
```

```
        """
```

```
        if key in self.col:
```

```
            return True
```

```
        else:
```

```
            return False
```

```
1 class MyHashSet:
```

```
2
```

```
3     def __init__(self):
```

```
4         """
```

```
         Initialize your data structure here.
```

```
         """
```

```
         self.col = set()
```

```
5
```

```
6     def add(self, key: int) -> None:
```

```
7         self.col.add(key)
```

```
8
```

```
9     def remove(self, key: int) -> None:
```

```
10         if self.contains(key):
```

```
11             self.col.remove(key)
```

```
12
```

```
13     def contains(self, key: int) -> bool:
```

```
14         """
```

```
         Returns true if this set contains the specified element
```

```
         """
```

```
         if key in self.col:
```

```
15             return True
```

```
16         else:
```

```
17             return False
```

```
18
```

```
19
```

```
20
```

```
21
```

```
22
```

```
23
```

```
24
```

```
25
```

```
26
```

题目描述

评论 (7.7k)

题解 (10.7k)

提交记录

Python3

智能模式

模拟面试

i

执行结果: 通过 [显示详情](#)

执行用时: 48 ms, 在所有 Python3 提交中击败了 18.24% 的用户

内存消耗: 14.9 MB, 在所有 Python3 提交中击败了 57.25% 的用户

炫耀一下:

写题解, 分享我的解题思路

```

1 class Solution:
2     def twoSum(self, nums: List[int], target: int) -> List[int]:
3         idx01 = 0
4         idx02 = 0
5         dic = {}
6         i = 0
7         for i in range(len(nums)):
8             dic[nums[i]] = i
9         i = 0
10        for i in range(len(nums)):
11            idx01 = i
12            if target - nums[idx01] in dic:
13                idx02 = dic[target - nums[idx01]]
14                if idx01 != idx02:
15                    break
16        return [idx01, idx02]

```

题目描述

评论 (4.7k)

题解 (5.2k)

提交记录

Python3

智能模式

模拟面试

i

执行结果: 通过 [显示详情](#)

执行用时: 92 ms, 在所有 Python3 提交中击败了 9.26% 的用户

内存消耗: 14.7 MB, 在所有 Python3 提交中击败了 87.06% 的用户

炫耀一下:

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	92 ms	14.7 MB	Python3
几秒前	通过	60 ms	15.1 MB	Python3
1分钟前	解答错误	N/A	N/A	Python3

```

9         t1 = l1
10        base = 1
11        while t1 is not None:
12            v1 = v1 + base * t1.val
13            t1 = t1.next
14            base = base * 10
15        print(v1)
16        v2 = 0
17        t2 = l2
18        base = 1
19        while t2 is not None:
20            v2 = v2 + base * t2.val
21            t2 = t2.next
22            base = base * 10
23        print(v2)
24        s = v1 + v2
25        print(s)
26        st = str(s)
27        L = len(st) - 1
28        t = ListNode(int(st[0]))
29        i = 1
30        while i <= L:
31            ln = ListNode(int(st[i]), t)
32            t = ln
33            i = i + 1
34        return t

```

```

class Solution:
    def addTwoNumbers(self, l1: ListNode, l2: ListNode) -> ListNo
de:

    v1 = 0
    t1 = l1
    base = 1
    while t1 is not None:
        v1 = v1 + base * t1.val
        t1 = t1.next
        base = base * 10
    print(v1)
    v2 = 0
    t2 = l2
    base = 1
    while t2 is not None:
        v2 = v2 + base * t2.val
        t2 = t2.next
        base = base * 10

```

```
print(v2)
s = v1 + v2
print(s)
st = str(s)
L = len(st) - 1
t = ListNode(int(st[0]))
i = 1
while i<=L:
    ln = ListNode( int(st[i]), t )
    t = ln
    i = i + 1
return t
```

🔍 题目描述

💬 评论 (4.1k)

🔑 题解 (5.1k)

📝 提交记录

Python3

智能模式

执行结果: 通过 [显示详情](#)

执行用时: **48 ms**, 在所有 Python3 提交中击败了 **99.40%** 的用户

内存消耗: **14.9 MB**, 在所有 Python3 提交中击败了 **74.11%** 的用户

炫耀一下:

[📝 写题解, 分享我的解题思路](#)

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	48 ms	14.9 MB	Python3
几秒前	通过	68 ms	15.1 MB	Python3

```
1 class Solution:
2     def lengthOfLongestSubstring(self, s: str) -> int:
3         L = 0
4         maxL = 0
5         sub = ""
6         for ch in s:
7             if ch not in sub:
8                 L = L + 1
9                 sub = sub + ch
10                if L > maxL:
11                    maxL = L
12            else:
13                pos = sub.find(ch)
14                #print(pos)
15                sub = sub[pos+1:] + ch
16                #print(sub)
17                #sub = ch + ""
18                L = len(sub)
19                if L > maxL:
20                    maxL = L
21
22        return maxL
23
```

4.

```
class Solution:
    def lengthOfLongestSubstring(self, s: str) -> int:
        L = 0
        maxL = 0
        sub = ""
        for ch in s:
            if ch not in sub:
                L = L + 1
                sub = sub + ch
                if L > maxL:
                    maxL = L
            else:
                pos = sub.find(ch)
                #print(pos)
                sub = sub[pos+1:] + ch
```

```

        #print(sub)
        #sub = ch + ""
        L = len(sub)
        if L > maxL:
            maxL = L

    return maxL

```

题目描述 | 评论 (2.5k) | 题解 (2.5k) | 提交记录

Python3 | 智能模式 | 模拟面试

执行结果: **通过** 显示详情 >

执行用时: **52 ms** , 在所有 Python3 提交中击败了 **70.69%** 的用户

内存消耗: **15.1 MB** , 在所有 Python3 提交中击败了 **30.46%** 的用户

炫耀一下:

写题解, 分享我的解题思路

```

1 class Solution:
2     def findMedianSortedArrays(self, nums1: List[int], nums2: List[int]) -> float:
3         merge = nums1
4         for i in nums2:
5             merge.append(i)
6         merge = sorted(merge)
7         L = len(merge)
8         if L % 2 == 1:
9             return merge[ L//2 ]
10        else:
11            return (merge[ L//2 ] + merge[ L//2 - 1])/2
12
13
14
15

```

```

class Solution1:
    def findMedianSortedArrays(self, nums1: list[int], nums2: list[int]) -> float:
        merge = nums1
        for i in nums2:
            merge.append(i)
        merge = sorted(merge)
        L = len(merge)
        if L % 2 == 1:
            return merge[ L//2 ]
        else:
            return (merge[ L//2 ] + merge[ L//2 - 1])/2

```

题目描述 | 评论 (2.7k) | 题解 (2.7k) | 提交记录

Python3 | 智能模式 | 模拟面试

执行结果: **通过** 显示详情 >

执行用时: **928 ms** , 在所有 Python3 提交中击败了 **69.35%** 的用户

内存消耗: **15.2 MB** , 在所有 Python3 提交中击败了 **39.37%** 的用户

炫耀一下:

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	928 ms	15.2 MB	Python3
几秒前	通过	900 ms	15.3 MB	Python3

```

1 class Solution:
2     def longestPalindrome(self, s: str) -> str:
3         if len(s) <= 1:
4             return s
5         if len(s) == 2 and s[0] == s[1]:
6             return s
7         L = len(s)
8
9         target = ""
10        for i in range(L):
11            print(i)
12            left, right, sub = self.expand(s, i, i)
13            if len(sub) >= len(target):
14                target = sub
15            left, right, sub = self.expand(s, i, i+1)
16            if len(sub) >= len(target):
17                target = sub
18        return target
19
20    def expand(self, s, left, right):
21        while left>=0 and right<len(s) and s[left] == s[right]:
22            left = left - 1
23            right = right + 1
24        return left, right, s[left+1:right]

```

```

class Solution:
    def longestPalindrome(self, s: str) -> str:
        if len(s) <= 1:
            return s
        if len(s) == 2 and s[0] == s[1]:
            return s
        L = len(s)

        target = ""
        for i in range(L):
            print(i)
            left, right, sub = self.expand(s, i, i)
            if len(sub) >= len(target):
                target = sub
            left, right, sub = self.expand(s, i, i+1)
            if len(sub) >= len(target):
                target = sub
            #print(target)
        return target

    def expand(self, s, left, right):
        while left>=0 and right<len(s) and s[left] == s[right]:
            left = left - 1
            right = right + 1
        #print('--', left+1, right-1, s[left+1:right])
        return left, right, s[left+1:right]

```

力扣

学习

题库

讨论

竞赛

求职

商店

新功能

下载 App

题目描述

评论 (3.3k)

题解 (3.8k)

提交记录

Python3

智能模式

模拟

执行结果: 通过

显示详情 >

执行用时: 44 ms

在所有 Python3 提交中击败了 52.61% 的用户

内存消耗: 14.7 MB

在所有 Python3 提交中击败了 75.59% 的用户

炫耀一下:

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	44 ms	14.7 MB	Python3
4 分钟前	解答错误	N/A	N/A	Python3
8 分钟前	解答错误	N/A	N/A	Python3

```

1 import math
2 class Solution:
3     def reverse(self, x: int) -> int:
4         if x>0:
5             st = str(x)
6             base = 1
7             v = 0
8             for i in st:
9                 v = v + base*int(i)
10                base = base * 10
11            if v > math.pow(2, 31)-1:
12                return 0
13            return v
14        else:
15            x = -x
16            st = str(x)
17            base = 1
18            v = 0
19            for i in st:
20                v = v + base*int(i)
21                base = base * 10
22            v = -v
23            if v < -math.pow(2, 31):
24                return 0

```

您上次编辑到这里, 代码已从您浏览器本地的临时存储中恢复了 还原默认代码模版

```
import math
```

```

class Solution:
    def reverse(self, x: int) -> int:
        if x>0:
            st = str(x)
            base = 1
            v = 0
            for i in st:
                v = v + base*int(i)
                base = base * 10
            if v > math.pow(2, 31)-1:
                return 0
            return v
        else:
            x = -x
            st = str(x)
            base = 1
            v = 0
            for i in st:
                v = v + base*int(i)
                base = base * 10
            v = -v
            if v < -math.pow(2, 31):
                return 0
            return v

```

题目描述

评论 (2.6k)

题解 (3.6k)

提交记录

Python3

智能模式

执行结果: 通过 [显示详情](#)

执行用时: 92 ms, 在所有 Python3 提交中击败了 19.63% 的用户

内存消耗: 15.1 MB, 在所有 Python3 提交中击败了 5.30% 的用户

炫耀一下:

写题解, 分享我的解题思路

```

1 class Solution:
2     def isPalindrome(self, x: int) -> bool:
3         if x<0:
4             return False
5         else:
6             num = x
7             v = 0
8             while num > 0:
9                 n = num%10
10                v = v*10 + n
11                num = num // 10
12            print(v)
13            return v==x
14

```

```

class Solution:
    def isPalindrome(self, x: int) -> bool:
        if x<0:
            return False
        else:
            num = x
            v = 0
            while num > 0:
                n = num%10

```

```

        v = v*10 + n
        num = num // 10
    print(v)
    return v==x

```

力扣 学习 题库 讨论 竞赛 求职 商店

新功能 下载 App Plus 会员 中

题目描述 评论 (2.5k) 题解 (2.7k) 提交记录 Python3 智能模式 模拟面试 i {}

执行结果: **通过** 显示详情

执行用时: **56 ms**, 在所有 Python3 提交中击败了 **67.33%** 的用户

内存消耗: **14.9 MB**, 在所有 Python3 提交中击败了 **55.56%** 的用户

炫耀一下:

与题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	56 ms	14.9 MB	Python3

```

1 class Solution:
2     def romanToInt(self, s: str) -> int:
3         dic = { 'I':1, 'V':5, 'X':10, 'L':50, 'C':100, 'D':500, 'M':1000}
4         size = len(s)
5         i = 0
6         re = 0
7         while i < size:
8             cur = dic[s[i]]
9             if i+1<size:
10                nex = dic[s[i+1]]
11                #
12                if cur >= nex:
13                    re = re + cur
14                    i = i + 1
15                else:
16                    re = re + nex - cur
17                    i = i + 2
18            else:
19                re = re + cur
20                i = i + 1
21
22        return re

```

```

class Solution:
    def romanToInt(self, s: str) -> int:
        dic = { 'I':1, 'V':5, 'X':10, 'L':50, 'C':100, 'D':500, 'M':1000}
        size = len(s)
        i = 0
        re = 0
        while i < size:
            cur = dic[s[i]]
            if i+1<size:
                nex = dic[s[i+1]]
                #
                if cur >= nex:
                    re = re + cur
                    i = i + 1
                else:
                    re = re + nex - cur
                    i = i + 2
            else:
                re = re + cur
                i = i + 1

        return re

```

题目描述

评论 (2.3k)

题解 (2.8k)

提交记录

Python3

智能模式

模拟面试

i

执行结果: **通过** [显示详情](#)

执行用时: 40 ms, 在所有 Python3 提交中击败了 73.03% 的用户

内存消耗: 14.9 MB, 在所有 Python3 提交中击败了 51.55% 的用户

炫耀一下:



写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	40 ms	14.9 MB	Python3
几秒前	执行出错	N/A	N/A	Python3

```

1 class Solution:
2     def longestCommonPrefix(self, strs: List[str]) -> str:
3         if len(strs) < 1:
4             return ""
5         pre = ""
6         i = 0
7         while True:
8             flag = True
9             if i >= len(strs[0]):
10                 break
11             ch = strs[0][i]
12             for sub in strs:
13                 if i >= len(sub):
14                     flag = False
15                     break
16                 if sub[i] != ch:
17                     flag = False
18                     break
19             i = i + 1
20             if flag:
21                 pre = pre + ch
22             else:
23                 break
24         return pre

```

class Solution:

```

def longestCommonPrefix(self, strs: List[str]) -> str:
    if len(strs) < 1:
        return ""
    pre = ""
    i = 0
    while True:
        flag = True
        if i >= len(strs[0]):
            break
        ch = strs[0][i]
        for sub in strs:
            if i >= len(sub):
                flag = False
                break
            if sub[i] != ch:
                flag = False
                break
        i = i + 1
        if flag:
            pre = pre + ch
        else:
            break
    return pre

```


力扣

学习

题库

讨论

竞赛

求职

商店

新功能

下载 App

Plus 会员

题目描述

评论 (2.9k)

题解 (4.1k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过

显示详情 >

执行用时: 40 ms

在所有 Python3 提交中击败了 66.68% 的用户

内存消耗: 14.6 MB

在所有 Python3 提交中击败了 88.47% 的用户

炫耀一下:

微信

微博

QQ

豆瓣

领英

写题解, 分享我的解题思路

```

1 class Solution:
2     def isValid(self, s: str) -> bool:
3         dic = { '(':1, ')':-1, '[':2, ']':-2, '{':3, '}':-3 }
4         stk = []
5         for ch in s:
6             v = dic[ch]
7             if v > 0:
8                 stk.append(v)
9             else:
10                if len(stk) == 0:
11                    return False
12                flag = v + stk.pop()
13                if flag != 0:
14                    return False
15            return len(stk) == 0

```

```

class Solution:
    def isValid(self, s: str) -> bool:
        dic = { '(':1, ')':-1, '[':2, ']':-2, '{':3, '}':-3 }
        stk = []
        for ch in s:
            v = dic[ch]
            if v > 0:
                stk.append(v)
            else:
                if len(stk) == 0:
                    return False
                flag = v + stk.pop()
                if flag != 0:
                    return False
        return len(stk) == 0

```

力扣

学习

题库

讨论

竞赛

求职

商店

新功能

下载 App

Plus 会员

题目描述

评论 (1.5k)

题解 (2.4k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过

显示详情 >

执行用时: 44 ms

在所有 Python3 提交中击败了 66.25% 的用户

内存消耗: 14.9 MB

在所有 Python3 提交中击败了 39.32% 的用户

炫耀一下:

微信

微博

QQ

豆瓣

领英

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	44 ms	14.9 MB	Python3
1 分钟前	解答错误	N/A	N/A	Python3

```

1 # Definition for singly-linked list.
2 # class ListNode:
3 #     def __init__(self, val=0, next=None):
4 #         self.val = val
5 #         self.next = next
6 class Solution:
7     def mergeTwoLists(self, l1: ListNode, l2: ListNode) -> ListNode:
8         if l1 is None:
9             return l2
10        if l2 is None:
11            return l1
12        #
13        h = None
14        h1 = l1
15        h2 = l2
16        while h1 and h2:
17            if h1.val > h2.val:
18                tmp = ListNode(h2.val, h)
19                h = tmp
20                h2 = h2.next
21            else:
22                tmp = ListNode(h1.val, h)
23                h = tmp
24                h1 = h1.next
25        while h1:
26            tmp = ListNode(h1.val, h)

```

```

# Definition for singly-linked list.
# class ListNode:
#     def __init__(self, val=0, next=None):

```

```

#         self.val = val
#         self.next = next
class Solution:
    def mergeTwoLists(self, l1: ListNode, l2: ListNode) -> ListNo
de:

    if l1 is None:
        return l2
    if l2 is None:
        return l1
    #
    h = None
    h1 = l1
    h2 = l2
    while h1 and h2:
        if h1.val > h2.val:
            tmp = ListNode(h2.val, h)
            h = tmp
            h2 = h2.next
        else:
            tmp = ListNode(h1.val, h)
            h = tmp
            h1 = h1.next
    while h1:
        tmp = ListNode(h1.val, h)
        h = tmp
        h1 = h1.next
    while h2:
        tmp = ListNode(h2.val, h)
        h = tmp
        h2 = h2.next
    #
    hh = None
    while h:
        tmp = ListNode(h.val, hh)
        hh = tmp
        h = h.next

    return hh

```

执行结果: 通过 显示详情

执行用时: 5768 ms, 在所有 Python3 提交中击败了 5.01% 的用户

内存消耗: 15.8 MB, 在所有 Python3 提交中击败了 40.33% 的用户

炫耀一下:



写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	5768 ms	15.8 MB	Python3
几秒前	通过	5600 ms	15.8 MB	Python3

```

1 class Solution:
2     def removeDuplicates(self, nums: List[int]) -> int:
3         L = len(nums)
4         if L < 1:
5             return 0
6         if L == 1:
7             return 1
8         i = 0
9         while i < L-1:
10             cur = nums[i]
11             j = i + 1
12             nex = nums[j]
13             while nex <= cur and j < L-1:
14                 j = j + 1
15                 nex = nums[j]
16             if nex <= cur:
17                 break
18             else:
19                 nums[j] = nums[i+1]
20                 nums[i+1] = nex
21                 i = i + 1
22             #print(nums)
23             return i+1
24

```

```

class Solution:
    def removeDuplicates(self, nums: List[int]) -> int:
        L = len(nums)
        if L < 1:
            return 0
        if L == 1:
            return 1
        i = 0
        while i < L-1:
            cur = nums[i]
            j = i + 1
            nex = nums[j]
            while nex <= cur and j < L-1:
                j = j + 1
                nex = nums[j]
            if nex <= cur:
                break
            else:
                nums[j] = nums[i+1]
                nums[i+1] = nex
                i = i + 1
        #print(nums)
        return i+1

```

力扣

学习 题库 讨论 竞赛 求职 商店

新功能

下载 App

Plus 会员

题目描述

评论 (2.0k)

题解 (2.6k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过 [显示详情](#)

执行用时: 32 ms, 在所有 Python3 提交中击败了 94.13% 的用户
 内存消耗: 15 MB, 在所有 Python3 提交中击败了 5.14% 的用户
 炫一下:

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	32 ms	15 MB	Python3

```

1 class Solution:
2     def removeElement(self, nums: List[int], val: int) -> int:
3         L = len(nums)
4         N = 0
5         i = 0
6         while i < L:
7             if nums[i] == val:
8                 N = N + 1
9                 i = i + 1
10
11        j = 0
12        u = 0
13        while j < L - 1:
14            k = j
15            cur = nums[j]
16            while cur == val and k < L:
17                self.update(nums, j)
18                k = k + 1
19            cur = nums[j]
20            u = u + 1
21            j = j + 1
22        print(nums)
23        print(L - N)
24
25        return L - N
26

```

```

class Solution:
    def removeElement(self, nums: List[int], val: int) -> int:
        L = len(nums)
        N = 0
        i = 0
        while i < L:
            if nums[i] == val:
                N = N + 1
                i = i + 1

        j = 0
        u = 0
        while j < L - 1:
            k = j
            cur = nums[j]
            while cur == val and k < L:
                self.update(nums, j)
                k = k + 1
            cur = nums[j]
            u = u + 1
            j = j + 1
        print(nums)
        print(L - N)

        return L - N

    def update(self, nums, i):
        L = len(nums)
        while i < L - 1:

```

```
nums[i] = nums[i+1]
i = i + 1
```

力扣

学习

题库

讨论

竞赛

求职

商店

新功能

下载 App

Plus 会员

题目描述

评论 (2.1k)

题解 (2.2k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过 [显示详情](#)

执行用时: 32 ms, 在所有 Python3 提交中击败了 96.68% 的用户

内存消耗: 14.9 MB, 在所有 Python3 提交中击败了 76.76% 的用户

炫耀一下:

[写题解, 分享我的解题思路](#)

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	32 ms	14.9 MB	Python3

```
1 class Solution:
2     def strStr(self, haystack: str, needle: str) -> int:
3         L1 = len(haystack)
4         L2 = len(needle)
5         if L1 < L2:
6             return -1
7         if L1==0 and L1==L2:
8             return 0
9         i = 0
10        while i<=L1-L2:
11            flag = True
12            j = 0
13            while j < L2:
14                if haystack[i+j] != needle[j]:
15                    flag = False
16                    break
17                j = j + 1
18            if flag:
19                return i
20            i = i + 1
21        return -1
```

```
class Solution:
    def strStr(self, haystack: str, needle: str) -> int:
        L1 = len(haystack)
        L2 = len(needle)
        if L1 < L2:
            return -1
        if L1==0 and L1==L2:
            return 0
        i = 0
        while i<=L1-L2:
            flag = True
            j = 0
            while j < L2:
                if haystack[i+j] != needle[j]:
                    flag = False
                    break
                j = j + 1
            if flag:
                return i
            i = i + 1
        return -1
```

力扣

学习

题库

讨论

竞赛

求职

商店

新功能

下载 App

Plus 会员

题目描述

评论 (1.9k)

题解 (2.6k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过

显示详情 >

执行用时: 32 ms

在所有 Python3 提交中击败了 94.86% 的用户

内存消耗: 15.3 MB

在所有 Python3 提交中击败了 48.96% 的用户

炫耀一下:

写题解, 分享我的解题思路

```

1 class Solution:
2     def searchInsert(self, nums: List[int], target: int) -> int:
3         L = len(nums)
4         i = 0
5         pv = nums[i]
6         while i < L:
7             if nums[i] == target:
8                 return i
9             if target < nums[i]:
10                return i
11                i = i + 1
12            return L
13

```

```

class Solution:
    def searchInsert(self, nums: List[int], target: int) -> int:
        L = len(nums)
        i = 0
        pv = nums[i]
        while i < L:
            if nums[i] == target:
                return i
            if target < nums[i]:
                return i
            i = i + 1
        return L

```

力扣

学习

题库

讨论

竞赛

求职

商店

新功能

下载 App

Plus 会员

题目描述

评论 (2.2k)

题解 (1.8k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过

显示详情 >

执行用时: 104 ms

在所有 Python3 提交中击败了 6.64% 的用户

内存消耗: 15.2 MB

在所有 Python3 提交中击败了 5.01% 的用户

炫耀一下:

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	104 ms	15.2 MB	Python3
几秒前	通过	104 ms	15.1 MB	Python3
几秒前	解答错误	N/A	N/A	Python3

```

1 class Solution:
2     def countAndSay(self, n: int) -> str:
3         if n == 1:
4             return "1"
5         line = "1"
6         i = 1
7         while i < n:
8             # 描述
9             pre = ""
10            pos = []
11            val = []
12            # cur = line[0]
13            j = 0
14            while j < len(line):
15                ch = line[j]
16                if ch != pre:
17                    pos.append(j)
18                    val.append(ch)
19                    pre = ch
20                j = j + 1
21
22            # 转换成字符串
23            n1 = ""
24            k = 0
25            while k < len(pos):
26                sub = ""

```

```

class Solution:
    def countAndSay(self, n: int) -> str:
        if n == 1:
            return "1"
        line = "1"
        i = 1

```

```

while i < n:
    # 描述
    pre = ""
    pos = []
    val = []
    #cur = line[0]
    j = 0
    while j < len(line):
        ch = line[j]
        if ch != pre:
            pos.append(j)
            val.append(ch)
            pre = ch
        j = j + 1

    # 转换成字符串
    nl = ""
    k = 0
    while k < len(pos):
        sub = ""
        if k == len(pos)-1:
            sub = line[ pos[k]: ]
        else:
            sub = line[ pos[k]: pos[k+1] ]
        nl = nl + str(len(sub))+str(val[k])
        k = k + 1
    line = nl
    i = i + 1

return line

```

力扣 学习 题库 讨论 竞赛 求职 商店

新功能 下载 App 会员 中

Python3 智能模式 模拟面试

题目描述 评论 (1.5k) 题解 (1.7k) 提交记录

执行结果: 通过 显示详情

执行用时: 1092 ms, 在所有 Python3 提交中击败了 5.03% 的用户

内存消耗: 15.4 MB, 在所有 Python3 提交中击败了 7.74% 的用户

炫耀一下:

与题解, 分享我的解题思路

提交时间 提交结果 运行时间 内存消耗 语言

```

1 class Solution:
2     def convert(self, s: str, numRows: int) -> str:
3         if numRows <= 1:
4             return s
5         padding = ' '
6         nr = numRows - 1
7         rd = numRows * 2 - 2
8         ns = ""
9         i = 0
10        L = len(s)
11        col = 0
12        while i < L:
13            ch = s[i]
14            nl = i%nr
15            if nl < numRows:
16                ns = ns + ch
17                col = 0
18            else:
19                col = col + 1
20                col = col % nr

```

class Solution:

```

def convert(self, s: str, numRows: int) -> str:
    if numRows <= 1:
        return s
    padding = ' '
    nr = numRows - 1
    rd = numRows * 2 - 2
    ns = ""
    i = 0
    L = len(s)
    col = 0
    while i < L:
        ch = s[i]
        n1 = i%rd
        if n1 < numRows:
            ns = ns + ch
            col = 0
        else:
            col = col + 1
            col = col % nr
            af = col
            bf = nr - af
            j = 0
            while j<bf:
                ns = ns + padding
                j = j + 1
            ns = ns+ch
            j = 0
            while j<af:
                ns = ns + padding
                j = j + 1
        i = i + 1
    print(ns)
    line = ""
    i = 0
    while i<numRows:
        step = numRows
        j = i
        while j<len(ns):
            if ns[j] != ' ':
                line = line + ns[j]
            j = j + step
        i = i + 1
    return line

```


题目描述

评论 (2.3k)

题解 (2.1k)

提交记录

Python3

智能模式

模拟面试

执行结果: 通过 显示详情 >

执行用时: 44 ms, 在所有 Python3 提交中击败了 67.68% 的用户

内存消耗: 14.8 MB, 在所有 Python3 提交中击败了 76.92% 的用户

炫耀一下:



写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	44 ms	14.8 MB	Python3
几秒前	执行出错	N/A	N/A	Python3
1 分钟前	解答错误	N/A	N/A	Python3

```
1 import math
2 class Solution:
3     def myAtoi(self, s: str) -> int:
4         s = s.strip()
5         L = len(s)
6         if L < 1:
7             return 0
8         first = s[0]
9         if self.isdigit(first) or first == '-' or first == '+':
10            sub = "" + first
11            for ch in s[1:]:
12                if self.isdigit(ch):
13                    sub = sub+ch
14                else:
15                    break
16            print('---', len(sub))
17            if sub == '-' or sub=='+':
18                return 0
19            v = int(sub)
20            mx = int(math.pow(2, 31) - 1)
21            mn = int(-math.pow(2, 31))
22            if v<mn:
23                return mn
24            elif v>mx:
25                return mx
26            else:
```

```
import math
class Solution:
    def myAtoi(self, s: str) -> int:
        s = s.strip()
        L = len(s)
        if L < 1:
            return 0
        first = s[0]
        if self.isdigit(first) or first == '-' or first == '+':
            sub = "" + first
            for ch in s[1:]:
                if self.isdigit(ch):
                    sub = sub+ch
                else:
                    break
            print('---', len(sub))
            if sub == '-' or sub=='+':
                return 0
            v = int(sub)
            mx = int(math.pow(2, 31) - 1)
            mn = int(-math.pow(2, 31))
            if v<mn:
                return mn
            elif v>mx:
                return mx
            else:
                return v
```

```

else:
    return 0

def isdigit(self, ch):
    if ch>='0' and ch<='9':
        return True
    else:
        return False

```

力扣 学习 题库 讨论 竞赛 求职 商店

新功能 下载 App 会员

题目描述 评论 (1.3k) 题解 (2.1k) 提交记录

Python3 智能模式 模拟面试

执行结果: 通过 显示详情

执行用时: 7580 ms, 在所有 Python3 提交中击败了 5.86% 的用户

内存消耗: 25.3 MB, 在所有 Python3 提交中击败了 5.03% 的用户

炫耀一下:

写题解, 分享我的解题思路

提交时间	提交结果	运行时间	内存消耗	语言
几秒前	通过	7580 ms	25.3 MB	Python3
几秒前	通过	7664 ms	25.3 MB	Python3
4 分钟前	解答错误	N/A	N/A	Python3

```

1 class Solution:
2     def maxArea(self, height: List[int]) -> int:
3         L = len(height)
4         i = 0
5         j = 0
6         mx = 0
7         H = max(height)
8         h = H
9         while h>0:
10             i, j = self.findIndex(height, h, L)
11             print(i, j)
12             if i>=0 and j>i:
13                 a = self.area(height, i, j)
14                 if mx < a:
15                     mx = a
16             h = h-1
17
18         return mx
19
20     def findIndex(self, height, h, L):
21         i = 0
22         j = L - 1
23         idx01 = -1
24         idx02 = -1
25         while i<L:
26             if height[i] >= h:

```

```

class Solution:
    def maxArea(self, height: List[int]) -> int:
        L = len(height)
        i = 0
        j = 0
        mx = 0
        H = max(height)
        h = H
        while h>0:
            i, j = self.findIndex(height, h, L)
            print(i, j)
            if i>=0 and j>i:
                a = self.area(height, i, j)
                if mx < a:
                    mx = a
            h = h-1

        return mx

```

```

def findIndex(self, height, h, L):
    i = 0
    j = L - 1
    idx01 = -1
    idx02 = -1
    while i<L:
        if height[i] >= h:
            idx01 = i
            break
        i = i+1
    print(idx01)
    while j>i:
        if height[j] >= h:
            idx02 = j
            break
        j = j-1
    if idx01 >=0 and idx02>idx01:
        return idx01, idx02
    return -1, -1

def area(self, height, i, j):
    h1 = height[i]
    h2 = height[j]
    h = 0
    if h1<h2:
        h = h1
    else:
        h = h2
    b = j - i
    return b*h

```