

# Follow Up Question

九章算法强化班 第7章



扫描二维码关注微信/微博  
获取最新面试题及权威解答

微信: [ninechapter](#)

微博: <http://www.weibo.com/ninechapter>

知乎: <http://zhuanlan.zhihu.com/jiuzhang>

官网: <http://www.jiuzhang.com>

1. Find peak Element 2 follow up
2. Subarray sum 3 follow up
3. Continuous Subarray Sum 2 follow up
4. Wiggle Sort 2 follow up
5. K Largest 6 follow up

# Find Peak Element

<http://www.lintcode.com/en/problem/find-peak-element/>

<http://www.jiuzhang.com/solutions/find-peak-element/>

# Find Peak Element II

<http://www.lintcode.com/en/problem/find-peak-element-ii/>  
[www.jiuzhang.com/solutions/find-peak-element-ii](http://www.jiuzhang.com/solutions/find-peak-element-ii)

# 用一道题足以 区分5类面试者

# 一道题可以区分5类面试者

Find Peak Element 只会 $O(n)$

Find Peak Element 会 $O(\log(n))$

Find Peak Element II 只会 $O(n^2)$

Find Peak Element II 会 $O(n\log(n))$

Find Peak Element II 会证明是 $O(n)$

只会写for循环

会优化

会优化不会举一反三

会优化会举一反三

会举一反三四

帮助大家从1, 2, 3 晋升到4, 5 档。

# Subarray sum

# Subarray sum

<http://www.lintcode.com/en/problem/subarray-sum/>

<http://www.jiuzhang.com/solutions/subarray-sum/>



# Submatrix Sum

<http://www.lintcode.com/en/problem/submatrix-sum/>  
<http://www.jiuzhang.com/solutions/submatrix-sum/>

# Subarray Sum II

<http://www.lintcode.com/en/problem/subarray-sum-ii/>  
<http://www.jiuzhang.com/solutions/subarray-sum-ii/>

# 循环连续子序列

# Continuous Subarray Sum

[www.lintcode.com/en/problem/continuous-subarray-sum/](http://www.lintcode.com/en/problem/continuous-subarray-sum/)

[www.lintcode.com/en/problem/maximum-subarray/](http://www.lintcode.com/en/problem/maximum-subarray/)

<http://www.jiuzhang.com/solutions/continuous-subarray-sum/>

{ -2, 11, -4, 13, -5, -2 }

# Continuous Subarray Sum II

<http://www.lintcode.com/en/problem/continuous-subarray-sum-ii/>

<http://www.jiuzhang.com/solutions/continuous-subarray-sum-ii/>

{ -2, 11, -4, 13, -5, -2 }

# 旋转排序

# Wiggle Sort

<http://www.lintcode.com/problem/wiggle-sort/>

<http://www.jiuzhang.com/solutions/wiggle-sort/>

# Wiggle Sort II

<http://www.lintcode.com/problem/wiggle-sort-ii/>

<http://www.jiuzhang.com/solutions/wiggle-sort-ii/>



# 反向思路的BFS

# Build Post Office

<http://www.lintcode.com/en/problem/build-post-office/>

# Build Post Office II

<http://www.lintcode.com/en/problem/build-post-office-ii/>

# Bomb Enemy

<http://www.lintcode.com/en/problem/bomb-enemy/>

# 第K大问题

# Kth Largest Element

<http://www.lintcode.com/en/problem/kth-largest-element/>

<http://www.jiuzhang.com/solutions/kth-largest-element/>

# Kth Smallest Number In Matrix

<http://www.lintcode.com/en/problem/kth-smallest-number-in-sorted-matrix/>

<http://www.jiuzhang.com/solutions/kth-smallest-number-in-sorted-matrix/>

# Kth Smallest Number In Two Array

Kth Smallest Number Sum In Two Array  
Kth Smallest Number Product In Two Array



# 需要做修改这道题 第K大其他问题

n个数组的第k大 done

n个数组多台机器第k大(k比较小) done

n个数组多台机器第k大(k比较大) done

(4-6月) 码农类 硕士 全职@Google - 内推 - 技术电面 | Fail fresh grad应届毕业生  
3面试的，今天接到电话说已经挂了。

gle的面试给我留下了很好的印象，因为做的比较规范，最后反馈也都是电话告知结果，非常正规的感觉。

i时面试的时候运气不错，是一个美国小哥，题目也蛮简单的。第一题是给两个string,其中一个string比另外一个多了个字母，返回这个字母。

题就用了最简单的比较，比较每一个字母（按照顺序就可以了），注意一下边界条件，最后一个的边界条件。

rw up是如果字母顺序打乱了怎么找。我先说了HashMap，要写的时候想起来另外一个方法，就是把所有的ASICII加起来，减一下，得到的就是多出来的那个。我就直接说了这个，然后写了这  
写完之后小哥问了一下时间空间复杂度，然后说你已经把本来要提升的东西做了、

给了个第三题，是说如果说要实现一个数据结构，要有insert（），delete（），medium（），mode（）方法，怎么写。

感觉是我最后被拒的原因，因为我感觉我没正确理解他的意思。因为这里你用LinkedList（）也好，ArrayList（）也好，总会有一个时间复杂度会很高，后面结束的时候想想可能是想让我比较  
构的优缺点。

i时是写了一个Arraylist的，他问如果找中位数怎么办，是不是会time complexity很高，然后我说每个结构都会有缺点，是个trade off。

i时间也不够，因为是额外加的，草草结束了这题。

答完感觉还可以，不过最后还是挂了。

e on了，不过HR还是很nice，今天电话告知我挂了之后问我要不要点学习资料，还发了学习资料，然后说8到12个月以后再见。整体对google的招聘印象很好

## Follow Up 常见方式

- 一维转二维
  - 可以套相同的思路试一试
    - Find Peak Element I/II
    - Trapping Water I/II
    - Subarray Sum/Submatrix Sum
- 数组变成循环数组
  - 循环数组小技巧
    - Continuous Subarray Sum
- 题目条件加强
  - 可能题目的解题方法会变化
    - Wiggle Sort I/II
- 换马甲(变一个描述, 本质不变)
  - 本质不变
    - Number of airplane on the Sky/ Meeting Room
    - BackPack Problem
    - Two Sum/ Three Sum/ Four Sum
- 描述完全不一样, 但是方法相同
  - 这种题目得去分析
    - 前向型指针的题目
    - Quick Sort/ Bolt Nuts Problem





Thank You!!!

