

透析热门互联网公司中的 FollowUp面试题

课程不允许录像, 否则将追究法律责任, 赔偿损失

九章强化班 第一讲



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://zhuoanlan.zhihu.com/jiuzhang>

官网: <http://www.jiuzhang.com>

版权声明

九章的所有课程均受法律保护，不允许录像与传播录像
一经发现，将被追究法律责任和赔偿经济损失



- 主讲:陈近南
- 本科毕业于国内TOP2学校, 硕士毕业于北美西部某S大CS专业
- 参加过国家信息学竞赛, 大学生程序设计竞赛
- 拿过国内和北美顶尖IT企业offer数11+
- 曾就职过3个顶尖互联网企业, 面试过110+面试者。



- 助教:ben老师
- 毕业于美国排名前四的计算机院校
- 算法竞赛全国一等奖
- ACM ICPC 大学生程序设计竞赛金牌
- 拿到7个北美IT企业offer。

- 直播课堂: www.gotowebinar.com
- 在线评测: www.lintcode.com
- 教学资料: www.jiuzhang.com/accounts/profile

- 课后答疑: 高级算法学员专属QQ群
- 私有天梯: <http://www.lintcode.com/zh-cn/ladder/4/>
- 九章问答:
 - 新学员必看 <http://www.jiuzhang.com/qa/3/>
 - 九章算法学员手册 <http://www.jiuzhang.com/qa/990/>

- 直播课堂: www.gotowebinar.com
- 在线评测: www.lintcode.com
- 教学资料: www.jiuzhang.com/accounts/profile

- 课后答疑: 高级算法学员专属QQ群
- 私有天梯: <http://www.lintcode.com/zh-cn/ladder/4/>
- 九章问答:
 - 新学员必看 <http://www.jiuzhang.com/qa/3/>
 - 九章算法学员手册 <http://www.jiuzhang.com/qa/990/>

- 聊聊面试当中的Follow Up问题
- 做题的常见误区
- 九章强化算法班正确打开方式
- 聊聊如何在更好的准备面试
- 后续课程安排
- 每日一鸡

聊聊面试当中的FollowUp问题

Two sum

<http://www.lintcode.com/en/problem/2-sum/>

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

1. Hash + 查询
2. 排序 + 两个指针扫描

Interviewer: Two sum II

<http://www.lintcode.com/zh-cn/problem/two-sum-ii/>

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

[Docs](#)

[3,4,5,6,7,8] target=9

| 全部集 | sum | 3 | 4 | 5 | 7 | 8 |
|-----|-----|---|---|---|----|----|
| | 3 | | 7 | 8 | 10 | 11 |
| | 4 | | | 9 | 11 | 12 |
| | 5 | | | | 12 | 13 |
| | 7 | | | | | 15 |
| | 8 | | | | | |

横坐标是i,纵坐标是j

| 第一步 | sum | 3 | 4 | 5 | 7 | 8 |
|-----|-----|---|---|---|---|----|
| | 3 | | | | | 11 |
| | 4 | | | | | 12 |
| | 5 | | | | | 13 |
| | 7 | | | | | 15 |
| | 8 | | | | | |

| 第二步 | sum | 3 | 4 | 5 | 7 | 8 |
|-----|-----|---|---|---|----|----|
| | 3 | | | | 10 | 11 |
| | 4 | | | | 11 | 12 |
| | 5 | | | | 12 | 13 |
| | 7 | | | | | 15 |
| | 8 | | | | | |

| 第三步 | sum | 3 | 4 | 5 | 7 | 8 |
|-----|-----|---|---|---|----|----|
| | 3 | | | 8 | 10 | 11 |
| | 4 | | | | 11 | 12 |
| | 5 | | | | 12 | 13 |
| | 7 | | | | | 15 |
| | 8 | | | | | |

| 第四步 | sum | 3 | 4 | 5 | 7 | 8 |
|-----|-----|---|---|---|----|----|
| | 3 | | | 8 | 10 | 11 |
| | 4 | | | 9 | 11 | 12 |
| | 5 | | | | 12 | 13 |
| | 7 | | | | | 15 |
| | 8 | | | | | |

Two sum II

1. Hash + 查询 -> 线段树(平衡二叉树) + 查询
2. 排序 + 两个指针扫描 -> 排序 + 两个指针

就这样就完了么？
Follow Up 会这么裸？



Triangle Count

<http://www.lintcode.com/en/problem/triangle-count/>

<http://www.jiuzhang.com/solutions/triangle-count/>

(4,3,6,7,8,9)

Triangle Count

面试的时候如果不给简单题，
直接给Triangle Count

您能联想到曾经做过的2 sum么？

Key:

- 排序
- 内层循环优化

题虽然增加，但是思路不会增多

LC题在增长是指数函数，但是你要准备东西的增长是Log函数

- Two Sum 模板
- 这一类通过对撞型指针优化算法, 根本上其实要证明就是不用扫描多余状态

```
1  给一个数组A
2  int left = 0, right = nums.length - 1;
3  while(left < right) {
4      if(A[left] 和 A[right] 满足某一个条件) {
5          // 做一些事情
6          right --; // 不用考虑[left+1, right-1] 和 right 组成的pair
7      } else if(A[left] 和 A[right] 不满足某一个条件) {
8          left ++; // 不用考虑[left+1, right-1] 和 left 组成的pair
9      }
10 }
```

- 2 Sum 类 (通过判断条件优化算法)
- 3 Sum Closest
- 4 Sum
- 3 Sum
- Two sum II
- Triangle Count
- Trapping Rain Water
- Container With Most Water



做题的常见误区

做了的题就过了。

不懂的题，看九章或者博客解答抄一遍然后就认为自己会了。

面试遇到做过的题又不会了

总觉得新题越来越多。

怎么解决不会做FollowUp问题

定期整理自己做过的题目

对自己提三个问题:

属于哪一类?

同类的题目有什么相似之处?

他们思考的思路是怎么样?

上这个课后如何提高 FollowUp能力？

关于强化班的正确打开方式

课前预习

上课之前浏览一遍当前课需要讲的内容。

最好是自己思考一下每道题的解法，如果时间不够，可以浏览一下每个题目的题意。这个非常有助于上课理解。

上课做笔记

笔记本+PPT

- 思维思考方式
- key关键点
- 一系列题目相应的总结

不懂提问题

gotowebinar问答里面赶紧提出来
主讲老师和助教老师会回答。

课后做训练

3-5min冥想复习

Lintcode Ladder

基础差的同学每周15~30小时

基础好的同学每周10~15小时

作业要点

- 先自己想, 再看笔记, 最后看code
 - 温故知新
- Lintcode做笔记记录思路

- 为什么要面FLAG或者Startup?
- 硅谷各档公司new grad工资收入

| | 公司 | 工资 | 股票 |
|-----------------|--|---------|----------|
| Pre IPO Startup | Uber, Snapchat, Airbnb, Pinterest | 10w-12w | 40w-100w |
| Big Name | Facebook, LinkedIn, Google, Twitter, Apple | 10w-12w | 15w-50w |
| Old IT | Oracle, EMC, Yahoo, SAP, Adobe, Cisco | 8w-11w | 0w-20w |

聊聊如何在更好的准备面试

总结题型

求第k小元素FollowUp

矩阵或者多个数组

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/>

是否需要遍历全部的元素呢？

见到集合求Min/Max
就要想到堆

马甲变换一

Kth Largest in N Arrays

<http://www.lintcode.com/en/problem/kth-largest-in-n-arrays/>
<http://www.jiuzhang.com/solutions/kth-largest-in-n-arrays/>



拿到数组先排序

Kth Largest in N Arrays

- 用什么数据结构？
 - Answer:堆
- 方法：
 - 把N个数组先排序
 - 排序过后把每个数组最大的数字放入堆里面
 - 然后堆里面只维护前k个元素
 - 堆pop k次得到答案。
 - 时间复杂度 $N * \text{Len} * \text{Log}(\text{len}) + K * \text{log}N$ (len 是平均每个数组的长度)

马甲变换二

Kth Smallest Sum In Two Sorted Arrays

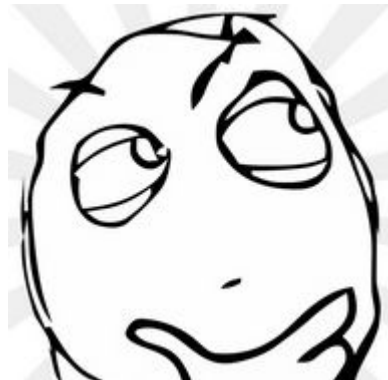
<http://www.lintcode.com/en/problem/kth-smallest-sum-in-two-sorted-arrays/>

<http://www.jiuzhang.com/solutions/kth-smallest-sum-in-two-sorted-arrays/>

要是给两个无序数组呢？

马甲变换三

Kth smallest **product** in two positive element arrays



小结三道题

- 三道题相似点：
 - 求矩阵/数组的第k大
- 可以总结的规律
- 规律1
 - 见到需要维护一个集合的最小值/最大值的时候要想到用堆
- 规律2
 - 第k小的元素, Heap用来维护当前**候选集合**。
- 规律3
 - 见到数组要想到先排序

总结题型

做过了相似的题型要总结出来
看看他们的规律
用了什么相似的算法
这样才能够把题目越做越少

什么样的人适合上这个课程？

- 上过九章算法班，算法上还想要一些深入
- 打算去面FLAG的时候更加稳拿或者Pre-IPO的startup 充满向往
- 希望强化班后很少有难题能够考倒你

- 1.透析热门IT公司中的FollowUp面试题
- 2.数据结构(上)
- 3.数据结构(下)
- 4.两个指针
- 5.动态规划(上)
- 6.动态规划(下)
- 7.如何解决困难的follow up 问题

2. 数据结构(上)

- 并查集
 - 并查集的基本原理
 - 并查集的相关运用
 - 并查集的拓展(带路径压缩)
 - 并查集的运用
- Trie 树
 - 具体结构
 - Trie 树的相关运用
- 扫描线
 - 扫描线的常规题目
 - 扫描线和其他数据结构结合的拓展

3. 数据结构(下)

- Heap的深入理解和运用
 - Trapping rain water
 - Building Outline
 - Heap重要拓展:
 - 带删除的堆hash-heap
 - Median 问题的拓展
- Deque
 - Sliding Windows问题总结

4. 两个指针 Two Pointers

- 对撞型指针
 - Two sum 类和灌水类
 - Partition 类
- 前向型指针
 - 窗口类
 - 快慢类
- 两个数组上的指针

5. 动态规划(上)

- 动态规划的空间优化
 - 滚动数组
- 动态规划时间优化
 - 划分类型
 - Local 和 Global
- 记忆化搜索
 - 普通记忆化搜索

6. 动态规划(下)

- 记忆化搜索
 - 区间动态规划
 - 博弈类动态规划
- 背包类动态规划
 - BackPack I/II
 - K sum
 - Minimum Adjustment Cost

7. 如何解决困难的follow up 问题

- Peak Element
 - Peak Element I
 - Peak Element II
- Iterator
 - Flatten Nested List Iterator
 - Zigzag Iterator
 - Binary Search Tree Iterator
- Subarray sum
 - Subarray sum
 - Submatrix sum
 - Subarray Sum Closest
 - Subarray sum II
- Wiggle Sort
 - Wiggle Sort I
 - Wiggle Sort II

- 题目难度
 - Medium 50% + Hard 50%
- 目标公司
 - FLAG + USPD
- 学习新的解题思路和较难的算法
 - Trie, 并查集, Hashheap, 动态规划优化
- 题目思路总结, 举一反三
 - 解决follow up

- 一共7次课
- 美西时间(PDT)每周六、日16:00-18:00。
- 美东时间(EST)每周六、日19:00-21:00。
- 北京时间(CST)每周日、一早上07:00-09:00。

如何付费

九章官网登陆 → 我的课程
付费之后即可开启 LintCode 阶梯训练权限

优惠码的获得？

关注微信“九章算法”
点击右下角“课程优惠”按照提示操作



课程提供视频么？

九章坚持的是在线直播这种课程教学模式。
学员的参与度更高，听课效果更好。每节课上座率均超过95%。
因为个人原因错过课程的，无法受理补课。

All Companies

| Company Name | Total LCAs |
|---|------------|
| infosys limited | 106951 |
| tata consultancy services limited | 50875 |

| Company Name | Total LCA |
|--------------|-----------|
| Apple | 6258 |
| Google | 14614 |
| Amazon | 6007 |
| Yahoo | 5529 |



QA



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://zhuanglan.zhihu.com/jiuzhang>

官网: <http://www.jiuzhang.com>

