

---

# Google 近期面经 2022



本资料由九章算法独家倾情整理

获取更多干货，欢迎关注**九章算法**微信公众号



九章算法，专业的 IT 求职面试培训，团队成员均为硅谷和国内顶尖 IT 企业工程师。

目前开设课程有：[《九章算法班》](#)[《系统架构设计》](#)[《九章算法基础班 Java+Python》](#)[《面试软技能指导》](#)[《九章算法高频题冲刺班》](#)等。



大厂真题任意刷，欢迎关注  
**LintCode 炼码**微信公众号

全球名企面试真题汇总，剖析当前科技行业热点动向，算法与人工智能专业训练，LintCode 领扣助你斩获理想 Offer!

---

## 电面

### 新鲜出炉狗家店面

刚做完狗家店面，给大家参考一下

不知道是利口第几题但是感觉有点似曾相识的感觉。题目是给一些 String 比如说{"do", "doing", "ing"}

然后有一个 API 叫 observe (char c); 每次 call 完如果刚好能和之前连着 call 过的 char 连起来是 String list 里的某个 String 就要返回符合的所有 String

比如说

```
observe ('a'); return nothing
observe ('b'); return nothing
observe ('c'); return nothing
observe ('d'); return nothing
observe ('o'); return {"do"}
observe ('i'); return nothing
observe ('n'); return nothing
observe ('g'); return {"doing", "ing"}
observe ('n'); return nothing
observe ('g'); return nothing
```

我的解法基本上就是把 input String list 变成了一个 Set，然后把每次 call 的字母都记下来，每 call 一次就和之前的所有字母都 check 一边看看有没有组成 set 里有的 String, runtime 应该是  $O(n^2)$ ，如果不算每次从 char 转成 String 所需要的时间，算的话应该就是  $O(n^3)$ 。

和面试官聊的还不错聊到了用 Trie 会更加高效，不过他没有问我 runtime lol，希望是过了。

### 谷歌 DS 面经

面试官国人很 nice 很有耐心的一点点 clarify

讲了一下现在的工作的 project

只有一个 data point 怎么样判断是不是从某个 distribution 来的

$y = ax_1 + bx_2$ ,  $y = a(x_1 + x_2) + b(x_1 - x_2)$  问这两个 regression 是不是一样，然后 ridge lasso

怎么样研究上不上大学对于收入影响

给搜索颜色的用户 survey 红黄蓝，问有什么问题

code，就是一些 numpy 的 manipulation

总体不是很难就是一些统计/计量的基础。面的时候一上来卡壳了第一题分不清楚 one tail 还是 two tail test 了，但是面试官还是挺耐心的，我之后还是搞明白了。我也不是 phd 所以

---

也没有啥很难的题，面试官也没有因为上来的 hiccup 挂我，希望之后也能好运哈

## 狗家 DS 店面

1. 设计一个调查红绿蓝的 survey, 如果只给有过 color related query 的用户发问卷, 有什么问题?

答: Sample error(convenience sampling or voluntary sampling?); 应该考虑其他的国家的 language; 应该考虑红绿色盲的人, 他们只会随便选一个答案, 甚至 not able to tell the answer; 可以结合其他的 Google product, gmail 还有其他的产品作为 sampling pool

2. coding:  $m \times n$  matrix, each column is from normal distribution

3. 问了 sample mean standard error, 假设是 normal distribution 和不是 normal distribution 的情况下的区别; 以及 sample median standard error. (bootstrapping)

4. coaching 的 top 100 and bottom 100

5. 1000 个数据, 900 个 feature 的问题

6. Regularization 解释一下。

7. 解释了一下 PCA, 如果 PCA 用全部的 principal component 的话会什么问题。

## VO

### Google VO 过经

google: vo

第一题:

一个二维数组, 0 代表水, 1 代表陆地, 陆地可以任意整数高度, 但是陆地不能超过邻居大于一, 邻居就是上下左右, 问最高的陆地可能有多高。

邻居是水的话, 邻居看作 0

第二题:

设计一个数据结构, 一个 queue, 需要实现 max 返回当前 queue 的最大值。

第三题

给定一个树, 每个 node 有 length field 叶子结点有 data field, length field 此节点的所有叶

---

子的 data 长度之和  
root(5) => left((3,"aaa"))  
=> right((2, "bb"))

this represents "aaabb"

1. find the k the character in this node
2. remove [l:r) from the tree.

#### 第四题

find valid set inside a collection of cards, the card has 4 attrs. each attr can be 0,1 or 2. given three card, if they have the same attr or completely different attr on each attr, they are called a set. only return one set if there're multiple

遇到一个很好的国人大哥，应该是抬了我一手的。第二题卡了好几个地方，都给我 hint 了，最后强行写完了=。=

## Google VO 挂

11 月 23 VO，这两天收到 hr 答复，挂。

题：

1. ABC 小哥，Topological sort，类似模版题。用 BFS 算法解决了，为了个简单的 follow up 也答上来了。感觉算 lc medium level。感觉做的还行，但是面试官似乎不大熟悉 BFS 的 topological 解法，解释了一下，我不知道他是不是只要 dfs 解法。我觉得我算做出来了，不差，然并卵。
2. 国人大哥，地里常见的飞机旅行问题，就是说疫情下有些机场被封了，给你一堆 graph edge，要你找到从 start 到 dest 的最短路径。第一问，路径 cost 都是一，BFS 解决。第二问，被封的机场解锁，问如何找到这样一个路径，就是要求通过被封的机场数目最小。方法是构建 edges，当 edge 进入被封机场时候，定义 cost 为 1。然后 Dijkstra。这轮两个题加一起，加上时间限制之类的，我觉得算是 hard 了，但是算 fair 的 hard。觉得算是做的最好的一轮了。哎。
3. ABC 大哥 感觉很 senior，给了道不难的题，类似 serialize array，真的不难就不多说了，follow up 问了不少 edge case 和 优化问题，要实现出来的。感觉这轮算是 medium，但是总觉得里面有很多隐藏测试，面试官一直都很 nice，感觉也算是正反馈。
4. 国人小哥，问题是其他面经里见过的 string replacement，就是说有一堆 key value pair，给你一个长 string，你给我把 key 换成 value。用 1 pass sliding window 做了。follow up 是如果有迭代的 key value 咋办，比如 value 中包含了 key，答 recursive call。这部分有一些 back and force 的探讨，估计算是收到了 hint，但我没要。。最后 follow up，要是 value 包含 key 的过程中，出现了死循环咋办，如何 handle。这块做的一般，解释了半天 idea，最后算是找到了可以的解法，但没时间实现了。这轮算真正的 hard，感觉我做的达标 6-70% 吧，毕竟有 discussion。达到 100% 应该需要全程自己 drive discussion 并且实现完吧。
5. BQ 轮，很常规，我也尽量答的比较 humble 和 team work。这真的不知道是不是这轮出问题了。

总的来说，这是菜 b 我最接近 Google 的一次了，可惜还是跪了。感觉挂的很无厘头，真的不知道是怎么回事，hr 的说法是 HC 告诉 not moving forward。然后我说我觉得我面的还行，

---

为什么反馈上差这么多，可否透露下。不可透露。好像事已至此就只能 move on 了吗？求大家给些 idea。。。

## 狗家 MLE 昂赛

一共 5 轮，3 轮 coding，1 轮 ML，1 轮 BQ，其中 ML 轮分配给了一个背景不相关的面试官，然后临时换人了。

1. BQ
2. 国人老哥，给定一个 0 1 矩阵，1 是障碍，起点 S 终点 E，Find the number of all the nodes that are on any shortest path between S and E。难点是把最短路径存下来。我只能想到直接暴力存所有路径，但复杂度确实高，面试官不让这么写，也不给提示，最后没做出来，应该是挂在这了。没搜到原题，欢迎大家讨论怎么做比较好
3. 两个简单 coding: 1. 给定整数 N，求 Sum of digits; follow up 是如果可以利用 cache，怎样让速度快一些。可以用 int array 存一些计算好的结果; 2. LC 50，问了一些可能要测试的方面，比如 overflow, precision; 后续问在 java 里怎么发现判断 overflow
4. ML 轮，Implement kmeans; follow up: how to use kmeans to detect plagiarism，如果有很多整句话的抄袭怎么做；可以用 N gram
5. 两道题，1. 先热身，给定一些 string 作为 votes，选出票数最多的作为 winner; 2. 和 LC 1366 有点像，但是逻辑复杂一些。在问 clarifying questions 的时候有些 case 面试官也没考虑到，写完之后没时间确认了

一周多一点后 down level 成了 L3，约等于挂了，和 recruiter 说了手上其他的 offer，我们都知不知道不会去了，就取消 team match 了。

## 狗家 VO 昂赛新鲜全面经

第一轮 BQ 常规题

第二轮 Coding

国人大叔，maxCPU

第三轮 Coding

Give a log of message. Each message has source and message as attribute.

要求公平的分配 message 到不同的 source. 每一个 source 能处存的东西是有上限的,上限来自于本来的 log message 分配到总数.

Ex:

maxLog = 4

LogMessage: ("log1",A) ("log2",B) ("log3",B) ("log4",A) ("log5",A)

logA 上限 = 3

logB 上限 = 2

---

Result = ("log1",A) ("log2",B) ("log3",B) ("log4",A)

楼主用  $O(N)$  走 logmessage, 再用  $O(N)$  分配.

用一个 hashmap 纪录个数, 在分配的时候一个个扣掉个数, 若个数为 0, 则移除当前的 hashmap.

Follow up: 楼主本来用  $O(2*N)$  解决问题, 要求  $O(N)$

第四轮 Coding

三哥, Given 一个 file system, 这边用一个 Map 来表示,  $\langle id, File \rangle$

每一个档案可能有档案, 或者资料夹。档案有 size, 给定一个 id 求问总大小

楼主直接 DFS 解

Follow up: 如何其中一个档案更新大小, 其他大小如何更新? 楼主解释用 logn 的方式, 类似 segment tree 的 update 方法

第 5 轮 Coding

力扣旧陵玖

## 狗家在职跳槽 onsite L3

第一轮 bq, 都是常规的 bq 问题, 没有问简历。一点建议, 如果面试官提出的情景你没有遇到过, 最好直接说我没有遇到过, let's make an assumption, 这样即使你回答的不完美面试官也会理解。

有没有 push back 客户的不合理需求? 怎么解决技术难题? 有没有 project 做到一半发现方向做错了等。

第二轮算法, 很类似力扣叁陆陆, 但是面试官要求要用他提示思路来遍历二叉树, 比如后序遍历和拓扑排序, 拓扑排序是按子节点数量计算入度, 每次取出入度为 0 的点, 再更新父节点入度, 以此完成遍历。

第三轮算法, 很类似 merge interval, 但是 merge 的是时间段而不是时间点。举个例子, merge 时间点是[1 点, 2 点], [3 点, 4 点] merge 之后还是[1 点, 2 点], [3 点, 4 点] merge 时间段是

1 2 3 4

## -> 数字表示[1, 2]

## -> 数字表示[3, 4]

merge 之后是[1,4] #####

当时题目没理解对, 最后仓促修改, 最后这轮没有拿到 hire

第四轮算法, 面试官原创题, 但是解法和壹叁陆捌很像, BFS 和 DFS 的结合, 每一层 BFS 搜索的时候做 DFS 把周围符合要求的元素放进队列, 属于考验基本功, 没有思考难度。followup 是按要求修改一下 dfs 算法, 用他提出的要求判断 dfs 到的元素要不要放进队列。

第五轮算法, 原创题, 按面试官的要求用 stack 和 queue 来放元素, 比如 queue 里面装一些 stack, stack 里面放数字, 或者用 stack 装 queue, queue 里面再装数字, 面试官会提出

---

一些要求，比如每个 stack 里的数字和不能超过多少之类的。

这轮几乎没有任何算法，但是感觉这个面试官注意交流和 corner case。需要主动提出一些 corner case 然后探讨解决方案，比如遇到一个特别大的数，是抛出异常然后 catch 还是直接无视。

followup 的要点是元素放进 stack 里再拿出来会颠倒一下顺序，类似于用两个 stack 实现 queue。之后又探讨了一下用双向队列形成一个类似滑动窗口的设计。

总结：感觉没有遇到那些很难想的 DP 和贪心算法，所有题目的思路都很直接，所以我就没有写具体题目内容而是直接写了思路，从我的看帖体验来说，直接写题目反而会产生歧义。题目都属于考察基本功和熟练度。45 分钟面试去掉 10 分钟谈话，剩下 35 分钟要完成题目和几个 followup

timeline 是 11 月初面试，直接跳过电面开始 onsite，两周后接到口头 offer，最后祝各位也拿到心仪的 offer。