

化学奶爸转码上岸微软分享内容整理

背景：

本科， PhD， 博士后都是化学。

5 年的仪器行业的工作经验， 无码。

2019 年 4 月离职， 在一个学校挂了一个 volunteer 的客座研究员。

2019 年 8 月决定大龄有娃徒手转码。

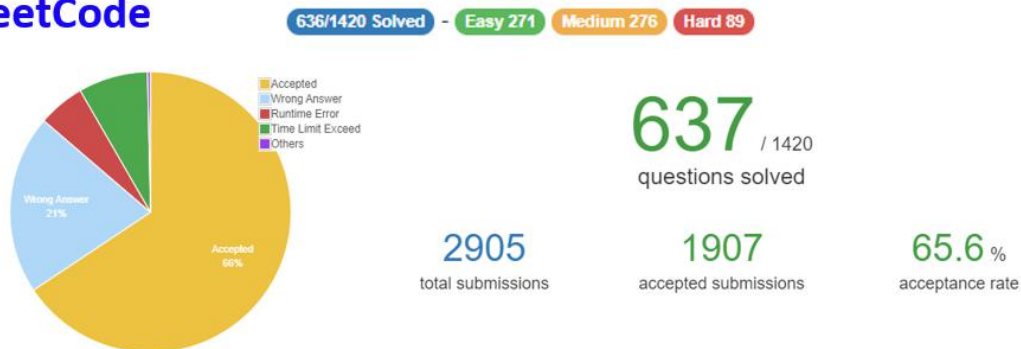
2020 年 1 月面亚麻。

2020 年 3 月面微软。

2020 年 4 月面 google。


最后拿到微软 Level 61 的 offer， 签了。

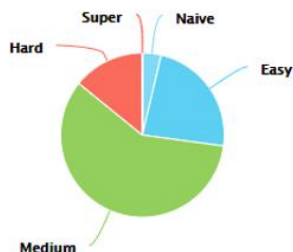
LeetCode



LintCode

The 2,334 submissions in the last year

AC Leaderboard		
Rank	Username	Accepted
333	 MysterionT	516



面试之前刷题 400 左右，基本都是上九章的课刷的 ladder。面试期间，针对三个公司，总共新增了 200 多。

Time line

转码之前：2017 年 9 月 - 2019 年 7 月

2017 年 9 月开始了解 DS 的东西。接下来这一年，大量的时间去学很多数学，MATLAB, R 和 python。R 和 python 从零基础开始。写了一些 data analysis 的包给客户，然后也用 python 做了一些 marketing data 的分析。还现学现卖的给同事做了 python 和 machine learning 的一些基础培训。

2018 年 9 月，随老婆搬到了湾区。

2018 年 11 月，通过内推，拿到了一家半导体公司的 Staff Data Analytics 职位的面试机会。被告知可能会考算法，但是很浅的那种，因为那个职位更注重 customer engagement。面试前一天，临时抱佛脚的看了 “Grokking-Algorithms-An-illustrated-guide-for-programmers-and-other-curious-people” 的卡通读物。算是有生之年第一次接触算法。

这个面试对我的打击挺大的，那时候不知道刷题是啥。有位同胞问了我一道 two sum 和开根号。如果是今天，就是赤裸裸的送分题。我没有答好，不过同胞也没有因为这个卡我。倒霉的碰到另一位变态面试官，让我介绍当时正在做的项目，接着让我重写 project code，我长长在纸上写了好几页。最后他说他要的是 MATLAB，不是 python。把我挂了。第一次体会到非大厂的面试的不确定性。面试完的总结就是，**我还不不如直接准备准备去面软件大厂好了。**

后来，因为出差实在太多，2019 年 4 月底离职，从此失业。

那段时间，朋友介绍去面一家小公司做 DS。直接挂算法上。问题很简单，smooth an image。

还有，FB 的 recruiter 找我面 DS。recruiter 一上来就问我三道题：SQL，统计，和 python。题目没有问题，但是我跟他说我不 ready，等我 ready 以后再找他。

6 月底，开始刷题，只是为了应付 DS 的职位。本来也就只是想刷点 easy 的。结果是爱上刷题。而且开始看 “Element of programming interviews in python” 和 《大话数据结构》，还上了清华大学严蔚敏老师的网课。

7 月中旬，在朋友的介绍下，报名了令狐冲老师的九章算法班。基本上，上这门课期间，有了转码的念头。不过那会还在折腾 data incubator bootcamp 的申请。一直到 8 月中旬，Data incubator 不给我 fellowship，才彻彻底底的决定转码。

决定转码之后：2019 年 8 月 - 2020 年 4 月

2019 年 8 月 - 11 月：

2. ACM 金牌进阶带刷班-提高
3. Java 基础班
4. 背包四讲
5. 系统设计
6. 线段树与树状数组
7. 高频题班
8. 面试软技能指导
9. 递归四讲
10. 面向对象设计
11. 全栈开发基础项目课
12. 动态规划专题班
13. Django 开发入门
14. 海量数据处理

2019 年 12 月：

组队做 citi bikeplus 的项目。做简历。开通 LinkedIn Premium

12 月 18 号：亚麻 recruiter 在 LinkedIn 找上了我，然后准备 OA，做了 OA

2020 年 1 月：

亚麻：系统复习了算法，OOD 和系统设计。1 月 31 号面试。

2020 年 2 月：

第一周和第二周：组队学习 deep learning，还有做做学校的项目。

2 月中旬，google recruiter 找我面 MLE，我电话里面争取了变成去面试 general SDE。

安排了电面，但是我在电面到来之前通过争取把 google 的电面给 skip 了。

2020 年 3 月：

google recruiter 给我直接安排 onsite，我约了 4 月初。但是后来因为疫情，老婆 WFH，小孩不能上学，导致我的准备大大受到影响，于是，我又推迟了两周。

这个月，在微软的师兄给我内推，然后，我把整个微软的电面 + V0 都走完。

还有，在我电面微软的前一天，隔了差不多一年，FB 的 recruiter 又找我。这次不再是 DS，而是让我面 MLE。我想争取换成 general SDE。recruiter 让我都面。她帮我也转给了 general SDE 的 recruiter，结果两次都不给我机会，那会因为疫情，其实 FB 开始只招 experienced 的。

2020 年 4 月：

很烦躁的一个月。最后还是非常幸运的拿到了微软的 offer。

拿到微软的 offer 以后，我取消了 FB 的面试。反正 MLE 我也到不了。当初还有其他大厂的 recruiter 也在联系我，我全部都婉拒了。我就只想着赶紧开始工作。唯一保留的就是 Google。

最后 Google 也没戏，很 close。recruiter 让我一年后直接 onsite，他说我的表现非常好，所以一年后不需要电面。

Timeline 的小结：

因为工作的需要和自己个人兴趣所在，2017 年 9 月到 2019 年 7 月，差不多两年的时间一直在准备 DS 的。而真正彻底决定转码是从 2019 年 8 月中旬才开始。决定转码以后，四个月的时间上了九章 10 几门课，然后 12 月底开始面亚麻，2 月份开始准备 google，3 月份面了微软。真正面试只有亚麻，Google，和微软这三家。算是 5 个月的准备，然后 4 个月面试三家一线公司，最后拿了一个 offer。

心路历程

为什么是从 DS 开始？

我和家里队友一直都爱折腾，年纪都不是问题，关心的只是自己的兴趣是什么。

我是化学的 PhD，博士期间是做合成，搞激光，研究太阳能。博士后期间换了更接近器件方面的。到了工作，最开始是做客服的，然后转到了研发搞仪器开发。每个阶段都不一样。跨度都有点大。

2016 年买了房子，娃出生，但是我们还是继续折腾。队友 2017 年和 2018 年每年都会面一次大厂，我都会帮着复习 DS 的东西。后来工作上自己也要用到，就自然而然的想转 DS。不过，后来发现自己真心不想做调包侠，所以对转 DS 一直心里没有底，甚至有点抗拒。尤其是，工作上，学会写代码以后，经常跟 software team 打交道，后来有些给客户的代码干脆自己写。发现这些比 DS 的调包有意思多了。后来知道了还有 apply scientist 和 machine learning engineer/scientist 这些职位存在，给了自己很多希望。但是很快就绝望，这些岗位需要有一定

的 SDE 的背景。正常一点的路子，应该是 SDE → DS → AS/MLE。这些都不是问题，真正的问题是，一直被洗脑，说码农是小年轻做的事情，所以最开始那两年非常踌躇和郁闷，真的浪费了那两年的黄金时间。

为什么决定转码？

上了九章算法班以后，结实了一些朋友，都是些有趣的灵魂，都是大佬，一路带我飞。

当然，也知道刷题在真正 SDE 的工作中不怎么用到，甚至用不到。后来，真正让我确定我的兴趣的还是因为令狐冲老师的系统设计课，文泰来老师的 Django 课，全栈设计课，以及跟小伙伴组队做的公益项目。

我在前公司的时候，996 太正常不过了的。如果碰到出差，一周 100 个小时也是家常便饭。所以刷题也好上课也罢，我一点都不觉得累。现在又有兴趣，做出来的东西又能马上帮助到别人。虽然我在之前的公司做了五年，到了 staff 的位置，可是薪酬还不如 SDE I 高。那么，就转呗。

为什么不骑驴找马？

一直都不是一个拿着工资然后三心二意的干活的人，还有之前的公司我那个位置要全世界到处飞，太难了。

但是我依然强烈建议转码的各位，尽量骑驴找马。9 个月，一个人窝在家里，刷题上课找工作，还要跟娃斗，不是那么容易的。

9 个月是怎么熬过来的？

首先是家里队友的支持。她一直秉承着她公司的理念在支持我：如果一份工作，是你喜欢的，你也能做好，又能帮到别人，那么就是一份适合自己的好的工作。

其次，通过九章这个平台，我认识很多有趣的灵魂，都是大佬，一路带我飞。

七八月份上算法班的时候，有一位“台湾小哥”在中部，每天晚上都跟我一起刷题到凌晨一两点，加州时间。。。建立了深厚的革命友谊。嗯，因为她网名有点刚，到了后来才发现她是传说中的大眼萧亚轩，号称优雅女神。

这位“台湾小哥”跟我组队上背包课，接着来了东北小伙子“天线宝宝”，这个组队上背包课的群，最后变成了今天的 40 个人左右的花花群。花花群是什么？之前给九章做分享的大佬，基本都被拉进去。我们无耻的把花花班班拉进群，让这位班班带领我们。

后来花花群里面上过动态规划班的说想复习一下，那我们就组了一个 dp 群，10 个人左右，这个群是后来的开车 2 群。2 群一直变，因为只要超过 20 个人，会重新组一个新的开车 2 群，只保留那些还活跃的人。

问题来了，有 2 群，那 1 群呢？

1 群总共 10 个人，不会再新增人。这个群，来源于教主带小仙女刷题，一步一步把小仙女带进了 Google。当初也是 X 哥和小仙女在带小蓝人，最后小蓝人也上岸 Intuit。在这个群里面，后来教主也想拖我进 Google，没有成功，遗憾。

这两个开车群，每天给了我无尽的动力和快乐。

开车群都是些什么人呢：1/3 + 是 PhD，CS，数学，化学，物理，好几个还是有博士后经历的。也有文科的艺术的。有前端，后端，全栈工程师，有数据分析师，工程师，科学家。有像我这种大龄有娃徒手转码的，有本科还没有毕业的，等等。大家都来自九章，都有一颗想要上岸大厂的心。现在，好多进了大厂。好多人都给九章做过分享活动。现在好多人手头有大厂 offer，比如 google，亚麻，oracle，intuit，snap，微软等等。当然，也有一部分正在努力面试中。

9 个月到底是怎样的一个经历

7 月中旬开始上算法班，那一个月真的打鸡血一样，每天基本睡不到 6 个小时。这个月有“台湾小哥”陪伴。

8 月份上了刷题班。应该是这个时候认识圆圆和花花班班的。圆圆带我进了一个亚麻分享活动，从那里认识了小天使。跟小天使一起刷题一起 mock。

9 月份，“台湾小哥”度假回来，在背包群，也就是后来的花花群，里面说大家一起来每天 5 道题。后来慢慢的把一些有兴趣的刷友拉进群。应该是在这个时候，跟几位天竺小哥在 slack 组队刷题，还有讨论系统设计。

10 月份，在教主的分享上认识了教主，把教主拉进了花花群，从此大家有了灵魂和奋斗方向。然后后面的九章分享，我们成天打着教主的旗帜，把其他做分享的大神拉进了花花群。我第一次找教主帮忙是让他指导我 DFS。他给了讲了一个小时的 DFS 入门。后来我写了一个文档，做了总结。那段时间，小仙女和小蓝人一堆面试。教主就一直负责给他们 mock。尤其是小仙女，因为要面 google，教主真得一步一步带着她，带着她刷进了 google。我蹭了所有的带刷。这个月，X 哥来了，一位“别人家的孩子”的优秀小伙子。从此以后开车群有了开车的灵魂。每天就看 X 哥和教主在顺便开个车，然后我们的“台湾小哥”优雅女神会在合适的时候问个算法问题给打住。

11 月份，上了全栈课，在 D 小美女这位数学 master 的建议下，在肠粉的带领下，拉了 Rita 这位艺术生转前端的，还有一位在读 DS master 小师妹，还拉了小天使和我高中同学做导师，我们做了一个 citi bike plus 的全栈 + ML + 实时数据分析的项目。相信这个项目最后都是大家简历上面很靓丽的一笔。

12 月份，事实上应该是从感恩节开始，大家比较有时间，每天都一起刷题讲题。教主回国度假之前，是他带我们，后来我们自己组织。那一个月，我们录了 100 多个视频。当时 Q 博士说，大家在 Zoom 里面都有一套别墅。那段时间 QY 博士给大家贡献了非常多高质量的整个系列的题讲：word 系列，coins 系列，序列化系列等等。

2020 年 1 月份，好几个人一起面亚麻。这个月我们 F 神也来了，我们现在公认的真正的神，横扫各个大厂。那会我们组织了亚麻的 OA 讲解，过去六个月的亚麻 leetcode 原题讲解，真题快速点题。QY 博士绝对是我们的 engine。QY 博士出品，必是优品。然后这段时间我们也完整复习了 OOD

和系统设计。OOD 我们把文泰来老师每节课两百多页的课件浓缩成不超过 10 页。小蓝人还用英语讲了几个 cases。系统设计我请了外援，我两位高中同学，其中一位还是某公司的 CTO。这段时间，QY 博士和 F 神每周都 mock 好几次，我们也领教了什么是神仙打架。

这个月，最要感谢的是小天使，好多个晚上，陪我聊 BQ 到半夜三更。聊出了 30 多个故事，最后还帮我精心准备了 10 个故事。

小蓝人把亚麻军规 14 条给大家朗读录音，我相信很多人每天睡觉前和起床第一件事都先听一听小蓝人那甜甜的声音给你读一遍军规 14 条。

这个月，我们开车 1 群也迎来了芒果女皇，从此这个群，格调大大上升。真有一种，她带着我们一群弱鸡出去闯江湖的感觉。气场太强大了。她有一个分享，讲怎样面对面试，我每次面试之前，电面或者 V0，都要从新看一遍。而且，我给每一个问我面试心得人都推荐一定要去好好看看这个分享。

2020 年 2 月份，J 同学带领我们过了一遍 FB 的好多题。她也帮我们系统过了 leetcode 所有括号题，然后她获封括号女神。FB 过完以后，正好一堆人要开始准备 google，然后我们进入 google 题目阶段。教主回国度假之前，给我们留下了一个 教主题库，70 多道题，道道精品，我们都全部一起刷了一遍。

2020 年 3 月份，群里面活动少了很多，但是各种小群衍生出来。大家已经非常非常熟了。熟悉到微信已经不够了，直接拨打电话。熟悉到，会说“等等，我正在 nn”这种比开车还亲昵的话。也是这个月，教主精心带我刷题。每个 topic 都刷一遍，教我怎么手动跑代码，培训我的 coding style 方方面面。每次都是两三个小时。大家都说，我进 google，第一个月的工资要给教主。那是没问题的。因为群里面开始 bid 排队，说如果教主能带进 google，可以贡献两个月，三个月，半年。。。工资，就差以身相许了，哈。反正有这些培训，面微软的时候，感觉在飞。除了。。。DFS。。。。

2020 年 4 月份，面完微软，教主又集中虐我 DFS。最后 google DFS 那一轮给的是 positive。非常赞。不过，真的被虐到想吐。Google 太难了。怎么个难法？我拿到微软 offer 的是时候，我有一种放弃 google 的冲动。

总之一句话，这 9 个月，我不是一个在奋斗。感谢这些小伙伴们。我在写这些文字的时候，心理面都透着各种欢乐。

说说九章的课

我最能够炫耀的是，我认识很多九章班班。。。我还加了令狐冲老师的微信。

我也了解过其他培训机构的课，九章是唯一一个不会说友商坏话的机构。我真心不了解，为什么九章的友商喜欢攻击九章，他们格调太低了。

下面具体说说我上过的课吧

1. 九章算法班：

这不是基础班，这是中级班。如果算法和数据结构基础的，建议先上基础班。然后再来上这个。否则你就会像我一样，当初每天睡不到 6 个小时，强度特别大。

随堂课是一定必须肯定要看的。令狐冲老师花了大量的时间整理出来的。

ladder require 题目都是上课讲过的，让你锻炼模板和套路的。上课听一听，看一看代码，自己实现一遍。第二天再做一次。那些选座和 **related** 才算作业题。**required** 的你做得很熟，剩下那些作业题大部分也不会太难。就比如，如果一个人连拓扑排序这种套路和模板都不知道，直接去做 **LintCode615** 这个 **course schedule** 题目，你是 **re-invent** 自己的 **wheel**，没有必要。

上课讲得基本都是 **require** 题目。基础差的，真不建议自己徒手做，听明白听懂能重复比较重要以后会用。就像让你自己证明勾股定理然后再去用一样。。。没有必要呀。

每次面试之前，我都会把 **ladder** 的题目至少 **require** 的全部重新做一遍。

2. ACM 金牌逐行带刷班-提高

这个太赞了，尤其其它的 **ladder**，不过已经停办。但是又新的版本。问圆圆去。

3. Java 基础班

帮我快速上手 **Java**。我整个准备过程，一直想要精通 **Java**，最后有心无力。不过这个课但是帮我打了一点点基础。

4. 背包四讲

适合入门的人。更加推荐侯卫东老师动态规划班的 **DP**。但是侯老师的 **DP** 课没有讲多重背包。这个我有整理笔记，可以分享。

5. 系统设计

令狐冲老师主讲的课，绝对推荐。如果你是徒手转码，一定要上。别去听什么面试不考。相信我，上了以后，你才跟那些面试官有共同语言。不然，面试官真会觉得你就只是一个只会刷题的白痴。但是个人建议，先上了 **Django** 或者全栈课再来上这个，概念会更加具体。建议买全套增值服务，一整个月无限回放那种。我反正那会，一个月听了好多遍。最近系统设计新增了一节 **API design**。这个非常重要，大部分人面试都被问到。

6. 线段树与树状数组

令狐冲老师推荐的课程，如果你想学好 **divide and conquer**，这个数据不错。而且真的能够在很多场景帮你从 n^2 优化到 $n \log n$ ，而且碰到跟区间有关的题目，都可以往这上面想。最赞的还是这门课的 **ladder**。

7. 高频题班

必上！必上！这个班是 **offer** 机器，如果你不上，你的九章人生不完整。上课的人大部分是正在 **actively** 找工作的人。你可以找到很多志同道合的小伙伴。

8. 面试软技能指导

沙加老师真的很牛。对于 **new grad** 和转码的，都建议上。帮你揭开 **SDE** 找工的各种迷雾。很多理念特别特别有用。我这种在工业界 5 年的，都受益匪浅！

9. 递归四讲

这个课开课太晚了。我转码第一个坎就是递归。我多么希望这个课早点开。非常适合碰到递归就怕的小伙伴。转码必上的小课。

10. 面向对象设计

这是另一个必上课。对于一位 SDE，不管面试考不考，OO 的东西是必备的基础。让你做个算法题，稍微复杂一点，就能够直接看出你是否有 OO 的思想。如果你上过 OO 的课，有了这些概念，会感觉写出来的代码就该长 OO 那样。详见芒果女皇的九章分享。

我一直觉得九章面向对象和系统设计都是转码的人必须上的。不然码工们在聊天，对你，你只会觉得是对牛弹琴。

11. 全栈开发基础项目课

如果你的简历缺项目，就去上一个。但是要用心上完。这个课，帮你理解了很多项目开发的概念

12. 动态规划专题班

如果你要面大厂，必上的课。侯卫东老师讲得真的太好了。把没有套路的东西讲出套路来。你会发现，动态规划也就那么回事。反正我面 google 前，把这个班的题目又做了一遍。最后还真被问到了，尽管更多是 DFS + memo

13. Django 开发入门

说自己会 python 的，都去上一个。用教主的话说，你说自己会 python，但是一个 framework 都不会，好意思吗？这个也可以是你简历上的靓丽一笔。

14. 海量数据处理

令狐冲老师主讲的课。这个是面试 follow up 最喜欢考的。我几乎每轮面试都会被问到当你的数据量很大的时候该怎么办。答案就在这门课。

十几位小伙伴们上过的课程的统计：

算法, 刷题班-提高, Java 基础班, 背包, 系统设计, 线段树与树状数组, 高频题, 面试软技能, 递归, OOD, 全栈课, DP, Django, 海量数据

算法, Django, 全栈课, OOD, 系统设计, 面试软技能

强化班, 系统设计, OOD

高频题, DP, 背包, 私教

算法, 高频题, 系统设计、OOD

算法集训营, 高频题, OOD, 系统设计, 面试软技能

算法, 高频题, 强化班, OOD, 背包, Django, Spark

算法, 高频题, 系统设计, OOD

高频题, 刷题班-提高, 系统设计, 海量数据

算法, 高频题, 刷题班-基础, 海量数据, 线段树与树状数组, 背包, 递归

算法, 线段树与树状数组, 海量数据, 背包, 递归

算法, 强化班, 递归, 系统设计

算法, 强化班, 系统设计, 背包, 递归

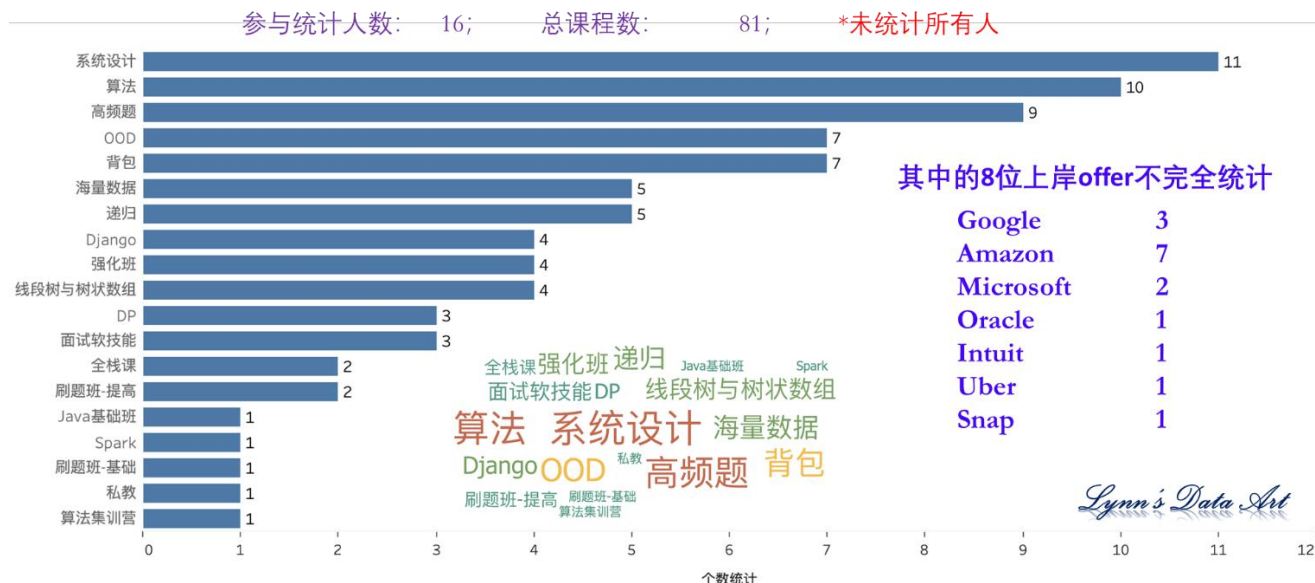
算法, 高频题, 系统设计, 背包, 线段树与树状数组, Django, 海量数据

DP

系统设计

全栈课 强化班 递归 Java基础班 Spark
面试软技能 DP 线段树与树状数组
算法 系统设计 海量数据
Django OOD 私教 高频题 背包
刷题班-提高 刷题班-基础
算法集训营

Lynn's Data Art



说说面试

不谈面试技巧。对于面试技巧，请看芒果女皇的分享。我只说我自己的经历。

亚麻就是玄学。

这是我决定转码以后第一个面试。各种准备，但是我一直就定位在去体验一下。最后也挂了。这种挂我是预料中的，但是最后不小心知道 **feedback**，我跌破眼镜。真的是。。。什么鬼。。。

四轮，每一轮上来都跟我说是 **SDE II** 的职位。

第一轮，经典的考某种数据结构。我挂在这个地方。说我不 **take feedback**。我 **tmd** 的无语。什么鬼。就因为我用了单向链表。虽然我能感觉到他想要双向链表，但是也没明确要我用啊。最后也说可以呀。然后，我每写一步都要我解释得清清楚楚。最后还说不错，**I learned something today**。最后给我个 **negative**，还建议冷冻我一年。我还以为这一轮肯定是最好的。确实是某国小哥，但是不要有成见。后面还有两位天竺小哥哥小姐姐在我表现很烂的情况下，意外的给我 **green**。只能说，碰到天竺小哥，你的运气比较随机。至少我这次面试，不该过的两位，给我过了。我觉得很好的，给我废了。无法评价。对了，**follow up** 也不是我能 **handle** 的。如果 **capacity** 要动态变化怎么办？怎样做 **code review** 等等。

第二轮，先告诉是 **SDE II** 的职位。然后面我个停车场。。。面试前，**recruiter** 三番两次要我准备好系统设计。面试前一天晚上又 **email** 我一次。然后面试前两周，我 **OOD** 一点都没有复习。然后就各种奇葩。最后也画了些图，讨论了很多，但是一举一动都非常不 **profession**。结果。。。居然给我过了。。。不过，后来跟小伙伴们一起总结，发现我的停车场的考法跟别人不一样。会涉及系统设计。比如，**membership**，比如手机推送，比如如果是西雅图级别的，**state** 级别的，**USA** 级别的，全球级别的，怎么处理。比如如何找到最近的停车场。比如如果 **app** 挂了，怎么处

理。比如一辆车进来，怎样最优使用停车场，等等。算法，数据结构，OOD，系统设计都有涉及。一开始 5 分钟，面试官带我做了类图。给我过 SOLID 五个原则。真的是他带我，我还一堆违反原则。我当时心早凉凉了。然后就全部是瞎 BB。我可能是瞎 BB 做得好。好几次我跟面试官说，我实在没有好的主意。要不给点 hint。他都想一会说，他也没有更好的办法，move on。

第三轮，bar raiser。第一次面 SDE，在 bar raiser 的各种高压之下，老掉坑，紧张到 bit 和 byte 乱说，大数据乱答，写代码的时候，function signature 也忘了写，然后。。。也给我过了。。。题目有点像 consecutive number，但是是 data stream，关键是如果数据量很大的时候怎么办？怎样做到时间和空间的优化权衡。后面基本瞎聊海量数据的东西。

第四轮，正常吧。但是有一个坑，问了跟前面的人一模一样的 BQ。我跟她说别人问过了，她说那你就重复回答一遍好了。后来别人跟我说，这是坑。亚麻最忌讳用同样的答案去回答同样的 BQ 题目。我一直以为这三姐是在坑我，最后居然也是给我 green。算法题是字符串处理的二分答案。如果不问清楚，挺难的。但是没有让我实现所有的代码，也不可能有多长时间。

如果我不知道 feedback 的话，我以为我挂在 2-4 那三轮，结果我只是挂第一轮我以为稳了的。我。。。

微软：

运气爆棚的一个面试，天时地利人和，除了网络有问题。

很强的内推

面试之前，各种猛刷数据结构，链表，tree，栈，哈希表等等做到吐。结构就考这些。

我没有多少时间准备，因为内部消息说可能会 hiring freeze，让我赶紧面。我周一电面完，周三跟我约 onsite，我约的是下周一，结果周四深夜，让我第二天周五的时候，赶紧改成周一面。所以只有三天的时间准备。

电面是 tree 两道，难度中等。

VO 的时候，

第一轮是链表，包装成 product development 来问。follow up 是 hard

第二轮，数据流或者字符串处理，好多东西要 clarify。这一轮有涉及一些计算机底层的东西，但是不多。实现本身不难，顶多 medium

第三轮，网路几乎处于挂的状态，而且一道很简单的 DFS 我被卡住了。

第四轮，绝对的 hard，用了 n 多种数据结构。整个过程就做题，但是做题过程中体现各种 OOP，还有数据结构很多底层的东西。

居然等了快两周才确定 heard count 下来。口头 offer 当天我就谈好，一个很 low 的包，但是疫情期，而且我确实有一轮做得不好，只能接了。第二天签了正式 offer。然后就进入 pre-onboard，特别神速。因为。。。疫情其不稳定，赶紧开始才是正道。

签了微软 offer 以后，其他所有面试，除了 google，我都推掉了。

,

Google:

技能大家要 get: 电面是可以免的。如果你能够找到 googler 给你个推荐。一般不超过 5 个。我反正有一帮给力的 google 小伙伴可以帮我写推荐。

第一轮，主要考 clarification，这一轮挂了。没有马上想到最优解。

第二轮，又是把算法包装成产品开发，然后中间不停改需求，我也改代码。最后情景模拟完了，才跟我说最后写的代码有 bug。一个很低级很大的 bug。我当时心凉凉了。

第三轮，BFS， follow up 是 Dijkstra

第四轮，数据结构 + DFS，很难的题目，但是不要求写完整代码，主要是各种 communication 和思路。

第五轮，BQ，就瞎 BB

送了 hiring committee，按照 recruiter 的意思说，挂在 coding efficacy inconsistent。他觉得是应该是疫情影响，不然如果正常时期，顶多是 downgrade 到 L3。他让我一年后联系他，到时候不需要电面，直接 onsite 就行。也算是一种肯定。

不过，我自己虽然因为结果很 close 有点 upset，但是是很满意的。我面试之前以为每一轮都要挂得凉凉的，结果远超出我预期。

准备 google 是什么心态？拿到微软以后，我想直接放弃。真的因为压力和心累，比高考压力还大。

一些心得:

没啥心得，都是拾人牙慧。先说一些我自己经常会回去看的九章分享吧。我也就没有必要重复了。大神们才是真的分享。

最全面和牛逼的总结，没有之一，请看芒果喵的分享：

<https://www.youtube.com/watch?v=exll4E-EGz4&t=487s>

Coding:

请找圆圆要教主的文字分享，还有他跟入职亚麻的 David 哥一起做的分享

教主跟老顽童老师的分享在这里：<https://www.youtube.com/watch?v=lytDznWI304>

如果你想短平快上手，推荐庄周老师的分享 <https://www.youtube.com/watch?v=ssGdqpmWPS0>

设计：

这个很赞：<https://www.youtube.com/watch?v=mSVuU6bdFv4>

这个我没有看，但很多人说很赞：<https://www.youtube.com/watch?v=VjnXZpZ7wvs>

项目：九章没有做过。其他培训机构我没有看到好的。

BQ：芒果喵的够了。

对于别人的经验，我的态度一直都是适合自己的才是有用的，否则对你都是 fun pea。不是不好，如果自己做不到或者就是没那个习惯，那就算了，不要强求。比如，我就很难做到众大神们那种优雅有条理的解题报告和各種 bug 记录。我做不到。年纪大了，实在很难去培养一些新的习惯。但是我有自己的方法。我会存整份代码，在代码上面标出 bug 的地方，一目了然。还有，比起解题报告，我更喜欢做 ppt。

对于刷题，我补充一点适合我自己的。我不喜欢闷头刷题。我也确实经历过那种憋了一天只做出了一道题的日子。后来觉悟了。很简单，不管面试和实际工作中，你都不是一个人，当今社会讲究合作。面试的时候是有人跟你沟通的，是为了找一位未来的同事。沟通更重要。所以，我慢慢的养成了拉人一起 team work 刷题的习惯。刷完题，就是拿出来讲的。讲题，是面试的基本功。你可以试想，你是跟别人 team work 一起一天刷 10 道题好呢，还是自己在那闷着卡在一道题目上面一整天好？当然，这不是让你去完全依赖别人。基本功还是需要靠你自己去打牢。比如手动跑代码，各种套路的熟悉，如何找 bug 等等，这些自己下来以后，还是要不停的自己锻炼。我非常感谢我们 QY 博士，每次我找 bug 超过 15 分钟，我就拉他帮忙，他绝对火眼金睛！

关于简历

我的简历事实上是一位 FB 的 software engineer manager 帮我修改，应该说完全重做。是一位经验很丰富的美国人。他是我的找工作过程的 mentor。在我的求职过程中，他一直给我提供很多非常有帮助的 advice。我在这里也只是把他告诉我的传达给大家。

最开始做简历的时候，有很多人给我很多建议。基本上都是说要隐藏各种细节。Concern 就是 PhD + 博后 + 5 年的工作经验，entry level 不敢招你，experienced 又嫌你技能不够。还有，就是各种文章啊，引用啊，不要提。最好就写个本科毕业，然后淡化各种经验。这些都是扯蛋。都是错误观念。你就是你。只要你把简历做得别人能看懂，没有必要去隐藏一些你牛逼的经历。比如，像我，发过很多文章，引用也还行，我不会去列那些文章，因为不相关。但是我的简历就简洁的写了这么一句话来突出你不是个废物，而是一个很 productive 的人。

PUBLICATIONS

- About 50 peer-reviewed scientific publications with 2900+ citations and an H-index of 25

一开始的 summary 或者 key qualifications 很重要。这个基本上就是面试的时候说的 tell me about yourself 你最想提的三四个要点。这三四句话，也是 recruiter 或者 hiring manager 会第一眼就看的。

比如我的：

KEY QUALIFICATIONS

- A tenacious PhD researcher with a track record of unique discoveries and impactful deliverables.
- Led the development of market-dominating products, generating multi-million annual revenues.
- Passionate about applying software development and machine learning to make business impact.
- Skills: **Python, R, Java, Machine Learning, Django, Angular, Spring, MySQL, AWS, Heroku** etc.

反正，按照我的 mentor 的话说：

1. 一定要在 10 秒钟以内让人能够 catch 你最牛逼的地方。
2. 把你最重要的东西放在最前面。写每一段简历也是。有工作经验的，一定要突出 business impact。
3. 数字数字数字。。。当面对一堆文字的时候，别人最喜欢找的就是数字。
4. 层次要分明，每一段经历，都有一个总结。然后才有 bullet points。如果全是 bullet points，别人抓不到重点。

比如我们跟小伙伴一起做的那个项目：

- **Software Engineer @ Public Welfare Project 11/2019 – present**
Developing Citi Bike Plus, using live data streaming from 13,000 bikes, nearly 800 stations, and over 2000 trip/hour to empower users with real time route analysis, dock availability prediction and event promotions.
 - Contributing to all aspects of this **full-stack** software development that mines (**Machine Learning, Apache Kafka**), serves (**RESTful API + Spring + MySQL**) and visualizes (**Angular**) actionable insights, from the low-level algorithms and data structures up to the cloud deployment (**AWS**).
 - Aiming for 5-star-rating user experiences and saving 25% of time for customer during rush hour.

我的 LinkedIn 就只是简历的重复。几乎每天都有 recruiter 找我。当然，找我的都是 Sr 或者以上的职位。就是需要简单处理一下，跟他们说想先尝试 SDE I/II 的职位。FB 找我的是 MLE 需要五年相关经验。Google 一开始也是，但是我让安排 SDE II，等等。至少机会进来了。

LinkedIn premium 有用吗？

如果你只是为了有一份工作，不是只冲着大厂的话，绝对有用，我也强烈建议你开通。也许是我的 LinkedIn 页面比较好，也许是我开通了 premium，真的是 recruiter 天天找你。各种小厂或者猎头公司，多得嫌烦。

我有一种直觉，对于大厂，好像没什么用。但是我可能是错的。当然，premium 也方便你去联系 recruiter。亚麻和 FB 是我还没有开始 premium 的时候，就找上我了。我就只有知道废了亚麻的

那天下午，去联系各个大厂的 recruiter，结果这些 recruiter 一个都不理我。然后我所有大厂的面试机会，却都是 recruiter 们自己主动找上我。

有 Gap 重要吗？

不重要。我一直挂着一个学校 guest researcher 的 title，帮他们做一个数据分析的软件。这个是我简历的一部分。

还有前面说的，我们群里面一起组队做项目。

面试如果问我这一年在干吗，我基本就回答这些。

到底是 DS，SDE 还是 MLE？

这个真的完全看个人。

我自己是不考虑那种哪个工作比较轻松，因为在前公司就喜欢折腾，过着比 996 还要忙的日子。去掉这个考量，那么我只考虑前面说过的，自己喜欢，自信能够做好，而且还能帮助别人，这三个原则。

另外，转到哪个方向，对其难易程度的判断也是取决于个人。我觉得 DS 太难了，基础数学，统计，编程，各种调包，还要 domain knowledge。对于我这种不喜欢形式主义的，臣妾做不到。

事实上，后来才知道，我心目中的 DS 其实是 AS 或者 MLE。但这些都需 SDE 的基础。所以，如果我最终走到 AS 或者 MLE，现在先积累几年 SDE 也是靠谱的。

还有，我自己觉得转 SDE 比较容易，因为需要准备的内容范围相对于 DS 来说，范围小好多好多。比如对 google 基本就是刷题。对亚麻，主要是 BQ，简单的设计和算法。考虑到 BQ 只需要准备一次亚麻 14 条，主要剩下的就是刷题。设计也都是有章可循的。

但是，一条忠告就是，你只能挑一个。我知道很多人贪心，想 DS 和 SDE 一起来，除非你是天才，否则请只走一条路。我因为贪心，前期浪费了不少时间。

需要先去小公司吗？

我曾经考虑过外包公司，也曾经考虑过去给别人免费干活，也考虑过 intern。当然，基本上没有付诸行动。因为后来觉得，一线公司也有至少 10 几个，全部先去折腾一遍，实在不行再回头找小公司，小公司还不行，我还可以回硬件里面去找软件。我自己的经验告诉我，在你没有身份，经济这方面的顾虑的前提下，一定要把时间放在刀刃上。而且，小公司靠人脉，大公司才是标准化的。用一位师兄的话说，就一直折腾大公司好了，折腾了一圈，其他拒你的大公司也开始解冻结束了。反正建议就只盯着大公司。

需要去读个学位吗？

我反正没有。也就是说，至少我自己就是一个不要学位的可行的例子。但是看你自己。有个学位再加上学校的一些资源，肯定找工作要方便很多，尽管有前期的时间和金钱的付出。

怎样跟娃斗？

我们没有长辈过来帮忙，都是自己带的。如果没有娃，我的找工时间可以大大缩短。对我影响最大的还是 Google。疫情来了，娃在家了。真的。。。家里鸡飞狗跳。。。各种 distraction。真正完全属于我的安静时间，要晚上 8 点半以后。。。解决办法就是，跟小伙伴们一起刷题讲题。你会发现，在跟别人交流的过程中，外部的干扰对你的影响会小很多。这就是为什么需要小伙伴。真的感谢小伙伴们这段时间的陪伴。

还有，家里有个靠谱给力的队友才是关键。

面试是一种什么样的体会？

看芒果喵的分享。简单的总结就是，只是一个交流的过程。有一个活生生的人在你面前，而且是一位沟通好，就能给你 hint 的人，你要想办法用上。也不难，你就处理成工作中的场景，大家一起来做一个 project。

致谢：

家里的队友。

我的求职 mentor。

以教主和小天使为首的开车 1 群，2 群，和花花群。【不加新人！！！谢谢！除非你说服教主和小天使】

高中大学博士期间的同窗，师兄师姐师弟师妹们。

以令狐冲老师为首的九章众恩师。

以圆圆为首的众班班。

感谢芒果喵的九章分享！！！！！！

Thank you for your attention! Questions?