

1 Amazon Summary

1.1 面经

1.1.1 Onsite, 23 Jan

1. Find lowest common ancestor, with and without parent nodes.
2. min stack; combination sum, two sum.
3. 概念问了 reflection, garbage collector 和几种 GC 的区别, abstract class 和 interface 的区别. 然后让一个 interface 实现两个功能: 一个 check input 单词拼写有没有错, 一个给出 alternative words.
4. 第一个是给一个 deck of cards, 实现两个 methods, 一个是 deal, 这个 method 可以发一张 random 的 card. 另一个 method 是 shuffle, 就是类似 reset deck. 第二个是 binary tree 找到距离最长的两个 leaf nodes 包含的 nodes 个数。

1.1.2 Onsite, 22 Jan

1. [D] 设计一个 truck tracking system, 要求实现查询 truck 的地点, 每天的行程之类的。一开始写了一个 truck class, 然后在引导下一点点的增加功能, 然后问了下 database 的 table 应该有哪些东西。之后问了个算法题给一个 array 找出里面出现奇数次的数字。我先用 hashmap, 后来用 int[], 最后小哥说 no extra space, 卡住, 然后就结束了。
2. linkedlist 每个节点多一个 random pointer, 问怎么复制, hashmap, 很简单, 没让写代码。给一个平方数, 求平方根。就 $\sqrt{2} / \sqrt{2} / \sqrt{2} \dots \log(n)$ 的复杂度
3. 问了个循环列表, 删除指定节点, 写完让想一些 edge case 测试一下代码正确性
4. 给你一个 array, 不用循环把所有元素复制到另一个 array 里, 用 recursive, 然后问了个火箭打彗星, 火箭每次发射之后需要五分钟准备时间, 雷达会监控彗星出现的距离, 每个火箭只能打最近的一个彗星, 问用什么 data structure 记录彗星的情况, 要求时间复杂度 $O(1)$ 的, 不会, 然后他说换一个题吧, 就让写了个 quick sort 伪代码, 写完一起举了两个例子测试了一下。

1.1.3 Onsite, 20 Jan

1. 第一题: BST lowest common ancestor, 注意判断下两个 node 是 ancestor 的特殊情况。第二题: Merge two sort linked list. 第三题: Two stack implement one queue
2. 说一个很长很长的 list 里面有很多 amazon 的产品, 每一种产品都出现了很多次, 所以有的产品比较 popular, 让你找出前 100 个最 popular 的产品, 并且输出。我一开始写 hashmap 构建了一半, 她叫我不要写了, 然后讨论思路, 我后面说用 priorityqueue, 搞个 comparator, 往里加然后 size100 个就行。
3. 第一题: 说只能用 recursive 的方法 reverse 一个 string, 比如 "I am a cat" -> "cat a am I". 我看错题了, 写了个 recursive reverse 整个 string, 后来发现是单词, 然后卡住。后来发现一开始就该 split。第二题: 有两个 array, a 和 b, 输出一个新的 array, 只包含 a 里的元素, 但是不包含 b 里的元素, 两个 array 都可能有 duplicate。
4. 问了一题: 【2, 3, 4, 6, 7, 8, 9, 12, 13, 14, 26, 27, 29】, 然后有个 class Range { int begin; int end; getter and setter}, 然后 Range 就是连续数的 object, 之前那个 array 中间有些 gap, 比如 4 和 6 中缺 5, 9 到 12 缺 10, 11, 然后一个 Range 就是比如 begin: 2 end: 4, 这样是一个 range, 因为连续。让你输出对于一个数组的所有 range 的 list。比如上面这个 array 就该输出: Range (2, 4), Range (6, 9), Range (12, 14), Range (26, 27), Range (29)

1.1.4 Onsite, 15 Jan

1. 第一题，就是给 array，找后面最近的比他大的那个数，如果没有找到输出本身。比如说 [1,5,3,6,9]，那么 output 是 [5,6,6,9,9]。因为看过面经，所以很快就给出 stack 解法，从后往前扫，我在白板给烙印演示，然后烙印看我会，就没叫我写代码。直接问了时间复杂度，我没过脑子，就直接说了 $O(n)$ 。然后居然还设个陷阱为难我，说他觉得 worst case 是 $O(n^2)$ ，因为有可能数组当前那个很大，那么 stack 就要一直 pop 到最末尾，比如说 [9,1,2,3,4,5,6,7,8]。
2. 问的 the largest repeating substring。比如 banana，输出 ana，然后我不会怎么优化，就只好一个一个 substring 从大到小得扫，碰到有重复就直接输出， $O(n^2)$ 。但当时真心想不到了，后来在图书馆才想到可以用 binary search 扫那个 substring 长度，这样复杂度能继续降一点。感觉貌似要用到 suffix tree 来做。
3. 图。有 user，每个 user 有 friends，有喜欢的书。就是把某一个 user 当成 root，往下扫，输出它朋友和朋友的朋友...一堆人的书（不重复的），那就直接 BFS+HashSet 扫了。当场写完面试官比较满意。
4. [D] OOP chess。这个就主要和面试官交流了，最后写完了，面试官说 good good，照了个相，就带我出去了。（OOP 大家还是多准备几个吧，啥 chess, elevator, parking, twitter, hotel reservation, restaurant reservation, traffic signal, chess, furniture 和 cracking 上面的，反正我是都写了一遍，最后幸运的考到了答出来也不慌，推荐一本书 Cracking The Java Interviews (Java 8), 2nd Edition: Targeted for Investment，谷歌上面有电子版的，书上有 oop 一些例子代码和讲解，感觉例子也是 amazon 常考的，大家可以参考

1.1.5 Onsite, 15 Jan

1. [D] 给定一个 string。比如说是 "hello"。判断输入的 string 是不是他的 substring。输入 "ell" 返回 true，输入 "eo" 返回 false。（我面完这题觉得很简单，但后来同学告诉我，这题要用 kmp 算法做）写完之后，问如果 "oh" 和 "ohel" 这种也算是原 string 的 substring，应该怎么办。（把两个元数组相加，其他代码不变）；OO design：要给一个家具工厂的所有家具做质量测试（压力测试，是否易燃等等）。
2. 输入是一个城市的地图的大小 (m, n)，和一个 list，里面包含所有有 locker 的地理位置。输出一个 $m \times n$ 的二维数组，每个单元的值到最近 locker 的距离。问时间复杂度（这题要从每个 locker 同时开始 bfs）
3. 给一个图片，有着亮的点和黑的点，判断一共有多少个黑暗的块儿。时间复杂度。（这题我觉得相当于找出一个图中有多少个联通分量，我用 bfs 做的，用一个二维数组记录每个点是否被访问过）
4. 问 linkedlist 怎么回事，hashmap 怎么回事；给一个二维数组，都是整数，每行都是从小到大排列，每列也是从小到大。（但是第二行的第一个不一定大于第一行最后一个），给一个 target，判断是否存在于这个矩阵中。

1.1.6 Onsite, Dec

1. given a string, output character and frequency in decreasing order of frequency (ignore other chars that's not letter)

```
input: "Aab s."
output:
a:2
b:1
s:1
```

2D matrix with only 1 and 0, continuous 1 means island, count the number of island

2. [D] project; goal that you spend long time to achieve. Design Elevator system
3. design data structure that have two methods: check_in and check_out. 就是相当于一个装有 integer 的 pool，check out 就顺序输出里面已有的数字，check in 就是放入数字。

```
e.g:  
check_out - output 1  
check_out - 2  
... (7 times check_out)  
check_out - 10  
check_in(2)  
check_in(7)  
check_out - 2  
check_out - 7  
check_out - 11
```

1.1.7 Onsite, 13 Jan

1. [D] 你在 project 中有没有开始本来没有想实现的功能，后来意外修复并且增加了意想不到的功能（我一听这是什么问题了，我怕我听错了，又确认了一下，好吧，我没听错，就开始瞎扯吧）；你在 project 有没有一些时候 teammate 没有时间完成 TA 的 part, 你帮 TA 完成了，怎么完成的，举例子（又开始瞎扯了）。然后他就说我们来个一个 design 的问题吧，我一听就知道小虚。。。。。题目是：设计一个 traffic management system. 大概就是一个十字路口，有好几个方向的汽车，红绿灯，行人，设计这个的 controller 吧，然后边讨论思路边写 controller 的 method 什么的，最后讨论讨论怎么 initialization 就草草结束了。
2. 一个 undirect graph, 每个 edge 都有一个 weight, 这个 weight 都不同或则相同, 给两个 node A, B, 问 B 到 A 的最短距离。一开始问有没有 cycle 什么的，他说有 cycle。然后讨论思路。
3. 给一个 unsorted positive integer array , 没有 duplicate, 输出是一个 integer 和这个 integer 后面第一个大于该 integer 的那个数

```
input(int[]): 4, 2, 6, 8  
output(List<String>): 4---6  
                     2---6  
                     6---8  
                     8--- -1 (因为8后面没有大于8的element了，用-1表示)
```

4. Validate Sudoku. hashset 时间复杂，我说 $O(1)$ 吧，他说重复怎么办，我说 worst case $O(n)$. 可不可以不用 hashset, 让查找变成 $O(1)$.

1.1.8 Onsite, 09 Jan

1. Given a grid which represents all location points, there are some Amazon lockers in the grid, given a customer's location in the grid, find the closest locker.
2. [D] OOD design a poker game and different methods in each class.
3. validate BST using three different methods, array vs linked list, how to implement hashtable.
4. given a list of player, each of them can choose not playing the game, playing with a specific player(preference) or playing but does not care about the preference. find a team with five players that meet all constraint.

1.1.9 Onsite, 06 Jan

1. 最后题意其实就是有个字典，里面有一堆单词，还有 a bag of chars, 要求返回 a list of words in dict where each char in the word is in the bag

2. multithreading, bfs, dfs, bfs, dfs 的 space cost, 还有啥情况用 bfs, 啥情况用 dfs 更省空间吧. quicksort 的时间复杂度, 那你知道 $O(n)$ 的 sorting algorithm 吗, max heap/min heap. 这里出了道题 find the largest 50 integers of an integer array, 然后把 integer array 里的元素改成一个 record class, 有 id 和 time stamp, 要求 find the largest 50 records based on time stamp with id = 'x'
3. 题目是 2d matrix, 从给定的一个坐标出发, 往四个方向 search, 搜到 2 返回 true, 搜到 1 能走, 搜到 0 不能走。问我懂不懂数据库。。不太懂。然后就问我懂 design pattern 么, 我说知道个 singleton, 让实现, 结果我挖个坑把自己埋了。
4. why amazon, 怎么赶 deadline. array 和 linked list 区别, 后面问了根据 url 访问网页, 你会采用啥 caching mechanism. 我说 LRU, LFU, 他说你实现一个吧, 开始写 LRU, 然后问 LRU 有啥缺点

1.1.10 Onsite, 06 Jan

1. 印度人, 题目是关于字符串内容的压缩处理, 即给了一个很大的 context 文本, 怎么能用比较少的 memory 去 storage, 老实讲不是很清楚他具体想问什么, 就跟他慢慢扯, 聊得过程中发现他是想建一堆 structure 去存储文字和相应的出现 index, 例如.

```
structure word{
    string name;
    vector<int> index;
}
```

然后讨论一下 trade off, 下一步又问能不能对单个词再压缩, 例如 stats, naan, 这样的词是对称的, 可以进一步改变 structure

2. 是个亚裔, 但是不是中国人, 来了又先问了一下 string 对称怎么判断, 大家注意 invalid input; 然后, 实现具有取出最小值功能的 stack; 然后, 用两个 stack 实现 queue.
3. 美国人, 上来先相互寒暄了一下, 然后开始做题, 给了一堆 date, productID, name, price, quantity 的数据行, 然后慢慢一遍解释需要用这些数据计算什么结果, 一边讨论用什么数据结构, 这一轮因为 stl 的一些底层实现我不是很清楚, 但是美国人是主要用 java 的, 所以交流有点吃亏, 感觉不是很好.
4. 最后一轮, 同样是给了一大堆 date, productID, name, price, quantity 的数据, 然后, 让你预测某个商品下周和下周的总销量, 大家都应该知道是 ml 的内容, 但是讨论过程中 HR 不断强调 30 分钟实现线性回归什么的复杂算法会写不完, 这样没有 credit 给我, 所以纠结了一下问能不能用平均值来代替, 同意 (不知道这里是不是设了陷阱让我跳?), 然后写代码, 写完了之后又聊了一些 behavior question.

1.1.11 Onsite, 18 Dec

1. 上来随便扯几句直接 LevelOrderZigZagTraversal, LeetCode 原题。follow up 问我为什么用 ArrayList 不用 Queue 等等的问题.
2. LRU Cache, 不要求直接写代码, 一步步讨论实现。会反复问“你觉得自己的代码没有问题了吗”, 没看出来就举个 example 引导, 压力略大的.
3. 坑爹的黄眼镜儿白人, 不知道哪儿的口音, 一进来就问问题, 问的题目之前也没见过。给你一个.csv 文件, 每个 entry 有 employer name, employer ssn, manager name, manager ssn, employer title 几个属性, 给一个 FileIterator 来读取每条 entry, 给一个 Person class, 里面有 String name, String ssn, String title 和 List reports (向这个人汇报的所有人), 要求用某种数据结构存这些个 entry, 然后在里面找谁是 CEO。哪位大大给分析下这道题具体怎么做?
4. nice 的白人大叔, 响应非常及时。题目是简单的给一个 Integer Array, 返回一个 duplicate, 加上各种 follow up, 比如 ArrayList 的 contains 怎么实现的, 换成什么其他数据结构比较好, hash function 怎么实现, hash

collision 怎么解决，不用 extra space 怎么办，Array 里面有一百万个元素怎么改，返回所有的 duplicates 怎么写等等。

1.1.12 Onsite, 10 Dec

1. binary tree 里给个 node，找出比他大的下一个 node.
2. 从 head，一次改一个字母，要求改完仍是一个有意义的词，找 path。这轮基本都在谈思路，感觉不是很好，我用 BFS 的. (word ladder)
3. 给一串没空格的字符串，判断是不是一句话，之后会叫你返回加上空格后的那句话。
4. 给一个 int 的 array，值代表高度。求这个东西能储存多少水. (trapping rain water)

1.1.13 Onsite, 18 Nov

1. 判断 subtree，说了常规方法。。然后不知道脑子怎么想的，说了一个自己都没写过的方法能到线性
2. log 系统 url 统计，没扣定，纯思路，中间夹着复杂度，数据结构，scale 各种问题。还有很多 behavior
3. 白 aws sde. serialize / deserialize tree，聊项目把自己用过的 aws service 扯了一遍，尽快跟小哥建立交流很重要。小哥思路很快，中间一直在研究我的方法然后每行都记录，然后兴奋的告诉我找到一个 bug。结束说我有地方方法不错，但是还可以怎么怎么做，然后 follow up
4. object two sum。扣定加各种讨论，前三轮完略有点放松，简单的地方也 buggy 了。。很多 follow up，各种数据结构，treemap，linkedlistmap 讨论

1.1.14 Onsite, 11 Dec

1. valid parentheses
2. polite number polite number 的意思是一个正整数可以表示为几个连续的正整数的和。

$3 = 1 + 2$ $15 = 1 + 2 + 3 + 4 + 5 = 4 + 5 + 6 = 7 + 8$
--

给你一个正整数，让你求出一共有多少种表示的可能性，比如 3 就返回 1，15 就返回 3。

3. 一个 cloud 里有 N 台 server，比如里面存了一个变量叫 $x=3$ ，和这个变量同时存储在 server 里的还有这个变量的 version。进行了一次写操作改成了 $x=5$ ，同是将相应的 version+1。由于写操作比较耗费资源因此每次写并不是把所有 N 台 server 的数据全改写了，只写了 W 台。现在另外一个 client 要读出 x 的数据。为了保证 consistency，必须得到 $x=5$ 而不是 $x=3$ 。假设需要读取 R 次（在已知 N 和 W 的情况下）我才能保证我得到的是最新的数据。那么 N,W,R 有一个什么样的关系？

1.1.15 Onsite, 3 Dec

1. multithread, dead lock
2. 设计 tic tac toe 游戏设计整个游戏不只是怎么判断赢
3. lc 原题 Longest Palindromic Substring
4. ctc 原题 8.9 Entry File Directory 那个具体题目忘了然后让写一个 search 函数

1.1.16 Onsite, 3 Dec

1. 白人老头。聊简历。最后问了一个关于测试的问题。就是说如果你的 Boss 告诉你说这个网站很慢，你应该怎么做？我也不知道，就瞎扯了一通。然后问了一个 OO design 的问题。设计一个 System 玩 21 点。可以有 AI 和有人玩。

2. 一个亚裔面孔的女性，说是在澳大利亚读书，然后来美国工作。问了一个和 amazon 工作实际相关的问题。给你一个函数返回所有 amazon 有的 Product，还有一个函数返回 amazon 仓库里面的所有 Product 记录。用这两个函数找出特定要求的货物。
3. 说是要面数据结构和算法。结果上来就是二叉树的先序遍历递归和非递归的方法。我自己傻逼了耽误了很长时间。然后问了问关于 Hash 的问题。让自己实现一个 Hash 的 contains 方法。
4. 一个印度人。纯设计题。设计一个 FC 仓库。要求有车，有一个 dock 用来卸货，有一个 FC 有好多 dock，有一个 controller 控制车去哪个 dock 卸货。

而且设计的时候要考虑 large scale。他家特别喜欢这一点。第一个和第四个都提到 large scale 了。第四个设计题是直接问如果是 large scale 你的系统要做出哪些调整。我说了个就是用 distributed controller 然后他就问那如果保证 controller 之间不会冲突，不会获取到同一个 dock。我说给 dock 加锁。然后他就问如何给 dock 加锁。因为没有时间了这一点就随便扯了扯。

1.1.17 Onsite, 11 Dec

1. 第一题是 cc150 原题, string compression, 但是要求字符个数为 1 的时候不输出 1, 只输出字符; 第二道题我没见过, 就是一圈 zombies, 给你一个 starting point, 然后每隔 k 步你 shoot 一个 zombie, 完后依次下去直到剩下一个 zombie, 让你输出最后剩下的 zombie。我用 linkedlist 做的, 然后 while loop 每次 shoot 一个 zombie, num-1 删除那个 listnode, 直到最后 num 变成 1 跳出循环。
2. 给你一个 sorted array, 完后输入一个数, 让你 output 它在 array 里面出现的次数。这道题比较 tricky 的地方是, corner case 的考虑。每个人都能想到 binary search, 但是在处理 binary search 的时候有些 corner case 要处理好。三哥一直提示我哪里要考虑 edge case 哪里有 tricky part 真的超级 nice。
3. 问了一些 project 的问题, behavior 问题例如 how do u deal with very tight deadline?(貌似亚马逊很喜欢问这个), 完后问了一道 leetcode 原题, integer to roman.
4. 貌似是 bar raizer, 带了一个三哥进来, 问了很多 project 的问题, 完后一道算法题给你一个 unsorted int array, 输出所有出现 odd 次数的 integer。我给出了 $O(n^2)$, $O(n\log n)$ 和 $O(n)$ 的算法, 完后让我实现 hashmap 的部分。完后让我比较 space complexity, 设计 test case。

1.1.18 Onsite, Dec

1. 设计一个 dog class, 实现其中的 breed() 函数, 意思就是追踪一个狗的品种的纯度。例如: 纯 dog a 和纯 dog b 交配得到的 dog c 就是 50% 的 a 纯度和 50% 的 b 纯度, 以此类推。
2. 设计一个 class, 然后写其中一个方法去 parse 销售记录, 记录自己定义, 其中包含 day productid quantity price。给 5 天的销售记录, 输出的第六天和第七天所有 product 的预测销售数量。可以用 ml 算法, 但跟面试官聊聊就行, 不用写。实现一个最简单的算法就行去预测就行。这个是 bar raiser 的题, 因为他是 senior manager
3. 就是面经里的题目: 一个矩阵有 0 和 1, 问 1 一共有多少堆。第二题是 leetcode 的 atoi。第三题是给一个 string, 转换成 n 进制, n 是参数在方法里有
4. 最后一轮很简单, 给一个 string, 返回 unique character, 用 hashmap 或者 int[256] 都行

1.1.19 Onsite, 23 Oct

1. 按行从左往右打印一棵树, 写代码。follow up 是 zigzag 打印树, 在之前代码基础上修改。最后是写代码打印树的最外一圈。
2. 这轮比较简单, 就是聊天加 OOD 设计, 聊一些 OOD 的概念和一些细节, 然后设计纸牌游戏, 还让画类图。
3. 二维迷宫 ($m \times n$ 矩阵), 每个点代表一个房间, 房间与房间之间有的联通有的不连通, 有的房间里有金币, 给一个

起点坐标，收集所有能到达的房间的金币，返回金币总和，这个很容易。follow up 是，每个点可能是一个房间，也可能被划分成多个子房间，子房间的性质也是一样的，房间之间有的联通有的不连通。重新设计数据结构存这个迷宫，并写代码收集金币。

- 标准 leetcode 的 two sum。我分别写了 $O(n^2)$, $O(\log n)$ 和 $O(n)$ (哈希表) 三种方法的代码，每种方法其实就 5-8 行代码就搞定，不费什么时间。然后面试官说，让我给代码加注释，然后写单元测试。。。

1.1.20 Non-OA, Onsite, Dec

- 问了个 anagram 的题目, leetcode 原题, 然后问了大量的 behavior 问题
- 问了个买卖股票一次获取最大 profit 的问题 leetcode 原题, 然后问了大量的 behavior 的问题和一些设计
- 问了个 matrix 的问题

```
0 0 0 0 0
1 1 0 0 0
1 0 0 0 1
0 0 0 0 0
```

0 表示空的 block, 1 表示 train station, 让你输出一个 matrix, 每个点的 value 是这点到离它最近的 train station 的距离.(上下左右都可以走)

这个输出如下

```
1 1 2 3 4
0 0 1 2 1
0 1 2 1 0
1 2 3 2 1
```

开始想 dp, 发现行不通, 因为上下左右都要考虑, 没想出 dp 怎么弄。用的暴力迭代, 先把 train station 的距离设为 0, 其他-1, 然后再把 train station 周边置为 1, 然后在依次迭代, 每次距离 +1, 所有的坑都填上。

- 问了个如何设计 twitter 的 design 完全不会
- opentable, design. 完全自己设计数据结构和接口。需求, 用户输入 restaurant, timeslot, 人数, 返回是否可以预订。一个 restaurant 可能有多个桌子, 每个桌子可能可以坐多个人。如果预订的时候, 人数大于一个桌子, 可以把相邻的桌子 combine。timeslot 30 分钟算一个, 一天就可以算 24×2 个, 编号从 0 到 47, 用户的输入只能有一个 timeslot。

1.1.21 Onsite, 20 Nov

- 一个 non-negative integer array, 只有一个数字是奇数个, 其他都是偶数个, 找奇数的那个数。integer 换成 object, 所以不能用异或。上一题, 可不可以不用 extra memory
- 设计一个 zoo
- 模拟一个社交网络, no cycle, 所以实质是一个 tree。每个人有一个 booklist。a. 找到 tree 里 top 3 most popular book; b. 找到 tree 里 top n most popular book
- a bulls & cows。b. 找到 integer array 里, 数字和 index 相等的数。c. 找到一个 2d integer array(都是 0 和 1) 里的 region 数。如:

```
[1,1,0,1,1]
[0,0,0,1,1]
[1,1,0,1,1] 有3个region, 因为有三堆1被0隔开。
```