

# BB 近期面经

---

## 1. 11.1

第一轮：只有一个印度人面我，整体气氛比较严肃，第一题利口伊散霸。

第二题设计题 given 一些公司的名称和 execute 数额的大小实现两个 function, `execute(share_name, execute_amount)` 和 `printTopK(share_name)`, 顾名思义 execute 就是把 `share` 加进去, `print` 就是把 `k` 个成交量最大的拿出来。

第二轮：美国白大叔，非常和蔼可亲。只有一道题设计题，非常 open ended。进来之前还说怎么上一轮只有一个人面我，说可能今天 run out of interviewer 了。题就是利口散霸零变形，设计 raffle system。楼主一开始没反应过来，给了不是最优解。最后看出来这个题的本质，就解出来了。

第三轮：早有耳闻 hr 小姐姐很好看，是真的很好看啊！扯了扯皮

第四轮：美国白人小哥 senior manager，一身西装。非常帅。又是问了我一个非常 open ended 的设计题。怎么设计电梯。这个楼主是从来没有看到过。就按自己的想法说了说。之后他说我这是第几轮，我说第四轮了，上轮 hr。他说我想去哪里看看吗，我就说想去顶楼看看，因为顶楼的景色听说很好。然后他就带我去楼上转了一圈，最后给我送下楼了。

体验整体感觉不错，酒店是小有点，不过楼主觉得也非常不错了。等了一个礼拜多还没有消息，今天催了下，hr 说一周以内会出消息。不知道为什么这么慢，看地里说基本很快都出结果的。

## 2. 11.5

只问了两道题也许因为第二题回答得太慢了，但其中一个面试官小姐姐真的人很好，以后可能再也遇不到这么耐心这么 nice 这么 friendly 的面试官了 T^T。

第一题 Binary Search Tree 求第 1, 2, 3, k 大的 node，因为对 BST 熟悉算是解出来了。

第二题是给一个 two dimensional array 模拟一个墙砖，怎么 cut 每个 row 使得破坏的 brick 的数量最少，return 最少的被破坏砖块的数量。但因为想复杂了，同时太紧张了花了太多时间，也不知道除了 linear 的解答是否有更好的方法。

## 3. 11.9

第一轮国人小姐姐，好看。很多银行开门关门时间，客户交易时间，问交易是否可以被完成，类似 **interval overlap**，不难。还有一道设计数据类型，不用写码，重交流。

第二轮国人大哥□南亚小哥。大哥功力深厚，人也很好，利口要散霸扩展到药散散，写了后者的代码。小哥问了尔酒舞，答的有些小瑕疵。

国人大哥走后来了一个回马枪把我领下楼，说我有些紧张，我决定还行，可能交流赘述让人感觉紧张了，还要多练。

## 4. 11.8

【第一轮】两个白人面试官，简称为没胡子和有胡子，看得出来没胡子更 **senior** 一些，有胡子给我感觉还比较看他脸色

第一题：BST 中找第二大的数，我说用 **inorder** 相反的方向遍历这个 **tree**，**global var** 记录最大的两个数，返回第二大的就好了，中间找到第二大之后持续 **return** 就好了，他说不允许用 **global var**，行吧，我给个 **list** 往里面 **add** 你管不着了吧

写完之后还一堆破事，问我你在阿里实习是吧，那么大的公司肯定也有自己代码准则，比如变量命名和 **comment** 之类的啊，你咋应用到你这段 **code** 上。我第一反应：你说尼玛呢，什么跟什么啊，反正跟他掰扯一阵子终于放过我了，楼主最不喜欢出简单题还净事儿的面试官，你给我一道 **hard** 我还愿意讨论讨论，你出这个题问的都是没话找话的烦不烦？

第二题：轮到有胡子了，有胡子的题目是地里有的交税，我第一遍给的解是把 **threshold** 和 **percentage** 都放 **array** 然后 **reverse go through**，后来说可以 **binary search**，找到比给定数字小的数中最大的，然后如果有多个人就做 **pre-calculation**，这样对每个人就再算一次就好了，看得出来有胡子挺满意的，就是碍于没胡子板着脸所以说不错的时候声音都很小，贼真实

【第二轮】白人小哥和华人小姐姐的组合，两个人都很 **nice**，舒服多了：

第一题：在 **String Array** 里找出所有的 **anagram**，答对所有的 **word** 做 **sort**，一样的利用 **hashmap** 放同一个 **list** 里，返回 **list of list** 即可，我说做 **quicksort** 如果 **word** 太长 **performance** 会很差，提出用 **bucket sort**，然后 **implement** 给他们了，后面的 **follow up** 分别是：忽略大小写怎么办，有特殊字符怎么办，有 **wildcard** 怎么办？都解决了之后两个人都说很满意

第二题：剩的时间可能不那么充裕，小姐姐怕我写不完让我说想法，其实这个题写起来很快肯定写的完的：一个画图，实现 **redo** 和 **undo** 以及 **fill**，前面给两个 **stack**，**fill** 就 **dfs** 或者 **bfs** 就好了

这里其实很有意思，面完之后看两个人就我坐在这儿还是回去好像得出了不同的结论，其实很好推测，第一轮没胡子肯定想让我两轮游，第二轮的两个人看了 **feedback** 之后来面我的，白人小哥觉得没胡子那个评论可能没后续了，但是小姐姐觉得我这轮还行就去找 **hr** 了，果然等来了 **hr** 和 **manager**（虽然是推测不过我觉得和情境很吻合）

【第三轮】HR 标准式问答，忽略

【第四轮】亚裔 **manager**，完全没口音判断不了，整个围绕我的 **intern** 项目屡了一遍，超详细，我觉得我简直差不多把系分给他讲了一遍，一直撸到四点，（甚至都没时间给我 **show terminal** 了）回去拿包的时候发现早上围在那一堆的包只剩我的了 ==

不得不说 **BB** 的办公环境真的是不错，养鱼也是很有想法了，听说他们家结果挺快的，就等吧...

## 5. 10.8

今天的纽约特别暖和，10点半到了bb的大楼开始tour，tour结束后大概11点，大家陆陆续续被面试官接走，只有我一脸期待地等到了最后都没人来带我去面试.....当recruiter小姐姐告诉我大家先去面第一轮了，我只用等到12点面第二轮的时候我是懵逼的，邮件上明明说的是两轮呀？

第一题：给一个undirected graph，每条边都有各自的cost，求一个能遍历完所有nodes并且total cost最小的路径，返回total cost

第二题：设计电梯

一个ood的经典题目，假设一栋楼五层，有三个电梯，人可以在任何楼层按电梯的上/下键去到另一层楼，然后问需要哪些class/variale/function来实现这个电梯系统

## 6. 10.5

### 1. 华人小哥+中东胡子小哥

第一题 word anagram + followup substring anagram,

第二题记得是lc题，doubly linkedlist带child，要求把child都flatten最后return doubly linkedlist

2. 面无表情 白人大哥，眼睛会不停的眨，可能不舒服，我觉得可能出问题的话就出在这一轮，大哥心中有个答案可能和我说的并不match

median of data stream 用俩pq做

LFU Cache

### 3. HR

HR小姐姐有点困，疲惫

吹

### 4. manager

18年大哥主面，12年大哥没讲几句

吹，

最后要求展示terminal，awesome, amazing

最后两个大哥送出电梯，告诉我会很快出消息



魂守蓝桥 发表于 2018-11-8 01:50

请问楼主manager面基本就是聊简历项目是吗？会问算法ood或者系统设计吗？

我是只聊项目 然后聊了下课 我也有朋友考了design 应该不会很难

## 7. 10 月底

一共四轮，前两轮技术面每轮一小时左右。第三轮 HR，半小时。第四轮 Manager 45 分钟至一小时。

前两轮一共四道题。

1. 设计消息处理系统一个含有 `set(key, value)`, `get(key)` 函数，并使得当系统内 `set` 的 `record` 超过 5 分钟后会自动 `expire`。
2. 给一个 `arr`，找最大的三个数乘积。原题。
3. 给一个二维 `grid`，每个 `cell` 内有一个 `int` 值，当相邻的 `cell` 有同样的值的时候可以认为这两块是连着的。求 `grid` 中连着的最大面积是多少。（很相似于岛的题）
4. 设计一个类似计算机处理的时间系统。按照时间线每当一个 `user hit` 了系统一下，系统都会记录一下 `count`。`count` 会一直累加。设计一个函数可以获取之前 1 个小时内 `hit` 的总数。比如 10 点时候 `hit` 了 10 下，9 点的时候 `hit` 了 9 下。那这段时间内一共就是 1 下。  
但要注意，从当前时刻往前退一个小时的时候不一定正好那个时刻有 `hit`。所以 `follow up` 要想怎么优化处理最快。

HR 面

单纯聊简历，聊了一些经历问了 `bq`。但是也没问到经典的 `why bb`。之后他向我介绍了一下 `bb` 进公司后 3 个月培训的 `program`。

Manager 面

聊了简历，聊了项目。之后做题。

1. 介绍 `BST`，然后写个算法找 `BST` 第二大的数。
2. 给你一个名为 `F2()` 的已知函数，函数输入不用管。输出是等可能 0 或者 1（50%）。让你利用这个 `F2()` 的函数设计 `F3()-FN()`。  
以 `F3()` 为例，`F3` 输出的应该是 0, 1, 2 且每个值的概率为 33.3%。

自我感觉 `BB` 家题不难，但是有很多小细节的 `follow up` 和优化要做好。

## 8. 10 月底

第一轮，亚洲小姐姐和白人大哥。

上来 10 分钟 `behavior`，`why bb`

做题 `candy crush` 和 `topo sort`。不用直接跑 `test case` 但是要口头过

`candy crush` 是简化版本，`AABBBA -> ""`, `ABBBA -> AA`

`topo sort` 是地里有题，`A -> B -> C -> D`, `E -> D ==> A -> B -> C -> E -> D`

题都不难，很基础。在 `hackerrank` 上写

写完也没什么 `follow up` 说一下 `space` 和 `time` 就好

第二轮，白人小哥和印度小哥。

上了还是先 **behavior**, **why bb**

上来包装过的 **LRU**，写完再说怎么写 **LFU**

第二题是 **word scramble**，给 **list of char**，**find the longest possible word**

全程印度小哥疯狂问问题，白人小哥都看不下去了，说差不多了

因为第一题被印度小哥怼了很久，第二题中间又被怼了好久，所以第二题没写 **code** 只说了思路

**HR**,

非常 **causal** 的面试，**HR** 英国小哥，就聊 **why bb**, **other offers**

**Manager**，印度大哥

很和蔼的老哥，各种 **behavior**,

然后一道 **design** 题，大概意思是记录每个股票在每个 **exchange** 的价格，两个 **map** 解

然后问 **map** 加锁，还要支持 **max min** 和别的一些需求，具体记不清了

问完让我问问题，结束送下楼。

另外 **BB** 比较看重 **why bb** 和类似下一份工作希望是什么样的这种问题，请各位注意准备。

## 9. 11.2

第一轮是一个中国小哥哥和一个白人小哥哥，中国小哥哥出的题目

一开始让我讲讲自己的 **internship**，然后做题。

给一个起点和终点，要往这段路上放水站，写一个 **class** 实现两个 **function** : **addWaterStation()** 和 **removeWaterStation()**, **add** 的时候要加在最大的 **interval** 中间（第一次 **add** 放在起点，第二次放在终点，），**remove** 的顺序要和 **add** 的顺序相反。一开始想用 **pq** 去找最长的 **interval**，但 **java** 里 **pq** 不太方便删，就用了 **bst**

我怕自己想的太复杂，一直和面试官讲了我的整个思路才开始写的，讲思路的时候小哥觉得可行，然后开始写代码，写的时候也一直在讲自己的思路，两个面试官都不懂 **java**，我一直在说我在干什么，这个数据结构在 **java** 里面是什么。写到最后，我说跑个 **test** 吧，中国小哥哥说不用了，他觉得可行。白人小哥哥让我分析一个时间复杂度，我详细分析了一遍。

问问题

问了他们为什么选择写一个自己的 **comdb2**，还要自己给各个语言做 **extension**，回答：他们想用 **rdms** 的时候这个东西还没成熟，他们就自己写了一个。

问他们的 **open api** 有没有一部分是 **open source** 的，回答没有，靠这个赚钱呢

问他们最喜欢 **bb** 什么：**work life balance**

第二轮是一个亚裔小哥哥和一个印度小哥哥\*，印度小哥哥出的题目：

先问了我的 **project**，我用 **angular express** 做的，让我详细给他讲了 **project** 的架构，问前端从提交开始整个 **project** 的 **life cycle**。问我 **mongodb** 和 **mysql** 的区别（主要讲了 **schema flexibility**），**angular js** 和 **angular** 的区别。然后问我怎么测试，我说 **angular** 用 **jasmine** 和 **kamar** 做 **unit test** 和 **user interface**，后端 **unit test**。问我如果 **database** 在云端怎么办，我说建一个 **local** 的 **db**，用少量数据测试。然后用一个 **environment file** 存不同的 **connection string**，

production 的用远程服务器， development 用本地的。

做题， leetcode 399。 准备谷歌店面的时候看了一下题目， 没做过。bb 现在都是这个难度了  
么。。。 现场 build graph 然后跑 dfs。 写之前先沟通了算法， 分析了可能的 invalid input 的情况。  
我写了四种：1 给的变量不存在 2 给的两个变量不连着 3 本身的图是 invalid 的 4 给的 ration  
是 0。 面试官说先 assume 这些都不存在， 写完之后问我怎么判断这些 invalid 的情况。 其他都和  
简单， 本身图是 invalid 的时候找 cycle， 如果 cycle 乘积不是 1 就不 valid。 跑了一个 test， 分析了  
复杂度。

问问题环节。

问了 development cycle

## 10. 11.2

把问题背景去掉了

第一轮，

1. 给一个 double 数组， 和一个值 v  
return 一个数组， 使得数组里所有元素的和大于等于 v，

要求：

- 1) 数组的长度尽可能短
- 2) 元素的和与 v 的差值尽可能小

第二轮。

1.LC 582,

2. 给一个 string 和一个 threshold value L, 返回一个 string such that no substring of the result  
contains more than or equal to L consecutive repeat characters.

比如： L=3

aabbccbbd-> aabbbbd->aad ,return aad

## 11. 10 月底

第一輪 ABC 小姊姊跟華人小姊姊

一上來輪流問了簡歷還有 why bb

第一題就是簡單的 DFS

就是給你兩個機場 要從機場 A 飛到機場 B 還有一堆不同機場間的班機 要 print 出機場 A 飛到機場  
B 的所有路徑

題目不難 但花了巨多的間在跑 test case LZ 用電腦打扣 但他們不要電腦 run 用手動 run code 導致  
第一題寫完根本沒剩多少時間

第二題是裡扣妻散舅

LZ 一開始忘了怎麼寫 只給出個  $O(n^2)$  暴力解 小姊姊叫我打暴力解的扣後突然想了起來 就用白板  
解釋一下算法就草草結束了

第二輪一個美國大叔跟歐洲小哥

也是輪流問了簡歷還有 why bb 美國大叔還說我認識有人在你實習工作的公司 你這產品我沒聽過  
是新的嗎？

第一題是 shuffle linked list

給你一個 linkedlist 先把他切成兩半 然後再把兩個 merge 成一個 linkedlist merge 時要 random 從第一個拿或從第二個拿 但 merge 後在原本在同個 linkedlist 的元素順序不變  
寫扣期間大叔不斷的打斷我 光一開始把 linkedlist 切一半這段就問了五六個問題 具體忘了 一直說這段扣需要嗎我們是不是可以優化  
或是表達對 shuffle 的定義的質疑 隨機從兩個 linkedlist 拿是對的嗎 是不是要按照剩餘多少元素的比例去求機率 不否認大叔提出的點都很精確 只是體驗不是很好  
最後問了我一個致命的問題 說要 shuffle 幾次我們才可以保證所有元素是隨機的 LZ 就隨口答了  $\log n$  次 說我也不知道為什麼 有沒有人可以跟我說答案是什麼？

test case:

1 -> 2 -> 3 -> 4 -> 5 -> 6

可以變成 1 -> 2 -> 3 -> 4 -> 5 -> 6 或 1 -> 2 -> 4 -> 3 -> 5 -> 6 或 4 -> 5 -> 6 -> 1 -> 2 -> 3 但就是不能變成 1 -> 3 -> 2 -> 4 -> 5 -> 6

第二題是 bank hours and transaction hours

給定一堆銀行開門的 interval 數字都在 0-24 之間代表 hour 問給定的 transaction interval 可不可以被執行 可以被執行的定義就是在這 transaction 裡面任意時間都要有銀行開門

解法就是先 sort and merge bank hours interval 然後在這 interval binary search 到小於等於 transaction 開始時間 然後確定 interval 結束時間是否 cover transaction 結束時間

最後小哥問了兩個 follow up

第一個是如果 interval 開始時間大於結束時間怎麼處理 也就是說這間銀行開門時間或是 transaction 時間跨過了半夜

第二個是怎麼設計 class 假設 bank hours 變動很少但 query 次數很多要怎麼讓這個 query 更快 也就是這個查詢不用每次都 merge bank hours interval

test case:

bank hours: [4, 7] [21, 3] [8, 11] [10, 15] transaction: [6, 9] return false, [9, 13] return true, [22, 1] return true, [20, 1] return false

最後等了十幾分鐘被 hr 送下樓 下樓前還帶我去拿了零食水果

隔天中午在機場收了拒信

因為學校地點關係要當天要回到學校最晚的飛機是五點 我就稍微提一下說是不是太趕 後來 BB 就提供 LZ 兩晚飯店



lukuang 发表于 2018-10-29 00:10

我感觉第二题因为是只有24小时，所以是不是用bucket sort更快？ $O(n)$

啊挺有道理的 說不定因為這樣我就掛了哈哈

## 12. 10 月中

第一轮：白人小哥+天竺小哥

聊了下简历，聊了下 MongoDB 的索引，以及简历中用到了 data 相关的技术栈

第一道题，找两个单词的距离，比如 "ab" 和 "abcd" 距离为 2，"bc" 和 "aacbe" 距离为 3（说实话这个也不能叫距离，但是面试官就是这么描述的）

第二道题，拓扑排序，类似 course schedule

第一轮结束马上约了第二轮面试。

第二轮：白人小哥+天竺老哥

天竺老哥问了一道 **anagram**, 写了 **hashmap** 解, 重点是他后面的 **follow up** 很奇怪, 问我还有什么其他方法, 还可以这么优化? 我说也可以 **sort**, 如果说不用 **hash** 的话。但是其实他想表达的是用 **hashmap** 情况下怎么优化, 即当第二遍扫描的时候 **map** 中某个字符的 **value** 变成 **0** 的时候就从 **map** 中消去, 实际上他想问的是如果两个 **string** 很长的情况, 但是因为他口音实在太重, 我当时就没想到他是想在这个方面优化, 所以这道题花了 **20** 分钟, 这时基本知道自己是凉了。

白人小哥上来是一道 **permutation**, 很快写了思路, 画出了图给他解释, 然后第二道题是给一个 **rand5()** 方程去实现 **rand7()**, 这题 **leetcode** 有类似的题目(思麒麟), 但是我没做到过, 所以答的不好, 然后第三题是 **LRU Cache**, 写了接口然后和他讲了一下思路。  
第二轮面完没有约 **HR** 面, 确定已经凉了。