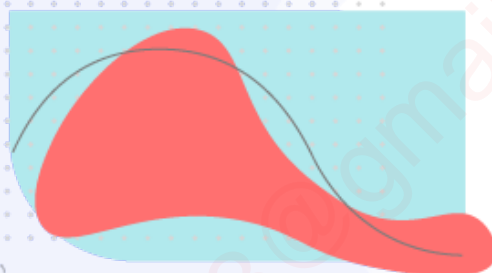
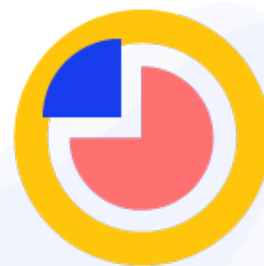




匠人学院  
jiangren.com.au



# FLAG面试机经





匠人学院<sup>TM</sup>  
jiangren.com.au

匠人学院成立于2017年，致力于帮助在澳洲华人，让找工作不再是难事，是澳洲学习和工作伙伴。超过10000的小伙伴加入我们，三年服务超过了5000名学生，帮助华人从学业困难，到就业困难，超过800多位同学拿到了Offer，进入了澳洲主流社会。我们希望成为下一代的肩膀，帮助更多的华人解决问题，也希望有同样目标，志同道合的人加入。

布里斯班

悉尼

阿德莱德

堪培拉

墨尔本

霍巴特

10000<sup>+</sup>

成员 导师

300<sup>+</sup>

80

家本地企业

offer数量

800<sup>+</sup>

100<sup>+</sup>

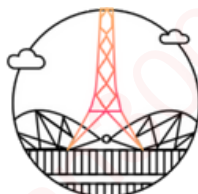
每年 场活动



Level 13b, 116  
Adelaide Street,  
Brisbane CBD



Level 8, 11 York  
St, Wynyard,  
Sydney CBD



Suite 4.03, 838 Collins  
St, Docklands,  
Melbourne



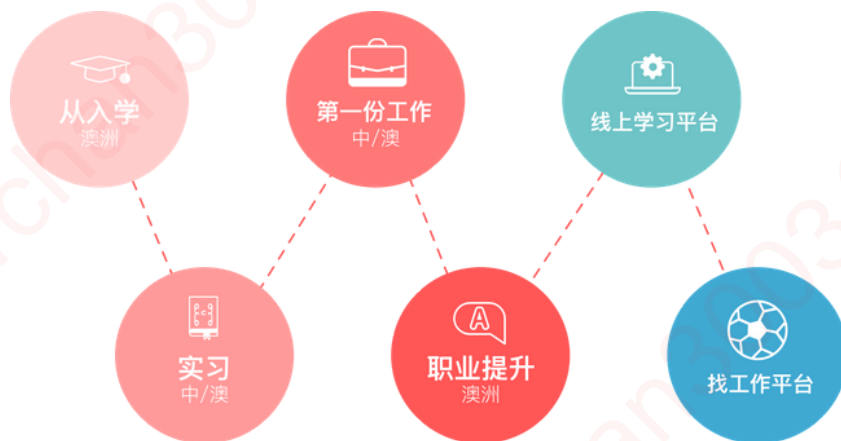
Business Hub, 155  
Waymouth St, Adelaide

学IT找匠人



匠人学院<sup>TM</sup>  
jiangren.com.au

从学业，实习，到就业培训，公司猎头，以及学习平台，更多全方位帮助学生



覆盖软件开发、Web开发、UI/UX、运维、人力资源、移动端开发、数据分析、数据科学、市场营销。



全栈工程师



数据分析



DevOps



数据科学



移动端开发



UI/UX设计师



人力资源



市场营销



学业指导



技能培训



项目实战



求职辅导



职位共享



社群交流



名企内推



精准猎聘

学IT找匠人

## 面试题

### Question: Square

一排房子,每个房子里有一定价值的东西,小偷不能偷相邻的两个房间。即如果小偷光临了房间  $i$ , 那么就不能再偷房间  $i - 1$  和房间  $i + 1$ 。要求返回小偷能偷到东西的总价值的最大值

**Sol:** A modified version of this problem is that all houses form a circle, whose solution is very similar. We need to run DP twice.

1st:  $f[0] = v[0]$ ,  $f[1] = 0$ ,  $f[i] = \max\{f[i - 1], f[i - 2] + v[i]\}$  for  $i = 2$ ,

3, ...,  $n - 2 \implies \text{ans1} = f[n - 2]$

2nd:  $f[0] = 0$ ,  $f[1] = v[1]$ ,  $f[i] = \max\{f[i - 1], f[i - 2] + v[i]\}$  for  $i = 2$ ,

3, ...,  $n - 1 \implies \text{ans2} = f[n - 1]$

return  $\max\{\text{ans1}, \text{ans2}\}$

### Question:

扑克牌问题:给一副扑克牌排序,先是按花色,同一花色按数字排序。主要是扑克牌这个 CLASS 应该如何设计,如何表示花色和面值。

FOLLOW-UP:现在你有一手牌,你要计算其分值,规则如下:如果两张牌相同,或这两张牌的面值和为 15,则计 2 分。ACE 可以是 1 或者 11。

### Question: Google

第一轮 1.1. Tokenize a string to words. Ignore any space and punctuator

1.2. Design an distributed file system to store files of TB size

Follow-up: How to find and store the top-k most frequent keywords among documents stored on all Google servers

第二轮 2.1. Given a list of words, find two strings  $S$  &  $T$  such that:

a.  $S$  &  $T$  have no common character

b.  $S.\text{length}() * T.\text{length}()$  is maximized

Follow up: how to optimize and speed up your algorithm

第三轮 3.1. Design an interface that can convert both a sorted linked list and a sorted array into a balanced binary search tree. Implement it in both bottom-up and top-down approaches

3.2. (Leetcode 原题) Given a matrix of size  $m * n$ ,  $\text{matrix}[i][j]$  stores the number of carrots in cell  $(i, j)$ . Now a rabbit starts from the left upper corner and wants to reach the right below corner. It can only move either

## 面试题

to the right or below. Compute the maximum number of carrots that it can collect along the way, and output that path.

Follow up: how many different ways are there?

第四轮 4. Given a byte array, which is an encoding of characters. Here is the rule:

- a. If the first bit of a byte is 0, that byte stands for a one-byte character
- b. If the first bit of a byte is 1, that byte and its following byte together stand for a two-byte character

Now implement a function to decide if the last character is a one-byte character or a two-byte character

Constraint: You must scan the byte array from the end to the start.

Otherwise it will be very trivial.

**Question:** Microsoft

1.1. What are the two ways to implement hash tables? How to add, delete, and lookup an key? How to deal with collision?

1.2. Given an integer, return the next prime number bigger than it.

Follow-up: If this function will be called frequently, how to optimize the performance?

2.1. What's a full outer join in database? Implement a full outer join given two tables.

Follow-up: If two tables are very big (i.e., no enough RAM to load them), how to deal with it?

2.2. Given `random()` that can return 0 or 1 uniformly, implement `random_new()` that can return 0 with 90%, and 1 with 10%.

3.1. Given an image represented by `byte[][]` image, return its mirror image.

3.2. Design a distributed LRU

4.1. Given an array `[a1, a2, ..., an, b1, b2, ..., bn]`, transform it to `[a1, b1, a2, b2, ..., an, bn]`.

Requirement: time complexity  $O(n \log n)$ , space complexity  $O(\log n)$

**Sol:** the base idea is to use quicksort techniques. Suppose the current array is A, whose size is  $2k$ .

## 面试题

1. Divide A into four segments:  $A = [A1\ A2\ B1\ B2]$ , where  $A1.size = B1.size = k / 2$ ,  $B1.size = B2.size = k - k / 2$ ;
2. Swap A2 and B1, and we get  $A = [A1\ B1\ A2\ B2]$ . In this step, we actually need to rotate  $[A2\ B1]$  to the right by  $k - k / 2$  items. This can be done by reversing  $[A2\ B1]$  first, and then reversing  $[A2]$  and  $[B1]$  respectively.
3. Recursive on  $[A1\ B1]$  and  $[A2\ B2]$  respectively.

Example:  $A = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10]$

$A1 = [1\ 2]$ ,  $A2 = [3\ 4\ 5]$ ,  $B1 = [6\ 7]$ ,  $B2 = [8\ 9\ 10]$

After 2nd step,  $A = [1\ 2\ |\ 6\ 7\ |\ 3\ 4\ 5\ |\ 8\ 9\ 10]$

For the 3rd step, process  $[1\ 2\ 6\ 7]$  and  $[3\ 4\ 5\ 8\ 9\ 10]$  respectively

- 4.2. Design: suppose you have a cluster, and each machine in this cluster has a large number of numbers. How can you find out the median of all the numbers on all the machines.

### Question: Amazon

1. Given a string, find the longest palindromic substring
2. Given a binary tree, find the length of the longest path in the tree. A path can start and end anywhere in the tree (i.e., not necessary from the root to a leaf).
3. Given a large number of integers, return the largest K numbers. How to process them using MapReduce?
4. Implement a priority queue: enqueue, getFront, dequeue
5. Given a set of points on a plane, and a list of circles centered at the original point, find the ring containing the most number of points.
6. Design: You have a HTML page, which contains many strings describing potions in a CSS file, how can to compress these strings to reduce the size of the HTML page.

Follow-up: Users complain that your website becomes slow recently, how can you find out the problems, and how to fix them?

7. Java OO concepts, dissertation and behavior questions from CC150.



## 面试题

**Google,**

**Phone Interview 09/19**

根据空格分隔字符串,但是引号内的是整体,不可分割

如果这个字符串是一个连续分布在很多机器上的大文件,每个机器知道其前后机器是谁并

且可以相互通信,那么如何继续分隔(引号可以分在两个机器上

**Baidu USDC,**

**Phone Interview 09/22(first)**

Reverse Words in a String, 要求 in place

**Phone Interview 10/07(second)**

Print diagonally matrix,. 1point3acres.com/bbs

**Airbnb,**

**Phone Interview 09/23**

给一个整数数组,求不相互挨着的数字可以相加得到的最大和

**Snapchat,**

**Phone Interview 09/24**

输出 Excel 所有的列名

**Sumo Logic**

**Phone Interview 09/29 (first), 10/03 (second)**

Anagrams

Decode Ways

**LinkedIn**

**Phone Interview 10/01**

设计一个类实现下面的接口

```
interface List{
```

```
public void add(T o);//add to the last. 1point3acres.com/bbs
```

```
public T get(int index); //get the index object
```

```
public int size();//return the size
```

```
public boolean remove(T o);//remove the first o and return true; if not  
exist, return false.
```

```
}
```

## 面试题

### Twitter.

#### Phone Interview 10/03

C++内存问题

```
void fun() {  
    Someobject *object = new Someobject();  
    int a = 3;  
}
```

两种变量如何分配内存空间,对于 object 要说出指针和 object 分别存储在栈和堆上。对于 Object 来说,如果没有连续内存空间分配,怎么办,举例说明

### Facebook

#### Phone Interview 10/06.

二叉树转双向循环链表

### Dropbox

#### Phone Interview 10/07

boolean wordPattern(String str, char[] pattern)-google

1point3acres

```
/* [a, b, b, a]
```

“dog cat cat dog” match

“dog cat cat fish” doesn’ t match

```
[a, a, a, a]
```

“dog cat cat dog” doesn’ t match

```
[a, b, b, a]
```

“dog dog dog dog” doesn’ t match

\*/. more info on 1point3acres.com

boolean harderWordPattern(String str, char[] pattern)

// True if there exists a splitting of the string and pattern s.t.

there’ s a match, else False

```
/* [a, b, b, a]
```

“dogcatcatdog” match

“dogcatcatfish” doesn’ t match

```
[a, a, a, a]
```



## 面试题

“dogcatcatdog” doesn’t match

[a, b, b, a]

“dogdogdogdog” doesn’t match

[a, b, b, a]. Waral 錦氫 鏈爰洿澶氫构筠◆,

“ABCBCA” match

. from: 1point3acres.com/bbs

[a,b]

“abab” match

### Two Sigma

#### Phone Interview 10/13

describe your most challenging project

difference between process and thread, methods for IPC

throughput & delay

hashtable implementation

判断一个数字的二进制表示中,1 的个数是不是质数

#### 面经:

#### Bloomberg

电面:mapreduce 一道题(和 search engine 有关,build

index),hashtable 实现相关问题,智力题(不难),开放题(match

persons and teams, each person has a rank for each team, each

team also has a rank for each person, design the matching

algorithm)

#### Rocket Fuel

#### code challenge: auto racer

电面 1:

第一题:贪心

Given a number, can you remove k digits from the number so that the new formatted number is smallest possible.

input: n = 1432219, k = 3

output: 1219

第二题:DP

BT(binary tree), want to find the LIS(largest independent set) of the BT

## 面试题

LIS: if the current node is in the set, then its children should not be in .

Waral 錦氫 鏈爰洿澶氫

构筠◆,

the set. So that the set has the largest number of nodes.

电面 2:

第一题:Median of Two Sorted Arrays

第二题:DP

一个二维数组,元素是 0 或 1,找出最大的由 1 构成的"X" 形状

**onsite:**

1. print all subsets

system design(N topics, publishers, subscribers, scalability, distributed).

the most frequent urls in the past minute, hour, day

2. manager interview

code review

3. shortest path between two nodes of a tree(no parent pointer)

4. machine learning

5. machine learning

**Google:**

**电面:**

remove duplicate lines of a file(what if the file is very large which could not be held in the main memory)

开关灯问题

Trapping Rain Water(leetcode)

sometimes a program works, sometimes it does not. Possible reasons

onsite:-google 1point3acres

1. clone directed graph(recursive, non-recursive)

longest common suffix of two linked list. from: 1point3acres.com/bbs

data structure design

2. how many (m, n) pairs such that  $m * m + n * n$

## 面试题

### F Onsite 面经

三轮

1. a) 给出加密的方法 'a' -> 1, ..., 'z' -> 26. 给一个数, 问有多少种解密的方法。  
b) 给你 n 个用户和 k, 找出发帖数最多的 k 个用户。
2. a) 给你棵二叉树, 节点上有权值, 问从一个叶子走到另外一个叶子的路里面权值最大的那条是什么。  
b) 给你数组  $a_1, a_2, \dots, a_n$ 。输出数组  $a_2 * a_3 * \dots * a_n$ ,  $a_1 * a_3 * a_4 * \dots * a_n$ , ...,  $a_1 * a_2 * \dots * a_{n-1}$ 。
3. 问简历, 问来想做什么工作。一道 coding 题: Read4k, leetcode 上那道 "Read N Characters Given Read4" 类似。

### fb intern 面经

Print arbitrary binary tree by vertical levels / columns from left to right

example tree

a

/ \

b c

/ \ \

d g z

\ /

e i

/

q

/

x

/

x1

/

x2

sample output.

x2

d x1

b e x

a g q

c i

## 面试题

### Facebook summer intern phone interview

// in: [ "rat" , "art" , "tar" , "bacefook" , "facebook" ,  
"w" ]

// out: [[ "rat" , "tar" , "art" ], [ "bacefook" , "facebook" ],  
[ "w" ]]

荷兰旗的变体:

// In: [2,9,4] -> [2,4,9]

// ^l^m^h

// Out: [2,4,9,5,1] -> [1,2,4,5,9] or [2,1,5,4,9], ...

/\*

sort(begin, end, cmp)

a low b

\*/

### linkedin,dropbox,facebook 面经

1 word ladder

2 design shorten url, 主要考察对 scale 的了解,怎么存,query 多时的处理,某一个 url 的

query 很多怎么处理。如何查询最近 1 个月来 query url 的 top 10, 如果要对 url 设过期时间

怎么处理,大概这些。其实题目只是个幌子,目的是考察你多方面的知识, hash, db,

concurrency.

3, print binary tree by level, BFS 和 DFS 方法都得写

4, record hits in last five minutes

5, combination sum. 时空复杂度

6 isomorphic

7 pow

8 the shortest distance between two words in a string

9 BST iterator

10 max points on the line

## 面试题

### Facebook onsite 1/26 面经

- 1, Jedi, behavioral + projects + one coding (phone number permutation)
- 2, Ninja, a) k closest points to a given point(quick select).  
b) dot product of sparse matrices.
- 3, pirate, Design Messenger.
- 4, Ninja, a) longest increasing contiguous sequence in an array (leetcode)  
b) longest increasing sequence in an array(sequence does not need to be contiguous). For instance, 1,2,4,3,2,7,8,9 .. return 1,2,3,7,

### FB 电面面经

- 1) 给个数组 seq, 和一个 total, 找 if there is a contiguous sequence in seq which sums to total.
- 2) palindrome String
- 3) decode ways.

### 电面题目是 task scheduler, 举例如下:

Tasks: AABABCD

Cooldown Time: 2

A\_AB\_ABCD

Output: 10

就是说同样类型的 task 之间至少要等 2, 每个 task 的执行时间是 1

followup: 如果 cooldown 是个参数, 也就是说有可能会很长时间, 怎么修改之前的程序

### Facebook 电面

We have a coding system from alphabets to numbers where A=1, B=2, ... Z=26. You are given a string of digits as an input. Write a function that will compute the number of valid interpretations of the input.

## 面试题

A=1

B=2

C=3

...

L=12

...

W=23

...

Z=26

—

f(“123”) = 3 // { “ABC” , “LC” , “AW” }

### fb 店面

Move all none zeros to left. [1, 0, 2, 1, 0, 0] -> [1, 2, 1, ?, ?, ?]

Print Binary tree all path.

### facebook intern

```
class IntFileIterator {
```

```
    boolean hasNext();
```

```
    int next();
```

```
}
```

```
class FileCompare {
```

```
    public boolean isDistanceZeroOrOne(IntFileIterator a, IntFileIterator b);
```

```
}
```

// return if the distance between a and b is at most 1..

1point3acres.com/bbs

// Distance: minimum number of modifications to make a=b

// Modification:-google 1point3acres

// 1. change an int in a

// 2. insert an int to a

// 3. remove an int from a

## 面试题

### onsite:

1.1 有很多 meeting with time intervals [start, end], 求出一个有最多 meeting 的时间点

1.2

有这么一个 class Contact, 里面有一个 string 的 name, 和一个 vector 装着 email address, 是这个 Contact 有的 address, 用一个 list 装着是因为一个人有可能有多个 email, 现在给你 vector, 比如

```
{ { "JohnS", { "john@gmail.com" } }, { "Mary",  
  { "mary@gmail.com" } }, { "John",  
  { "john@yahoo.com" } }, { "John", { "john@gmail.com",  
    "john@yahoo.com",  
    "john@hotmail.com" } }, { "Bob", { "bob@gmail.com" } }
```

2 System design: shorten URL

3.1 isPalindrome(string str) 3.2 dot product of sparse vectors 3.3 find the first bug version

4 behavior + bianry plus one

5 System desgin: point of interests – given a point, find intersetting points within 5 miles

### 一面:

给定任务 AABCB, 冷却时间 k(相同任务之间的最短距离时间), 任务顺序不能变, 问完成任务的总时间。

例子: AABCB, k=2, A\*\*ABC\*B, 时间为 8.

解法: 用 hashtable 保存上次的时间。

Followup1: 如果 k 很小, 怎么优化?

解法: 之前的 hashtable 的大小是 unique task 的大小, 如果 k 很小, 可以只维护 k 那么大的 hashtable。

Followup2: 如果可以改变任务的顺序, 最短的任务时间是多少?

例子: AABBC, K=2, AB\*ABC, 时间为 6.

解法: 根据每个任务出现的频率排序, 优先处理频率高的。但是具体细节没有时间讨论。



## 面试题

### First round:

Flatten list of lists

A - B - C - D - E

| | | |

M W P O

| |

N Q

Return:

A - M - N - B - W - Q - C - D - P - E - O

Node {

String v;

Node\* right;

Node\* down;

}Second round

1, Given O 20 (bad)

|

...

|

O

|

|

O 15 (good)

int isBad(int rev);

Find first bad version in git versioncontrol

2, Tree to doubly linked list

4

/ \

/ \

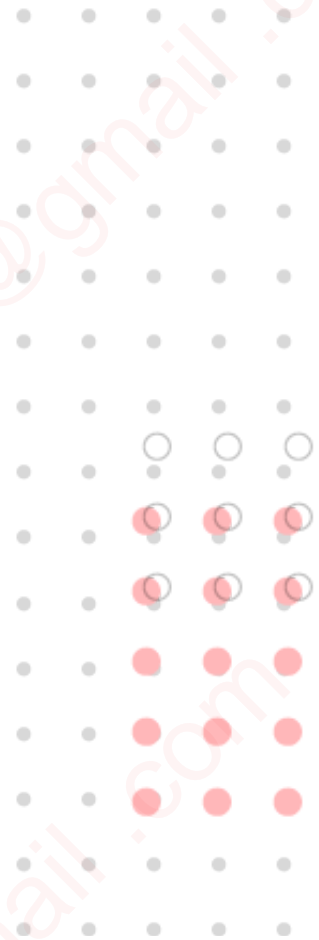
10 7

/ \

. 1point 3acres 璁哄 / \

3 2 1

3 <-> 10 <-> 2 <-> 4 <-> 7 <-> 1



## 面试题

**Question:** given a series of meetings, each with a start time, end time and priority, schedule these meetings into a limited number of rooms, dropping the meetings that are lowest priority

for example:

INPUT:

3(# of rooms)

5(# of meetings)

1, 1300, 1400, 100(priority) "meeting 1"

2, 1345, 1445, 100, "meeting 2"

3, 1330, 1350, 100, "meeting 3"

4, 1500, 1700, 75, "meeting 4"

5, 1300, 1400, 90, "meeting 5"

OUTPUT:

1 3 2 4 (会议的序列)

**Onsite:**

1. ML design 设计一个根据用户信息的搜索,然后能辨识出来他的语言。

2. LC 原题, 3Sum + Trap water

3. System design: instagram

4. Coding: Sliding window maximum + LC 原题, sliding window 包含所有字符那个

5. Culture fit: 有 200M 个用户,现在让你进行分组,将他们分成大概 20 个组,每个

组里大概有 10M 的用户,尽量让用户 interaction 多的在一起

**Question:**

remove minimum number of left and right brackets and return the string with

valid bracket pairs. given (a))(), should return (a)(), or (a()); given ())(), should return either ()() or (); given ()(9()), should return either ()(9()) or ()((9)).

## 面试题

### Question:

convert integer to english words

e.g.,

123 -> "One hundred and twenty three"

1234567 -> "One million two hundred and thirty four thousand five hundred and sixty seven"

12345 -> "twelve thousand three hundred and forty five"

### Onsite:

五轮 Onsite 没有 coding,全是问实际问题怎么解决和 design。

1. 如何设计一个 priorityqueue service,client 可以 submit job request 然后 server 按照 priority 执行

2. 需要一个 key-value store with 1M qps,most read,1ms 99% latency,如果用 HBase 的话会有什么问题,怎么解决

3. 给很多整数,如何用 mapreduce 找 median,如果是很多 float 数,可以有一定的误差,如何找

4. Programming Test 的扩展,如果 soduku matrix 非常之大怎么做

### 电面:

1. 给平面一堆点,把所有在同一条直线上的点 group 在一起,求出所有的 group

2. 一种 encoding 的方法,如果一个 byte 第一个 bit 是 0,比如 00000000,那它自己表示一个字符,如果一个 byte 第一个 bit 是 1,比如 10000000,那它和它后面紧跟的 byte 表示一个字符,现在给一个 byte array,判断最后一个字符是一个 byte 还是两个 byte 组成。

3. parse message from byte stream,message format 是前 4 个 bytes 组成的 int 值表示 message 的长度 L,然后后面连续的 L 个 byte 是 message 真正的内容,每个 message 都是这样表示,需要一边读 byte stream 一边 parse 每个 message

4. 两个 table 做 join 有哪几种方法,分别有哪些 drawback

5. merge two sorted list

6. sqrt(double number, double epsilon)

7. auto completion implementation using trie

8. edit distance

9. Implement blockingqueue

10. how is a hive query transferred to mapreduce jobs

## 面试题

### Onsite:

1. given a list of pairs, pair.first 表示 parent, pair.second 表示 child, reconstruct the tree, return the root node.
2. auto completion – design the service
3. design a service, accept stream of events, each event has a type and timestamp, need to support the query of top k most frequent types in a query specified [start, end] time range.
4. closest number to target in BST
5. validate sudoku / solve sudoku, and optimizations
6. 给一个 json object, 给一个 wildcard path with ‘?’ as arbitrary name, 比如 a.?.b 找到所有符合 path 的 objects

### Onsite:

1. OOD Restaurant Reservation System
2. Merge K Sorted List
3. K Sized Sliding Window Sum/Minimum Value
4. 给一个 css file 里面很多 class, 然后 class name 里面其实很多重复的, 怎么 compress 用尽量最小 size 的 string 来表示, 这样传输的 byte 比较少。
5. shorten url system design
6. longest palindromic substring
7. robot moving from topleft to bottomright corner of a matrix, matrix 里面有些 cell 是障碍物不能通过, 只能往下或者往右走, 有多少种方法。

### Onsite:

1. topological sort
2. design web crawler system, how to scale, what would be the bottle neck and how to solve the problem
3. 如何用 semaphore 或者 condition variable 实现 3 个 process p1, p2, p3, p2 必须要 p1 结束才能运行, p3 必须要 p2 结束才能运行
4. bloom filter 如何 implement, estimate false rate
5. what is the best design pattern do you think and why

## 面试题

### Onsite:

1. 有意思的题目 1,设计 Bi-directional LRU cache data structure,既可以 lookup key to get value,也可以 lookup value to get key,还支持 set(key, value)操作,后面又加了条件,concurrent 的情况下,会有什么问题,如何改进,假如 set 这个操作的频率远远小于 get 这个操作的频率,需要写代码实现。
2. robot from topleft to bottomright LC 原题,无障碍和有障碍
3. given a list of sets,find all pair of sets having any intersection
4. 有意思的题目 2,设计 caltrain system,要实现 caltrain 上车下车刷卡扣钱整个功能,assume 每个 station 都跟一个 central server 相连,要处理如果有 network partition 怎么办,eventually 车费还是要 charge 到账户上,但是不能影响 partition 的 station 正常运作。要处理某些人下车没刷卡怎么办,followup 可以非常多
5. 有意思的题目 3,仍然是设计一个 concurrent 环境下的 time leased cache,但是有些区别,假如 delete 操作是一个 daemon thread 来做不用太多考虑,但是 get(key)操作的逻辑是如果 key 不在 cache 里面,需要一个非常 expensive 的操作把对应 value load 进来,如何让这个 load 的操作对同一个 key 尽量少发生,需要写代码实现。

### Onsite:

1. OOD astroid game,就是飞机打石块的游戏,石块可以任意形状可以移动,飞机撞上就挂了,飞机可以发射子弹,子弹打上石块会把石块分成多个小石块按照不同方向和速度移动。要写伪代码。
2. 每个 person 有一个 list of intervals,表示 busy 的时间段,问最 busy 的一段时间分别都是谁 busy。
3. 一个描述起来不算简单的题目,但是算法不难,在版上看到过但是细节记不清了,好像是给一堆 stock profile 然后算 profit
4. 一个 2d matrix,被分成好几个区域,区域之间都是 value 为 0 的 cell,每一堆 connected 的非 0 cell 算是一个区域,问和最大的区域是哪个,要设计 API,怎么用 json return 结果。
5. system design 又是 distributed key-value store,万年不变的题

## 面试题

### Onsite:

1. log\_hit(), get\_last\_5mins\_hits()那个题目,concurrent 怎么搞
2. token bucket,假设每 x 秒 提供一个 token,然后外面可以申请任意数量的 token,如果 token 不够就 block,要求 concurrent 情况下,不能有专门的 thread 产生 token,怎样用最简单的方法实现
3. web crawler,要分析可能的 bottleneck,然后转化成 concurrent 运行的版本,写 runnable 代码。
4. system design

### Onsite:

1. 一个正整数可以表示成其他几个正整数的平方和,给任何一个正整数,求最少的那几个正整数,平方和是给定的数,比如  $14 = 1^2 + 2^2 + 3^2$ ,如果给的数是 14,应该返回(1,2,3)
2. 给一个dictionary,然后可以 support 的 query 是,给一个 string,返回在 dictionary 里面包含给定string的所有 character 的最短的 string
3. 如何设计 google login system
4. web crawl 的时候如何判断两个 document 是相同/相似的。

1. Write a function that takes as input integers P and Q and returns P to the power of Q. Note any assumptions you make and the complexity of the algorithm. We expect you to do better than  $O(Q)$ .
2. Write a function that takes as input an array of 1 million integers, such that  $1 \leq x \leq 10$  for every element x in the array, and returns the sorted array. The sort does not need to occur in-place. Obviously you can just call a standard sorting function like quicksort, but can you do better?
3. You are given an alphanumeric string. Write an algorithm that will segment the string into substrings of consecutive integers or numbers and then sort the substrings. For example, the string "AZQF013452BAB" will result in "AFQZ012345ABB" .
4. Write a function to determine the largest palindromic subsequence of a string. A palindromic string is a string which is the same when read in either the forward or reverse direction. For example, "ABBA" is a palindromic string and the largest palindromic substring of "TABBA" is "ABBA" .



## 面试题

### Onsite:

1. graph deepcopy
2. use normal lock to implement readwrite lock
3. design question, how to scale web application
4. given a list of iterators which iterates over sorted lists, write a MergeIterator class which iterates over the merged list, e.g.

```
class MergeIterator<T>
{
    MergeIterator(List<Iterator<T>> iterators)
    {
    }
    boolean hasNext()
    T next();
}
```

### hedge fund 2:

1. friend circles – give a matrix, Y in cell means i and j is friend, N otherwise, find how many friend circles in the matrix, e.g. 1 is friend of 2, and 2 is friend of 3, then 1,2,3 is in same friend circle.
2. StringChain, give a dictionary, the string chain is by remove a char in the string, and if the new string is in the dictionary, then continue, e.g. dict = { a, b, ab, abc, add} then the longest chain is (a, ab, abc) or (b, ab, abc). The char can be removed from any place in the string.

### Onsite:

1. multiply 2 numbers, the digits of the numbers are given as int array, e.g. int[] product(int[] num1, int[] num2);
2. given a list of intervals, each interval is defined as 2 integer (start/end), find min set of points, for those points, each interval at least cover 1 point. e.g. given intervals as [1, 4], [2, 3], [5, 6], we just need 2 points, (2, 5), and each interval will either cover point 2, or point 5. need O(nlogn) solution.



## 面试题

3. given binary search tree, each tree node contains piont of (left, right, parent, leftChildTreeSize), write a function to find the number of nodes which has value less than the given node, e.g. int  
findNumberofLess(Node current, Node root);
4. process 2 stream of data and output result, basic merge sort implementation.

### tech company 2:

1. have N offices globally. each office have a local calendar with holidays. you are allowed to move every weekend to different office, how to get max numbers of holidays. follow up, if for each office, there are only certain set of offices are reachable, e.g. if you are in NYC this weekend, you can move to SF, or London. If you are in SF, you can move to NYC and Beijing, etc. how to max the holidays.
2. Binary tree find the longest consecutive path.
3. how to check 2 rectangles overlap. Give a very large set of segments (each segment is defined by start point and end point), given a function which given 2 segments, returns the intersection of the 2 segment if they intersect, or null if not. How to find all the intersections, cannot do the double loop in memory since the dataset is too big to fit in memory.
4. give a string array, find the 2 string which don't share any char, and have the max product of the lengths. e.g. given string abc, aagh, def, the max product is  $\text{len}(\text{abc}) * \text{len}(\text{def}) = 3 * 3 = 9$
5. design question, how to generate unique sequence number using distributed system. e.g. you have a set of machines which is running this sequence number generator, client can connect to any machine, and get the next sequence number which is guranteed to increment for same client.

## 面试题

### tech company 3:

online coding:

1. find kth minimal number in tournament tree. sample of tournament tree (2

beat 4, 3 beat 5, 2 beat 3 and become champion)

2

2 3

4 2, 3, 5

2. word distance, e.g. given an array of words, and give 2 words, find the min distance of index those 2 words

### Onsite:

1. deepIterator, e.g. given list {1, 2, {{3, 5}, 4}, 6}, write an iterator class which will iterate through the deep list.

2. check whether 2 tree is identical, can you do it iteratively?

3. roman string to int, and int to roman string

4. adding a list of intervals, each interval is defined by start point and end point, find the total coverage of the intervals, e.g. intervals: { 1, 4}, {2, 5}, {7, 10}, total coverage is 1 to 5 and 7 to 10, which is 7.

5. design question, design a system which can rank the url sharings, e.g. users will share urls, we want to rank the most shared urls for the last 10 minutes, for last hour, for last day, etc. there are total 100 millions url sharing happen every day..

### onsite

F:1.find bad version, 比如 isgood(version 1) = true, isgood(version 30) = false, 找出第一个出错的 version

2.BST inorder tranverse

3. 把 string 转化成 floating number(stof)

behavior question 的最后烙印来了一道按列打印 tree, follow up 是不用 hashmap 存 node 的水平距离, 用 vector 存, 如何做, onepass, 不准先求树的 width

4. system design: 每个 record 有个很大 field, 比如年龄, 性别, 爱好等。给一个 field 的组合, 比如小于 25 岁, 爱好体育, query 满足这些组合条件的用户个数

## 面试题

L:

1. max point on line/ (如何不是整数坐标如何处理, 需要改写 hashmap 的 compare)
2. special container add/remove/removeRandom at O(1): array + hashmap
3. k-way sort given a stream iterator, vector,
4. product of other elements; 考虑 1 个 0 和 2 个 0 的情况
5. 实现 movemem( void\* src, void\* dest)
6. system design: tiny url
7. host manager 那轮最后问了一个, 如何在不影响功能的情况下, 把一个 data center 的数据复制到另外一个新的 data center 去。

G:

1. find all rotation symmetric numbers less than N digits, 16891 -> 16891,
2. give integer, 12345, 返回 32154  
give a target string and list of strings, find the longest string that has target as prefix, follow up, stream of target string, 用 trie, 每个节点保留最长 string 信息。
3. integer array add one rotation abc->bcd->cde, give a list of strings, group them if they are rotations.
4. given grid of colors, coordinate of a point and its color, find the perimeter of the region that has the same color of that point.  
print all morse code given the length constraints, short “\*” takes one, long “—” takes two. (find a bug in the code) 就是排列组合的典型题
5. design: chromecast, how to know which app can be supported?  
There is a cloud that can give the information to the chrome cast, appID, deviceID, cache design.

1. LRU cache
2. 一个整数数组, 先递增然后递减, 也有可能只有递增或者递减。查找某个整数在不在数组里。
3. 设计 Boggle 游戏
4. OO design, 用树形结构表示表达式。注意 operator 要用多态实现。
5. 2 sum, 一个元素只能用一次

## 面试题

6. 1)判断一个数组中是否有 3 个元素和为 0,元素可以重用。 2)merge k sorted array. 3)稀疏向量的点乘。
7. 一个数组,把非 0 的元素移动到开头。
8. 1) maximum subarray 2)树里两个节点的最低公共祖先 3)LC subset
9. 设计 fb newsfeed
10. 大数相乘
11. 1) 给一段话,再给两个单词,求这两个单词在这段话里的最小距离 2)打印二叉树(level order 遍历)
12. 随机洗牌算法
13. 1)给一个字符串,返回每个字符及其个数。比如:aaabcc-> 3a1b2c 2)给字符及其个数,返回原本的字符串 3)median of an array,有哪些方法,如果数据太多内存装不下怎么办
14. 用一个  $m*n$  的矩阵表示一副图片,其中每个元素对应 pixel 的灰度。Smooth the image with  $m1 * n1$  scale。也就是每  $m1 * n1$  的矩阵里面的值求平均,放到中间的那个像素里。如何节省内存?
15. 一个 server,多个 client。client 给 server 发任务,每个任务带有过期时间。server 要按顺序处理这些任务,直到该任务做完或者被取消或者过期。实现提交,取消和查看任务状态的 API。
16. 二位数组的 Zig zag traversal
17. 一个数组长度未知。如果访问超过长度的 index 会产生 out-of-bound 异常。查找某个元素,如果不在数组内则返回-1
18. 给出左下角和右上角坐标,画出矩阵
19. 如何检测数据库的死锁
20. k-means 算法实现
21. 一个数组有  $n+1$  个元素,每个元素都在 1 到  $n$  之间,只有一个元素出现了两遍,找到这个元素

### 电面:

- 1) search in rotated sorted array, with or without duplicates  
Binary search, 问复杂度解释的时候要逻辑清楚,写码的时候要清晰简洁。

## 面试题

2) find same color area in 2D array

GGGG

GGBC

BBBD

ECED

return 6

DFS

### onsite:

第一轮:给定 N 个 2D 坐标(可以设想为餐厅的位置),要求输入任意坐标,可以返回方圆 d 距离内的所有餐厅

第二轮:讨论 research

第三轮:输入一堆 job,比如 ABAC,相同的 job 有 d 的 cool down time,就是执行完 A,必须等 d unit time 才能执行下一个 A。假如 d=3,对于输入 ABAC,一个合法的 schedule 就是 AB\_ \_AC,需要 6 unit time。编程求对于一个输入,需要最少多长的时间执行。Follow up: 可以任意更改输入 job 的顺序,求最短执行时间。

第四轮:

每个人对应几个 email,把所有相同的人 group 起来输出

例如:

“John” ==> “1@gmail.com, 2@gmail.com, 3@gmail.com”

“Mary” ==> “4@gmail.com, 5@gmail.com, 2@gmail.com”

“Tom” ==> “6@gmail.com”

“Jerry” ==> “5@gmail.com”

就要输出 vector<vector> = {{ “John” , “Mary” , “Jerry” },  
{ “Tom” }}

### Pinterest Phone:

第一轮:1: count the number of words in a file

2: 给定一堆人名,{ “a” , “b” , “c” , “d” , “e” }, 输出 “a, b, c, d and e” . Follow up: 只输出前

K 个。比如 K=2, 输出 “a, b and 3 others”

## 面试题

第二轮:

输入一个 matrix,每个 cell 的值是当前 cell 的硬币数目,负值代表当前 cell 是 blockage。

问从左下角走到右上角最多可以 collect 多少硬币,并且输出 collect 最多硬币的路径(这

个比较麻烦,不用输出路径的简单 dp 就搞定了)

比如:[[2,4,6], [1,-1,100]], 最大值是 13 (1+2+4+6)

### Question:

Implement a method which takes an integer array and returns an integer array (of equal size) in which each element is the product of every number in the input array with the exception of the number at that index.

\* Example:

\* [3, 1, 4, 2] => [8, 24, 6, 12]

\*/

```
public int[] selfExcludingProduct(int[] input) {
```

```
// implementation...
```

```
}
```

```
/*
```

```
* Returns true if the input string is a number and false otherwise
```

```
*/
```

```
public boolean isNumber(String toTest)
```

```
{
```

```
// implementation here
```

```
}
```

## 面试题

1. public interface PointsOnAPlane { /\*\* \* Stores a given point in an internal data structure \*/ void addPoint(Point point); /\*\* \* For given 'center' point returns a subset of 'm' stored points that are \* closer to the center than others. \* \* E.g. Stored: (0, 1) (0, 2) (0, 3) (0, 4) (0, 5) \* \*

findNearest(new Point(0, 0), 3) -> (0, 1), (0, 2), (0, 3) \*/ Collection

findNearest(Point center, int m);}

2. public interface InfluencerFinder { /\*\*

\* Given a matrix of following between N LinkedIn users (with ids from 0 to N-1):- @, B5 C&

~\$ p, { . i; p

\* followingMatrix[i][j] == true iff user i is following user j

\* thus followingMatrix[i][j] doesn't imply followingMatrix[j][i].

\* Let's also agree that followingMatrix[i][i] == false

\*

\* Influencer is a user who is:

\* – followed by everyone else and

\* – not following anyone himself

\*

\* This method should find an Influencer by a given matrix of following,

\* or return -1 if there is no Influencer in this group.

\*/

int getInfluencer(boolean[][] followingMatrix)

### Question:

1. 2D matrix, sorted on each row, first element of next row is larger(or equal) to the last element of previous row, now giving a target number, returning the position that the target locates within the matrix



## 面试题

2. Given a binary tree where all the right nodes are leaf nodes, flip it upside down

\* and turn it into a tree with left leaf nodes.

\*

\* for example, turn these:

\*

\* 1 1

\* //

\* 2 3 2 3

\* /

\* 4 5

\* /

\* 6 7

\*

\* into these:

\*

\* 1 1

\* //

\* 2—3 2—3

\* /

\* 4—5

\* /

\* 6—7

\*

\* where 6 is the new root node for the left tree, and 2 for the right tree.

\* oriented correctly:

\*

\* 6 2

\* //

\* 7 4 3 1

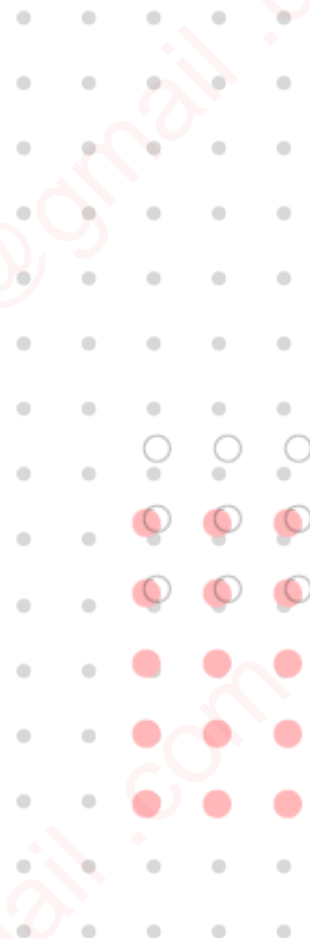
\* /

\* 5 2

\* /

\* 3 1

\* /



## 面试题

### Question:

Given an array of positive integers and two players. In each turn, one player picks up one number and if the sum of all the picked up numbers is greater than a target number, the player wins. Write a program canIWin() to print the result.

### Answer:

```
enum Result {Win, Lose, Draw}
public class PickUpNumbers {
    public static Result canIWin(int[] numberPool, int target) {
        if (target <= 0) return Result.Lose; boolean isEmpty = true; for (int data :
        numberPool) if (data >
        0) isEmpty = false;
        if (isEmpty) return Result.Draw;
        else {
            for (int data : numberPool)
                if (data >= target) return Result.Win;
            Result drawFlag = Result.Draw, rivalWinFlag = Result.Win;
            for (int i = 0; i < numberPool.length; ++i) { if (numberPool < 0) continue;
            int data = numberPool[i]; numberPool[i] = -1; Result rivalResult =
            canIWin(numberPool, target - data); //
            rival's turn if (rivalResult != Result.Win) rivalWinFlag = rivalResult; if
            (rivalResult !=
            Result.Draw) drawFlag = rivalResult; numberPool[i] = data; } if (drawFlag ==
            Result.Draw) return
            Result.Draw; if (rivalWinFlag == Result.Win) return Result.Lose; //
            whatever number i choose,
            rival wins return Result.Win; } } } Test Code: public static void main(String[]
            args) { int[]
            numberPool1 = {1, 2, 3};
```

## 面试题

```
System.out.println(PickUpNumbers.canIWin(numberPool1, 5));
System.out.println(PickUpNumbers.canIWin(numberPool1, 4));
System.out.println(PickUpNumbers.canIWin(numberPool1, 8)); int[]
numberPool2 = {1, 2, 3, 4,5};
System.out.println(PickUpNumbers.canIWin(numberPool2, 11)); int[]
numberPool3 = {1, 2,3, 4, 5, 6};
System.out.println(PickUpNumbers.canIWin(numberPool3, 17)); }
Result: Win Lose
Draw Lose Win
```

### Question:

```
/* This class will be given a list of words (such as might be tokenized
 * from a paragraph of text), and will provide a method that takes
two
 * words and returns the shortest distance (in words) between those
two
 * words in the provided text.
 * Example:
 * WordDistanceFinder finder = new
WordDistanceFinder(Arrays.asList( "the" , "quick" ,
    "brown" , "fox" , "quick" ));
 * assert(finder.distance( "fox" , " the" ) == 3);
 * assert(finder.distance( "quick" , "fox" ) == 1);
 */
```

### Question:

Implement a (Java) Iterable object that iterates lines one by one from a text file.

**\*\* A reference to a file. \*/**

```
public class TextFile implements Iterable
{
```

```
    public TextFile(String fileName) { // please implement this
```

```
    /** Begin reading the file, line by line. The returned Iterator.next() will
    return a line. */
```

## 面试题

@Override

public Iterator iterator() { // please implement this

Print binary tree by level

/\*\*

\* Sample input:

\*

\* 1

\* /\

\* 3 5

\* /\

\* 2 4 7

\* /\

\* 9 8

\*

\* Expected output:

\* 1

\* 3 5

\* 2 4 7

\* 9 8

\* =====

\*/

### Question:

\* Given a nested list of integers, returns the sum of all integers in the list weighted by their depth

\* For example, given the list  $\{\{1,1\},2,\{1,1\}\}$  the function should return 10 (four 1's at depth 2, one 2 at depth 1)

\* Given the list  $\{1,\{4,\{6\}\}\}$  the function should return 27 (one 1 at depth 1, one 4 at depth 2, one 6 at depth 2)

\*/

/\*\*

## 面试题

- \* This is the interface that represents nested lists.
- \* You should not implement it, or speculate about its implementation.
- \*/

```
public interface NestedInteger
{
    // Returns true if this NestedInteger holds a single integer, rather than a
    // nested list
    public boolean isInteger();
    // Returns the single integer that this NestedInteger holds, if it holds a
    // single integer
    // Returns null if this NestedInteger holds a nested list
    public Integer getInteger();
}
```

解答:

```
public int getSum(NestedInteger ni)
{
    if (ni.isInteger()) return ni.getInteger();
    else return getListSum(ni.getList(),1);
}
private int getListSum(List lni, int depth)
{
    int sum = 0;
    NestedInteger ni = null;
    while (lni.hasNext()) {
        ni = lni.next();
        if (ni.isInteger()) sum += ni.getInteger()*depth;
        else sum += getListSum(ni.getList, depth+1)
    }
    return sum;
}
```

## 面试题

### 电面:

1. 给一个二叉树,返回它的镜像实现一个 thread-safe blocking queue
2. 一个嵌套 Map, 就是一个 HashMap, 它的 value 可以是一个 element 也可以是另外一个嵌套 map 或是空的 map. 实现一个 iterator 来遍历这个 map 里面的所有 element。 就是类似树遍历一样的方法

### Onsite:

- 第一个: 给两个单词, 比如 head, tail: 找到一个最短的转换,从 head 到 tail,每次只能变一个字母,path 上的 word 都必须是有有效的英文单词,我用的 Graph shortest path
- 第二个: memcpy: 源区域和目标区域可能有重叠BST 插入和删除操作实现BST iterator 实现
- 3: 实现两个函数: H() and O(), 这两个函数会被多线程调用。 当一个线程调用 H 或 O 时,如果当前已经有至少两个线程 call H 和一个线程 call O。 那么让两个 call H 和一个call O 的线程返回(产生一个水分子),其他的都 block。
- 4: Given a social graph, find if there is a path between two persons with at most 2 steps (3rd level connection), how to handle it in distributed way (large graph stored at a large number of nodes, minimize cross-communication)
- 5: 设计题: a restful server with 4GB,  
given a request such as: http://seq=4?len=60?xxxxdata  
the system will store the binary data with that sequence number.  
given a request: http://startseq=3?maxLen=100, the system returns all data objects with sequence  $\geq 3$  with total data length less equal than 100. multiple clients calling simultaneous what data structure, concurrency, locking, etc..

### Question:

1. public double pow(double a, int b) { 1point3acres.com/bbs
2. Given a sorted array that has been transposed (that is, a portion has been removed from one end and attached to the other), write a function to determine if a given number is present in the array.  
Examples: [6 7 1 2 3 4 5] => find 1 or 4; [9 13 32 54 1 2 5] => find 1 or 8  
public static boolean isInList(float targetValue, float[] list)

## 面试题

### 1. wordList Distance:

/\* This class will be given a list of words (such as might be tokenized from a paragraph of text), and will provide a method that takes two words and returns the shortest distance (in words) between those two words in the provided text.

\* Example:

\* WordDistanceFinder finder = new

WordDistanceFinder(Arrays.asList( "the" , "quick" , "brown" ,  
"fox" , "quick" ));

\* assert(finder.distance( "fox" , "the" ) == 3);

\* assert(finder.distance( "quick" , "fox" ) == 1);

\*/

### 2 & 3. Max sum subarray & Max production subarray

### 4. Add interval:

public interface Intervals {

/\*\*

\* Adds an interval [from, to] into internal structure.

\*/

void addInterval(int from, int to);

/\*\*

\* Returns a total length covered by intervals.

\* If several intervals intersect, intersection should be counted only once.

\* Example:

\*

\* addInterval(3, 6)

\* addInterval(8, 9)

\* addInterval(1, 5)

\*

\* getTotalCoveredLength() -> 6

\* i.e. [1,5] and [3,6] intersect and give a total covered interval [1,6]



## 面试题

\* [1,6] and [8,9] don't intersect so total covered length is a sum for both intervals, that is 6.

\*

\*/

```
int getTotalCoveredLength();  
}
```

### Question1:

/\*\*

\* Given two words as Strings, determine if they are isomorphic. Two words are called isomorphic

\* if the letters in one word can be remapped to get the second word.

Remapping a letter means replacing all

\* occurrences of it with another letter while the ordering of the letters remains unchanged. No two letters

\* may map to the same letter, but a letter may map to itself.

\*

\* Example:

\* given "foo" , "app" ; returns true

\* we can map 'f' -> 'a' and 'o' -> 'p'

\*

\* given "foo" , "boa" ; returns false

\* we can map 'f' -> 'b' , 'o' -> 'o' , we can't map 'o' -> 'a'

\*

\* given "bar" , "foo" ; returns false

\* we can't map both 'a' and 'r' to 'o'

\*

\* given "turtle" , "tletur" ; returns true

\* we can map 't' -> 't' , 'u' -> 'l' , 'r' -> 'e' , 'l' -> 'u' ,  
'e' -> 'r'

\*

\* given "ab" , "ca" ; returns true

\* we can map 'a' -> 'c' , 'b' -> 'a'

\*/

## 面试题

### Question2:

```
public interface FirstCommonAncestor {
```

```
/**
```

```
* Given two nodes of a tree,
```

```
* method should return the deepest common ancestor of those nodes.
```

```
*
```

```
* A
```

```
* / \
```

```
* B C
```

```
* / \
```

```
* D E
```

```
* / \
```

```
* G F
```

```
*
```

```
* commonAncestor(D, F) = B
```

```
* commonAncestor(C, G) = A
```

```
* commonAncestor(E, B) = B
```

```
*/
```

```
Node commonAncestor(Node one, Node two);
```

```
}. more info on 1point3acres.com
```

```
class Node {
```

```
final Node parent;
```

```
final Node left;
```

```
final Node right;
```

```
public Node(Node parent, Node left, Node right) {
```

```
this.parent = parent;
```

```
this.left = left;
```

```
this.right = right;
```

```
}. 1point3acres.com/bbs
```

```
bool isRoot() {
```

```
return parent == null;
```

```
}
```

```
}
```

## 面试题

### 电面:

1. 给一个二叉树,返回它的镜像实现一个 thread-safe blocking queue
2. 一个嵌套 Map, 就是一个 HashMap, 它的 value 可以是一个 element 也可以是另外一个嵌套 map 或是空的 map. 实现一个 iterator 来遍历这个 map 里面的所有 element。就是类似树遍历一样的方法

### Onsite:

第一个: 给两个单词, 比如 head, tail: 找到一个最短的转换,从 head 到 tail,每次只能变一个字母,path 上的 word 都必须有效的英文单词,我用的 Graph shortest path

第二个: memcpy: 源区域和目标区域可能有重叠

BST 插入和删除操作实现

BST iterator 实现

3: 实现两个函数: H() and O(), 这两个函数会被多线程调用。当一个线程调用 H 或 O 时,如果当前已经有至少两个线程 call H 和一个线程 call O。那么让两个 call H 和一个 call O 的线程返回(产生一个水分子),其他的都 block。

4: Given a social graph, find if there is a path between two persons with at most 2 steps (3rd level connection), how to handle it in distributed way (large graph stored at a large number of nodes, minimize cross-communication)

5: 设计题: a restful server with 4GB,

given a request such as: http://seq=4?len=60?xxxxdata

the system will store the binary data with that sequence number.

given a request: http://startseq=3?maxLen=100, the system returns all data objects with sequence  $\geq 3$  with total data length less equal than 100.

multiple clients calling simultaneous

what data structure, concurrency, locking, etc..

## 面试题

### Question:

1. 查找 2 个单词的距离

/\*

\* Example:

\* WordDistanceFinder finder = new

WordDistanceFinder(Arrays.asList( "the" ,  
"quick" , "brown" , "fox" , "quick" ));

\* assert(finder.distance( "fox" , " the" ) == 3);

\* assert(finder.distance( "quick" , "fox" ) == 1);

\*/

2. 洗牌 要求 in-place

\* Return the smallest character that is strictly larger than the search character,

\* [ 'c' , 'f' , 'j' , 'p' , 'v' ], 'a' => 'c'

### Question:

1, 给一个 array,生产一个新的 array,新 array 中的每个元素都是上一个 array中除了这个位置之外其他的元素的乘积。

原题链接:<http://www.geeksforgeeks.org/a-product-array-puzzle/>

这个题从原始 array的两边往中间搞,把乘的结果存起来,然后新的 array 利用左右的结果相乘得出,时间复杂度  $O(n)$

2, Find the shortest distance between two words in a string, 好像也是 leetcode 原题,记不清了。

input: { "you" , "he" , "she" , " her" , "you" }, "you" , "her"

output: you 和 she 的最短距离是 1

### Question:

1. 层序打印 binary tree

2. 实现 BlockingQueue 的 take() 和 put()

public interface BlockingQueue

{

## 面试题

```
/** Retrieve and remove the head of the queue, waiting if no elements  
are present. */
```

```
T take();
```

```
/** Add the given element to the end of the queue, waiting if necessary  
for space to become available. */
```

```
void put (T obj);
```

```
}
```

3. 实现一共 TwoSum interface

```
public interface TwoSum {
```

```
/**
```

```
* Stores @param input in an internal data structure.
```

```
*/
```

```
void store(int input);
```

```
/**
```

```
* Returns true if there is any pair of numbers in the internal data  
structure which
```

```
* have sum @param val, and false otherwise.
```

```
* For example, if the numbers 1, -2, 3, and 6 had been stored,
```

```
* the method should return true for 4, -1, and 9, but false for 10, 5,  
and 0
```

```
*/
```

```
boolean test(int val);
```

```
}
```

### Onsite:

1. behavior with manager

2. system design – tiny url

3. max points in a line, minimum window substr, array of products without itself

4. technical communication

5. print all factor combinations of a given value, maximum value path on a grid starting from (0,0) to (n-1, n-1), can only move right and down

## 面试题

### Question:

/\*

Question Description: You are to write an abstraction layer for a persistent buffer. Provide an implementation of the following abstract class:

\*/

```
public abstract class pBuffer {
    protected final int BLOCK_SIZE = 1024;
    protected final int BLOCK_COUNT = 1024;
    protected byte[] buffer = new byte[BLOCK_COUNT * BLOCK_SIZE]; // A sample
    1mb buffer, to be allocated in 1k chunks.
    public pBuffer() {
        fillBufferFromFile(); // Reads the buffer from file and dumps the
        contents into the array, restoring the state to what it was when onShutdown(
        ) was called
    }
    // Returns a Location for a free block of the buffer, suitable for passing
    to put, get, and free
    public abstract Location allocate() throws NoAvailableSpaceException;
    // Stores up to BLOCK_SIZE bytes of data in location l. Data beyond BLOCK_
    SIZE bytes should be truncated
    public abstract void put(Location l, byte[] data);
    // Returns the BLOCK_SIZE bytes of data stored at location l, or null if l
    is unallocated
    public abstract byte[] get(Location l);
    // Indicates that an area of the buffer is no longer needed, and can be
    reused
    public abstract void free(Location l);
    // Called on shutdown
    private void onShutdown() {
        writeBufferToFile(); // writes the full contents of the buffer to disk,
        for reading when later invoked by the constructor
    }
}
```

要求实现 allocate, put, get, free 的内容. 已知条件, pBuffer 这个类在初始化过程中已经调用了 fillBufferFromFile. 此外有个 onShutdown()函数要把 buffer 的内容写到 disk.

## 面试题

### Coding Question:

Given an array with relative indexes, determine if there is a single complete cycle or not.

For example:

$x = [3, 1, 2, -1]$  – false

$y = [-5, 1, 4, 6, 4, 2, -1, 5]$  – true

### Solution:

```
static boolean isCycle(int []arr){
    if(arr.length==0)
    { return false;
    }
    boolean []visited= new boolean[arr.length];
    int current=0;
    int count=0;
    while(count<arr.length)
    { current=(current+arr[current])%arr.length;
    }
    if(current<0) { current=current+arr.length; } visited[current]=true; count++;
    } for(int i=0;i
```

### Question:

1. Reverse Array:  $len+=1$  改成  $len-=1$
2. Manchester Array:  $result=(a[i]==a[i-1])$  改成  $result=(a[i]!=a[i-1])$
3. checkgrades: 两个  $||$  改成  $|||$
4. 根据奇偶 replace value:  $i \leq len$  和  $j \leq len$  改成  $i$  换成  $< 6$ . 另一种排序: 选择排序,  $arr[\min] > arr[x]$  改成  $arr[\min] > arr[y]$
7. count occurrence: while 里漏掉了  $i++$



## 面试题

### Question:

1. accountOccurancy while loop 最后加 i++
2. 计算分数 “||” -> ” ”
3. 整数处以位数取余 要 int 一个变量 value 来存储 num 最后用 value%count
4. sortarray “>” -> “<” 5. 若输入奇数换为 1,偶数换为 0 “i<=len, j<=len” -> “i” -> “<” 7. printpartternfor 少了大括号

### Question:

1. countOccurance, 2. SelectionSort, 改 arr[index\_of\_min] > arr[x] 为 arr[index\_of\_min] > arr[y]
3. DesendingSort, 改大小于号
4. replaceValue, 改 i +1 为 i = i +1 或者 i++
5. printPattern, 比如 input 给 5, 要打出  
11  
1111  
111111  
11111111  
1111111111  
错误是加括号
6. printPattern(忘了名字了), input num 如果给的是偶数,就打出前 num 个偶数,如果input num 是奇数就打前 num 个奇数  
错误是括号和 i = i +2
7. checkGrade  
改||为

### Question:

1. ReverseArray, arr[len-1]改 arr[len-i-1],删除 for loop 里 len += 1
2. C[size=14.666666666666666px]ount occurrence, while 里加 i++
3. Sort Array in place, > 改 < 4. check grade, || 改 5. replace values 两个 for loop 里面 i<=len , j  
<=len 改 i<len,j 改 < check grade, || 改 补充内容 (2016-1-3 10:15):  
replace values 两个 for loop  
里面 i<=len , j <=len 改 i</len,j

## 面试题

### Question:

Given a 2 dimensional matrix where some of the elements are filled with 1 and rest of the elements are filled. Here X means you cannot traverse to that particular points. From a cell you can either traverse to left, right, up or down

Given two points in the matrix find the shortest path between these points

For example if the matrix is

1 1 1 1 1

S 1 X 1 1

1 1 1 1 1

X 1 1 E 1

1 1 1 1 X

Here S is the starting point and E is the Ending point

### 算法题

1. 根据两个 log file(一个昨天,一个今天)找出两天都出现 customer,具体背景是,他听说 lz 是做广告的,然后就让 lz 统计哪些 customers 昨天,今天都访问了统一网站。log file 纪录的就是网站的访问记录。然后问了时间复杂度

2. find target in ascending and descending array (先 ascending 后 descending)然后要求 code module 化,最后问了复杂度。

### 第一题:bst 最短路径

```
public static int minSum(TreeNode root){ if(root == null){ return 0; }
if(root.left == null
root.right == null){ return root.val; } if(root.left != null root.right == null){
return
minSum(root.left) + root.val; } if(root.left == null root.right != null){ return
minSum(root.right)
+ root.val; } return Math.min(minSum(root.left), minSum(root.right)) +
root.val; }
```

节点的每种情况都要写出来

## 面试题

### 第二题: round robin

```
public static float waitingTimeRobin(int[] arrival, int[] run, int q) { if
(arrival == null || run == null || arrival.length != run.length) { return 0; }
int waitTime = 0; int curTime = 0; int index = 0; int len = run.length;
LinkedList queue = new LinkedList(); while (!queue.isEmpty() || index <
len) { if (!queue.isEmpty()) { Proccess curProccess = queue.poll();
waitTime += curTime - curProccess.arrTime; curTime += Math.min(q,
curProccess.runTime); while (index < len arrival[index] <= curTime) {
queue.offer(new Proccess(arrival[index], run[index])); index++; } if
(curProccess.runTime > q) {queue.offer(new Proccess(curTime,
curProccess.runTime - q)); } } else { queue.offer(new
Proccess(arrival[index], run[index])); curTime = arrival[index++]; } }
return (float) waitTime / len; } private static class Proccess { int arrTime;
int runTime; public Proccess(int arrTime, int runTime) { this.arrTime =
arrTime; this.runTime = runTime; } }
```

### A (Airbnb)

1. 2D array, 访问顺序必须是‘回’字的方式,就是从外圈转到里圈,写出 class, Iterator, hasNext(), next().
2. 电话号码和计费的一个 log, 去 parse 看规定时间内哪个号码产生费用最高。
3. leetcode anagram 的一题变种
4. 有很多个 sorted queue 存在不同服务器上,如何有效的读取到一个 sorted 大 queue 里 (google 也面到了这题)
5. 设计 db, 如何存取房东和房客 reviews, 如何 maintain 他们之间的关系。

### A (Amazon)

1. leetcode tree 的一题,就是每层的 nodes 横着也是连着的
2. 设计搜索,在 amazon 搜索如何设计。
3. 写一个 class 可以把树存入到 db 里。
4. 设计游戏的背包 ood.

## 面试题

### G(google)

1. 一段话,里面有几个关键词可以被替换成别的词,比如 \$Foo 可以换成任意的词,设计 class 搞这个。
2. 一道图的题,打印出所有的环。
3. 有很多个 sorted queue 存在不同服务器上,如何有效的读取到一个 sorted 大 queue 里
4. 在一个 2d 数组里,打印出某一块矩形所框范围内的所有值的和。
5. 2d 数组里走格子,给你 A 点位置,有的格子不能走(石头)有的能走,问最短路径从A 到 B.

### Question:

```
int row;
int col;
Point(int row, int col) {
    this.row = row;
    this.col = col;
}
public class Maze {
    public static int solution(int[][] matrix) {
        if (matrix == null || matrix.length == 0 || matrix[0].length == 0) {
            return 0;
        }
        if (matrix[0][0] == 9) {
            return 1;
        }
        if (matrix[0][0] == 0) {
            return 0;
        }
        final int[] dx = { -1, 0, 0, 1 };
        final int[] dy = { 0, 1, -1, 0 };
        int m = matrix.length;
        int n = matrix[0].length;
```

## 面试题

```
Queue q = new LinkedList();
matrix[0][0] = -1;
q.add(new Point(0, 0));
while (!q.isEmpty()) {
    Point curPoint = q.poll();
    for (int i = 0; i < 4; i++) { int nextRow = curPoint.row + dx[i]; int nextCol =
    curPoint.col + dy[i];
    if (nextRow >= 0 nextRow < m nextCol >= 0 nextCol < n) { if
    (matrix[nextRow][nextCol] == 9)
    { return 1; } else if (matrix[nextRow][nextCol] == 1) { q.add(new
    Point(nextRow, nextCol));
    matrix[nextRow][nextCol] = -1; } } } return 0; } }
```

### Debug:

reverseArray: arr[len - 1] -> arr[len - i - 1] remove len += 1;  
sortArray: max < arr[j]; repalceValues: 去掉 for 循环里的 =, 改成 i <  
len; j < len printPattern:  
for 循环加 {} manchester Array: A == A[i - 1] 改成 A != A[i - 1];  
countOccurence: while 里面最  
后加上 i++; selectionSort: arr[index\_of\_min] > arr[x] 改成 > arr[y]

### Logica:

3, 11, 25, 45? 71

985: 874 :: 763 ? 652

FASTER:HCUVGT SLOWER ?UNQYGT

KML, PRQ, NPQ, TVU ? NPQJOHN:LSNV:: MARK: OEXS

小明向北走 11miles, 然后向南走 30miles, 然后又向北走 30miles, 问小明  
位置

### Question:

1)

```
class Solution {
    ListNode* merge2Lists(ListNode* l1, ListNode* l2) {
    ListNode* dummy = new ListNode(0);
```

## 面试题

```
ListNode* cur = dummy;
while (l1 != NULL l2 != NULL) {
    if (l1->val < l2->val) {
        cur->next = l1;
        l1 = l1->next;
    } else {
        cur->next = l2;
        l2 = l2->next;
    }
    cur = cur->next;
}
cur->next = (l1 == NULL) ? l2 : l1;
return dummy->next;
}
```

```
2)
int pathSum(TreeNode* root) {
    if (!root){
        return 0;
    }
    stack st;
    stack stVal;
    minS = 1 << 15; st.push_back(root);
    stVal.push_back(root->val);
    while(!st.empty()) {
        TreeNode* top = st.top();
        int val = stVal.top();
        st.pop();
        stVal.pop();
    }
}
```

## 面试题

```
if (top->left == NULL top->right == NULL) {
    minS = min(minS, val);
}
if (top->right val + top->right->val < minS) { st.push_back(top->right);
stVal.push_back(val + top->right->val);
}
if (top->left val + top->left->val < minS) { st.push_back(top->left);
stVal.push_back(val + top->left->val);
}
}
return minS;
}
```

3)

```
int cacheMiss(int array[], int len, int size) {
    list store;
    map::iterator> m;
    int count = 0;
    for (int i = 0; i < len; i++) { int val = array[i]; // hit if (m.count(val)) {
store.erase(m[val]); } else
{ //miss count++; if (store.size() == size) { m.erase(store.back());
store.pop_back(); } }
store.push_front(val); m[val] = store.begin(); } return count; }
```

### Onsite

1: topological sort

2: 一个 two-dimentional image, 有 black 和 white 两种数值,把它最后用一个 quadtree 表

示。 每四个方型 pixel 可以合成一个,如果四个都是黑,合成一个黑,四个都是白,合成

一个白,否则合成灰。 recursion 解决

3. 写一个 hashmap,要求底层实现是 bst,get(key) 函数和 set(key, value) 函数

4. 地里有的类似 print company organization 那题



## 面试题

### Onsite:

五轮 Onsite 没有 coding,全是问实际问题怎么解决和 design。

1. 如何设计一个 priorityqueue service,client 可以 submit job request 然后 server 按照 priority 执行
2. 需要一个 key-value store with 1M qps,most read,1ms 99% latency,如果用 HBase 的话会有什么问题,怎么解决
3. 给很多整数,如何用 mapreduce 找 median,如果是很多 float 数,可以有一定的误差,如何找
4. Programming Test 的扩展,如果 soduku matrix 非常之大怎么做,然后还有一大堆针对hadoop 的各种情况下怎么 optimize 的问题

### 电面:

1. 给平面一堆点,把所有在同一条直线上的点 group 在一起,求出所有的 group
2. 一种 encoding 的方法,如果一个 byte 第一个 bit 是 0,比如 00000000,那它自己表示一个字符,如果一个 byte 第一个 bit 是 1,比如 10000000,那它和它后面紧跟的 byte 表示一个字符,现在给一个 byte array,判断最后一个字符是一个 byte 还是两个 byte 组成。
3. parse message from byte stream,message format 是前 4 个 bytes 组成的 int 值表示 message 的长度 L,然后后面连续的 L 个 byte 是 message 真正的内容,每个 message 都是这样表示,需要一边读 byte stream 一边 parse 每个 message
4. 两个 table 做 join 有哪几种方法,分别有哪些 drawback
5. merge two sorted list
6. sqrt(double number, double epsilon)
7. auto completion implementation using trie
8. edit distance

### Onsite:

1. given a list of pairs, pair.first 表示 parent, pair.second 表示 child,reconstruct the tree, return the root node.
2. auto completion – design the service

## 面试题

3. design a service, accept stream of events, each event has a type and timestamp, need to support the query of top k most frequent types in a query specified [start, end] time range.
4. closest number to target in BST
5. validate sudoku / solve sudoku, and optimizations
6. 给一个 json object, 给一个 wildcard path with '?' as arbitrary name, 比如 a?.b 找到所有符合 path 的 objects

### Onsite

1. topological sort
2. design web crawler system, how to scale, what would be the bottle neck and how to solve the problem
3. 如何用 semaphore 或者 condition variable 实现 3 个 process p1, p2, p3, p2 必须要 p1 结束才能运行, p3 必须要 p2 结束才能运行
4. bloom filter 如何 implement, estimate false rate
5. what is the best design pattern do you think and why

### Onsite

1. 设计 Bi-directional LRU cache data structure, 既可以 lookup key to get value, 也可以 lookup value to get key, 还支持 set(key, value) 操作, 后面又加了条件, concurrent 的情况下, 会有什么问题, 如何改进, 假如 set 这个操作的频率远远小于 get 这个操作的频率, 需要写代码实现。
2. robot from topleft to bottomright LC 原题, 无障碍和有障碍
3. given a list of sets, find all pair of sets having any intersection
4. 设计 caltrain system, 要实现 caltrain 上车下车刷卡扣钱整个功能, assume 每个 station 都跟一个 central server 相连, 要处理如果有 network partition 怎么办, eventually 车费还是要 charge 到账户上, 但是不能影响 partition 的 station 正常运作。要处理某些人下车没刷卡怎么办, followup 可以非常多
5. 仍然是设计一个 concurrent 环境下的 time leased cache, 但是有些区别, 假如 delete 操作是一个 daemon thread 来做不用太多考虑, 但是 get(key) 操作的逻辑是如果 key 不在 cache 里面, 需要一个非常 expensive 的操作把对应 value load 进来, 如何让这个 load 的操作对同一个 key 尽量少发生。

## 面试题

### Question:

题目是 move robot

大意是输入指令是一个字符串 “NWES” N 表示向北移 1, W 表示向西移 1.

所以 “WN” 的结果是(-1,1)。“WS” 结果是 (-1,-1)

特殊字符 X 表示撤销上一次操作, 比如 WNX 表示 W, 输出时(-1,0)

每个字符前可加数字, 表示移动多步, 比如 7N3W, 输出时(-3,7)

7N3WX 输出时(0,7), X 表示撤销上一次 3W 的操作

7N3WXX 输出是 (0,0)

### Snapchat

- (1) Big integer (negative included)
- (2) Topological sort
- (3) Manager behavior question + N-queen II
- (4) Unique BST I, II + lots of

### Amazon

- (1) Given an array of integers, return the result after calculate square of each element(don't worry overflow): eg [1,2,3] => [1,4,9]
- (2) System Design yahoo news
- (3) 给一个 matrix 和字典, matrix 每一个 cell 是一个字母;从 matrix 里面找到所有的字典里的单词返回
- (4) OOP 餐厅等位系统
- (5) Behavior question + background + deep copy of a graph
- (6) TRIE + lots of related questions

Asana:

- (1) Given an array, return an array of product without current value  
example:  
given [1,2,3,4] => return [24,12,8,6]
- (2) OOP: 如何 solve 拼图
- (3) Regular expression match, 不是 leetcode 的那个题, 主要考点是计算 reverse index, 没让写 code, 主要讨论想法
- (4) (1) 不用除号实现除法 (2) 设计 data structure 存储 java script file (3) 拓扑排序
- (5) Powof4, OOP design 国际象棋

## 面试题

### Question:

给一个 array, 找出最高点或最低点, 例子如下

【1,2,3,2,1】 => 3

【3,2,1,2,3】 => 1

[1,2,3,4,5] => -1

第二题是 simple calculator (leetcode)

(3) 设计一个 cache, 要求实现如下功能:

1. Add
2. Search
3. Delete
4. Delete all

要求每个 function 的时间都是  $O(1)$ , 这个 cache 只会存储 1 - 500M 的数字

(4) Manager behavior questions

### Question:

1. LRU cache
2. 一个整数数组, 先递增然后递减, 也有可能只有递增或者递减。查找某个整数在不在数组里。
3. 设计 Boggle 游戏
4. OO design, 用树形结构表示表达式。注意 operator 要用多态实现。
5. 2 sum, 一个元素只能用一次
6. 1) 判断一个数组中是否有 3 个元素和为 0, 元素可以重用。2) merge k sorted array。3) 稀疏向量的点乘。
7. 一个数组, 把非 0 的元素移动到开头。
8. 1) maximum subarray 2) 树里两个节点的最低公共祖先 3) LC subset
9. 设计 fb newsfeed
10. 大数相乘
11. 1) 给一段话, 再给两个单词, 求这两个单词在这段话里的最小距离 2) 打印二叉树(level order 遍历)
12. 随机洗牌算法

## 面试题

13. 1)给一个字符串,返回每个字符及其个数。比如:aaabcc-> 3a1b2c 2) 给字符及其个数,返回原本的字符串 3)median of an array,有哪些方法,如果数据太多内存装不下怎么办

14. 用一个  $m*n$  的矩阵表示一副图片,其中每个元素对应 pixel 的灰度。Smooth the image with  $m1 * n1$  scale。也就是每  $m1 * n1$  的矩阵里面的值求平均,放到中间的那个像素里。如何节省内存?

15. 一个 server,多个 client。client 给 server 发任务,每个任务带有过期时间。

server 要按顺序处理这些任务,直到该任务做完或者被取消或者过期。实现提交,取消和查看任务状态的 API。

16. 二位数组的 Zig zag traversal

17. 一个数组长度未知。如果访问超过长度的 index 会产生 out-of-bound 异常。查找某个元素,如果不在数组内则返回-1

18. 给出左下角和右上角坐标,画出矩阵

19. 如何检测数据库的死锁

20. k-means 算法实现

21. 一个数组有  $n+1$  个元素,每个元素都在 1 到  $n$  之间,只有一个元素出现了两遍,找到这个元素

Round 1: Big integer add, follow up: add more than two integers in any base (not only decimal)

Round 2: A lot of questions about networking stack, TCP/UDP/HTTP. Write method to simulate network requests.

Round 3: Given a list of list of char. Input is a char, output is all chars which are in the same list of input.

Example: List: ((a, b, c), (b, d, e), (e, f), (g, h))

Input: b, output: a, c, d, e

Input: f, output: e

Preprocessing could be slow, but find method should be constant complexity

Round 4: OS related: processes and threads. Socket IPC and design problem about chat server. (Multiple sockets sending large files might cause delay/ jitter. How to handle this) Implement Game of life.

[http://en.wikipedia.org/wiki/Conway's\\_Game\\_of\\_Life](http://en.wikipedia.org/wiki/Conway's_Game_of_Life)

## 面试题

Round 1: Find all amicable numbers; Big integer multiplication.

Round 2: Game of life (same as yelp). Follow up: from 2-dimensional to N- dimensional. What if multiple users are accessing the same game board, and they have different viewports.

Round 3: Discussions about design servers for snapchat. Requirement: guarantee that the messages are reliably delivered; also need to guarantee deletion after reliable delivery.

Design an elevator.

Round 4: Flatten BST to in-order doubly linked list.

Palantir:

03/27 电面

Warmup: Read input in format: city, state, population (tab delimited file. population is of form: integer number always followed by a 'k'). Note that city name can contain spaces. Output the total population in each state in descending order

Given an array of integers which is initially increasing and then decreasing, find the maximum value in the array.

Given a stream of integers, find top K (heap)

Given an array of integers, find top K. top K elements do not need to be sorted. (quick select the Nth, then scan the array again, linear complexity)

Round 1: 写一个 Tree iterator

Round 2: Longest substring which repeated in a string, example: banana, return ana Flatten BST to in-order doubly linked list

Round 3: Given stocks with dates and values. For multiple companies. For each date, return the current total amount of stock you have. (a variation of Merge K sorted array)

Round 4: Word search in leetcode

Round 5: Convert map > to map > >

Find k missing numbers in an array 1-N



## 面试题

### Amazon (Boston office):

04/16 电面

Find how many times an array has been rotated

Input: [4, 5, 6, 7, 8, 1, 2, 3]

Output: 3

Find longest substring which has been appeared more than once (the same with Palantir round 2...)

Single number (leetcode)

Round 1: OOD, design a furniture class, type: table/chair, material: steel/ wood, methods: addWeight/putFire 测试家具强度,返回 true or false 判断家具是否可用

Round 2: Basic data structure problems. Stack with min, find common parts of two files (not sorted, files could be very large)

Round 3: Merge two sorted array (and eliminate duplicates), letter combination, basic design pattern questions.

Round 4: Given an array of intervals. Input: an integer, output: true if there exists at least one interval containing this integer.

Round 5: Given a board with 0 and 1, find number of all components (connected 1s).

Example:

1 0 0 0

1 1 0 1

0 1 0 1

0 0 1 0

Output: 3.

Serialization/Deserialization of a Binary Tree.

### Question:

design a string class , with implementation of charAt() and substring(b,e), with substring() requires O(1) time and O(1) space complexity

followup:



## 面试题

a new method `setCharat(index, char)` is added, a substring must keep the changes of parent string that are made before its creation, but both the parent string and the substring will not affect each other after the creation of the substring, requires  $O(1)$  space complexity.

### Question:

1: Find first repeating letter in a string.

比如输入 “abcba”, 返回 “a”

2: Merge 2 arrays in 1 array.

`static void mergeArray(int[] a, int[] b, int M)`

两个 sorted array 都有  $M$  个元素,但是  $a$  的 capacity 是  $M$ ,  $b$  的 capacity 是  $2M$ ,最后是把  $a$  中的元素加入到  $b$  中,保持 sorted。

3: Stock Maximize

### Question:

**Social Graph :** In this problem, we want to parse a members social group on the hot new social network. For the unfamiliar, each member in the network has  $n$ -many friends. Each of those friends has  $n$ -many more friends and on and on and on. We would like to write a program that given a social graph, outputs each level of friends. Each friend should only be output once, at the first level they are encountered. The root friend should not be output. Input will consist of several lines, where each line represents a member and their friends. The final line will be the student of whom we will begin our traversal. For instance, a member and their friends will be represented as a string: `A:B,C,D` Where  $A$  is the member and  $B$ ,  $C$  and  $D$  are the friends of  $A$ . If a member is listed without a colon, you can assume they have no friends of their own. Input 4 `A:B,C,D` `B:A,D,E` `C:E,B` `A` output: `B:C,D` `E`

## 面试题

### Question:

aa 1. Design a minesweeper game (<http://minesweeperonline.com/>)

a. A minesweeper game has a board with  $M \times N$  squares. The squares are initially covered.

b. A square contains one of the following objects

i. A mine

ii. A number – tell how many mines lay hidden in the eight surrounding squares

iii. Nothing – an empty square

c. User click on a square to uncover a square

i. Uncover a mine – game ends

ii. Uncover an empty square – continue playing, all the surrounding empty squares got uncovered

iii. Uncover a number – continue playing

5×6 Tiles

1 1 1 – – –

1 M 1 – 1 1

1 1 2 1 2 M

– – 1 M 2 1

– – 1 1 1 –

d. Focus on

i. Board initialization (randomness, uniformly distributed)

ii. A logic to uncover a square

1. stem-font”  $>n * m$  的二维数组。检查数组中的每个元素,如果以此元素为起点走  $k$  步中是不是有重复的元素。如果有输出 YES,没有输出 NO。

以下输入输出用例都是我搬地里同学的,因为是一样的。

输入:

4 //数组行数

1 2 3 4

5 6 7 8

9 10 11 12

13 14 15 16

3 //k

## 面试题

输出:

NO

输入:

4

1 2 3

. 1point3acres.com/bbs4 5 6

7 8 9.

10 11 12

2

输出:

NO.

2. shift Matrix

二维 square 数组顺时针往右 shift 一位。如果输入的不是 square 数组,需要输出 error, 最后 print 输出处理后的二维数组。

输入:

2 //接下来的输入行数

1 2 // 每个元素之间以 space 隔开

3 4

输出:

3 1

4 2

输入:

2

1 2 3

4 5 6

输出:

ERROR

### 一轮电面

1) A and B have a game. There are 7 games in total, who wins 4 games first will succeed the whole game and then the game ends. Given that A has a probability  $P$  to win a single game, and A already lost the first 2 games, what's the probability that A still wins the whole game.

## 面试题

2) Given that someone said A has 80% probability to win the whole game, what is the posterior that A win the whole game.

3) A specific scenario, Logistic regression, how to interpret the parameters.

4) R coding

1. a data set of transactions in a supermarket, select those who spend more than 100 dollars, pay attention to someone may have multiple transactions that are more than 100, do not duplicate the customers.

2. add the prices of all the transactions for each customer.

5) when you type a letter, say A, in google search, some suggestions will appear below, such as America, Apple, Amazon.... there are two models for those suggestions, how do you know which model is better. He refers to A/B test.

### Debug:

1. 降序: 代码里面就一个判断语句, 把判断语句里的 < 改成 >。

2. Manchester code: 里面就一个判断语句, 把判断语句的 == 改成 !=

3. selection sort: 这个要说一下因为面经里关于这题的答案很少。代码就简写了 for(x = 0 ; x < n ; x++) // n 是给的 list 的长度 { for(x = 0 ; x < n ; x++) // n 是给的 list 的长度 { int index\_of\_min = x; for(y = x ; y < n ; y++) { if(arr[index\_of\_min] > arr[y]) { // 这里错了 >>>>>> 正确

### 答案应该是

```
if(arr[index_of_min] <= arr[y]) { // '>' 变为 '<=' y = index_of_min; y = index_of_min; } int temp = arr[x]; // 把元素交换语句放到 if 中 int temp = arr[x]; // 这三句代码是要调换元素位置, 但是代码应放在 if 中 arr[x] = arr[index_of_min]; arr[x] = arr[index_of_min]; arr[index_of_min] = temp; arr[index_of_min] = temp; } } } }
```

4. 一个数 num, num 是奇数, 输出从 1 开始的 num 个奇数, num 是偶数, 输出 num 个从 0 开始的偶数: 题中有两个 for 循环(奇数情况一个 for 循环, 偶数情况一个 for 循环), 这俩循环没加 {} debug 部分就这些要说的

## 面试题

Reasoning: 1.QDXM:SFUN::UIOZ:?==》WKPA

2. AIE: FNJ::KSO:? ==> PXT

3. PSRQ, MNPO, SVUT, KNML ==> MUPO

4. LKJI, XYWV,WVUT,KJIH ==> XYWV

5. JOHN:LSNV::MARK:?==>OEXS

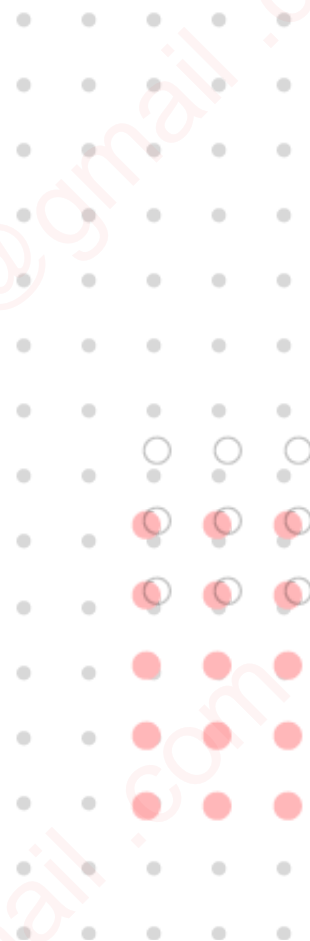
6. ASS ERT IVE NES S: SEN SSA EVI S TRE::MUL TIN ATI ONA L:?

==>ANO LUM ITAL NIT //三个字母为一组,最后多出来一个字母,所以一个单词里有 5 组,三个字母一组的(4 个)组内字母倒序,同时这五组也相互调换顺序,具体调换规则请自行总结

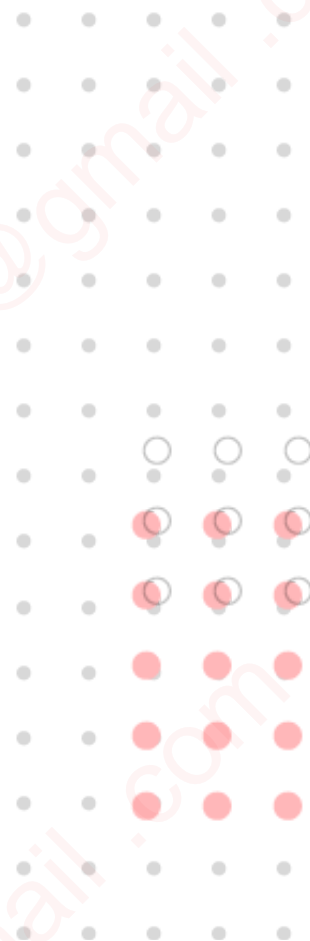
7.AG6,BI7,FA5,HC2,==>HC2

8.10,74,202,394,?==>650

## 面试题

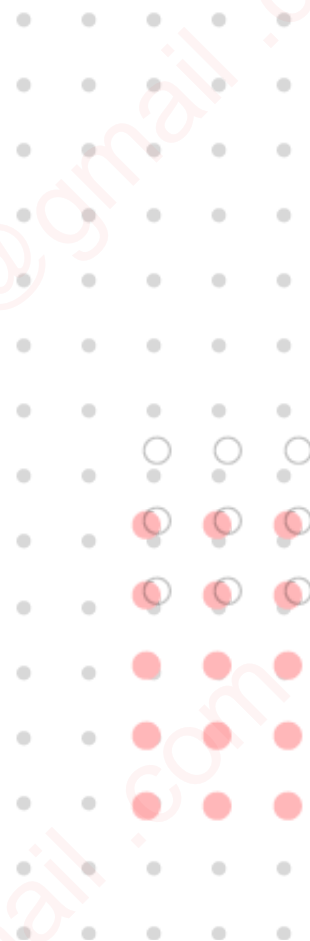


## 面试题





## 面试题



## 面试题

