

Mitbbs JobHunting Interview Questions

deepwaterooo

April 26, 2015

Contents

1	数组	5
2	字符串	7
2.1	写 atoi 函数, 测 nodepad	7
2.2	reverse words in a sentence, 使用如下函数。	7
2.3	covert interger number to date string, for example, 20090130 -> "01/30/2009"	7
2.4	第三道自己的做法是先取得 30, 然后 01, 然后 2009 然后组合成一个 string。问题	7
3	Linked List	9
3.1	复制 linked list。已知每个节点有两个 pointer, 一个指向后一个节点, 另一个指向其他任意一节点。O(n) 时间内, 无附加内存, 复制该 linked list。(存储不连续)	9
4	Sort	11
5	Search	13
6	Backtracing	15
7	DP	17
8	brain teaser	19
8.1	50 个黑球 50 个百球, 2 个罐, 要求你放这 100 个球在这 2 个罐, 使得别人随机从 2 个罐中任意拿一个球是黑球的几率达到最大。	19
8.2	heard on the street 上的男人出轨题, 简单逻辑推理。	19
8.3	2 个人商量好策略, 然后一个从 52 张牌里面随机抽 5 张, 看牌, 考虑。。。然后排在桌上, 摊开前 4 张, 第 5 张面朝下, 由第二个人判断第 5 张牌。问这个策略。	19
9	系统	21
10	c++	23
11	c++ and data structure	25
11.1	single linked list, find nth from the end	25
11.2	Overwriting and Overloading	25
11.3	Stack vs. Queue	25
11.4	Array of integers, all integers appear even times except one, find the one appears odd times. (some following up questions for this one)OOD	25
11.5	Do you approve the following design?	25

11.6 An open question.Takes more time than any other questions. It is related to the project they are working on, you should not be asked.	25
12 OOD OOP	27
13 Google / MS	29
13.1 continental divider	29
13.2 删除一个 singly linked list 节点, 但不知道 head. 不知道 head, 怎么找之前的那个节点阿? 又没说有 loop. 谁知道 trick 在哪?	29
13.3 Write code for finding number of zeros in n! OR Find the first	29
13.4 you have a billion google searches a day, design a data structure which lets you pull out the top 100 unique ones at the end of the day. 我的想法是 create hashtable. scan billion data 一次, 在 hashtable 纪录每个 query 的次数, 然后再 scan billion data 一次, 通过 heap 和 hashtable 找到 top 100, 不过这样的话,billion data 会被 scan 2 次, disk i/o 会很大不知道有没有什么 scan billion data 一次就可以找到 top 100 的办法	30
13.5 What is the best assertion that should be used as a post-condition?	30
13.6 The difference of following two expressions: Test A or Test B()	30
13.7 When a derived class is destructed, at what stage will the base class's destructor be called?	30
13.8 OOP constructor & destructors	31
13.8.1 什么样的情况下用 virtual destructor?	31
13.8.2 virtual function 是如何工作的? virtual table 是如何实现的?	31
13.8.3 virtual function 具体调用哪个 function 是在编译的时候, 还是在代码执行的时候决定的?	31
13.8.4 类的 copy constructor 和 assignment operator "=" 有什么区别? 有什么主意事项?	31
13.9 什么网站有 C++ 的测试题库, 哪有 free 的可以练练手? bloomberg 考的 C++ 问题哪里能得到?	31
13.10 给定一个二叉树的一个 node, 编程返回中序遍历的下一个 node。如果最后一个, 返回 null, 怎么做?	31
13.11 一堆数, 其中一些数出现了一次, 一些数出现了两次, 只有一个数出现了三次找出那个出现了 3 次的数 hash 方法很 trivial 就不说了。如果用 bitwise operator, 怎么高效的做? 除了 XOR, 是不是还得用点别的办法?	31
13.12 If the probability of rain tomorrow is twice than no rain.What is the probability of rain tomorrow: A grass, 3 cow can eat 3 days. 2 cow can eat 6 days. How long can one cow finish the grass?	31
13.13 Given a document and a query of K words, how do u find the smallest window that covers all the words at least once in that document? (given you know the inverted lists of all K words, that is, for each word, you have a list of all its occurrences). This one is really hard. Could someone propose an algorithm in O(n)?	32
13.14 Desgin an algorithm to find whether a given sting is formed by the Intealeaving of two given strings. 注意, 原来的两个 given strings 的本身的 character 的顺序不能变。	32
13.15 Given a set of points (x,y) , find all pairs of points whose distance is less than a given number, say, K. 这个题 brute force: 对每个点, 求和其他点距离, O(N ²), 不知道哪位大侠有高见啊	32
13.16 {1, 5, -5, -8, 2, -1, 15 } 要把负的扫到左边, 正的扫到后边。不能改变顺序得到 {-5 -8 -1 1 5 2 15} 这个题有 time 低于 n ² space=O(1) 的解法吗	32

13.17Mempool design with 30k limit.mempool 是应该在一开始就 allocate 30k 连续的内存, 然后分配和管理? 或者是每次 call allocate(n) 的时候再通过 operator new[] 来分配内存, update size member? 如果是的话, free(ptr, n) 怎么写呢? 貌似 operator delete[] 不能带 size 参数啊? 总之我就是对 memory design 这块很不熟悉。。。	33
13.18Implement put/get methods of a fixed size cache with LRU replacement algorithm. 这个是不是用 fixed size 的 max heap 来实现? 每个元素定义一个 key, 表示距离上次使用的时间, 每使用一个元素, 就相当于的是把它的 key 更新为比当前最小值更小的数, 然后做 heapify() 操作? 每 put 一个元素, 就 assign 新元素一个最小的 key, 然后用新元素替换掉堆顶点, 然后做 heapify?	33
13.19Write a function to implement a buffer for DataOutputStream. 这个我完全没啥概念了。。。求指点一下 ~	33
13.20How do you write malloc and free to detect memory reference violation?	33
13.21flag a block of memory as used by putting some bit pattern at the beginning of the block. What bit pattern will you use? 这两个问题也是摸不到头脑的。。。	33
13.22How to implement singleton without using static/global variable?	33
13.23其实这些题也适合别的 OS, 只不过面试的这个职位是基于 Linux 的。	34
13.23.1 buffer overflow 的工作原理:	34
13.23.2 关于 TCP 的实现的问题 (1): 操作系统中 TCP 的实现用到了几个 timer, 分别是什么? 这个题怪怪的, 谁能知道这样的细节?	34
13.23.3 关于 TCP 的实现的问题 (2): TCP packet header 中的 Window size (接受方的 window size) 的 update 是多久进行一次? 就是问接收方在什么情况下, 或者是多么频繁向发送方 update 新的 windows size?	34
13.23.4 关于 socket: TCP connection 用 socket 建立后, 有可能有很长时间通讯的双方没有任何数据来往, 比如 telnet client 登录 telnet server 后, 可能人会离开很长的时间, 这个时候 TCP server 怎么知道 TCP client 是 alive 还是 crashed? 如果你设计一个自己的应用程序, 你该如何处理? 就是问在你自己的 client 和 server 建立 connection 后, 你是如何 check whether the socket is still alive or not? 是在你自己的应用程序中定时的发一些类似于"Hello" 的 packet 作为查询呢? 还是 OS 的 socket 能够自动的提供 the information about the socket status?	34
13.23.5 如何用 C 语言实现 object oriented programming?	34
13.23.6 关于 kernel synchronization: 在 SMP 系统下, 用 spinlock, 还是用 semaphore 来作 synchronization 比较好? 为什么? 俺只知道如果你的代码不能 sleep 的时候必须用 spinlock, 比如在 interrupt handler 里面。还有就是如果用了 spinlock, 你要能够处理的很快。别的就想不出有什么区别了? 不过 Jonathan Corbet 的"Linux device drivers" 一书中说在 Linux kernel 的实现中, spinlock 引入的主要目的是为了 Linux 在 SMP 系统里运行的更有效, 不知这是为什么?	34
13.24如题, 职位是 web engineer, 希望有人可以用到。第一次电话是 recruiter 的, 按清单问了些问题:	35
13.25there are only 6 db connections in the pool, all 6 are being used,	35
13.26For the db connection. min = 10 and max = 40.	36
13.26.1 If we start with using 13 connections, when all the jobs are done, how many will be kept in the pool? 10 or 13?	36
13.26.2 If we start with using 45 connections, when all the jobs are done, how many will be kept in the pool? 40 or 45?	36
13.27suppose there are n cities, and there may / may not be flight route between c1 to c2. Design data structure to store this information and write a function that receives two cities name, and return whether or not there is a flight between them (either directly or through connections)	36

13.28不 trivial Given a 3x3 square:	36
14 Google interview question	39
15 google interview question from glassdoor	41
16 LinkedIn	45
17 Amazon	47
17.1 为什么对 Amazon 感兴趣。	48
17.2 自己最近的 Project。	48
17.3 说出自己会的编程语言并打分 (1 — 5)。	48
17.4 有没有开发 Mobile application 的经验。	48
17.5 几个常见 Data structure 的 Lookup 操作的时间复杂度。	48
17.6 HTTP post 和 get 的区别。	48
17.7 Design Pattern: Singleton, Factory, Lazy initialization。	48
17.8 Multi-threaded programming, deadlock 之类。	48
17.9 对 Unix 环境是否熟悉, 几个常见命令, ls, ps 之类。	48
17.10 Reflection 的概念, Java reflection, C++ 里面是不是有 reflection。	48
17.11 如何实现 Garbage Collection。 Reference counting 的缺点 (cycle), 如何解决, JVM 有 没有解决。	48
17.12 C++ 里面 virtual destructor 的用途, 于一般 virtual function 的区别。	48
17.13 写一个函数实现两个整数相除, 不用"/" 和"%", 返回商和余数。写完读给他听。	48
17.14 算法设计: 一个 Galaxy, 每个星星用一个三维座标表示, 找出离地球最近的 1000 个。	48
18 amazon	49
18.1 那道 wood steel table chair furniture 的题目	49
18.2 题目	54
18.3 学习了 backtrack(回溯法)	55
18.4 题:	57
19 bloomberg's phone interview question	61
19.1	61
19.2	61
20 hedge fund	63
21 Adobe	69
22 cloud service	73
23 FULL TIME SDE.	75
24 ftware Development Engineer	77
24.1 问候之后, 首先问了一下我的 research, 让我具体的阐述我提到过的算法然后 OOD 的 一道题, 其实不难, 但我感觉自己答得不好	77
24.2 后面的问题比较简单:	77
24.2.1 given a deck of cards, how to shuffle it?	77
24.2.2 having a web application, front-end, middle layer and database. How to scale database to accommodate increasing traffic?	77

24.3	consider a function which, for a given whole number n , returns the number of ones required when writing out all numbers between 0 and n . For example, $f(13)=6$. Notice that $f(1)=1$. What is the next largest n such that $f(n)=n$?	78
24.4	请解释 garbage collection? 在 garbage collection 中, 对 circular reference 的你怎么办?	78
24.5	栽在一道编程题上:	78
24.6	Given n points in the form $(x_1, y_1, z_1) \cdots (x_n, y_n, z_n)$, find the k closest points to the origin.	78
24.7		78
24.8	and(5) generates a random integer number between $[1, 5]$, how do you generate a random integer number between $[1, 7]$ when you can only call <code>rand(5)</code> ?	79
24.9	Given an integer, print the closest number to it that is a palindrome	79
24.10	Given a value and a binary search tree. Print all the paths(if there exists more than one) which sum up to that value. It can be any path in the tree. It doesn't have to be from the root. 我理解是这个 path 可以是其中任意一截, 不用包括头尾	79
24.11	given a character string, print the number of occurrence of each charcater inorder. ie. if the string is "ceabcw", then you should print something like:a 1 b 1 c2 e 1 w 1.she asked the possible data strucutre to approach. I gave array, hashtable, and BST. she asked me to use BST, and using no recursive. Also how to handle unicode.	79
24.12	题目:	80
25	Website Links	81

Chapter 1

数组

1. 一个 sorted interger Array[1...N], 已知范围 1...N+1. 已知一个数字 missing。找该数字。把原题改为 unsorted, 找 missing 数字。performance。
2. 一个 party N 个人, 如果一个人不认识任何其他人, 又被任何其他人认识, 此人为 celebrity。用 $O(n)$ 时间找到此 celebrity。

Chapter 2

字符串

2.1 写 atoi 函数，测 nodepad

2.2 reverse words in a sentence, 使用如下函数。

```
char* reverseWord(const char* str)
```

2.3 covert interger number to date string, for example, 20090130 -> "01/30/2009"

2.4 第三道自己的做法是先取得 30, 然后 01, 然后 2009 然后组合成一个 string。问题

是这样的话，月和日的 01 前的 0 可能会丢失，使得最后结果可能不对。后来想正确的做法应该是先把整形转成 string, 然后使用 substr 并组合。另外，面试者问有 stl 有什么可以替换 itoa, 自己答不出来。后来查了下，应该是可以使用 stringstream 来实现，如下：

```
stringstream ss;  
ss << intVal;  
ss.str() 就是我们要的结果。
```


Chapter 3

Linked List

3.1 复制 **linked list**。已知每个节点有两个 **pointer**，一个指向后一个节点，另一个指向其他任意一节点。 $O(n)$ 时间内，无附加内存，复制该 **linked list**。（存储不连续）

Chapter 4

Sort

1. an interger array containing millions of elements with min 0 and max 1000, how to sort it?

第二道使用 counting sort 应该就可以。面试者要求描述算法，不需写代码。

Chapter 5

Search

· 一来就是比较高难度的，给你一个字节数组（注意取值范围），数组长度可能非常长，如何找到第一个只出现了一次的数字。开始没什么思路，和他讨论了一会，边问还边问复杂度和数据结构的问题，后来发现应该进行数出现次数，这样复杂度就是 $2n$ ，结果出来了要求手写出代码。

Chapter 6

Backtracing

1. 设计一个函数，返回一组数字的组合，combination，不同的是，你每调用它一次，它就返回下一个组合，而不是一次全返回。注：你不能一次算完，把他们存起来，而必须临时算！

Chapter 7

DP

1. 手写 $\text{fab}(n)$ 函数，不是算，而是输出，递归或者循环都可，不过递归不高效大家应该知道

Chapter 8

brain teaser

- 8.1 50 个黑球 50 个白球，2 个罐，要求你放这 100 个球在这 2 个罐，使得别人随机从 2 个罐中任意拿一个球是黑球的几率达到最大。
- 8.2 heard on the street 上的男人出轨题，简单逻辑推理。
- 8.3 2 个人商量好策略，然后一个从 52 张牌里面随机抽 5 张，看牌，考虑。。。然后排在桌上，摊开前 4 张，第 5 张面朝下，由第二个人判断第 5 张牌。问这个策略。

Chapter 9

系统

1. 古老的三角形问题：输入 3 边，看是什么三角形。一个 mobile device 可以从服务器上传和下载图像，怎么测试这个系统？
2. lunch meeting 之后回办公室打开电脑，说他们现在开发的某产品有问题，每次要 loading 很久，差不多 10 秒的样子。问怎么测试并找出这个 bug？这个把我难住了，胡乱讲了一通，然后说太困难了；于是他换了个题目，画了一个 plotter 软件的界面，问怎么测。

coding 的题目是 Path Walk，给一条路径，写一个函数来走通它。其实这个题目我没搞明白什么意思，先沟通了很久，最后开始写（还是不太明白。汗…），写完了觉得不正确，正想再改改，被打住了，说给个 test case 一起来看看程序怎么执行。每句代码跑了一通，却发现 code 写正确了:-)

1. 一开始是个 IQ 题，把一堆数字填到格子里，满足一些条件，比如 1 和 2 不能相邻。测一个记事簿软件。有 scheduler 和 notifier 两部分，可以从 scheduler 输入时间和内容，然后 notifier 到预定时间会给出提醒。coding 题目很容易，找到单链表倒数第 N 个节点。

系统设计和经验：

1. 设计一个库，提供 timer 的功能。deltalist/hash，或类似 linux kernal 的 timer 设计。效率要比较高。
2. 一个类似 chord 的 DHT 设计。
3. 你有一个奇怪的程序，有时有 bug，有时没有，说出尽可能多的可能原因。
4. printf 来 debug 有何不妥。
5. process 和 thread。process 之间的 IPC 有那些种？process 间是否也可以 sharememory. 何时选 thread 或 process。

http://www.mitbbs.com/article/JobHunting/31393101_3.html

Chapter 10

c++

1. template 中用 typename 和用 class 有什么区别?
2. unix 下执行 shell 脚本和执行可执行文件有什么区别? 哪个更快, 为什么? 脚本语言程序 (如 javascript) 和可执行文件程序有什么区别? shell 和这两者却别呢?
3. 如何对 const data member 做 assignment?

```
class A {  
    const int a;  
public:  
    A():a(0){};  
    A(int m_a):a(m_a){};  
};  
  
int main(){  
    A a(1);  
    A b;  
    b = a; //how to implement assignment for this?  
}
```

4. 如果把 base class 对象赋给 derived class 对象, 会怎么样? compiler 报错还是执行错误?

```
class A{  
public:  
    int a;  
};  
  
class B : public A{  
public:  
    int b;  
};  
  
int main(){  
    A a;  
    B b;  
    b = a; //what happend? cout << b.b << endl;  
    B* b2;  
    b2 = &a; //how about this? cout << b->b << endl;  
}
```

1. 两个 C 的程序问题: 先是 char* 指针问题

```
char *dosth() {
    char s[256];
    char * p = r;
    p = "some new string";
}
```

- . 然后问了一堆变量的值，比如 s, *s, *(s+2), &p, etc.
- . 另外一个 switch 程序找错，没有加 break 之类，还有就是 return local variable 地址的问题

1. 逻辑问题：八个水罐称重

- 1, C vs C++
- 2, struct in C v.s. in C++ v.s. class in C++
- 3, virtual function, pure virtual function, abstract class what is the advantages of using virtual function
- 4, new v.s. malloc()
- 5, memory for a process (code, static data, stack, heap)
- 6, how to know the stack is growing in the direction of address increasing or decreasing
- 7, virtual memory

第一道被输入 const 给搞死了。先是没有注意 const, 直接按照常规非 const 做，没有写完就被叫停了；然后是被平时强调的 malloc 后必须及时 delete 规则搞死，坚持认为在函数里 malloc 一块内存然后在函数外 delete 是不好的习惯；最后当面试官提出如果定义一块内存，如 char tmp¹，然后使用会怎么样？自己提到可以在函数外 strcpy 函数返回结果，却忘了 arr 大小实际是无法指定的，所以这种方法是不可接受的。总之，很多的 trick 在里面没有注意到。

C++: effective c++ 上的东西若干；exception 相关；继承和子父类指针若干。十五分钟左右。

1. 大文件随机 sample, one pass.
2. sudoku solver.
3. logn 解 x^y ,
4. DP 题
5. 1Billion query 里选出时间最近 5 分钟内最 frequent 的 1000 个, one pass (我以前在 amazon 见到过这题)。
6. 两个排序数组找共同中值。递归和非递归解法。7. 斐波那契数列。100 层楼梯下楼，可以一步也可以两步，多少种下法？递归和非递归。8 贝叶斯后验概率。9。多少人在一起，生日可能出现重复概率大于 0.5? (算法导论原题，我只记得个答案，直接说了。。。)

1. 一个数组，找最大值比较次数？同时找最大值和最小值比较次数？找最大值和次最大值比较次数？(他问我是否知道这题，我说是作业题。后来和师兄聊说是这他常拿来用的面试题。)

¹DEFINITION NOT FOUND.

Chapter 11

c++ and data structure

11.1 single linked list, find nth from the end

11.2 Overwriting and Overloading

11.3 Stack vs. Queue

11.4 Array of integers, all integers appear even times except one, find the one appears odd times. (some following up questions for this one)OOD

11.5 Do you approve the following design?

```
Class Furniture{  
Some functions related to the property of furnitures;  
};
```

4 classes derived from Furniture

```
Class wood_chair  
Class steel_chair  
Class wood_table  
Class steel_table
```

What if you need to design a lot of other furnitures like desks... with other materials like plastics

11.6 An open question. Takes more time than any other questions. It is related to the project they are working on, you should not be asked.

http://www.mitbbs.com/article/JobHunting/31487819_3.html

1. given n strings with equal length, say x. find the substring shared by

all of them. For example, abcx, abdx, abea, then ab is shared by all of them.

1. the gmail page loads very slow. any suggestion for improvement?
2. we want to check the number of queries obtained from the world in the last minute and the last hour, what data structure should you use for that? If there are billions of records, i.e, too many records for the main memory, what suggestions do you have?

马上就要第二轮店面了，求大家的题目和建议，我们也可以私下交流。谢谢！

http://www.mitbbs.com/article/JobHunting/31487921_3.html

你有一种语言的 dictionary, 你有一大串 string, 没有 delimit, 你如何 interpret 成字典中的字呢?

http://www.mitbbs.com/article/JobHunting/31488093_3.html

Given a binary tree

```
struct node {
    struct node* leftChild;
    struct node* rightChild;
    struct node* nextRight;
}
```

The nextRight points to the right node to the current node in the same level. Ask you populate the nextRight pointers in each node.

http://www.mitbbs.com/article/JobHunting/31491521_3.html

今天 facebook 第一面，现在 hr 都开始问技术问题了。。。问我会什么语言，我就说 C++ best, 她就问我一些很基本 C++ 的问题，还有两个 bubble sort best case 的复杂度和一个排序的思路。

面完之后给了 puzzle 的 link, 要求做 meal 和 buffet 里挑一个。

http://www.mitbbs.com/article/JobHunting/31494081_3.html

1. Java 里如何比较两个 objects 是否相等
2. 怎样找出一个 list 是否包含循环
3. inheritance 和 composition: 什么时候需要用到哪种?
4. 一个 int array 如何找出 subarray, 使得元素之和最大比如 {-2,3,-1,3,-4} 那么答案应该是 {3,-1,3}

http://www.mitbbs.com/article/JobHunting/31494489_3.html

Chapter 12

OOD OOP

1. 一堆关于 OO 概念的问题，多态，继承，封装，接口和抽象类的区别，复写和重载（包括 C++ 具体怎么实现的）

然后就是一个智力问题，三个囚犯黑帽白帽，之前没见过，所以用了不少时间才想出来，大家可以搜搜，有现成的。最后反问问题后结束。

Chapter 13

Google / MS

13.1 continental divider

给一个矩阵，其中 0 代表海洋，其他数字代表高度，秉着水往低处流的原则，求出能够流向任意海洋的点。比如说

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

那么就要给出第二行的 4（这有这点出发，能够找到连通道四个 0 的区域的一条非递增路线），当然也有可能找不到这样的点，或者找到多个点。

http://www.mitbbs.com/article_t/JobHunting/32882153.html

13.2 删除一个 singly linked list 节点，但不知道 head. 不知道 head, 怎么找之前的那个节点阿？又没说有 loop. 谁知道 trick 在哪？

record the next node of it delete its next node do the assignment to copy saved next node to it

http://www.mitbbs.com/article/JobHunting/31454761_3.html

13.3 Write code for finding number of zeros in n! OR Find the first

non-zero digit from the right in 100! (Factorial of hundred). Can an int store hundred factorial. What size of array should be sufficient to solve the above problem. Write a code for the same.

http://www.mitbbs.com/article/JobHunting/31454839_3.html

13.4 you have a billion google searches a day, design a data structure which lets you pull out the top 100 unique ones at the end of the day. 我的想法是 create hashtable. scan billion data 一次, 在 hashtable 纪录每个 query 的次数, 然后再 scan billion data 一次, 通过 heap 和 hashtable 找到 top 100, 不过这样的话, billion data 会被 scan 2 次, disk i/o 会很大不知道有没有什么 scan billion data 一次就可以找到 top 100 的办法

http://www.mitbbs.com/article/JobHunting/31455781_3.html

```
list<int> L;
list<int>::iterator in_range = find_if(L.begin(), L.end(),
                                       compose2(logical_and<bool>(),
                                                  bind2nd(greater_equal<int>(), 1),
                                                  bind2nd(less_equal<int>(), 10)));
```

13.5 What is the best assertion that should be used as a post-condition?

```
assert(in_range == L.begin() || (*in_range >= 1 && *in_range <= 10));
assert(in_range == L.end() || (*in_range >= 1 || *in_range <= 10));
assert(*in_range >= 1 && *in_range <= 10);
assert(in_range == L.end() || (*in_range >= 1 && *in_range <= 10));
assert(in_range == L.end() && (*in_range >= 1 && *in_range <= 10));
```

http://www.mitbbs.com/article/JobHunting/31456679_3.html

13.6 The difference of following two expressions: Test A or Test B()

```
Class Test;
Test A;
or
Test B();
```

http://www.mitbbs.com/article/JobHunting/31456597_3.html

13.7 When a derived class is destructed, at what stage will the base class's destructor be called?

the answer varies on a case-by-case basis It will automatically be called after the destructors for the derived class data members. It will automatically be called before the destructors for the derived class data members. It should explicitly be called at the beginning of the derived class destructor. It should explicitly be called at the end of the derived class destructor

http://www.mitbbs.com/article/JobHunting/31456485_3.html

13.8 OOP constructor & destructors

13.8.1 什么样的情况下用 **virtual destructor**?

13.8.2 **virtual function** 是如何工作的? **virtual table** 是如何实现的?

13.8.3 **virtual function** 具体调用哪个 **function** 是在编译的时候, 还是在代码执行的时候决定的?

13.8.4 类的 **copy constructor** 和 **assignment operator** "=" 有什么区别? 有什么注意事项?

13.9 什么网站有 C++ 的测试题库, 哪有 **free** 的可以练练手?
bloomberg 考的 C++ 问题哪里能得到?

http://www.mitbbs.com/article/JobHunting/31457805_3.html

13.10 给定一个二叉树的一个 **node**, 编程返回中序遍历的下一个 **node**。如果最后一个, 返回 **null**, 怎么做?

http://www.mitbbs.com/article/JobHunting/31459733_3.html

13.11 一堆数, 其中一些数出现了一次, 一些数出现了两次, 只有一个数出现了三次找出那个出现了 **3** 次的数 **hash** 方法很 **trivial** 就不说了。如果用 **bitwise operator**, 怎么高效的做? 除了 **XOR**, 是不是还得用点别的办法?

http://www.mitbbs.com/article/JobHunting/31460327_3.html

就是一直一篇文章, 球可以覆盖所有单词的最小窗口, 记得 bbs 有几个人提过这个题, 但是没人给过解, 希望牛人能够赐教!! 感激!

http://www.mitbbs.com/article_t/JobHunting/31460569.html

13.12 If the probability of rain tomorrow is twice than no rain. What is the probability of rain tomorrow: A grass, 3 cow can eat 3 days. 2 cow can eat 6 days. How long can one cow finish the grass?

http://www.mitbbs.com/article/JobHunting/31461095_3.html

13.13 Given a document and a query of K words, how do you find the smallest window that covers all the words at least once in that document? (given you know the inverted lists of all K words, that is, for each word, you have a list of all its occurrences). This one is really hard. Could someone propose an algorithm in $O(n)$?

http://www.mitbbs.com/article/JobHunting/31461767_3.html

13.14 Design an algorithm to find whether a given string is formed by the interleaving of two given strings. 注意, 原来的两个 given strings 的本身的 character 的顺序不能变。

这个题不简单, 因为你不能简单的用 3 个指针分别指向三个 string, 遇到 string A 的就拷贝到 dst string, 遇到 string B 的就拷贝他的。最麻烦的在于遇到 A,B 都相同的, 你不能 advance both ptrs until they are different and then move one of them back. The point is who is to be moved back? You cannot simply randomly choose one.

For example,

stringA: ABCEF...

string B: ABCA...

dst string : ABCABCEF...

那么, 如果取 B's ABCA 就错了。

http://www.mitbbs.com/article/JobHunting/31463527_3.html

13.15 Given a set of points (x,y) , find all pairs of points whose distance is less than a given number, say, K . 这个题 brute force: 对每个点, 求和其他点距离, $O(N^2)$, 不知道哪位大侠有高见啊

http://www.mitbbs.com/article/JobHunting/31463131_3.html

13.16 $\{1, 5, -5, -8, 2, -1, 15\}$ 要把负的扫到左边, 正的扫到后边。不能改变顺序得到 $\{-5 -8 -1 1 5 2 15\}$ 这个题有 time 低于 n^2 space= $O(1)$ 的解法吗

http://www.mitbbs.com/article/JobHunting/31464055_3.html

这些东西我很都不熟悉。希望有高手指点指点, 呵呵

- 13.17 Mempool design with 30k limit.**mempool 是应该在一开始就 allocate 30k 连续的内存, 然后分配和管理? 或者是每次 call allocate(n) 的时候再通过 operator new[] 来分配内存, update size member? 如果是的话, free(ptr, n) 怎么写呢? 貌似 operator delete[] 不能带 size 参数啊? 总之我就是对 memory design 这块很不熟悉。。。
- 13.18 Implement put/get methods of a fixed size cache with LRU replacement algorithm.** 这个是不是用 fixed size 的 max heap 来实现? 每个元素定义一个 key, 表示距离上次使用的时间, 每使用一个元素, 就相当于它是把它的 key 更新为比当前最小值更小的数, 然后做 heapify() 操作? 每 put 一个元素, 就 assign 新元素一个最小的 key, 然后用新元素替换掉堆顶点, 然后做 heapify?
- 13.19 Write a function to implement a buffer for DataOutputStream.** 这个我完全没啥概念了。。。求指点一下 ~
- 13.20 How do you write malloc and free to detect memory reference violation?**
- 13.21 flag a block of memory as used by putting some bit pattern at the beginning of the block. What bit pattern will you use?** 这两个问题也是摸不到头脑的。。。
- 13.22 How to implement singleton without using static/global variable?**

完全没思路, design pattern 我基本上就是临时抱佛脚都还没抱上。。。

http://www.mitbbs.com/article/JobHunting/31464509_3.html

13.23 其实这些题也适合别的 OS，只不过面试的这个职位是基于 Linux 的。

13.23.1 buffer overflow 的工作原理：

问的特定环境是：在 client-server 的 model 下，client 是如何通过网络造成 server 上的 buffer overflow，从而在 server 上制造出 security hole？

俺只知道，肯定是 client 给 server 发的 packet 中，故意把特定的 field(比如长度) 弄错，使得 server 上的程序在 copy 的时候，造成 buffer overflow (因为一个特大的 length)，谁能说说到底 buffer overflow 是如何产生的？有什么好的文章，或者网站 link 介绍这个问题的？觉得搞 network security 的同学应该很明白。

13.23.2 关于 TCP 的实现的问题（1）：操作系统中 TCP 的实现用到了几个 timer，分别是什么？这个题怪怪的，谁能知道这样的细节？

13.23.3 关于 TCP 的实现的问题（2）：TCP packet header 中的 Window size (接受方的 window size) 的 update 是多久进行一次？就是问接收方在什么情况下，或者是多么频繁向发送方 update 新的 windows size？

13.23.4 关于 socket: TCP connection 用 socket 建立后，有可能有很长时间通讯的双方没有任何数据来往，比如 telnet client 登录 telnet server 后，可能人会离开很长的时间，这个时候 TCP server 怎么知道 TCP client 是 alive 还是 crashed？如果你设计一个自己的应用程序，你该如何处理？就是问在你自己的 client 和 server 建立 connection 后，你是如何 check whether the socket is still alive or not？是在你自己的应用程序中定时的发一些类似于"Hello" 的 packet 作为查询呢？还是 OS 的 socket 能够自动的提供 the information about the socket status？

13.23.5 如何用 C 语言实现 object oriented programming？

13.23.6 关于 kernel synchronization：在 SMP 系统下，用 spinlock，还是用 semaphore 来作 synchronization 比较好？为什么？俺只知道如果你的代码不能 sleep 的时候必须用 spinlock，比如在 interrupt handler 里面。还有就是如果用了 spinlock，你要能够处理的很快。别的就想不出有什么区别了？不过 Jonathan Corbet 的"Linux device drivers" 一书中说在 Linux kernel 的实现中，spinlock 引入的主要目的是为了 Let Linux 在 SMP 系统里运行的更有效，不知这是为什么？

13.24. 如题, 职位是 *WEB ENGINEER*, 希望有人可以用到。第一次电话是 *RECRUITER* 的, 按清单问了些问题

13.24 如题, 职位是 **web engineer**, 希望有人可以用到。第一次电话是 **recruiter** 的, 按清单问了些问题:

1. say some http methods?
2. get/put difference?
3. what does DTD for xml mean?
4. common protocol used in layer 4?
5. describe different ways to use css in html
6. difference between well-formed and valid xml?

前两天第二轮 technical phone interview:

1. why and how did u get into web development?
2. what do u like about web development? not like about it?
3. why do u want to work for google? 我扯到 ajax 的推广, 他顺着问 ajax principle, security issue
4. what language are you comfortable with? talk about it. why and how did people design it?
5. explain 3 components of MVC
6. what happens when a user types google.com in URL bar and press enter?(dns, http get, tcp connection establishment, etc)
7. what may slow down the rendering of html page when its contents have been downloaded from server? (load other resources like css,js and parse them, etc)
8. read n lines of random numbers(space as delimiter) from a file, lines with same numbers are treated as duplicated lines, regardless of the order. check and print non-duplicate lines. performance time analysis.

顺带问一个转身份的问题: 如果我 $H1 \rightarrow F1 \rightarrow H1$, 重新转回 $H1$ 的申请被拒了, 那是不是还停留在正常的 $F1$? 同样, 其它转身份, 例如 $F1 \leftrightarrow F2$, $F2 \leftrightarrow H4$ 互转之类, 如果申请被拒, 是正常停留在之前的身份吗? 前几天看到个帖子, $H1$ transfer 被拒, 身份就黑了

http://www.mitbbs.com/article/JobHunting/31467259_3.html

13.25 there are only 6 db connections in the pool, all 6 are being used,

another request needs to connect to DB, it does not want to wait. How to do it? One solution is to make a new connection and add it to the pool. But the interviewer wants standby solution. Anyone knows the standby solution? Thanks

13.26 For the db connection. min = 10 and max = 40.

Will 10 connections be created at server start up time?

13.26.1 If we start with using 13 connections, when all the jobs are done, how many will be kept in the pool? 10 or 13?

13.26.2 If we start with using 45 connections, when all the jobs are done, how many will be kept in the pool? 40 or 45?

http://www.mitbbs.com/article/JobHunting/31467451_3.html

13.27 Suppose there are n cities, and there may / may not be flight route between c_1 to c_2 . Design data structure to store this information and write a function that receives two cities name, and return whether or not there is a flight between them (either directly or through connections)

http://www.mitbbs.com/article/JobHunting/31469019_3.html

13.28 不 trivial Given a 3x3 square:

```
1 2 3
4 5 6
7 8 9
```

You are allowed to do circular shift on any row, and circular shift on any column, as many times as you please. Question: can you switch position of 1 and 2 with the allowed circular shifts?

http://www.mitbbs.com/article/JobHunting/31469459_3.html

通常看到这种题目都感觉有点头疼。比如，design a messaging system. an online poker room. 大家说说看

http://www.mitbbs.com/article/JobHunting/31470087_3.html

刚刚 on-stie 面试完某大公司。面试了 7 个人，大概问了 20-30 道题，有 1 道题不会，尽管其他的都打上来了，很是郁闷，本以为自己准备的足够好了，哎。但是这道题不会，很不甘心，希望大侠们帮助!!!

In our indexes, we have millions of URLs each of which has a link to the page content, now, suppose a user type a query with wild cards *, which represent 0 or multiple occurrences of any characters, how to build the index such that such a type of query can be executed efficiently and the contents of all corresponding URLs can be displayed to the users? For example, given a query http://www.*o*ve*ou.com. You may need to find [iloveyou.com](http://www.iloveyou.com), [itveabcu.com](http://www.itveabcu.com), etc. 以前我见过类似用 wild card 来做 query 的，就 * 来说，一个方法是用 *split query into a few parts, for example, *o*ve*ou => o, ve, ou, 然后分别用 o, ve, ou 查询，但是似乎不适合这道题。另外，如果对 Index 里的每一个 URL 建 suffix tree，然后对每个 query check against 所有的 suffix tree, 这样实际上就是 scan all urls, 明显也不合适。但是排序？我想不出来。

http://www.mitbbs.com/article/JobHunting/31472965_3.html

Chapter 14

Google interview question

Design a system to store heap on multiple machines ? What is avg number of machines accessed per operation and number of elements stored in a machine ? First greater number in an array. Given a large array of positive integers, for an arbitrary integer A, we want to know the first integer in the array which is greater than or equal A . $O(\log n)$ solution required

```
ex [2, 10, 5, 6, 80]
input : 6      output : 10
input : 20     output : 80
```

Given an N-by-N array of black (1) and white (0) pixels, find the largest contiguous sub-array that consists of entirely black pixels. In the example below there is a 6-by-2 sub-array.

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

http://www.mitbbs.com/article/JobHunting/31487235_3.html

Given a log file, which contains a series of websites, which the user has visited, find the most frequent path of 3 websites.

e.g: If this is a log file

```
A B C D E
A C D B E
C D E B A
A C D E B
C D E A B
```

clearly, C D E in the most frequent website?

http://www.mitbbs.com/article/JobHunting/31493409_3.html

1. find a pair that add up to a given sum
2. find all phone numbers in the html pages in a folder (and subfolder). something else, and self-introduction stuff

http://www.mitbbs.com/article/JobHunting/31493961_3.html

Chapter 15

google interview question from glassdoor

Design and describe a system/application that will most efficiently produce a report of the top 1 million Google search requests. You are given:

You are given 12 servers to work with. They are all dual-processor machines with 4Gb of RAM, 4x400GB hard drives and networked together. (Basically, nothing more than high-end PC's)

The log data has already been cleaned for you. It consists of 100 Billion log lines, broken down into 12 320 GB files of 40-byte search terms per line. You can use only custom written applications or available free open-source software.

http://www.mitbbs.com/article/JobHunting/31483445_3.html

u are given a binary search tree, each node has a parent, left and right do pre-order/in-order traversal without stack. cannot change the structure of Node. test cases: 8 6 7 5 4 9 10 11 12 test your codes using the test case above. http://www.mitbbs.com/article/JobHunting/31483789_3.html

关于排列组合的程序问题, 我一只都没理解太清楚, 现在厚脸皮来请教一下. 这些问题一般都要涉及到递归, 我这里不是问的算法的问题, 而是程序的实现问题. 我一直不知道怎么实现才是对的. 比如, 5 选 3 的全组合, a,b,c,d,e.

1 中间结果怎么保存, 是用一个 vector 来保存, 还是用多个 vector 来保存?

2 如果用一个 vector 来保存, 递归的时候, 最终状态是什么? 何时 pop, 何时 push, ?

http://www.mitbbs.com/article/JobHunting/31484637_3.html

Given an array, find the longest subarray which the sum of the subarray less or equal then the given MaxSum.

```
int[] FindMaxSumArray(int[] array, int maxsum)
for example, given array: {1, -2, 4, 5, -2, 6, 7}
maxsum=7
the result would be: {1,-2, 4, -2, 6}
```

http://www.mitbbs.com/article/JobHunting/31484653_3.html

given a integer, output its previous and next neighbor number which has the same number of bit 1 in their binary representation. 下面为什么去判断 (number & 3) != 2?

```
while ((number & 3) != 2) { // for right neighbor, change this line to
// (number & 3) != 1
```

http://www.mitbbs.com/article/JobHunting/31485091_3.html

要求当场写 code.

1 下面的 int * takeaddress() 有没有问题, 啥问题?

2 写个效率高的 takeaddress 出来, 实现同样的功能

```
int * paddress, address1, *r;
paddress = takeaddress(); /* defined below */
```

```

address1= paddress[0];

int * takeaddress()
{int address[8];
/* The address are defined here */
return address;
}

```

http://www.mitbbs.com/article/JobHunting/31485465_3.html

一个字符串，要求返回重复次数最多且最长的子字符串（假设源字符串中最长重复次数最多的子字符串只有一个）。例如“abcbcdabcbcd”要求返回“abcbcd”。因为“abcbcd”重复次数最多且最长。俺只想到两个土办法：

1) 找到所有字符串组合（例如 a, ab, abc, abca, b, bc, ...），都放入 hash table，找重复次数最多的且最长的。

2) 用 Dynamic Programming 找 LCS 的办法，两个字符串都是源字符串，然后在那个 2D array 里面找最长 match，并计算重复的次数，然后输出结果。

感觉两个方法的 time complexity 都挺大的，不知大家有没有什么别的想法？

http://www.mitbbs.com/article/JobHunting/31485529_3.html

很多都是老题，不过我专门整理了一下：

1. string match: string Text, Pattern; find a substring of Text matches with Pattern.

解法纲要: Rabin-Karp, KMP, suffix tree

变种 1b: multiple match: string Text, PatternSet[n]; find a substring of Text matches with any one pattern in the set;

解法纲要: Rabin-Karp

2.LCSubstring: string A,B; find the longest common consecutive substring;

解法纲要: DP(A.len*B.len 复杂度), suffix tree(A.len+B.len 复杂度)

3.Longest Palindrome: string A; find the longest substring of A which is a palindrome;

解法纲要: 类似 2

4.Wild card match:

4a: Pattern contains '?'(s)

4b: Pattern contains '*'(s)

4c: Pattern contains both;

//以下是与 dictionary 有关的题目

1. dictionary + wild card search(一般都需要做适当预处理):

第一种 search: search 所有 match 结果

第二种: 返回某个特定的结果, 比如, 所有 match 中最长的单词

5a: pattern = ??a????b* (指定某些位上的字母)

5b: pattern = abcde* (指定 fixed/unfixed length 的前缀)

5c: pattern = ?a*bcd*e?f* (? 和 * 任意混合搜索)

解法: 待探讨

1. dictionary + 包含字符集合:

Letter_{Set} = "aabbcd";

第一种 search: 所有至少包含 2 个 a, 3 个 b, 1 个 c, 1 个 d 的单词

第二种 search: 所有至少包含这个字母集合的单词中最长的/最短的

解法: 待探讨

1. convert a valid word to another valid word of the same length, by replacing one letter in one step, every intermediate word must also be valid;

解题思路：相同长度的单词构建一个图 + BFS

1. edit distance (misspell correction): type a misspell word, give top10/all suggestions of correct words;

解题思路：首先定义计算 edit distance 的 metrics，然后从每个 valid 单词计算出到它距离 \leq 某给定值的所有 misspell 的单词 (类似 BFS 的一层一层的算)

1. find a matrix with max area: each row and each column of the matrix must be a valid word;
2. 朴素搜索，在 dictionary 中搜索一个单词是否存在：

解题思路：hash; trie;

10b. shortest unique prefix: give a string, find its shortest prefix, which doesn't match with any prefix of any valid word in dictionary;

for example:

cat against {dog, be, cut} is ca

cat against {dog, be, cut, car} is cat

cat against {dog, be, cut, car, cat} is null

解题思路：trie/prefix tree;

1. solve a crossword puzzle;

http://www.mitbbs.com/article/JobHunting/31485923_3.html

1. N 台机器，每台机器有 N 个数找 median (2 个数组找 median 的扩展版)
2. 已知 coin denominator set，例如,2cent, 3cent, 5cent...给定一个目标数，比如 126centsk 这个题我以前问过一次，没人回。。。我觉得是很好的题，贪心，回溯，DP 都可以试试。但是我一直没找到最满意的解。
3. 一个整数数组，找 3 个数满足勾股定理。求比 $O(n^2)$ 更好的解

http://www.mitbbs.com/article/JobHunting/31486805_3.html

glassdoor 上看到一道题目：

Given a file of unknown size, devise an algorithm to give equal probability randomization to choosing a single line given a one line buffer space.

http://www.mitbbs.com/article/JobHunting/31487119_3.html

1. 很长的 log file 记录了用户访问 amazon.com 的过程，两列分别为 userID 和 pageName.

log 从上倒下按照点击发生的时间顺序。找出最 popular 的 3 连击。

```
zhang welcome
Li Hello
Wang welcome
Li books
Wang Hello
zhang books
Li shopping cart
Li checkout
zhang shopping cart
Wang camera
zhang checkout
```

最 popular 的 3 combo 是 books -> shopping cart -> checkout

1. Permutation of a string. 这题最郁闷，我把 programming expose 里的 code 默写了出来。但这个方法是不管字符重复的，假设都是不同的。现在考官要不显示重复的，而且他要求不能先都列出来再剔除，而要在发现重复的时候及时制止。没想出来
2. Design a flight ticket booking system.
3. 老板说网站很慢怎么办？老板说数据库很慢怎么办？

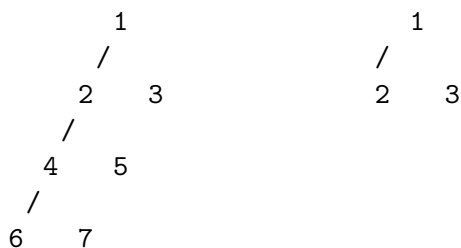
http://www.mitbbs.com/article/JobHunting/31487345_3.html

Chapter 16

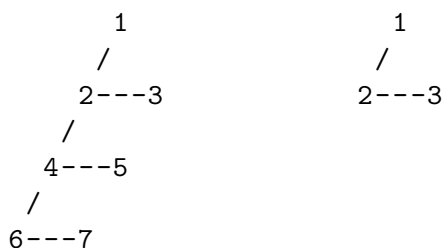
LinkedIn

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

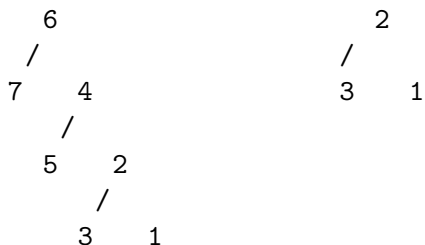
1. Given a binary tree where all the right nodes are leaf nodes, flip it upside down and turn it into a tree with left leaf nodes.
for example, turn these:



into these:



where 6 is the new root node for the left tree, and 2 for the right tree.
oriented correctly:



1. 电面不用 Gdoc, 用 CollabEdit

2. 第一题其实是 LC 原题的变种，等于的边界情况稍微处理一下就可以了

http://www.mitbbs.com/article_t1/JobHunting/32775405_0_1.html

1. 层序打印 binary tree
2. 实现 BlockingQueue 的 take() 和 put()

```
public interface BlockingQueue<T>
{
    /** 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);
}
```

1. 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);
}
```

http://www.mitbbs.com/article_t/JobHunting/32802467.html

Chapter 17

Amazon

- 17.1 为什么对 Amazon 感兴趣。
- 17.2 自己最近的 Project。
- 17.3 说出自己会的编程语言并打分（1 — 5）。
- 17.4 有没有开发 Mobile application 的经验。
- 17.5 几个常见 Data structure 的 Lookup 操作的时间复杂度。
- 17.6 HTTP post 和 get 的区别。
- 17.7 Design Pattern: Singleton, Factory, Lazy initialization。
- 17.8 Multi-threaded programming, deadlock 之类。
- 17.9 对 Unix 环境是否熟悉，几个常见命令，ls, ps 之类。
- 17.10 Reflection 的概念，Java reflection, C++ 里面是不是有 reflection。
- 17.11 如何实现 Garbage Collection。Reference counting 的缺点 (cycle)，如何解决，JVM 有没有解决。
- 17.12 C++ 里面 virtual destructor 的用途，于一般 virtual function 的区别。
- 17.13 写一个函数实现两个整数相除，不用”/”和”%”，返回商和余数。写完读给他听。
- 17.14 算法设计：一个 Galaxy，每个星星用一个三维坐标表示，找

Chapter 18

amazon

18.1 那道 wood steel table chair furniture 的题目

```
#include <iostream>
using namespace std;

class stuff {
public:
    stuff() {}
    virtual ~stuff() {}
    virtual void info() = 0 ;
};

class table : public stuff {
public:
    table() {}
    ~table() {}
    void info() {
        cout << "Table " << endl;
    }
};

class chair : public stuff {
public:
    chair(){}
    ~chair(){}
    void info() {
        cout << "Chair" << endl;
    }
};

class wood: public stuff {
public:
    wood(stuff * s): stf(s) { }
    void info() {
        cout << "Wood ";
        stf->info();
    }
};
```

```

    }
private:
    stuff* stf;
} ;

class steel : public stuff {
public:
    steel(stuff *s) : stf(s) {}
    void info() {
        cout << "Steel ";
        stf->info();
    }
private:
    stuff* stf;
};

int main() {
    stuff * wood_chair = new wood(new chair);
    stuff * wood_table = new wood(new table);
    stuff * steel_chair = new steel(new chair);
    stuff * steel_table = new steel(new table);
    stuff * wood_steel_chair = new wood(new steel(new chair));
    wood_chair->info();
    wood_table->info();
    steel_chair->info();
    steel_table->info();
    wood_steel_chair->info();
    delete wood_chair;
    delete wood_table;
    delete steel_chair;
    delete steel_table;
    delete wood_steel_chair;
}

```

output:

```

Wood Chair
Wood Table
Steel Chair
Steel Table
Wood Steel Chair

```

如果需要 plastic, 只需要再从 stuff inherit 一个 plastic 类就行了

===== 另一种方案 =====

```

#include <iostream>
using namespace std;

class material {
public:
    material() {}
    virtual ~material() {}
    virtual void info() = 0 ;
};

```

```
class wood : public material{
public:
    void info() {
        cout << "Wood ";
    }
};

class steel: public material {
public:
    void info() {
        cout << "Steel ";
    }
};

class furniture {
public:
    furniture() {};
    void setMaterial(material *m) {
        this->m = m;
    }
    virtual ~furniture() {};
    virtual void info() = 0 ;
protected:
    material * m;
};

class table : public furniture {
public:
    table() {};
    void info() {
        m->info();
        cout << " Table" << endl;
    }
};

class chair : public furniture {
public:
    chair() {};
    void info() {
        m->info();
        cout << " Chair" << endl;
    }
};

int main() {
    table *wood_table = new table();
    wood_table->setMaterial(new wood());
    chair *steel_chair = new chair();
    steel_chair->setMaterial(new steel());
    wood_table->info();
```

```

    steel_chair->info();
    delete wood_table;
    delete steel_chair;
}

```

output is

```

Wood   Table
Steel  Chair

```

http://www.mitbbs.com/article/JobHunting/31494857_3.html

suppose that you have a set of nodes with no null pointers (each node points to itself or to some other node in the set), given a pointer to a node, how to find the number of different nodes that it ultimately researches by following links from that node, without modifying any nodes. DO NOT use more than a constant amount of extra memory spa

http://www.mitbbs.com/article/JobHunting/31495985_3.html

1. 给定一个首尾相连的排过序的单链表，首节点最大尾节点最小，给出链表中任意一个节点，要求返回链表中间节点；
2. 一摞未排序的扑克中间有重复，用最有效的方法找出并删除重复者

http://www.mitbbs.com/article/JobHunting/31496467_3.html

昨天去某公司面试 Software Engineer 碰到的最后一道题：

有一种新语言，只能做三种操作。

X=0; 给变量赋值为 0;

X++; 递增

LOOP (x) {。。} 给定一个变量值就循环 X 次，循环 block 可以嵌套定义的三种操作。

题目是给定 B，求 A=B-1。

http://www.mitbbs.com/article/JobHunting/31496897_3.html

1. online skill assessment, Dec 2009 Some like GRE questions.
2. phone interview with 2 people in R&D, Dec 2009
3. how to find 1 missing number from 0 to N in an array of N numbers.
4. brainteaser, 5 jar problems.
5. how to calculate sqrt(N) without using sqrt function. Binary search tree problem.
6. some behavioral problem. Like, How do you know about BB? Why you wanna work in BB? Why you wanna work in industry?
7. onsite interview, Jan 2010 1st meet 2 people in R&D
8. train, tunnel, people escaping problem
9. 6 digits number, each changes from 0 to 9. Find the odds that sum of first three is the same as the sum of last three. A: 2 do loop.
10. Find 1 missing number from 0 to N. But notice that it is possible the sum would overflow. Think about a way to avoid the overflow.

11. Tricky problem. I do not think anyone else would know the answer except the one who gives the problem. Nothing to do with math, statistics.
12. Same 5 jars problem. That is their favorite.

2nd meet a lady in HR.

Ask 15-20 Behavioral problems. Cover most commonly behavioral problems.

3rd meet a senior manager in R&D

Ask one question, how to find the first unique number in an array of byte. and write a code to realize it.

http://www.mitbbs.com/article/JobHunting/31497519_3.html

Suppose there is a C function to count and return the number of nodes in a linked list.

What cases would you cover in unit tests of this function?

I can only think of two testing cases

(1): An empty list.

(2): An extrem long list with the length of the maximum value of unsigned int.

http://www.mitbbs.com/article/JobHunting/31499799_3.html

1.behavior question, Why you want to join BB?

2. 一个 windows 系统, 一个 unix 系统, unix 系统里有 100 个数据库, 总共 1TB, 如何在 1 小时内从 unix 系统转移到 windows 系统中 3. 找出一个字符串中最早出现的非重复字母 4. 两个鸡蛋测试那层楼丢下来会碎 5. 问了些做过的 project 的具体内容另, 我想再联系下我的面试官, 我知道名字, 怎么找到他的邮箱地址? 谢谢。

http://www.mitbbs.com/article/JobHunting/31499929_3.html

有一个循环链表 $a \rightarrow d \rightarrow b \rightarrow c \rightarrow e \rightarrow \dots \rightarrow a$, 每一个节点都是一个整数, 且不重复 (除了首尾节点外)。现在这个链表被拆断开了, 每 2 个相邻节点被存在一个 cell 里面, 但这些 cell 不是有序的。就是说链表被拆成了 $a \rightarrow d, c \rightarrow e, \dots, d \rightarrow b, \dots, b \rightarrow c, \dots$ 。我想重新把链表建立起来, 应该用什么样的算法?

http://www.mitbbs.com/article/JobHunting/31500287_3.html

合并两个 BST 要求 $O(n+m)$ 时间, n 和 m 为两棵树的大小。有什么好的解法么?

http://www.mitbbs.com/article/JobHunting/31500627_3.html

面试了一个小时左右。

1. 用两个 stacks 来实现一个 queue, 题不是很难, 但是要求逐行念代码, 精确到冒号分号, 尖括号怎么说不知道。。。耗了好久。
2. 一些关于 multi-threading, critical section, 等等。
3. SQL 的一些问题, 我不怎么会, 就 skip 了。
4. OOD 问题, 如何设计 parking garage, 大家有什么好的想法吗?

http://www.mitbbs.com/article/JobHunting/31501235_3.html

给你一个字典 array of strings (you may preprocess it if necessary) 任意一个单词, 求最小的 edit distance 一个单位的 distance 定义为:

a. replace a letter

b. delete a letter

c. insert a letter (also at any position)

快速的 code 出来~你就可以拿 facebook 面试了

http://www.mitbbs.com/article/JobHunting/31501445_3.html

18.2 题目

题目 1. LIS. 一个任意的数组，找出一个严格单调递增的最长子序列。例如: {3,0,1,7,2,4,5,9} -> output: {0, 1, 2, 4, 5, 9} 很简洁巧妙的算法，能在 $O(N \log N)$ 时间和 $O(N)$ 空间做出来！方法就是始终保持一个单增的序列，然后新来的数如果比当前最大还大就 append 在后面，否则在单增序列里面做 binary search，替换相应位置的数。

题目 2. 玻璃杯/鸡蛋 drop 问题。有 N 层楼，假定是在 i 层楼扔鸡蛋，如果没有碎，那么在所有 $\leq i$ 楼层扔鸡蛋都保证不会碎，反之如果碎了，那么保证在所有 $\geq i$ 楼层扔鸡蛋都必碎。通过若干次尝试扔鸡蛋，找到某个鸡蛋碎/不碎的“临界”层。允许你扔鸡蛋的总次数是 D ，允许你打碎的鸡蛋数是 B 。

问题的描述是：对一组给定的数 (N, D, B) ，如果存在一个策略保证能在 D, B 的限制下，在 N 层楼中找到“临界”层，那么称此 (N, D, B) 是 Solvable 的。接下来相关联的三个问题就是：

(a) 给定 D, B ，求满足 (N, D, B) Solvable 的最大的 N_{max} 。例： $D=4, B=1$ ，策略是从第一层开始一层层往上。 $N_{max}=D=4$ 。

(b) 给定 F, B ，求最小的 D_{min}

(c) 给定 F, D ，求最小的 B_{min}

这个问题相当容易找到看似最优的解，但是绝大部分的方法都不是最优的（最快最高效）。而且最迷惑人的是，(a)(b)(c) 三个问题中，必须先从其中某一个下手开始解决，如果你不幸的先从另外的两个问题下手，多半离最优解遥遥无望。

如果你找到了正确的入手点，有了正确的思路，最后的答案会异常的简单！入手点就是首先解决 (a) 问题，并且可以递归的来解决：假设 D, B 对应的答案是 $F(D, B)$ ，那么考虑在某一层摔一个鸡蛋后，如果碎了， $D-, B-$ ，如果没碎就只是 $D-$ ， B 不变。这样很容易写出递归方程，算出 F 关于 D, B 的 table。

题目 3. 经典的概率悖论。3 扇门，一扇背后有羊，你选中一扇门后，现在另外一扇门开了，里面是空的。问你是否应该重新选择。

分析：据观察，有一部分的人坚持认为一定要重新选择，另一部分的人认为是否重新选择都一样。另外少部分的人能看出，这个问题很巧妙的隐含了意识 (主观 intention)，信息和概率的关系！

题目 4. 很简单的， N 个数的数组，找出最大的和第二大的数，只用 $N + \log N - 2$ 的比较次数，不需要额外空间。这个是典型的问题本身就是答案提示的题目 - 基于比较又有 $\log N$ ，很显然思路涉及二分法，继续下去，剩下的问题就仅仅是找一个符合要求的 Implementation 了。

题目 5. 找 $N!$ 最后一个非零的数字。巧妙的方法可以在 $\log N$ 时间内找出来，一个 hint 是利用 $5^{k(\log_5 N)}$

题目 6. 任务分配，假设有 N 个任务，每个任务需要 W_i, M_i, w_i, \dots ，(大致上，最多只差一天)。

题目 7. 计算 Fibonacci 数 $F(n)$ ， $O(n)$ 的算法是很 trivial 的。但是有很漂亮简洁的 $\log(N)$ 算法，思路是利用 2×2 矩阵表示 Fibonacci 递推式，然后用二分法的思想求矩阵的 N 次方。

题目 8. 一颗 BinaryTree，每个节点有个 NULL 指针，要求把每个节点和在 BFS 中它的下一个节点串起来。其他 BinaryTree 的常见题有比如非递归的实现遍历，用 parent or stack。思考这些题的经验是，对于这一类的树的题目，有很强的递归性/规律性，通常都是 $O(N)$ 的复杂度，那么把 N steps 的问题，放在某个单 step 来研究，会把思路变得更清晰。另外一点就是，完全可以假设在做这一单步之前，在做这一步之前的问题已经最大可能的正确解决了，这样能够以一种数学归纳法的思想去利用之前的结论。比如这个题里面，假设节点 i 之前的节点都已经串好了，如何把 i 串到下一个节点。这个问题就是看一眼草图就能知道的了。最后一点经验是，在效率相当的算法的基础上，不同版本的实现，已经有能够互相启发的地方。

http://www.mitbbs.com/article/JobHunting/31502251_3.html

第一题。给一个数组 a^1 到 $a[n]$ ：例如 1, 2, 3, 4, 5, 6. 现在随机生成 a 的一个 permutation:

¹DEFINITION NOT FOUND.

b^2 到 $b[n]$ (例如: 3 1 5 2 4 6) 问, a 和 b 数组在每一位上都不相同的概率是多少? 假设 a 本身没有重复的数

主问题: $F(n)$ = 给定长度为 n 的 a 数组, b 数组有多少种取法辅助问题: 结果用 $f(n)$ 表示。 b 数组是 $\{1 \cdots i-1, x, i+1 \cdots n\}$ 的一个排列, 其中 $x \neq i$, 满足 a, b 在每一位上都不相同, 有多少种 b ? 例如, $a = 1, 2, 3, 4$; b 是 $\{1, 2, 5, 4\}$ 的一个排列。换句话说, 组成 b 的元素中, 有且只有一个数不在 a 中。这样定义了 $F(n), f(n)$ 后, 很显然有递推关系:

$F(n) = (n-1) * f(n-1)$ //解释: 第一位有 $n-1$ 种选择, 任意一种选择后, 问题变为一个 $n-1$ 规模的辅助问题

$f(n) = F(n-1) + (n-1)*f(n-1)$ //情况一, 在 b 数组的第 i 位置填入 x , 考虑剩下的 $n-1$ 个位置, 即是一个 $n-1$ 规模的主问题; 情况二, i 位置填入非 x 的数, 考虑剩下的 $n-1$ 个位置, 即是一个 $n-1$ 规模的辅助问题。

简化一下表达式就是:

$$F(n) = (n-1)(F(n-1)+F(n-2))$$

第二题, 一个 binary tree, 逆序打印 BFS 序列。不能同时用两段存储空间 (不同时用 queue 和 stack)

解法, 用一个 vector (array) 模拟 queue+stack。queue 的 push 操作即 vector 的 $push_{back}$, $q.pop()+stack.push()$ 的操作则是, vector 的 index 往前走一步! 最后把 vector 从尾到头打印一遍即可。

第三题, 网上看的答案, 超级巧妙, 生成一个 0-255 二进制数有多少位是 1 的查询表

```
static int BitSetCount256[256] = {
#define B2(n) n, n+1, n+1, n+2,
#define B4(n) B2(n), B2(n+1), B2(n+1), B2(n+2),
#define B6(n) B4(n), B4(n+1), B4(n+1), B4(n+2),
    B6(0), B6(1), B6(1), B6(2)
}
```

不得不说, 这个宏递归的方法用的太妙了!!! 附带赞一个巧妙度略低一些的计算二进制数有多少位 1 的方法

```
int bitSetCount(unsigned int i){
    int c=0;
    while (i) {
        c++;
        i &= (i-1); //这一步很赞, 每次保证清除最低一位; 1
    }
    return c;
}
```

http://www.mitbbs.com/article/JobHunting/31502237_3.html

18.3 学习了 backtrack(回溯法)

之前做了一些回溯的题, 比如打印 permutation, 打印任意 n 对括号等等, 都是瞎蒙的。还真凑巧, 上午做了打印 n 括号的题, 下午就看见有人说到回溯法, 想想自己还没系统学过这个, 找了本基础的中文算法书来看了看, 虽然书上讲的很浅显, 发现自己貌似瞎蒙还蒙对了思路, 呵呵。正好凑巧的是, 刚刚看了一点点, 网上就有个人问怎么做 Vertex Cover 的问题, 正好让我来做做练习。

1. 打印任意合法的 n 对括号:

```

void printParenthes(int N, int left, int right, stack<char> &stk) {
    if (left == N && right == N) {
        printStack(stk);
        return;
    }
    if (left > right){
        stk.push(')');
        printParenthes(N, left, right+1, stk);
        stk.pop();
    }
    if (left < N){
        stk.push('(');
        printParenthes(N, left+1, right, stk);
        stk.pop();
    }
}

```

1. Vertex Cover (NPC 问题), 图 G 中找一个顶点的最小子集, 覆盖图的所有边。

```

int current_k = N; //global
void VC(int k, int start_v){
    if (all_edge_covered(G) && k < current_k) {
        current_k = k;
        return;
    }
    if (k == current_k - 1) return; //剪枝
    for(; start_v <= N; start_v++) {
        if (!edge_list[start_v].empty()){ //剪枝
            list<int> temp_edge_list = edge_list[start_v];
            clear_edge(start_v, G);
            VC(k+1, start_v+1);
            if(current_k == k+1) return; //剪枝
            reset_edge(start_v, temp_edge_list, G);
        }
    }
}

```

想了想, 其中的 for 循环其实是不必的, 对于解空间树是子集树的问题, 只需要考虑《当前顶点“选”“不选”》两个情况改进后的算法是:

```

void VC2(int k, int start_v){
    if (k < current_k && all_edge_covered(G)) {
        current_k = k;
        return;
    }
    if(k >= current_k - 1) return; // 剪枝
    if(start_v == N) return; //没有下一个顶点了
    if(!edge_list[start_v].empty()){ //如果
        list<int> temp_edge_list = edge_list[start_v];
        clear_edge(start_v, G);
        VC2(k+1, start_v+1);
    }
}

```

```

    if(curent_k == k+1) return; // 剪枝
    reset_edge(start_v,temp_edge_list,G);
}
VC2(k, start_v+1); //不选这个顶点start_v
}

```

http://www.mitbbs.com/article/JobHunting/31502231_3.html

18.4 题:

1. 我们知道，从一个数组里找一段 (连续的) 子数组求最大和，是一道经典的面试题，方法很简单，只要 $O(n)$ 的时间。把这个问题变一下，假设是一个循环数组呢？找一个 $size \leq n$ 的子数组 with 最大和。

分析，很容易想到第一步，找个地方把循环数组切断，回到了原来的问题，然后在考虑一下额外的情况。额外的情况就是：有可能最大和的子数组是跨越了切断点的？这种情况的最大和怎么求呢？一个 naive 的方法能做到 $O(n)$ ，但是需要 $O(n)$ 的空间。巧妙的解法就是，注意到所有数的和是固定的，考虑切断后的非循环数组，找一段从首开始 + 一段从尾开始的两个子数组 with 最大和，等价于找一段子数组 with min sum.

总结，要擅长利用等价性转换问题，从而将新的问题转变为一个已知有好 solution 的旧问题。利用已知的经典问题来解决新问题，可以说是面试题目中相当重要的一个技巧

1. largest rectangular problem: 问题是这样的，一个 $N \times M$ 的棋盘，上面的数字要么是 1，要么是 0，那么要：

a) 最大的一个正方形全是 1 填充，

b) 最大的全是 1 的矩形。

a) 是用动态规划做，虽然方法也很好，但是这里就不提了。

b) 问题感觉上要比 a 难很多，为什么呢，因为 rectangular 比 square 有更大的自由度。不好用 DP 来做，分治也不合适。

这题的奥妙就在于，利用经典问题。什么经典问题呢？其实是另外一道面试题，其本身也是有一定难度的题，题目是：给你一个统计直方图，假设每根柱子都是单位宽度，从图的最左边一个紧挨一个排到图的最右边，求在这个图里找到一个最大矩形，它不跟任何直方柱相交 (边缘接触是允许的)。为什么提起这个题呢，故事是这样的，我之前没有做出 $O(N \times M)$ 解法的 largest rect 题，后来有一天遇到了这个直方图的题目，找到了很漂亮的 $O(N)$ 解法，猛然回顾起那道 largest rect 的题，这次就很轻松的搞定了。

3 (鸣谢 mitbbs jobhunting 版上的一位面试官贡献自己出的题) 有 n 个房间，小偷每天偷一间，偷的规律简单说就是随机行走，如果今天偷了第 i 间屋子，明天有一半的几率偷 $i-1$ ，一半的几率偷 $i+1$ ，注意如果刚好偷到了边界上，那么第二天只有唯一的选择。如果你是警察，你只能每天选择一个房间蹲守，并且贼的手段相当高明，偷了一个房间后，没有任何人能发觉该房间是否曾经被偷过。

提示：奇偶性。总结：注意观察题目中隐含的性质。

1. wild card 匹配 + 搜索：假设你有一个 dictionary(原题中是 URL 集合)，你要搜到所有与 $*a*bc*d$ 这样的输入所匹配的 words。这里， $*$ 是通配符，可以当成是任意个任意字符 (包括空)，怎么预处理 + 搜索？如果输入是 $???a???b???cde$ 这类呢？‘?’代表单个任意字符。如果输入是 $? *$ 的混合呢？

http://www.mitbbs.com/article/JobHunting/31502229_3.html

有 m 个 nuts, n 个 bolts, 规格大小都不相同

只能 nut 和 bolt 之间比较

怎么把他们排序? 要求复杂度最小

http://www.mitbbs.com/article_t/JobHunting/31502045.html

题目其实都不难, behavior 问了你最喜欢的 CS 的东西是什么, 我就说算法, 然后他还居然提了下我简历上量子计算, 估计他不会怎么感兴趣, 我就只简单提了一点。

然后 coding 题都是很基本的,

一个串 in place 删除某些字符, code 完了后, 反过来, 一个串在某些地方插入字符, 期间我脑子短路了一阵子, 还好过了一段时间后接上来了。。。

然后是个超级老题, 数组里面找唯一一个出现了奇数次的整数, 我这次很诚实, 直接说我知道这类 trick 的。。。不过他还是让我接着说了一下笨办法怎么做。

接下来又是一个老题, 楼层扔鸡蛋问题, 这个我前面的日记里面都贴过的, 所以除了表述上可能有些不清楚外, 算法本身肯定是 optimal 的了。接下来问了面试官几个 cliché 的问题就结束了, 至少表面上面试官还是比较满意的。。。

整个面了下来居然没有问 design 的题目, 也不知道是幸运还是不幸 (一次都还没被问过, 缺少实战经验)。。。

http://www.mitbbs.com/article/JobHunting/31502227_3.html

面的一般, 关键感觉那面试官比较冷, 问他也不说对错, 让人挺郁闷。

1. 如何寻找二叉树 (not binary search tree) 的 least common ancestor.
2. 如何测试一个计算器。
3. 如果你想打电话面试一个人, 拿到那个人的简历, 却发现他没有提供电话号码, 你

能想到几种方法找到他的电话号码。

最后一个问题还是挺好的, 不知道大家能想到几种方法。第一个问题我知道如何做, 就是找到从 root 到两个节点的 path, 然后比较 path 就可以了。但他问的很细, 可能我讲的不是很清楚。我现在想像这样具体到代码的问题, 咱们能不能说在电脑上写好了发到他邮箱, 然后再解释啊? 不然像这个问题, 一行一行解释真地挺难的。

http://www.mitbbs.com/article/JobHunting/31502699_3.html

very easy, but I think I have to say goodbye

1. deadlock's four condition
2. what's virtual memory
3. how to increase the virtual memory
4. when should we maintain v2p page table, when should we maintain p2v page table.
5. what's mmap & lazyloading. when should we use mmap to allocate memory
6. multi-level page table
7. implement a stack with lock to ensure thread safe

http://www.mitbbs.com/article/JobHunting/31356292_3.html

大部分的题都是板上贴过的, 再贴贴吧:)

1. C++ 和 C# 的最大区别?

2. 问到了 C++ 和 Java 在 Runtime Environment (没太听清, 可能是这个词) 的区别。后来我问了, 好像是在 virtual function 上的区别。我答得是 Java 里所有函数都是 virtual 函数, C++ 要显示标明。
3. TCP 和 UDP 的区别前面 3 个应该是根据我简历来问的, 其实我 Java 和 C# 也没用太多:(
4. 给定一个 int 数组和一个 int 变量叫 sum, 返回是否有数组中的 2 个数的和等于 sum。这个问题我最开始没处理好正好数组里有一个数等于 sum/2 的情况:(
5. n 级台阶, 一次可以上 1 级或 2 级, 有多少种上法老题目了, fibonacci number
6. reverse the words in a sentence, but don't reverse the words.

http://www.mitbbs.com/article/JobHunting/31344095_3.html

1. Given a random generator which can generate integer number from 1 to 5 with uniform probability. how to generate number from 1 to 7 with uniform probability.
2. Find the shortest path to convert one string to another using the minimum edits with each transformation string being a valid dictionary word in a dictionary.

for example: for->fork->ford->word->sword

http://www.mitbbs.com/article/JobHunting/31429703_3.html

how do you dynamically allocate space for a two dimension array in consecutive bytes?

should be easy

http://www.mitbbs.com/article/JobHunting/31432089_3.html

given N points in a place with their (x,y) co-ordinates. Find two points with least distance between them.

http://www.mitbbs.com/article/JobHunting/31437667_3.html

plz implement a non-recursive post order tree traversal. I think this is difficult. It is kinda simple for pre-order and in-order, but post-order is tough.

http://www.mitbbs.com/article/JobHunting/31455707_3.html

How do you measure context switch time in OS? any ideas?

http://www.mitbbs.com/article/JobHunting/31465291_3.html

you have a data structure of integers, which can be negative, zero, or positive, and you need to support an API with two public methods, insert(int) and getmedian(). Describe a data structure you would use to support this API and describe the running time of the two methods.

http://www.mitbbs.com/article/JobHunting/31472621_3.html

How to sort an array with only {0, 1, 2} possible values in O(n) without extra space?

Ex: an array {0, 1, 2, 2, 1, 0}

http://www.mitbbs.com/article/JobHunting/31472623_3.html

given a string, how to do a string rotation without using extra memory?

http://www.mitbbs.com/article/JobHunting/31473311_3.html

Q1) Given a list of characters and an int which is the distance between the same characters.

Eg: input- aaaaabbbbcc and distance as 2. One of the outputs can be- aabaababbcc

Come up with an algorithm and Code it.

Q2) Find out if two inputs are Anagrams with HUGE HUGE input (like thousand of terabyte)

Q3) Given lots and lots of points in a 2D space find all the line with most points on it.

http://www.mitbbs.com/article/JobHunting/31480153_3.html

dictionary is given. You have a word which may be misspelled. How will you check if it is misspelled?

http://www.mitbbs.com/article/JobHunting/31485125_3.html

Chapter 19

bloomberg's phone interview question

19.1

1. How to implement garbage collector (what data structure)
2. How to implement c++ smart pointer
3. Pro and Con of multi process and multi-thread
4. How many stack/heap does a multi-thread program with 10 threads have?

10 stacks? 1 heap?

19.2

1. 为什么加入 bloomberg? 2.introduce your experience. 3. 比较 c++ 和 java 的区别。内存管理, garbage class...compiler, 编译的作用? (gcc) (不懂 java。java 貌似编译了以后各个平台上都可以运行吗? virtual machine? C ++ 编译了以后的汇编语言只能在特定的系统上运行?) 4.C++ 中什么 function call delete a object? destructor 5. 谈到了 shallow copy deep copy.

1.

```
int main()
{
    return () main;
}
```

will this compile? will this run?

1. 利用一个写好的函数 putchar(char A)which prints out the character you

entered. 写一个 putlong (long A). 只能调用 putchar, 不能调用其他任何函数 (可以用 STL, 但是不能用 STL 中的函数)。后来还问到了如何 test 你的程序。这一道题折磨了我 15 分钟。only-putchar/ 所有问题不告诉你回答正确与否。他没有固定的几道题要问, 你谈到了什么就往深里面问。今天有 50 多人面试, 各个背景的都有。CS/EE 的我感觉就问编程, 非 CS/EE 的问智力题。再有就是通知巨突然, 周四晚上收到邮件, 让周六早上面试。。。体会: 是不是编程牛人, 一写程序就知道。。。这次又去当炮灰了。

<http://wuhrr.wordpress.com/2007/11/09/how-to-print-a-long-integ>

http://www.mitbbs.com/article_t/JobHunting/31500097.html

Chapter 20

hedge fund

职位是 junior financial engineer, 公司是一 hedge fund, 其实面完就感觉不太好, 一共见了 6 个人, 有两个人问得技术问题答得不太好, 也怪自己事先准备面试下的功夫不太到家, 准备得重点没有把握好。以下是一些能想起来的问题: 1.C++ 中的 virtual destructor 是啥? 为啥要用? 2.quick sort, merge sort 的复杂度. 3.Structure 和 class 的区别是什么? (我晕, 这个我居然给答反了) 4. 关于 C++ 处理异常的方法. (基本上一头雾水) 5.Monte Carlo method in american style option pricing. (我说的用 least square regression method,blah……)

1. $\int_0^T W(t) dW(t)$ (一看见这个, 贼激动阿, 熟悉的 ito's formula)

2. Stonivich intergral 是啥? 为什么用 Ito's 不用 stonivich? (不知道拼得对不对)

1. 一个国家所有的人如果生了一个男孩以后就停止生育, 生了女孩以后就继续生, 直到生出男孩才停止生育, 问多年以后男孩多还是女孩多? (要联系上 stopping time 的概念)。

1. 什么是 AR model? 啥时候用 AR model?

2. American option 的 up bound? (我说是 stock price, 被直接鄙视了, 说更精确的, 只好答没有研究过, 当时一头雾水)。

还有就是, 关于自己的简历上面的 Project 工作经历, 一定要熟练再熟练, 有些人问得那叫一个细啊, 而且基本上我所有的 Project 都被问到。这次面试的前 4 个人主要问计算机和金融方面的技术问题, 第 5 个 HR, 问些 personality 的问题, 最后是 hiring manager, 因为之前电话面试过我, 就没有问问题, 简单聊聊。整个面试花了 5 个小时, 雷死了, 脑子到后面都已经不转了。虽然结果让人遗憾, 不过就当是学习了, 贴点信息和大家共享下, 希望自己能早日找到工作, 也希望还在努力找工作的 XDJM 们再加把劲, 大家一起加油。 http://www.mitbbs.com/article/JobHunting/31406731_3.html

CS 方向, 希望对大家准备面试有帮助

1. 用 stack class 来实现 queue, 具体用几个 stack 不限。完了以后问怎么实现 thread safety, 然后是怎么测试。

2. 实现 strstr(str1, str2), 如果 str2 是 str1 的子串, 返回 true, 否则返回 false。实现完了以后问如何测试。

3. 给定一个 integer array with both positive and negative numbers, return a contiguous subarray with the largest sum. 我本来想用 dynamic programming 实现, 但面试官希望按照他的一个更 heuristic 的思路来解, 最后勉强搞定。

4. 给定一个排好序的 linked list, 删除其中所有的重复元素。比如给定 1->2->3->3->4->4->5, 返回 1->2->5。给定 1->1->1->2->3, 返回 2->3。看起来简单, 一边写一边发现许多细节需要小心应对, 好在最后搞定。
5. 给你三个烤箱, 每个烤箱可以同时烤两片面包, 需要的时间分别是 3 分钟, 4 分钟和 3 分钟。但第三个烤箱有一个 slot 出了点问题, 每次只能烤面包的一面。所以这个烤箱三分钟后只能算烤好一片半面包, 你需要把那半片翻个面, 在同一个 slot 里再烤一次才算一片完整的。现在给你这三个烤箱, 问烤好 21 片面包最少需要多少时间? 如果是 2100 片呢? 如果是任意给定的 N 片, 要求 $O(1)$ 时间内给出最少需要的时间。
6. 给你三根棍子, 每根都需要一个小时才烧完, 但每根燃烧的速度都不一样, 也不均匀。问只有这三根棍子和火柴, 如何精确的得到 1 小时 45 分钟的计时?
7. 在一个 party 上, 每个人可能认识别人, 也可能不认识。现在其中有一个人是名人, 定义就是所有的人都认识他, 但他不认识其余的任何人。现在要求你去找这个名人来。但你只可以通过一个方法, 就是问 A 是不是认识 B, 回答是表示 A 认识 B, 不是表示 A 不认识 B。你可以任意去问这样的问题, 问最少需要多少次能找出这个名人? 思路有了之后要求写代码实现, 可以调用 `knows(A, B)`, 代表上面的那个问题。实现完了以后问如何测试
8. 测试 `copy` 这个命令。然后自己问了一些 clarifying questions, 搞清了实际上是 `copy src dest`。src 可以是文件, 也可以是目录。dest 可以存在, 也可以不存在。

http://www.mitbbs.com/article/JobHunting/31410833_3.html

OO 设计题,

1. 怎么做一个十字路口的 traffic light.
2. 怎么不用 recursion 做二叉树 in order 遍历。

http://www.mitbbs.com/article/JobHunting/31421129_3.html

1. Write a function that returns a node in a tree given two parameters: pointer to the root node and the in order traversal number of the node we want to return. The only information stored in the tree is the number of children for each node.
2. Input a message and a text, find if the message can be composed by the text. If the text is in a magazine (two pages/a paper), how to design an algorithm?

http://www.mitbbs.com/article/JobHunting/31422009_3.html

1. When casting an object of a polymorphic class from a base class type, which one of the following casts performs the task only if the cast is valid?
 - a. `static_cast` b. `(void*)` c. `dynamic_cast` d. `const_cast` e. `reinterpret_cast`
- 1.

```
class A {
public:
    void f();
protected:
    A() {}
    A(const A&) {
    }
};
```

why are the default and copy constructors declared as protected?

- to ensure that instance of A can not be created via new by a more derived class
- to ensure that instance of A can only be created by subclasses of A
- to ensure that instance of A can not be copied
- to ensure that A cannot be used as a base class.
- to ensure that A cannot be instantiated on the stack

1.

```
template<class T1; class T2; class T3>
int Product(T1 a, T2 b, T3 c) {
    return a*b*c;
}
```

what is wrong with the sample code above?

- templates must be class definitions
- the template parameters should be separated by commas.
- the template definition is missing a pair of braces.
- template parameters must be pointer types.
- the * operator has not been defined for T1, T2, and T3.

1.

```
class FOO {
    char * buf;
public:
    Foo (const char *b = "default") {
        if (b) {
            buf = new char[std::strlen(b) + 1];
            std::strcpy(buf, b);
        } else
            buf=0;
    }
    ~Foo() {
        delete[] buf;
    }
};

Foo func (Foo f) {
    return f;
}
```

when the function fun is called, the program may crash or exhibit unexpected behavior, what is the reason of this problem?

- the destructor may attempt to delete the string literal "default"
- the destructor needs to check that the value of buf is not 0.
- the class does not allocate a long enough buffer.
- the function needs to return Foo& instead of Foo.
- the class needs to specify a copy constructor and assignment operator.

http://www.mitbbs.com/article/JobHunting/31426509_3.html

1. 请书写一个程序，将整型变量 x 中数字左右翻转后存到另外一个整型变量 y 中，例如 x = 12345 时，y 为 54321，x = - 123 时，y 为 - 321。其中 x 的个位不为 0。

```
void reverse (int x, int* y);
```

(1) 请实现该函数，以上函数原型是用 C 语言写的，你可以用你熟悉的语言；

(2) 请写出一段代码验证该函数在各种情况下的正确性。

2. 对集合 $\{1, 2, 3, \dots, n\}$ 中的数进行全排列，可以得到 $n!$ 个不同的排列方式。现在我们用字母序把它们列出来，并一一标上序号，如当 $n=3$ 时：0. 123 1. 132 2. 213 3. 231 4. 312 5. 321 现在，请书写一个函数 `void print (int n, int k)`，（函数原型是用 C 语言写的，你可以用你熟悉的语言）在已知 n 和序号 k 的情况下，输出对应的排列，并简要阐述思路。

3. 一维数轴上有 n 条线段，它们的端点都是已知的。请设计一个算法，计算出这些线段的并集在数轴上所覆盖的长度，并分析时间复杂度。例如，线段 A 的坐标为 $[4, 8]$ ，线段 B 的坐标为 $[1, 5.1]$ ，那么它们共同覆盖的长度为 7。请尽量找出最优化的算法，解释算法即可，不必写代码。

http://www.mitbbs.com/article/JobHunting/31428195_3.html

Given a sorted integer array and a number, find all the pairs that sum up to the number.

这个很简单，但现在多了一个条件 What if the array is sorted by absolute value, for example $\{1, -2, 4, -9\}$, find the answer in $O(N)$. 这样有什么好的思路么？

http://www.mitbbs.com/article/JobHunting/31430593_3.html

How do you find sequences of consecutive integers in a list that add to a particular number. Array 里面正负数都有. 这个能在 $O(n)$ 时间内解决吗？

http://www.mitbbs.com/article/JobHunting/31431861_3.html

A $m*n$ matrix of integer, all rows and columns are sorted in ascending order. Find the most efficient way to print out all numbers in ascending order.

http://www.mitbbs.com/article/JobHunting/31434325_3.html

一次面世 Google，问到 hash table 是怎么实现的。我说了一个取尾数 (round) 的方法，他说这个方法很 naive，工业界一般用其他的方法，比方说 STL 的 map。我想了半天没有想出来，到这里问问。hash table 具体怎么实现的啊？

http://www.mitbbs.com/article/JobHunting/31434401_3.html

49 辆赛车. Assume for each one, it travels the track in the same amount of time every time. Also assume no two finish the track in the same amount of time. Suppose you have 7 tracks, but no timer. Design races to find the 25-th fastest with minimal number of races.

http://www.mitbbs.com/article/JobHunting/31434523_3.html

How do you know the bloomberg?

What position do you expect?

What language do you want to answer with? (I choose C.)

What kind of questions do you meet for the online assessment?

what is static in C? how is it implemented by the compiler?

write the definition of a function that returns both the max and min.

why do you use the condition variable?

how to implement a lock?

Under what condition must you use linked list instead of array?

what data structure can you use to store elements dynamically and access them efficiently? The complexity of finding any element in a linked list in the worst case. multi-thread library programming: did you write your multi-thread library with p-thread? is there any problem you have with you library?

did you do your projects on linux? If you want to find a string in a file,

what command should you use?

do you know vector in C++?

a question about real-time programming (I forgot)

what is buffer overflow?

一些问题是针对我的简历里面提到的内容，所以，简历里面的内容要尽量吃透。

http://www.mitbbs.com/article/JobHunting/31434685_3.html

Given two classes:

```
class B {
public:
    B(args_1);
    B(args_2);
    // and many constructors with different arg lists
};

class D : public B {
public:
    D(args_1) : B(args_1) {}
    D(args_2) : B(args_2) {}
    // and many constructors with different signatures similarly implemented
    // some additional stuff specific to D
};
```

Assume that the arg list for B's constructors are quite long and may be revised pretty often in the future, in which case D's constructors have to be recoded correspondingly. Duplicating the update by copy-and-paste will certainly work here. Can you propose a better way so that the update can be done in one place without copy-and-paste duplication?

http://www.mitbbs.com/article/JobHunting/31434891_3.html

Given a large string (haystack), find a substring (needle) on it. 感觉这道题不就是 scan 一遍吗? 有什么 time and space complexity 上面的 trick 吗?

http://www.mitbbs.com/article/JobHunting/31435419_3.html

准备了很久, 看了很久算法的书。。 结果被问了一个怎么 optimize memcpy().. 傻眼了。。碰到了女老印, 倒霉~~~~~

http://www.mitbbs.com/article/JobHunting/31435587_3.html

给一个 substr, 如何判断它在不在给定的 str 里面。substr 有两个新的符号可能在里面:

- (1) * : 0-n 个任意字符
- (2) ? : 1 个任意字符

太紧张了, 所以面试者简化了题目, 说去掉 "?", 然后让 code 和测试: 基本框架出来了, 但是好多特殊情况没有处理到, 比如 substr 以 "?" 起头。后来又问如果加入 "*" 有没有思路, 刚说了两句就 out of time 了。

http://www.mitbbs.com/article/JobHunting/31436721_3.html

给定 $X[1..n]$ and $Y[1..m]$ 两个 arrays, 已经 sort 好了. 如何找到 $X \lt \text{Union} \gt Y$ 的 median? 我说用 merge sort, 要 $O(m+n/2)$ 时间。面试官明显不满意。这个已经 lineal 了? 难道还有更快的?

http://www.mitbbs.com/article/JobHunting/31437417_3.html

given a 32 bit number N and 2 numbers(A & B) that determine 2 different bit positions of N how do you make all the bits between A and B equal to another given integer k. given (A,B is in the range [0 to 31] and $k \leq 2^{(B-A+1)}$ (so that k fits between B-A+1 bits). Give an $O(1)$ solution forth is e.g if $N=9$ (1001), $A=0$, $B=2$, $K=5$ (101 then the result should be 1101 (1.e 13) 这个题是什么意思啊?

http://www.mitbbs.com/article/JobHunting/31437907_3.html

在做 careercup 上面的题目, 有两个问题没有看懂, 希望有人指点下

- 1 一个 BST, 给定一个值, 打印出所有的 path, 使 path 上所有节点的值等于给定值;
- 2 一个 tree, 如何高效的找出最长的 path?

这都是 amazon 的题目吧

1.sum of all nodes in a path = givenValue

2.<http://www.careercup.com/question?id=87897>

http://www.mitbbs.com/article/JobHunting/31441709_3.html

第一道是写一个函数，两个参数 (String prefix, String s)，返回 true 如果 s 有 prefix

第二道是写一个函数，两个参数 (int[] a, int sum)，找出数组里加起来是 sum 的几个数我第一题算是答出来了，第二题没做完，没有好的思路。。。

http://www.mitbbs.com/article/JobHunting/31446979_3.html

Chapter 21

Adobe

Went to Adobe to interview a Senior SW Engineer position, 总的 interview 的不错, 但被下面问题问倒了, 让回去想想,

Q1: "We need to compare thousands text files with each other, they are not big, less than 100K each. They are in a directories tree, with a few levels of subdirectories, how to speed up the comparing process?" My answers: We can read them all of these files into memory once so that we can reduce the number of diso I/O. [Feedback: That is a good approach].

Q2: How to read these files into memory (on MS Windows platform) ? how do you maintain directory structure in memory ? My answer: I talked some garbage ...

Q3: If someone already wrote the code in slow way, read each file from disk, do some thing, close the file, read another one, etc. Can you make a "portable layer API" library so that with minimal effort, old code can still work but much faster ? (of course, we need to recompile the code).

Please help with Q2 and Q3, thanks a lot.

http://www.mitbbs.com/article/JobHunting/31448285_3.html

今天把 M 的 onsite 给拒了, 实在没有时间面这么多 company, 又不想浪费别人的时间。不过心里还是觉得有点可惜, 啧啧。贴一下 M 的经历吧。On campus 就一轮, 30 分钟。Interviewer 是个老中, 一上来看我 resume, 问为啥 phd 了还来面 sde。然后开始问 resume 上的东西, 我借机会 sell 了一下自己。

Technical 问题只有两个:

1) Coding 题非常 old 了。两个 string 找最长 common substring。这个当场肯定 code 不了 subffix tree。于是就用暴力的方法, 三下五除二搞定。然后问 complexity, 如何改进, bla bla bla。竟然忘记了说可以用 DP, 低级失误啊。不过面试官还算满意。

2) 你认为 bing 有什么可以改进的 (我投的是 bing)? 你 research 做的东西有没有可以 apply 的?

http://www.mitbbs.com/article/JobHunting/31451397_3.html

今天又做 coding 面试了, 上来就要写个函数返回二叉排序树的第 k 个最小的 node。我写了一半, 感觉不对劲。请大侠赐教。

http://www.mitbbs.com/article/JobHunting/31451705_3.html

you are given a M x N matrix with 0's and 1's find the matrix with largest number of 1,

1. find the largest square matrix with 1's
2. Find the largest rectangular matrix with 1's

http://www.mitbbs.com/article/JobHunting/31452521_3.html

有几个同学问面筋, 不太记得起来, 很多版上是有的, 所以觉得那些面你的人水平挺一般的, 下面贴两个印象深刻的:

1. 这道题被好几个不同的公司面到过: Fibonacci 数列, 一般让你给一个 recursive 的版本, 然后写个 iterative 的版本, 然后问有没有更快的可能性。我记得以前在某个版讨论过, 参考 wiki: 这样的方法, 可以在 $O(\log(N))$ 的时间和 $O(1)$ 的空间复杂度内算好。要写程序的话, 用类似下面的方法:

```
Matrix2x2 F[][2] = {{1, 1}, {1, 0}}, Fn[2][2] = {{1, 0}, {0, 1}};
while (N) { if (N & 1)      mul(F, Fn, Fn);      // Fn = Fn x F;
  mul(F, F, F);           // F = F^2; N = N >> 1;
}
```

2. 另外一题很简单, 但是蛮 tricky 的。How to test if a number "a" is power of 2 return (a-1) & (a) == 0; 网上经常有问怎么样判断一个数里面有多少个 1 的位数, 这个只是其中一个最简单的特

http://www.mitbbs.com/article/JobHunting/31452533_3.html

于完成了 F 公司历时 2 个月的所有 interview, 总算可以松口气了, 据称他们下周一开会讨论, 希望最终会修成正果。来说点经历吧。多亏好朋友 Z 帮忙 forward resume, 很快就来了第一轮 phone interview。编程题还有点老:

[Coding Q1]: Given an array A, output another array B such that $B[k] = \text{product of all elements in A but } A[k]$. You are not allowed to use division. 其实这题 interview 之前在本版 JHQ 看过, 可是当时看的题目太多, 没有去想 solution。所以刚开始听到这题还 surprise 了一下。我觉得这个不能用除法的限制太无聊了 (建议改个 problem 来问这个 algorithm), 于是忍不住问 why not division, 顺便拖延一下时间想算法。面试官说除法慢...显然不是什么很 convincing 的理由, 我说那乘法也慢啊。说完我已经想到怎样做了, 于是顺利过关。接着就来了比较衰的第二轮, 题目是这样的:

[Coding Q2]: You are given a string e.g. "face" and a set of mutation rules, e.g. a->@, e->3, e-E. Print all the possible strings that can be generated by the rules, e.g. f@c3, fac3, etc. 其实就是 BFS 再加上 hash table 判断是否重复 print。马上就想到 algorithm, 面试官说好, 你开始写吧。然后问题就来了, 太久没写 c++ 忘了 hash table 的函数定义。好像依稀记得 hash table 还有几个版本, 想了一会没想起来, 又不好意思问, 汗! 最后还是忍不住问了, 他说你随便给个函数名和接口吧。最后磕磕碰碰总算把程序写完了, 却给人留下了很不好的印象, 感觉写程序很不熟! 据说最后这个人给了我一个 borderline, 还算好, 没把我 fail 掉。真惭愧啊, 可怜我还是写 c++ 起家的...因为第二轮不太理想, 本来应该两轮过后就 onsite, 结果 hr 来信说要第三轮 phone, 还很好人的说 We do have three phone interviews at times. We are constantly evaluating our process so I apologize for the change. 第三个面试官又临时换人, 最后居然是同系师弟, 不过之前没见过面, 不然可以套近乎了...言归正传, 换人大概还是因为他们组想看看我 match 与否。他问的都是 machine learning, 风格和原来完全不同, 还多多少少有点 surprising 的。

[Coding Q3]: Implement one step of decision tree which splits the node into two subtrees. 之后还讨论了一些 learning 的问题, 我问了他们用的 technique, 有什么存在问题等等, 相谈甚欢。

关于 onsite, 因为签了 NDA, 不方便透露题目。请大家也不用发信来问了, 做人还是要讲信用的。只看面经的各位看官可以略过以下了...至于那个 onsite 可谓一波三折。本来订了机票周四晚上到 sfo, 周五中午 onsite, 挺好的 schedule。结果某 airline 居然机件故障, 把飞机拖去修了几个小时, 又不肯调其他飞机来, 白白 miss 掉了从 vegas 飞 sfo 最后一班航班, 被迫在 vegas 住了一晚。下了飞机都晚上 12 点了, 随便找个 airport 旁边的 hotel 住下, 改了第二天最早一班机。结果第二天又晚点两个小时! 据称 SF 大雾, traffic control...折腾了半天到了 F 都周五下午两点多了, 又累又紧张。连 HR 也只好说: it's hard to visit us...不过不管那么多, 灌了杯 coffee 就上阵了, 结果还好, 没有想象中 intensive, 也发挥自己的水平了。因为 onsite 去的太迟, 没见到 manager, HR 又说 schedule TWO more follow-up, 其中一个是 manager。OMG, 我说好, 那就 back-to-back 吧。上周终于面完真是 relieved 啊, 前后 5 轮, 历时 2 个月。

关于面试的经验教训, 我的感觉是

1) F 的 interview 是比较严谨的, phone interview 就要 candidate 在 white board 上把 code 写出来, 不是说说算法就算了, detail 也问得很仔细。因为他们要求员工 follow 整个 project, 从 idea, 到 algorithm, 到 implementation, 而不是自己想个东西出来让别人写 code 实现就完了。各位像我一样

平时写 research matlab code 多于写 c++ 的 phd 要注意多练练手了，小心阴沟翻船。

2) Never give up 无论 interview 多么不顺利。不要被外界的不利因素 distract 自己，该准备什么就好好准备，我相信 life has miracles.

PS: 貌似 F 的同学们也会上来 job hunting 版。文中若有冒犯之处，请多多包涵。F 的 S 同学，你那题大概说了也不要紧吧，没理解背下来也没啥用的。F 的 Y 同学，我没有泄漏你的面试题，以后还可以继续用，哈哈。

http://www.mitbbs.com/article/JobHunting/31452725_3.html

5. Given a graph (any type - Directed acyclic graph or undirected graphs with loops), find a minimal set of vertices which affect all the edges of the graph. An edge is affected if the edge is either originating or terminating from that vertex. The time should be less $Q(n^2)$ 这个题就是最小顶点覆盖问题吧？或者是我对最小顶点覆盖问题理解有误？或者对这题理解有误？

http://www.mitbbs.com/article/JobHunting/31452961_3.html

00*100 部分有序矩阵数组的排序, 有 100 个有序数组 (从小到大), 每个里面有 100 个数。设计一个算法合并这个一百个有序数组, 中间步骤只允许多申请一个大小为 100 个数的空间 (也就是一个数组的大小)。

http://www.mitbbs.com/article/JobHunting/31453089_3.html

1. How to call C++ code in C? How to call C code in C++?
2. In which three cases, initialization list has to or is preferred to be used for a constructor?
3. Can we design singleton by setting all the data member and method of a class to be static?
4. Is overloading allowed in C? If not, how to differentiate them?
5. Default methods that are generated by a class in C++.
6. Difference of struct and class in C++
7. Given a class has first name, last name, SSN and etc. Need to query according to first name, first name + last name, what STL should be used? (If map/multimap, what should be the key?) How the query should be? How to query all the first name that initialed as "J"?
8. Meaning of static in C and C++
9. Meaning of inline in C++; where should it be used?

几天上午一个面试的问题。有些东西没用过，虽然以前看过，但是还是没有答出来；都去准备其它的了，没想到全是问 c/c++ 的问题。还是有些不服气，move on。

http://www.mitbbs.com/article/JobHunting/31454759_3.html

Chapter 22

cloud service

面的是 ELASTIC COMPUTING CLOUD 组的 SE, 我是做网络的, 对他们的这个 cloud service 有些兴趣, 以为会问算法和系统的问题, 结果问了一个 OOD 的问题, 说一个大楼, 10 层, 4 个电梯, 怎么设计类来实现这样一个系统? 题目 career cup 上有, 不过没想到他会问这个, ECC 又不是做应用的. 刚好是我的弱项, 一直做 research, 对算法和语言还算了解, 对应用系统和设计那是一片空白. 面的是一塌糊涂. 有要面 amazon 的参考一下. Windows Live Experience 组: 1. 美国人. 上来随便聊聊, 然后出了个 coding 题目一个数组, 找出第一个重复的数我给了三种方法, 最后用 hash 写的, 然后问 test case 之类的

2. 印度人上来问我会什么 C# 还是 C++, 我说 C# 会的多一些. 然后他上来问了四五个简单的语法问题. 正好我还都会, 心理还窃喜着呢. coding 也很简单, 给一个 01 字符串, 转化成整数. 写完后 test case. 第二个题目是两个函数互相调用, 无限循环了, 然我找出毛病, 问怎么解决. 然后午饭跟这个印度人吃, 随便聊聊. 就过了

3. 欧洲人, 不知道哪国. 女的, 人很好, 跟她聊的最开心. coding 题目是个没见过的, double bytes string 实现 delete 键功能. 这个比较难解释, 她开始也跟我解释了很长时间. 就是删除字符的时候如何确定是删一个字节还是删两个字节的. 我给出算法, 然后她有提示有哪些特殊情况要考虑, 也做出来了. 然后她就问我给一个一般的 application 如何测试, 又随便说了一通, 结束了

4. 美国人 senior test lead coding 很简单, 给一个句子, 把里边所有的单词自身 reverse 然后给我看他们的产品, 问我怎么测试. 聊的也挺好

5. 欧洲人 director 面到这个人的时候, 我都快累趴下了, 都不想面了, 实在是累. 心理还想着, offer 拿不拿得着无所谓, 别把老子给累死了. (看来真得努力锻炼身体, 不然面试都挺不住) 题目也很简单, 找 1-100 的素数. 我就给了最简单的方法, 然后我说要 check 一些边界情况, 他说不用了. 然后让我做到他的椅子上, 打开 excel, 问我怎么测设置字体这个 feature. 说完了问我有什么问题没有

给你一本 dictionary, 任意给你七个 letters, 让你找出包含这七个字母的、最长的单词. 条件: 可以 pre-processing, 这样每次给你不同的 letters 时, 可以 very efficient 我当时想了好久也没给出完整答案... naive 的解法当然就是每次 scan dictionary, 每次 $O(n)$... pre-processing 那就是建 index, 但 index 怎么建? 怎么操作?

Chapter 23

FULL TIME SDE。

1. REVERSE LINKLIST.

2. 给了 N 个数，值域 $[1, N-1]$ ，如何找出第一个重复的数

3. 算 POLYNOMIAL，比如 $5x^4 + 6x^3 - 7x^2 - 8 = ?$

4. 给一个 URL，如何把空格这种字符转换成 %20 这种

5. 给一个 LINKLIST, VALUE 的指针指向其他 NODE，复制他今天 RECRUITER 发邮件通知给 OFFER 了，漫长的两周。。。希望收回 OFFER 这种事不要发生。。

http://www.mitbbs.com/article/JobHunting/31387663_3.html

Chapter 24

ftware Development Engineer

24.1 问候之后，首先问了一下我的 **research**，让我具体的阐述我提到过的算法然后 **OOD** 的一道题，其实不难，但我感觉自己答得不好

have a furniture class, some child classes like table, chair, etc. they want to extend the class hierarchy, as there are wood table, steel table, wood chair, steel chair, and so on.

我首先给出 class + interface 的 design:

```
furniture (table, chair, ...)
table ( woodtable extends table implements wood, steeltable extends table
implements steel)
chair ( similar as table)
```

然而 interviewer 立刻指出这样的话，如果要加 fire, 或者和重量有关的 functions 时，会有 code duplication (因为 wood 可燃，steel 不可燃, assuming fire function is defined in wood and steel interfaces). 我最后说那把 wood 和 steel 也改成 class (c++ multi-inheritance), 这样可以 avoid code duplication. Interviewer 又问那么如何实现 woodsteeltable? 我说就 inherit woodtable and steeltable. 自己对这个问题总体感觉不好。大家有什么好的 design?

24.2 后面的问题比较简单:

24.2.1 given a deck of cards, how to shuffle it?

24.2.2 having a web application, front-end, middle layer and database.
How to scale database to accommodate increasing traffic?

http://www.mitbbs.com/article/JobHunting/31471911_3.html

24.3 Consider a function which, for a given whole number n , returns the number of ones required when writing out all numbers between 0 and n . For example, $f(13)=6$. Notice that $f(1)=1$. What is the next largest n such that $f(n)=n$?

http://www.mitbbs.com/article/JobHunting/31471823_3.html

24.4 请解释 garbage collection? 在 garbage collection 中, 对 circular reference 的你怎么办?

http://www.mitbbs.com/article/JobHunting/31470831_3.html

24.5 栽在一道编程题上:

Find a longest increasing subsequence in an integer array. 问问题的人要求朋友拿出 $O(n\log(n))$ 的算法, 但朋友只给出了 $O(n^2)$ 的 dynamic programming 的方法。其实我觉得给出 dynamic programming 算法足够进入下一轮了。那个 $O(n\log(n))$ 的算法好歹也值当年一篇 paper, 而且貌似不是那么直观。电面就想出来不容易。不过多半是我段位不够, 还不够 Google 的要求。或者朋友的 dynamic programming 其实错了 (这道题要倒过来找, 稍微绕一点点)。

http://www.mitbbs.com/article/JobHunting/31473303_3.html

24.6 Given n points in the form $(x_1, y_1, z_1) \cdots (x_n, y_n, z_n)$, find the k closest points to the origin.

Given the same points as above, find the K closest points to each other.

http://www.mitbbs.com/article/JobHunting/31473965_3.html

24.7

1. one array filled with numbers from 1 to N , but one number is missing. what's the most efficient way to find the missing item? what about two or more numbers are missed?
2. find the repetative chars in a string and delete them
3. find the binary tree from its preorder and inorder traversal

http://www.mitbbs.com/article/JobHunting/31474331_3.html

24.8. AND(5) GENERATES A RANDOM INTEGER NUMBER BETWEEN [1, 5], HOW DO YOU GENERATE

24.8 and(5) generates a random integer number between [1, 5], how do you generate a random integer number between [1, 7] when you can only call rand(5)?

http://www.mitbbs.com/article/JobHunting/31476251_3.html

24.9 Given an integer, print the closest number to it that is a palindrome

input: 1224

return: 1221.

http://www.mitbbs.com/article/JobHunting/31477969_3.html

24.10 Given a value and a binary search tree. Print all the paths(if there exists more than one) which sum up to that value. It can be any path in the tree. It doesn't have to be from the root. 我理解是这个 path 可以是其中任意一截, 不用包括头尾

http://www.mitbbs.com/article/JobHunting/31478003_3.html

24.11 given a character string, print the number of occurrence of each character in order. ie. if the string is "ceabcw", then you should print something like: a 1 b 1 c 2 e 1 w 1. she asked the possible data structure to approach. I gave array, hashtable, and BST. she asked me to use BST, and using no recursive. Also how to handle unicode.

然后问了一些测试题, 让我测试她们的一个产品。细节忘了, 总之她对我不满意。我也觉得基本没戏了。

第二个是个印度人。编程题: given a matrix (assume it is a bitmap), print all cells that is on. 做的不好。后来问了一些测试题。

第三个是个白人。开始问测试的问题, 回答得一般。因为觉得已经没有戏了, 所以也不大有精神。编程题很简单, 是实现阶乘。不过有个问题没有考虑到, 就是 overflow 怎么处理。总之非常惨, 第一次面试这么惨。也请给位说说自己的想法怎样解答那些问题

http://www.mitbbs.com/article/JobHunting/31481069_3.html

两个玩家, 一堆石头, 假设多于 100 块, 两人依次拿, 最后拿光者赢, 规则是

1. 第一个人不能一次拿光所有的

2. 第一次拿了之后，每人每次最多只能拿对方前一次拿的数目的两倍

求先拿者必胜策略, 如果有的话

http://www.mitbbs.com/article/JobHunting/31482015_3.html

24.12 题目:

从一个 string 变到另一个，比如”study”->”world”（字数相等），要求

1. 每次变一个字母
2. 每次改变后的 string 必须是一个词典里面能查到的英语单词，比如你不能把 study 变成 atudy

http://www.mitbbs.com/article/JobHunting/31482527_3.html

Chapter 25

Website Links

http://www.mitbbs.com/article_t/JobHunting/31342084.html
http://www.mitbbs.com/article_t/JobHunting/31347263.html
http://www.mitbbs.com/article_t/JobHunting/31347264.html
http://www.mitbbs.com/article_t/JobHunting/31348374.html
http://www.mitbbs.com/article_t/JobHunting/31348607.html
http://www.mitbbs.com/article_t/JobHunting/31350186.html
http://www.mitbbs.com/article/JobHunting/31354737_3.html
http://www.mitbbs.com/article/JobHunting/31356298_3.html
http://www.mitbbs.com/article/JobHunting/31368921_3.html
http://www.mitbbs.com/article/JobHunting/31373641_3.html
http://www.mitbbs.com/article_t/JobHunting/31376671.html
http://www.mitbbs.com/article/JobHunting/31383513_3.html
http://www.mitbbs.com/article/JobHunting/31387661_3.html