前一阵微软面试两次。第一次on-site fail, 第二次过了。On-site照例先HR谈话，说我  
们今天只安排3轮，别以为到中午走人就是fail等等假话套话；3轮过后，如果过不了就走  
人。留下来的lunch interview, 吃完饭和同一个面试官再谈20～30分钟，然后再来一个  
最高的manager面试。下面是面试技术题目的总结。其他问题包括介绍一下自己，怎么处  
理同事间矛盾，对我们公司会有什么贡献，等等，都是套路。  
  
电话面试  
1)      heap/stack的概念，区别？  
2)      virtual function? pure virtual function? virtual desctruction? 概念？  
何时用？  
3)      Base class 和derived class之间的赋值(比如A是base, B是derived, A p1 =   
new B, 行不行？反过来呢？)   
4)      class的概念，为什么要用OOP? protect/public/private，什么时候用它们？  
5)      设计一个parking lot的class  
6)      线程之间怎么通讯？什么是critical section/semaphore/mutax，区别？  
7)      process和thread的区别？

**8)      怎样检测link-list中的circle?**

#include<iostream>

using namespace std;

struct ListNode

{

int data;

ListNode \*next;

};

class LinkList

{

ListNode \*head; //指向链表的头指针

int count; //结点数目

ListNode \*interP; //存放制造环时的中间节点

public:

LinkList();

LinkList(int n); //手动创建一个含有n个元素的链表

~LinkList();

void ReverseList(); //逆序链表

void PrintList(); //打印当前的链表

bool IsCircle(); //判断链表中是否有环

//制造环——方法是将链尾指向链中第n个节点后面(条件:count>n)

void GeneCircle(int n);

void DegeneCircle(int n); //拆环，防止不能析构

};

LinkList::LinkList()

{

head = NULL;

count = 0;

}

LinkList::LinkList(int n)

{

count = n;

head = new ListNode;

ListNode \*p = head;

for(int i=0;i<n;i++)

{

ListNode \*newPtr = new ListNode; //新建一个结点

cout<<"Input Node "<<i+1<<": ";

cin>>newPtr->data; //新节点赋值

p->next = newPtr; //当前结点指向newPtr

p = p->next; //p变为当前位置的下一个位置

}

p->next =NULL;

interP = NULL;

}

LinkList::~LinkList()

{

ListNode \*p = head->next;

ListNode \*tmpPtr;

while(p != NULL)

{

tmpPtr = p->next;

delete p;

p = tmpPtr;

}

delete tmpPtr,p,head,interP;

p = tmpPtr = head = interP = NULL;

cout<<"Deconstructor!"<<endl;

}

void LinkList::ReverseList()

{

ListNode \*pCurr = head->next; //pCurr指向待逆序链表的表头结点(即p指向第一个节点)

if(pCurr == NULL || pCurr->next == NULL) //只有头指针或者只有一个节点

return ;

head->next = NULL; //清空原链表(head仍为逆序后的表头结点)

//将原链表中的结点依次进行逆序链接

while(pCurr != NULL)

{

ListNode \*pTmp = pCurr->next; //pCurr指向待处理的结点,pTmp存放pCurr的下一个节点

//将pCurr插入到head和第一个节点之间(即将pCurr作为新的第一个结点)

pCurr->next = head->next; //将新的待处理的节点pCurr插入到原先第一个节点(head->next)之前

head->next = pCurr; //头结点指向新的节点pCurr

pCurr = pTmp; //下一个待处理结点

}

}

void LinkList::PrintList()

{

if(count == 0)

{

cout<<"List is empty!"<<endl;

return;

}

ListNode \*p = head->next;

cout<<"data of the List is:"<<endl;

while(p != NULL)

{

if(p->next)

cout<<p->data<<"-->";

else

cout<<p->data;

p = p->next;

}

cout<<endl<<"length of the list is: "<<count<<endl;

}

bool LinkList::IsCircle()

{

ListNode \*p = head->next;

ListNode \*q = head->next;

while(p->next && q->next)

{

p = p->next;

if(NULL == (q = q->next->next))

return false;

if(p == q)

return true;

}

return false;

}

void LinkList::GeneCircle(int n)

{

if(n >= count)

{

cout<<"can not generate a circle!"<<endl;

return;

}

ListNode \*temp,\*p = head->next; //p将要指向第n个（队尾）元素

for(int i=1;p->next;i++)

{

if(i == n) //找到第n个元素

{

temp = p; //先用temp暂存第n个元素的指针p

break;

}

p = p->next; //p不断指向后面的元素

}

interP = p->next;

p->next = temp; //p的下一个元素指向temp（暂存原先p指向的元素），这样就形成了一个环

}

void LinkList::DegeneCircle(int n)

{

ListNode \*p = head->next;

for(int i=1;p->next;i++)

{

if(i == n)

break;

p = p->next;

}

p->next = interP;

}

int main()

{

int N,pos;

cout<<"Input length of the List: ";

cin>>N;

LinkList ls(N);

ls.PrintList();

ls.ReverseList();

ls.PrintList();

ls.ReverseList();

ls.PrintList();

cout<<"Input the position where to generator circle:";

cin>>pos;

ls.GeneCircle(pos);

if(ls.IsCircle())

{

cout<<"Circle Linklist!"<<endl;

//必须使用这个函数恢复链为无环状态，否则因为环的存在，

//链表的析构函数不能正常退出，导致程序死循环

ls.DegeneCircle(pos);

}

else

cout<<"Non-Circle LinkList!"<<endl;

return 0;

}

**9)      写一个reverse string的代码，怎么测试？(网上用livemeeting的面试)**

**void reverse\_str(char \*ch) /\*使用中间变量\*/**

**{**

**int len;**

**int i;**

**len = strlen(ch)-1;**

**char ctemp;**

**for(i = 0; i < len-i; i++)**

**{**

**ctemp = ch[i];**

**ch[i] = ch[len-i];**

**ch[len-i] = ctemp;**

**}**

**ch[len+1] = 0;**

**}**

**-------------------------------------------------------**

**void reverse\_str2(char \*ch) /\*不用中间变量\*/**

**{**

**int len;**

**int i;**

**len = strlen(ch)-1;**

**char ctemp;**

**for(i = 0; i < len-i; i++)**

**{**

**ch[i] = ch[i] ^ ch[len-i];**

**ch[len-i] = ch[i] ^ ch[len-i];**

**ch[i] = ch[i] ^ ch[len-i];**

**}**

**ch[len+1] = 0;**

**}**

**10)     写代码，计算一段plain text中有多少单词，以及计算出现频率最高的单词(网  
上用livemeeting的面试)。怎么测试？**

#include <stdio.h>

#include <stdlib.h>

#include <ctype.h>

#include <string.h>

typedef struct Word {

char w[20];

int k;

struct Word \*next;

}pWord;

int main(int argc, char \*argv[])

{

FILE \*fp = fopen("input.txt","r");

struct Word \*Head = NULL;

while (!feof(fp)) {

char \*p = (char \*)malloc(20\*sizeof(char));

fscanf(fp, "%s", p);

if(Head == NULL){

struct Word \*temp = (struct Word \*)malloc(sizeof(struct Word));

strcpy(temp->w, p);

temp->k = 1;

temp->next = NULL;

Head = temp;

} else {

struct Word \*pp = Head;

while (pp != NULL) {

if (strcasecmp( pp->w, p) == 0){

++pp->k;

break;

} else {

pp = pp->next;

}

}

if (pp == NULL){

struct Word \*temp = (struct Word \*)malloc(sizeof(struct Word));

strcpy(temp->w, p);

temp->k = 1;

temp->next = Head;

Head = temp;

}

}

}

struct Word \*q = Head;

while (q != NULL) {

printf("%s ", q->w);

printf("%d\n", q->k);

q = q->next;

}

return 0;

} /\*------end of main------\*

11)     A. Write a program in C which takes a string and tests whether any   
permutation of its letters could be a palindrome. Optimize for speed. Make   
sure your code handles all error conditions. Example : if input string is "  
LIILR" then your program should output TRUE. Since one of its permutations is   
LIRIL which is a palindrome. B. How would you change the above program if you   
also had to print out the palindrome. Example : if input string is "LIILR"   
then your program should output TRUE, LIRIL., 限时45min, email寄回。  
12)      Deadline邻近，程序出现重大问题，要么是没时间修改，要么是程序员拒绝修  
改，怎么办？  
13)     Deadline邻近，没有时间做足够测试，怎么办？如果连排在priority list中的  
项目都没时间测试，怎么办？  
14)     boost pool memory概念  
15)     java monitor概念  
  
Onsite (基本所有代码都要说出复杂度):  
1)      给一段代码，找错误  
**2)      什么是deadlock, 怎么避免？**

进程无限期等待不可能得到的资源，就是死锁。  
　 2.死锁的防止（要求达到“简单应用”层次）。   
　　（1）系统出现死锁必然同时保持的四个必要条件。   
　　（2）死锁的防止策略：静态分配、按序分配、抢夺式分配。   
　　 3.死锁的避免（要求达到“简单应用”层次）。   
　　（1）安全状态。   
　　（2）区分死锁的避免与死锁的防止。   
　　（3）银行算法是怎样避免死锁的。   
　　 4.死锁的检测（要求达到“领会”层次）。   
　　（1）什么是死锁的检测。   
　　（2）怎样实现死锁的检测。   
　　（3）检测到死锁后的恢复工作。

**3)      代码：Reverse一个link-list， 怎么测试？**

class Node

**{**

public**:**

Node**(**int value**) :** value**(**value**),** next**(**NULL**) {}**

public**:**

int value**;**

Node**\*** next**;**

**};**

Node**\*** reverseList**(**Node**\*** head**)**

**{**

Node**\*** newList **=** NULL**;**

Node**\*** current **=** head**;**

while **(**current**)**

**{**

Node**\*** next **=** current**->**next**;**

current**->**next **=** newList**;**

newList **=** current**;**

current **=** next**;**

**}**

return newList**;**

**}**

**4)      代码：link-list按照value合并/排序，怎么测试？**

#include <stdio.h>

#include <stdlib.h>

struct node {

int number;

struct node \*next;

};

/\* add a node to the linked list \*/

struct node \*addnode(int number, struct node \*next);

/\* preform merge sort on the linked list \*/

struct node \*mergesort(struct node \*head);

/\* merge the lists.. \*/

struct node \*merge(struct node \*head\_one, struct node \*head\_two);

int main(void) {

struct node \*head;

struct node \*current;

struct node \*next;

int test[] = {8, 3, 2, 6, 1, 5, 4, 7, 9, 0};

int i;

head = NULL;

/\* insert some numbers into the linked list \*/

for(i = 0; i < 10; i++)

head = addnode(test[i], head);

/\* sort the list \*/

head = mergesort(head);

/\* print the list \*/

printf(" before after\n"), i = 0;

for(current = head; current != NULL; current = current->next)

printf("%4d\t%4d\n", test[i++], current->number);

/\* free the list \*/

for(current = head; current != NULL; current = next)

next = current->next, free(current);

/\* done... \*/

return 0;

}

/\* add a node to the linked list \*/

struct node \*addnode(int number, struct node \*next) {

struct node \*tnode;

tnode = (struct node\*)malloc(sizeof(\*tnode));

if(tnode != NULL) {

tnode->number = number;

tnode->next = next;

}

return tnode;

}

/\* preform merge sort on the linked list \*/

struct node \*mergesort(struct node \*head) {

struct node \*head\_one;

struct node \*head\_two;

if((head == NULL) || (head->next == NULL))

return head;

head\_one = head;

head\_two = head->next;

while((head\_two != NULL) && (head\_two->next != NULL)) {

head = head->next;

head\_two = head->next->next;

}

head\_two = head->next;

head->next = NULL;

return merge(mergesort(head\_one), mergesort(head\_two));

}

/\* merge the lists.. \*/

struct node \*merge(struct node \*head\_one, struct node \*head\_two) {

struct node \*head\_three;

if(head\_one == NULL)

return head\_two;

if(head\_two == NULL)

return head\_one;

if(head\_one->number < head\_two->number) {

head\_three = head\_one;

head\_three->next = merge(head\_one->next, head\_two);

} else {

head\_three = head\_two;

head\_three->next = merge(head\_one, head\_two->next);

}

return head\_three;

}

5)      代码：设计一个vector(矢量)的class

6)      代码：reverse一段明文中的各个单词, 怎么测试？

7)      代码：程序接受command line input, 比如说程序是mycalc.exe, 用户运行”  
mycalc 3+2\*3\*3-1”，输出答案。假设: 1. 没有除法； 2. 用户输入正确(不会出现“  
mycalc 4+-5”类似情况)；3. 只考虑integer。 代码写完当场输入计算机测试

8)      4x4的正方形，有一些空格已经填入数字(1~4)，条件：每行每列以及每个2x2的  
小矩形(只考虑边缘4个小矩形)，1~4四个数字只能出现1次。设计算法。  
9)      有个party, 请了个人过来演讲，这个人不认识听众的任何一人，任何听众都认  
识他。现在演讲者已经混入人群，你可以问任何一人”你认识哪些人？”(当然不能问“  
谁是演讲者”)，优化算法，最快找到这个演讲者  
10)     写代码的时候，往往会给array定一个max\_number, 如果现实中有可能出现高于  
这个数字，怎么测试？比如找出邻近的wifi网络数，max\_number再怎么大，总有可能超出  
，怎么测试呢？而且比如这种情况还很难模拟(很难在现场设几百个网络吧)，怎么测试？  
不准用动态数组。

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.

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

编程题  
given a character string, print the number of occurence of each charcater in  
order. 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.  
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

Given a document of text, return the 10 most frequently occurred words in   
the text.

比如说叫design一个parking lot, elevator，online shopping system.  
  
这些问题应该怎么approach,主要是考察什么方面的知识和能力呢？ OOP和design patt  
ern?那么是不是说，一定要考虑到composition,还是inheritance,各自的优缺点要说下  
？尽量避免coupling之类的?要不要说个design pattern之类的，比如，singleton,或者  
factory.

Q2) 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.  
  
好吧，我承认，给我的人说他也不记得了，当时就是挺confused的。  
  
来另一个吧：  
  
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.

很简单，就是check 一个array里面有没有两个元素和为sum,  
  
我用了两种方法，一个是sort之后，keep两个指针i,j一头一尾,相加如果小于sum就前移  
，否则就后移。O(nlgn) time  
  
另一种方法是hash table.其中一个trick就是要确信没有自己加自己。这个是唯一一个  
主意点吧。O(n) with O(n) space

1. sort stack using only pop(), top(), push(), isEmpty(),isFull(). Do not   
   use any auxiliary stacks or arrays.  
   感觉不用辅助空间做不到啊，这题是真的有很巧妙的方法？还是玩玩文字游戏--比如用  
   linked list然后宣称它不是stack也不是array...  
     
   2. Given a set of coin denominators, find the minimum number of coins to   
   give a certain amount of change  
   贪心和DP,回溯法应该都能做的，无非就是状态空间搜索，但是我想到的都是伪多项式  
   的算法，感觉这道题应该能利用某些数论的知识在多项式时间解决

. 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  
  
3. 手写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  
  
4. 逻辑问题：八个水罐称重  
  
5. 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）  
  
6. 反馈问题  
  
第二波一个项目经理  
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。  
  
然后就是一个智力问题，三个囚犯黑帽白帽.

实常见，好象没见过好解。  
**Write a program to evaluate a simple mathematical expression like 4 + 2\*a/b**   
– 3  
基本idea就是要用stack来记录原来的operand, 然后新的operand和在stack上的  
operand进行比较。可能要计算中间值（一旦precedence确定），然后再push 进stack.  
但是具体怎么操作觉得没想明白，哪位大写指点一下？？多谢！！

1. Given a set of n points, find the line that intersects the most number of   
   points  
   想不出有比n^2更好的办法  
     
   2。Given n integers, find two of them that has max xor result  
   如果假定a xor b takes time len(a)+len(b)), then the naive way will take time  
   n^2 \* avglen  
   my idea: first align all the numbers to same length, say it is x, then   
   distribute by most significant bit, as 1, 0, to two array one and zero. we   
   will keep dividing one into oneone, onezero, zero into zerozero, zeroone.  
   then to achieve the max xor, we must pair oneone with zerozero, zeroone with  
   onezero if possible. and keep doing this. the expected time should be nlgn\*  
   x if groups of words with prefix 00, 11, 01, 10 have the same size.

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(logn) 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

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。  
eg:   
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  
  
2. 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  
  
3. Design a fight ticket booking system.  
  
4. 老板说网站很慢怎么办？  
   老板说数据库很慢怎么办？

1 什么是标准方差  
2 什么是标准error  
3 operator new什么功能  
4 virtual function干什么用用的, 为什么要有virtual function  
5 dynamic binding如何实现的  
6 如何debug 程序, 如果是多线程的如何debug  
7 malloc和calloc的区别  
8 什么是abstraction, 如何实现abstraction  
9 写一个程序计算fabonaci, 如果不recursive怎么写, 如何继续优化,我最后给的答案是用两个变量就行了.

1. phone interview with 2 people in R&D, Dec 2009  
   1) how to find 1 missing number from 0 to N in an array of N numbers.  
   2) brainteaser, 5 jar problems.  
   **3) how to calculate sqrt(N) without using sqrt function.**

|  |  |
| --- | --- |
| #include <iostream> |  |
| 02 | #include <math.h> |  |

|  |  |
| --- | --- |
| 03 |  |
| 04 | using namespace std; |  |

|  |  |
| --- | --- |
| 05 |  |
| 06 |  |

|  |  |  |
| --- | --- | --- |
| 07 | float sqroot(float m) |  |
| 08 | { |

|  |  |
| --- | --- |
| 09 | float i=0; |
| 10 | float x1,x2; |

|  |  |  |
| --- | --- | --- |
| 11 | while( (i\*i) <= m ) |  |
| 12 | i+=0.1; |

|  |  |
| --- | --- |
| 13 | x1=i; |
| 14 | for(int j=0;j<10;j++) |  |

|  |  |
| --- | --- |
| 15 | { |
| 16 | x2=m; |  |

|  |  |
| --- | --- |
| 17 | x2/=x1; |
| 18 | x2+=x1; |

|  |  |
| --- | --- |
| 19 | x2/=2; |
| 20 | x1=x2; |

|  |  |
| --- | --- |
| 21 | } |
| 22 | return x2; |  |

|  |  |
| --- | --- |
| 23 | } |
| 24 |  |  |

|  |  |  |
| --- | --- | --- |
| 25 | int main() |  |
| 26 | { |

|  |  |  |
| --- | --- | --- |
| 27 | cout<<"Enter a Number:"; |  |
| 28 | int no; |

|  |  |
| --- | --- |
| 29 | cin>>no; |
| 30 | cout<<"Square Root using sqroot()= "<<sqroot(no)<<endl |  |

|  |  |  |
| --- | --- | --- |
| 31 | <<"Square Root using sqrt()  = "<<sqrt(no); |  |
| 32 |  |

|  |  |  |
| --- | --- | --- |
| 33 | return 0; |  |
| 34 | } |

1. Binary search tree problem.  
     
     
   3. onsite interview, Jan 2010  
   1st meet 2 people in R&D  
   1) train, tunnel, people escaping problem  
   2) 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.  
   3) 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.  
   4) 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.  
   5) 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  
   Talk a lot about my research.  
   Ask one question, how to find the first unique number in an array of byte.   
   and write a code to realize it.

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<=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 >=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+logN-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LogN，  
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 LogN 时间内找出来，一个hint  
是利用 5^k（和log\_5）划分问题  
题目6. 任务分配，假设有N个任务，每个任务需要W\_i工作量，M个人，每人每天能做工  
作量w\_i，如何安排工作，使得所有工作能最快完成。这个问题其实更像一个开放性问  
题，因为一个合理的贪心策略，最后的结果跟最优结是很接近的(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trival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球矩阵的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。

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

Given two arrays A [1..n] and B[1..m], find the smallest window in A that co  
ntains all elements of B. That is, find a pair <l,k> such that A[l..k] conta  
ins B[1..m]  
  
For example, given A = 3,1,5,7,3,5,2 and B = 5,3 then the smallest window is  
[3,5].

. 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. 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）  
  
3. 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。   
  
4. 给中序后续，构建树。

Brain teaser:

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。  
  
2. heard on the street 上的男人出轨题，简单逻辑推理。  
  
3. 这个没答上来，后来给了提示做出来了，但是回头想想还是不对。上来请教一下  
。  
  
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。

. 写atoi函数，

// Test2.cpp : Defines the entry point for the console application.  
//

#include "stdafx.h"  
#include <cassert>  
#include <iostream>  
#include <string.h>  
#include <stdio.h>  
#include <vld.h>

using namespace std;

double Strtod(char \*str, char \*\*endstr)  
{  
    double num1 = 0.0;  
    double num2 = 0.0;  
    double point = 0.1;  
    int sign = 1;

    int len = strlen(str) + 1;  
    \*endstr = new char[len];  
    memset(\*endstr, 0, len);

    if (\*str == '-')  
    {  
        sign = -1;  
        ++str;  
    }

if (!isdigit(\*str))  
    {  
        strcpy(\*endstr, str);  
        return 0.0;  
    }  
    while (\*str && isdigit(\*str))  
    {  
        if (\*str == '.')  
        {  
            ++str;  
            while (\*str && isdigit(\*str))  
            {  
                num2 += point \* ((\*str) - '0');  
                point \*= 0.1;  
                ++str;  
            }  
            strcpy(\*endstr, str);  
            break;  
        }  
        else  
        {  
            num1 = 10 \* num1 + \*str - '0';  
            str++;  
            if (!\*str || !isdigit(\*str))  
            {  
                strcpy(\*endstr, str);  
                break;  
            }  
        }  
    }  
      
    return (num1 + num2) \* sign;  
}

int Atoi(const char \*pstr)  
{  
    int sign = 1;  
    int num = 0;

    while (\*pstr == ' ' || \*pstr == '\t')  
    {  
        pstr++;  
    }

    if (\*pstr == '-')  
    {  
        sign = -1;  
        pstr++;  
    }

while (\*pstr)  
    {  
        if (\*pstr >= '0' && \*pstr <= '9')  
        {  
            num = 10 \* num + \*pstr - '0';  
        }  
        else  
        {  
            return num \* sign;  
        }  
        pstr++;  
    }  
    return (num \* sign);  
}

double Atof(const char \*pstr)  
{  
    double sign = 1.0;  
    double num1 = 0.0;  
    double num2 = 0.0;  
    double point = 0.1;

    while (\*pstr == ' ' || \*pstr == '\t')  
    {  
        pstr++;  
    }

    if (\*pstr == '-')  
    {  
        sign = -1;  
        pstr++;  
    }

    while (\*pstr)  
    {  
        if (\*pstr == '.')  
        {  
            pstr++;  
            while (\*pstr >= '0' && \*pstr <= '9')  
            {  
                num1 += point \* (\*pstr - '0');  
                point \*= 0.1;  
                pstr++;  
            }  
        }  
        else if (\*pstr >= '0' && \*pstr <= '9')  
        {  
            num2 = num2 \* 10 + \*pstr -'0';  
        }  
        else  
        {  
            return (num1 + num2) \* (sign);  
        }  
        pstr++;  
    }  
    return (num1 + num2) \* (sign);  
}

int main( void )  
{  
    char str[] = "    -1234565kljh";  
    int num = Atoi(str);  
    cout<< num << endl;

    char pstr[] = "    112adfaf3  43224.569877aa";  
    double n = Atof(pstr);  
    printf("%6.8f\n", n);

    return 0;

}

2. 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？  
  
3. 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:-)   
  
4. 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。  
   测一个记事薄软件。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。  
  
5. 最后是hiring manager，问了一些behavior问题，然后打开一个网站，问怎么测试。

在onsite面试中实际遇到的。  
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; }

2. 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  
  
3. 手写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  
  
4. 逻辑问题：八个水罐称重  
  
5. 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）

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

1. **reverse words in a sentence，使用如下函数。  
   char\* reverseWord(const char\* str)**
2. #include <iostream>
3. using namespace std;
4. void rev(char \*l, char \*r);
5. int main(int argc, char \*argv[])
6. {
7. char buf[] = "the world will go on forever";
8. char \*end, \*x, \*y;
9. // Reverse the whole sentence first..
10. for(end=buf; \*end; end++);
11. rev(buf,end-1);
12. // Now swap each word within sentence...
13. x = buf-1;
14. y = buf;
15. while(x++ < end)
16. {
17. if(\*x == '\0' || \*x == ' ')
18. {
19. rev(y,x-1);
20. y = x+1;
21. }
22. }
23. // Now print the final string....
24. printf("%s\n",buf);
25. return(0);
26. }
27. // Function to reverse a string in place...
28. void rev(char \*l,char \*r)
29. {
30. char t;
31. while(l < r)
32. {
33. t = \*l;
34. \*l++ = \*r;
35. \*r-- = t;
36. }
37. }

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

void csort(int array[], const int end,

const int max, const int min)

{

int i;

const int range = max-min+1;

int count[range+1],

scratch[end];

for(i=0; i<range+1; i++)

count[i] = 0;

/\* Set the value of count[i] to the number of

\* elements in array with value i+min-1. \*/

for(i=0; i<end; i++) {

int c = array[i]-1-min;

count[c]++;

}

/\* Update count[i] to be the number of

\* elements with value less than i+min. \*/

for(i=1; i<range; i++)

count[i] + = count[i-1];

/\* Copy the elements of array into scratch in

\* stable sorted order. \*/

for(i=(end-1); i>=0; i--) {

int c = array[i]-min;

int s = count[c];

scratch[s] = array[i];

/\* Increment count so that the next element

\* with the same value as the current element

\* is placed into its own position in scratch. \*/

count[c]++;

}

for(i=0; i<end; i++)

array[i] = scratch[i];

}

3.covert interger number to date string, for example, 20090130 -> "01/30/  
2009"  
  
说说教训：  
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；最后当面试者提出如果定义一块  
内存  
，如char tmp[2048]，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。  
  
第二道使用couting sort应该就可以。面试者要求描述算法，不需写代码。  
  
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。

为什么对Amazon感兴趣。  
\* 自己最近的Project。  
\* 说出自己会的编程语言并打分（1－5）。  
\* 有没有开发Mobile application的经验。  
\* 几个常见Data structure的Lookup操作的时间复杂度。  
\* HTTP post和get的区别。  
**\* Design Pattern: Singleton, Factory, Lazy initialization。**

class Singleton

{

private:

static Singleton \_instance;

Singleton() {}

~Singleton() {}

Singleton(const Singleton &); // intentionally undefined

Singleton & operator=(const Singleton &); // intentionally undefined

public:

static Singleton &getInstance();

};

// Source file (.cpp)

//

// Static member initialization.

//

Singleton Singleton::\_instance;

Singleton &Singleton::getInstance()

{

return \_instance;

}

/////////////////////////////////////////////////////////////////////////////////////////////////////

class Computer

{

public:

virtual void Run() = 0;

virtual void Stop() = 0;

};

class Laptop: public Computer

{

public:

virtual void Run(){mHibernating = false;}

virtual void Stop(){mHibernating = true;}

private:

bool mHibernating; // Whether or not the machine is hibernating

};

class Desktop: public Computer

{

public:

virtual void Run(){mOn = true;}

virtual void Stop(){mOn = false;}

private:

bool mOn; // Whether or not the machine has been turned on

};

class ComputerFactory

{

public:

static Computer \*NewComputer(const std::string &description)

{

if(description == "laptop")

return new Laptop;

if(description == "desktop")

return new Desktop;

return NULL;

}

};

////////////////////////////////////////////////////////////////

#include <iostream>

#include <string>

#include <map>

using namespace std;

class Fruit {

public:

static Fruit\* getFruit(const string& type);

static void printCurrentTypes();

private:

static map<string,Fruit\*> types;

string type;

// note: constructor private forcing one to use static getFruit()

Fruit(const string& t) : type( t ) {}

};

//definition needed for using any static member variable

map<string,Fruit\*> Fruit::types;

/\*

\* Lazy Factory method, gets the Fruit instance associated with a

\* certain type. Instantiates new ones as needed.

\* precondition: type. Any string that describes a fruit type, e.g. "apple"

\* postcondition: The Fruit instance associated with that type.

\*/

Fruit\* Fruit::getFruit(const string& type) {

Fruit \*f = types[type]; // try to find a pre-existing instance, or std::map'll create one if not found

if (!f) { // if it was created by map automatically, it'll be pointing to NULL

// couldn't find one, so make a new instance

f = new Fruit(type); // lazy initialization part

types[type] = f; // Registering the newly created Fruit in the types' map for later use.

}

return f;

}

/\*

\* For example purposes to see pattern in action

\*/

void Fruit::printCurrentTypes() {

if (!types.empty()) {

cout << "Number of instances made = " << types.size() << endl;

for (map<string,Fruit\*>::iterator iter = types.begin(); iter != types.end(); ++iter) {

cout << (\*iter).first << endl;

}

cout << endl;

}

}

int main(void) {

Fruit::getFruit("Banana");

Fruit::printCurrentTypes();

Fruit::getFruit("Apple");

Fruit::printCurrentTypes();

// returns pre-existing instance from first

// time Fruit with "Banana" was created

Fruit::getFruit("Banana");

Fruit::printCurrentTypes();

return 0;

}

\* Multi-threaded programming, deadlock之类。  
\* 对Unix环境是否熟悉，几个常见命令，ls, ps之类。  
\* Reflection的概念，Java reflection，C++里面是不是有reflection。  
\* 如何实现Garbage Collection。Reference counting的缺点(cycle)，如何解决，JVM  
有没有解决。[http://hi.baidu.com/abusemind/blog/item/abb6774528141f2ccefca35f.html](http://www.google.com/url?q=http%3A%2F%2Fhi.baidu.com%2Fabusemind%2Fblog%2Fitem%2Fabb6774528141f2ccefca35f.html&sa=D&sntz=1&usg=AFQjCNHRjtcUPeD2MZRoWQ2KjjXW7F3E_Q)

\* C++里面virtual destructor的用途，于一般virtual function的区别。  
\* 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。  
\* 算法设计：一个Galaxy，每个星星用一个三围座标表示，找出离地球最近的1000个

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?
2. 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问题。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 effcient

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

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。  
  
算法/编程：1. 大文件随机sample，one pass.  2. sodoku 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。。。）10. 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。）  
  
  
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share  
memory.何时选thread或process。

。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......）  
6. Int\_0^T W(t) dW(t) （ 一看见这个，贼激动阿，熟悉的ito' s formula）  
7. Stonivich intergral 是啥？ 为什么用Ito's 不用 stonivich? (不知道拼得对不  
对)  
8. 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）。  
  
9。什么是AR model? 啥时候用AR model?  
10. 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)。

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。现在要求你去找出这个名人来  
   。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。

OO设计题，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

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. 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

1When casting an object of a polymorphic class from a base calss 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  
  
  
2.  class A  
{  
  
public:  void f();  
protected: A() {}  
       A(const A&) {}  
};  
  
why are the default and copy constructors declared as protected?  
  
a. to ensure that instance of A can not be created via new by a more derived  
class  
b. to ensure that instance of A can only be created by subclasses of A  
c. to ensure that isntance of A can not be copied  
d. to ensure that A cannot be used as a base class.  
e. to ensure that A cannot be instantiated on the stack  
  
  
3. 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?  
  
a. templates must be class definitions  
b. the template parameters should be separated by commas.  
c. the template definition is missing a pair of braces.  
d. template parameters must be pointer types.  
e. the \* operator has not been defined for T1, T2, and T3.  
  
4. 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 ofr this problem?  
  
a. the destructor may attempt to delete the string literal "default"  
b. the destructor needs to check that the value of buf is not 0.  
c. the class does not allocate a long enough buffer.  
d. the function needs to return Foo& instead of Foo.  
e  the class needs to specify a copy constructor and assignment operator.

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, …, 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。

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++?

给定 X[1..n] and Y[1..m] 两个 arrays，已经sort好了.   
如何找到X <Union> 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？

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

是现场写code的面试。  
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

* 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。

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;

[Coding Q1]: Given an array A, output another array B such that B[k]=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 b  
y 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 evalua  
ting 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

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?  
     
   Some others:   
   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?

什么样的情况下用 virtual deconstructor？  
2. virtual function是如何工作的？virtual table 是如何实现的？  
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？  
4. 类的copy constructor 和 assignment operator "=" 有什么区别？有什么主意事  
项？

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？

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？

1. 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。。。  
     
   2. 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?  
     
   3. Write a function to implement a buffer for DataOutputStream.  
   这个我完全没啥概念了。。。求指点一下~  
     
   4a. How do you write malloc and free to detect memory reference violation?  
   4b. 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。。。  
     
   5. How to implement singleton without using static/global variable?
2. **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**
3. **. 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**
4. 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  
     
   2. For the db connection. min = 10 and max = 40.  
   Will 10 connections be created at server start up time?  
   2.1 If we start with using 13 connections, when all the jobs are done, how   
   many will be kept in the pool? 10 or 13?  
   2.2 If we start with using 45 connections, when all the jobs are done, how   
   many will be kept in the pool? 40 or 45?

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?  
后面的问题比较简单：  
given a deck of cards, how to shuffle it?  
having a web application, front-end, middle layer and database. How to scale  
database to accommodate increasing traffic?

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(nlog(n))的算法

. one array filled with numbers from 1 to N, but one number is missing. wha  
t'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

given a character string, print the number of occurence of each charcater in  
order. 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.

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;  
}

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 “abcabcdfabcdf”要求返回“abcdf”. 因为“  
abcdf”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。

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  
  
5. 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  
  
6. dictionary + 包含字符集合:  
Letter\_Set = "aabbbcd";  
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  
  
7. 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  
  
8. edit distance (misspell correction):  
type a misppell 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<=某给定值的所有misspell的单词(类似BFS的一层一层的算)  
  
9. find a matrix with max area: each row and each column of the matrix must   
be a valid word;  
  
10. 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;

11. solve a crossword puzzle;

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126cents  
   最少需要多少个coin。  
     
   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

c++ and data structure  
\*\* single linked list, find nth from the end  
\*\* Overwriting and Overloading  
\*\* Stack vs. Queue  
\*\* Array of integers, all integers appear even times except one, find the   
one appears odd times. (some following up questions for this one)  
  
OOD  
\*\* 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   
  
\*\*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.

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.  
   2. the gmail page loads very slow. any suggestion for improvement?  
   3. we want to check the number of querys 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?

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？

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

* 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

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
   1. how to find 1 missing number from 0 to N in an array of N numbers.  
      2) brainteaser, 5 jar problems.  
      3) how to calculate sqrt(N) without using sqrt function. Binary search tree   
      problem.  
      4) some behavioral problem. Like, How do you know about BB? Why you wanna   
      work in BB? Why you wanna work in industry?  
        
      3. onsite interview, Jan 2010  
      1st meet 2 people in R&D  
      1) train, tunnel, people escaping problem  
      2) 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.  
      3) 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.  
      4) 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.  
      5) Same 5 jars problem. That is their favorite.

.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

. 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？

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

1.calculate the Depth of Binary Tree  
2.Graph Connectivity(Adjacency List)  
3.An array of integers, only one integer appears odd times, all others   
appear even times, find it  
4.Process Vs. Thread and Memory allocation(stack and heap)  
5.How to check singly linked list is a circular linked list.  
6.Override Vs. Overloading  
7.check a number whether is Power of 2  
8.Given an array of integer and a target number. please find out two   
number that add up to the target.  
9.Factorial and optimization  
10.Design a file system  
11.Check whether a binary tree is BST(if the binary tree is very large,   
you can not simply in-order print all the nodes out.)   
12.Design a Crossroad signal system  
13.implement atoi()   
14.Pattern Matching, if '.' is used as a wildcard, which means '.'can   
represent any character.  
15.Design a game(algorithm): which transform a word to a target word.   
for example: from head to tail, each step, you just can replace one   
character, and the word must be valid.  
16.In a clock, calculate the angle between hour and minute handle  
17.Given a singly linked-list, and a pointer to node, how to delete the   
node.  
18.File external sorting

C: pointer, call by value/pointer, return the pointer of a local variable,  
string manipulations, source code of some important C string subroutines  
(strcpy, strtok, etc), itoa, atoi, static variable and fuction, name   
mangling,  
memory allocation  
http://www.eskimo.com/~scs/C-faq/faq.html  
  
C++: namespace, abstract class, polymorphism, dynamic binding, virtual  
function, virtual destructor, constructur, copy constructor, assigment  
operator,  throw vs exit, C vs C++, Java vs C++, static attributes and  
methods, multiple inheritance vs interface, struct vs class, templates,  
template specialization. 如果你再会bit operations, union就更牛了。  
http://www.icce.rug.nl/documents/cplusplus/  
http://www.parashift.com/c++-faq-lite/  
  
OS 基础 : 常用shell commands, like top, ps, sort, jobs, diff,  
find。Synchronization， deadlock, race condictions, mutual excusive,  
semiphore, signals, multiple processing vs multiple threading。Communication  
between processes or computers, IPC, pipe。  
  
Data stucutures and algorithms:  array, largest N integers, find duplicates,  
missing value. linked list (必考！). sorting algorithms and complexity  
analysis, space vs speed. hash table, array vs linked list, linked list  
traversal, reversion, insertion, deletion and sorting, loop in LL. String  
reversion, remove duplicates, reverse words. Tree traversal, traversal   
without  
recursion, breadth-first traversal, compare two tree, find common ancestor.

Create a command-line program (GUI not necessary) in C++ to do the following:  
•Takes in 3 arguments from the command line:   
    o  Input file name: The input file will contain several rows of comma-  
separated name/age pairs; one pair per line (see below)   
    o  Operation type: one of 3 values specifying what type of operation to   
perform on the entries in the given input file:   
        “n” for sort by name   
        “a” for sort by age   
        “f” for filter out entries whose age is less than 21 (no sorting   
required)   
    o  Output format: one of 2 values specifying how to format the output of  
the operation.  
        “h” for HTML table.   
        “c” for comma-separated values (i.e. the same format of the input   
file)   
•Has one output:   
    o  The output should be the sorted or filtered list of name/age pairs,   
in the output format requested; dumped either to stdout or a file.   
  
Example input file contents:  
Jim, 50  
Abe, 20  
Mike, 33  
John, 21  
Abe, 18

First phone interview:  
1. what's the difference between overloading and overwriting  
2. what is polymophism?  
3. Design pattern--singleton  
4. design elevator system  
5. An array which has some elements (each element appear even number of   
times except one, find the one that occurs odd times)  
6. coding -- convert $134.35 to one hundred thirty-four and thirty-five   
  
Second phone interview  
1. talk about the data structures you know  
2. talk about sorting algorithms you know  
3. Design question: furniture, woodchair, steelchair, woodtable ...  
4. Algorithm:   
   1) Exchange two variables without using a temporary variable  
   2) Find elements in an array that sum up to a given number