

Akuna Capital面经收集，内容来自论坛

Onsite

参加的Akuna Capital Super Day，一天总共十个人左右吧。

第一轮HR，就简单过一遍简历

第二轮technical，设计memory pool的class，做到用户要memory的时候给，一般是用malloc，用户free的时候存在class里面，下次user要的时候直接给以前的。要求generic type。会用到template，copy constructor，constexpr等等。

然后做一个coding challenge，就是写一个battle ship的游戏策略，没玩过的可以去这个网站看看规则一样，<http://www.mathplayground.com/battleship.html>

游戏框架都给你搭好了，你只要写策略就行，怎么打对方。注意下用的是linux，默认只有vim，你可以问问面试官能不能用别的text editor，我看好像有notepad++。写这个好像只有一个半小时写完游戏以后code review，给他过一遍的你的代码，讲思路。

然后完了，攒人品！

当时真的面的很糟糕，不管怎样把经验发给大家，为接下来的M昂塞特攒人品

1.白人胡子叔

设计memory pool,写出完整代码实现，followup处理特殊请求情况，比如说连续请求超大内存后又请求超小内存等等，针对这些情况优化代码。from: 1point3acres.com/bbs
设计路口，纯吹水。从FSM吹到用户体验，最后说好不写代码又让人写socket。。。

2.HR

基本上就是behaviour问题啦

3.坑爹游戏锦标赛

十几个人一起参加的competition.

他编写好了一个server端口，要求面试者实现一个client，来与其它的面试者编写的游戏算法对打，看最后谁来晋级

游戏规则类似battle ship，tricky的点是place ship和shoot这两个方法的算法实现。

给你提供的机子上面只有emacs和vim，gdb可能有....

时间是1个多小时。

当时光优化算法了。。。结果程序跑出来是有问题的。。。

当时就觉得挂定了。。。。

4.白人饮料哥

全程就像在酒馆里和他打屁，他喝一口问一句

先是前一轮coding的代码review,一行一行给他讲算法，以及实现

对于我这样没写出来的人简直是羞耻play

Feel free to modify

经常会challenge你，比如说你为啥用这样的算法
四周墙壁都是白板，就在上面给他画图就行
review完了开始疯狂followup，C++组是编写trading engine的，开始向着方向靠拢。
比如说两个玩家，一个人每秒钟向服务器发一个请求，一个人每秒钟发100个请求，你的程序该怎么改善
比如你其中的某一个STL函数变的特别慢了，你该怎么改善程序
又比如你用了set,还能优化吗？
unordered_set?他说还不够好。
黑人问号，O(1)还不够好
想了想，flat_set,他终于满意了。。。

出去的时候整个人是崩溃的

补充内容 (2016-10-21 23:22):

还有个follow是问你多核CPU怎么优化

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=206893&page=1&authorid=133721>

过了第一个OA之后，是1小时电面，Behavioral 和 tech，OOD，STL对比，都是说了多少遍的东西了，无悬念的过了

Onsite：芝加哥. From 1point 3acres bbs

一共五轮

4个年轻：主要都是地里能查到的题目，我觉得我做的也都还可以，有些因为准备的好，回答的太快了，小哥有点震惊，我觉得还是演技不够. [1point3acres.com/bbs](http://www.1point3acres.com/bbs)

有个简单的coding就是那个打印出

1

23

456

我的方法跟他不太一样，他开始不理解，看了看才明白了，不知道是不是感觉我们不好合作，以后code review麻烦？

最后一个manager，问我那个交易系统的设计缺陷和改进，多线程问题。这个也是老生常谈了，不知道他到底想要听到什么

说来说去他总是不满意，总说：这个想法不错，那我们怎么改进更好呢？

估计是经理决定，其它问题大家都答得差不多，所以Failed

move on

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=198592&highlight=akuna>

Feel free to modify

昨天在芝加哥昂赛特，碰到了两轮新题

常规问题：

聊简历，简历中的project，对金融市场的了解和认识，兴趣点。

二进制数据流求5求结果 (用状态机画出“10010010”这类字符串作为输入的运行结果。这题目非常理论。。)

数据流求中位数

新问题：

白板编程题：根据上面画出来的FSM，写程序实现，并进行优化

设计题：给出一个交易所的价格表格，求需要的最新价格？进而问，如何把消息发送给trader？

这两个问题我都是跟他们沟通半天，不知道他们到底是想要往什么方向回答

没有靠谱思路

最后又聊了金融的知识，和兴趣

转了一圈，26层景色确实不错

竟然还看到好几个吊床，就在程序员那边，有点压力山大。。。

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=201462&highlight=akuna>

真是冰天雪地跪谢 gougou9901 同学昨晚江湖救急发了面经，今天全中了 = = 只有一道新题。我来详细描述一下过程和题目包括我面的新题。

第一轮HR面，聊一下简历，聊一些behavior question，比如why akuna capital, how do you solve the conflicts with your teammate 之类的。

第二轮技术面，两个工程师，面试题只有前两题写代，其它说idea：

- Median of data stream leetcode原题
- 给一个时间（小时，分钟），求时针分针形成的锐角角度。要写代码，楼主忘记在返回前处理锐角钝角了，经过提醒才修复了这个bug，如果挂了估计挂在这里。
- 给一个树，可以insert和look up，怎么保证thread-safe，用reader & writer lock。
- 给一段代码，找出bug，并修复。bug就是一个1的变量一直乘以自己然后永远不会大于1000然后返回。
- 一个linkedlist，有get(int index), insert(int index, int value), append(int value), remove (int value)。四个方法如何优化，分析tradeoff，最后我说和index有关的都没法优化时候他问那如果你能换一个数据结构你换什么，我说arraylist。他就点头了 = =
- 最后一个给你看一页C#代码，基本就是把一个小数的string转换成double输出。举个例子，输入是"4-5",10 输出就是4.5 因为整数部分是4，小数部分是5，然后10是分母，就是 $4 + 5 / 10 = 4.5$ 。认真看代码不懂的直接问他们，就没问题了。然后他们会让你写test case，你自己要写positive和negative的然后要说输出结果。

第三轮Operation Manager面，也是问了简历，问什么时候可以入职，然后让你挑一个项目详细讲一讲。

然后每一轮都留有答疑时间，基本就是这样了，芝加哥今天真是冷成狗 = =

Feel free to modify

~~~~~你们看我面经写的这么详细拜托给点米呀~~~~~

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=145918&highlight=akuna>

---

看到大家都在如火如荼的面各个大公司，地里也都是大公司的面经。发现我自己是非主流了。

上一个在芝加哥的面经，是做hedge fund的公司，叫akuna captial。算是应s兄的要求吧。

这是一家小公司，面之前真不知道这公司做什么。不过onsite见到9个candidates。我们住在palmer house。据说是chicago比较老的一家酒店。第一天到了之后直接被拉去吃饭，聊天。所以各种问software的问题。

整个工程师团队只有9个人。都是有经验的老兵。像是个精英团队。招人招的比较慢。两年里估计工程师也就招了3.4个。

第二天上来写一个battleship game的ai所有frameworks都写好了。两个小时时间里就不停写就好了。

然后就是hr面和技术面。技术面很简单，就是链表有环，tree的最接近某一value的node。秒杀。。。

最后整个工程师团队（就九个人。。。）和我们九个candidates挤在会议室里，让我们讲上午写的ai代码。。。我们大概讲了讲。然后两两pk淘汰直到找出winner。

candidates来自[stanford](#), cmu, uiuc, michigan, vanderbilt还有别的学校，记不清了。

找到winner后，工程师就退散了。我们就飞机回家。。。

---

web

周五刚面完“阿酷娜·卡皮头”这家奇葩公司的昂赛特，揉个面筋给你们吃 公司信息：地理位置：在芝加哥密歇根大道附近，CNA大厦26层，地理位置极佳，从窗口看出去就是密歇根湖，挺养眼的。公司性质：奇葩公司，直接拿自己的钱炒期末、股票、原油、金属...什么来钱炒什么。简单地说，就是黑心倒卖商。目前人员构成：白色人种为主，少量天竺人种，少量华夏人种 web职位性质：他们公司主要做的是实时交易，web攻城狮的作用主要是为数据做可视化。刷以下框架会大量增加与该公司的亲密度，大量增加得到昂赛特的概率。后台框架：python - tornado 可视化：d3js 面试流程：欧诶 -> 店面 -> 昂赛特 网申：这家公司的网申是我见过最TM简单的，填了名字+扔了简历 就行... 欧诶：没啥好说的，地里大多数人都发过了。给WEB的是10道题，TM全是python的。考python继承，dict、list的用法... python什么最基础考什么 电面：没啥好说的，都是概念题 主要是CSS, javascript, python少量 这家公司特别喜欢用non-blocking io, python yield, 和他们的信息系统的代码结构有关... 昂赛特：地里这家公司昂赛特信息比较少，我重点在此说明一下今年的情况 面试的小恩小惠给的还是不错的，包吃包住包机票。酒店就在公司附近的palmer house, Hilton系列。第一天晚上纯吃喝玩乐，带你去了家叫zed451的餐厅吃烤肉。依我的经验，这家店的水准近乎米其林一星，和芝加哥另一家Sepia的肉差不多。坐我旁边的天竺哥哥不

Feel free to modify

出肉，主厨特地给做了全素的一盘，还是很体贴的。第二天面试，就面一上午（TMBB了那么久，终于来重点了）：流程：HR -> interview1(resume+tech) -> coding challenge -> interview2 -> 滚蛋 面试candidates组成：2/3 华夏人种，1/3 天竺人种 \*

HR 老掉牙的BQ。遇到HR老主管的，听说比较惨 \* interview1 问你resume相关的，问我的问题如下：

0. 智障题：让你谈谈用 ajax 加载页面，与后台直接render页面的利弊。这个小哥描述不清楚，他希望的得到的是两种方式对用户体验的利弊。我都跟他说了这两个方法的整个加载过程，那白人小哥还不满意--我TM讲了10分钟的performance，你却跟我讲起了UX。

1. 送分题：对一个在线相册设计RESTful api，对这个系统我有如下功能：a. create 一个相册 b. 对一个相册 insert 照片 c. 对一个相册 delete 照片 d. delete 一个相册 follow up: 相册需要时私人的，其他人无法直接访问我的相册

2. 送命题：打开了公司网站的一个页面，说这个页面是外包给其他公司做的（你们web组干什么吃的）。然后求你如何优化这个页面？不会，呵呵。他指着一堆加载的css说，你看css那么多，合并成一个可以么？ interview 1... GG \*

coding challenge 描述：这是一个桌游，两人对战策略游戏。每人一个10\*10的棋盘，5个棋子。棋子大小(5\*1, 4\*1, 3\*1, 3\*1, 2\*1)

before starting: 每个人将棋子摆在棋盘上，随便怎么摆。横着竖着任意选择，不能斜着。

in the game: 两人交互check对面的棋盘的一个点，比如(b4)，如果棋盘上的那个点被棋子覆盖，则告诉你这个点被覆盖了。如果没被覆盖，告诉你这个点没被覆盖。

胜利条件：一方率先check掉对面覆盖的所有点。

代码：给了你一个python或C++文件，上面有demo怎么去check点，怎么摆棋子。另外给了一个基于socket的server程序，让你验证你写的东西。

要求：写上你的策略，怎么摆你的棋子，怎么去check对方的棋子。就那么简单

时间限制：1小时45分钟左右。没有严格招表

这个coding challenge没有正确答案，但会有一个比赛。每个人的代码和其他人PK，淘汰赛机制。在下不才，拿了那组的第一（捂脸）。

这个算法嘛，当时我用了启发式算法。具体什么的，说起来很麻烦。做完问了一下周围的 \*

interview 2 一个小哥review之前写的代码，求你讲出具体思路，实现。然后follow up说如何改进。最后，期望我能拿到amazon的onsite。

<https://instant.1point3acres.com/thread/194306>

链接: <https://instant.1point3acres.com/thread/194306>

来源: 一亩三分地

---

2. Akuna Capital. more info on [1point3acres.com](https://instant.1point3acres.com)

芝加哥的一个trading firm，看起来想金融公司，其实基本算是个tech公司，因为这类公司主要就是靠优化trading system赚钱，也有很多牛仔裤+拖鞋的码农。我投的是junior developer，以为也是要去开发trading system，不过Onsite的时候recruiter说这个职位主要是做web application，用C#。。。略坑啊。。面完recruiter就是两个人的技术面：

a. Find medium of streaming data (leetcode)

Feel free to modify

b. given hour and minute, return the degree between the hands of the clock

c. 找bug。代码大概如下，bug就是那个time一直是1

```
time = 1;. from: 1point3acres.com/bbs
while (!foundHeartBeat()) {
    if (time > 1000) return false;
    Thread.sleep(time);. From 1point 3acres bbs
    time *= time;
}
```

d. 给你一个二叉树，然后说有很多thread在insert node和读node，问如何线程安全。其实就是reader&writer问题。我给的答案就是最基础的那种(第一个reader读得时候wait writer lock，然后最后一个reader signal writer lock，同时需要一个mutex来给reader\_count++加锁)，说完之后又follow up问如何更加efficient。我不太懂，就扯了下增加writer同时write的数量。他们点点头就没深究。

e. 读一页纸的C#代码，说出主要function，然后写unit test。这题跪了，代码没怎么看懂。。

---

## Phone Interview

---

题目也是做得乱七八糟的 打开codepair 已经有一堆实现好的类和函数 大致就是一群鸟，和鸟的位置

实现如下四个函数：

1. 所有的鸟往leader飞. 鏗磋鏗或滑@1point 3 acres
  2. 所有的鸟往一个方向飞
  - 3.所有的鸟聚拢起来
  - 4.所有的鸟separate
- 鸟是一个array[bird]

```
struct bird{
two-demension position;
two-demension velocity;
};
```

---

"Implement a circular queue, two thread safe"

一看题这肯定是考察多线程啊。然后就开始写，先写一个 mutex类（有库函数，但是我实在想不起来怎么用了。。），然后再写一个cir\_queue 类。然后实现一些函数：constructor(), destructor(), push(), pop(), is\_empty(). . 1point 3acres 璁哄漕



Feel free to modify

以为很简单，然而坑就来了，让分辨mutex和semaphore的区别；new和malloc的区别（还有delete和free）；stack和heap的区别；等等C++的基础知识。很繁琐。感觉面的一般，因为基础知识不太扎实。

不过从他家OA也能看出来，OA全都是C++概念，基本没啥算法。。。求二面。。

第二轮也是oa，不过有个网站，点进去会录像，把答的题录成视频，题目不多，就是有的题时间不太够，一共6题：. [1point3acres.com/bbs](http://1point3acres.com/bbs)

behaviour

1. 你调过的最难调的bug是啥，你最后怎么确定这个bug被fix了
2. 简述你最近的一个teamwork的project，并且你负责了什么

math

3. 从8个men和10个women中选出3个men和3个women有多少种选法. [1point3acres.com/bbs](http://1point3acres.com/bbs)
4. 有两个学生考试，得分差\*分，低分的那个占总数的\*%，问两个人分数多少（数字不记得了，就是个二元一次方程组）

算法

5. hashtable, linkedlist和red black tree，搜索顺序从大到小排序
6. 给了三个简单程序，问时间复杂度是多少

我觉得中国人吃亏，这面试兼考口语，我是挂了，分享希望对大家有帮助。

最challenging的project；

stack和heap的区别和用法；

new和malloc的区别和具体用法；

. 鐳浠汉浜戣浼,涓€€浜†竺錄啣泣

a stream of number, keep track of median；follow up 1: 如果数字都在[0, 100]之间如何改进；

follow up 2: 如果99%数字都在[0, 100]之间如何改进；

解释mutex, 怎么用;. visit [1point3acres.com](http://1point3acres.com) for more.

lock具体如何实现.

感谢地里小伙伴提供的面经，基本都是原题。上来先聊下most challenge project。然后new和malloc区别。第三个问题就是median of datastream。 . [1point3acres.com/bbs](http://1point3acres.com/bbs)  
然后多线程问题。什么是mutex. 我讲到了lock, 面试官让我讲讲lock的底层实现，这里答得不太好。然后就结束了面试，希望能有onsite机会。我把自己总结的以前出现过的面经都传到Google docs, 希望对后来人有所帮助。

1 what is virtual memory

Feel free to modify

In [computing](#), virtual memory is a [memory management](#) technique that is implemented using both hardware and software. It maps [memory addresses](#) used by a program, called [virtual addresses](#), into physical addresses in computer memory. [Main storage](#) as seen by a process or task appears as a contiguous [address space](#) or collection of contiguous [segments](#). The operating system manages virtual address spaces and the assignment of real memory to virtual memory. Address translation hardware in the CPU, often referred to as a [memory management unit](#) or MMU, automatically translates virtual addresses to physical addresses. Software within the operating system may extend these capabilities to provide a virtual address space that can exceed the capacity of real memory and thus reference more memory than is physically present in the computer.

The primary benefits of virtual memory include freeing applications from having to manage a shared memory space, increased security due to memory isolation, and being able to conceptually use more memory than might be physically available, using the technique of [paging](#).

## 2 compiled language && interpreted language

A compiled language is one where the program, once compiled, is expressed in the instructions of the target machine. For example, an addition "+" operation in your source code could be translated directly to the "ADD" instruction in machine code.

An interpreted language is one where the instructions are not directly executed by the target machine, but instead read and executed by some other program (which normally is written in the language of the native machine). For example, the same "+" operation would be recognised by the interpreter at run time, which would then call its own "add(a,b)" function with the appropriate arguments, which would then execute the machine code "ADD" instruction.

You can do anything that you can do in an interpreted language in a compiled language and vice-versa - they are both Turing complete. Both however have advantages and disadvantages for implementation and use.

I'm going to completely generalise (purists forgive me!) but, roughly, here are the advantages of compiled languages:

- Faster performance by directly using the native code of the target machine
- Opportunity to apply quite powerful optimisations during the compile stage

And here are the advantages of interpreted languages:

- Easier to implement (writing good compilers is very hard!!)
- No need to run a compilation stage: can execute code directly "on the fly"
- Can be more convenient for dynamic languages

Note that modern techniques such as bytecode compilation add some extra complexity - what happens here is that the compiler targets a "virtual machine" which is not the same as the underlying hardware. These virtual machine instructions can then be compiled again at a later stage to get native code (e.g. as done by the Java JVM JIT compiler).

Standard C does not support function overloading



Feel free to modify

An abstract class in C++ just needs to have a pure virtual function inside it. This means that abstract classes in C++ are not explicitly declared to be abstract – having a pure virtual function implicitly makes a class abstract in C++

### 3 Override vs Overload

You overload functions for three reasons:

1. To provide two (or more) functions that perform similar, closely related things, differentiated by the types and/or number of arguments it accepts. Contrived example:

2. `void Log(std::string msg);` // logs a message to standard out

`void Log(std::string msg, std::ofstream);` // logs a message to a file

3. To provide two (or more) ways to perform the same action. Contrived example:

4. `void Plot(Point pt);` // plots a point at (pt.x, pt.y)

`void Plot(int x, int y);` // plots a point at (x, y)

5. To provide the ability to perform an equivalent action given two (or more) different input types. Contrived example:

6. `wchar_t ToUnicode(char c);`

`std::wstring ToUnicode(std::string s);`

In some cases it's worth arguing that a function of a different name is a better choice than an overloaded function. In the case of constructors, overloading is the only choice.

---

Overriding a function is entirely different, and serves an entirely different purpose. Function overriding is how polymorphism works in C++. You override a function to change the behavior of that function in a derived class. In this way, a base class provides interface, and the derived class provides implementation.

4 what is cache ? It is actually high speed caching

5 what is mutex?

In [computer science](#), mutual exclusion is a property of [concurrency control](#), which is instituted for the purpose of preventing [race conditions](#); it is the requirement that one [thread of execution](#) never enter its [critical section](#) at the same time that another [concurrent](#) thread of execution enters its own critical section

It can use both hardware and software to avoid this problem.

6 we get the fruit from mix. If we get apple from it. It shows the current correct label should be orange. And the orange label bag should be mix. And the apple label should be orange.

7 the difference between malloc and new

new/delete

- Allocate/release memory

1. Memory allocated from 'Free Store'

2. Returns a fully typed pointer.

3. new (standard version) never returns a NULL (will throw on failure)

4. Are called with Type-ID (compiler calculates the size)

5. Has a version explicitly to handle arrays.

6. Reallocating (to get more space) not handled intuitively (because of copy constructor).

Feel free to modify

7. Whether they call malloc/free is implementation defined.
8. Can add a new memory allocator to deal with low memory (set\_new\_handler)
9. operator new/delete can be overridden legally
10. constructor/destructor used to initialize/destroy the object

malloc/free

- Allocates/release memory
- 1. Memory allocated from 'Heap'
- 2. Returns a void\*
- 3. Returns NULL on failure
- 4. Must specify the size required in bytes.
- 5. Allocating array requires manual calculation of space.
- 6. Reallocating larger chunk of memory simple (No copy constructor to worry about)
- 7. They will NOT call new/delete
- 8. No way to splice user code into the allocation sequence to help with low memory.
- 9. malloc/free can NOT be overridden legally

Table comparison of the features:

| Feature                  | new/delete                                                | malloc/free                  |
|--------------------------|-----------------------------------------------------------|------------------------------|
| -----+-----+-----        |                                                           |                              |
| Memory allocated from    | 'Free Store'                                              | 'Heap'                       |
| Returns                  | Fully typed pointer                                       | void*                        |
| On failure               | Throws (never returns NULL)                               | Returns NULL                 |
| Required size            | Calculated by compiler                                    | Must be specified in bytes   |
| Handling arrays          | Has an explicit version                                   | Requires manual calculations |
| Reallocating             | Not handled intuitively                                   | Simple (no copy constructor) |
| Call of reverse          | Implementation defined                                    | No                           |
| Low memory cases         | Can add a new memory allocator   Not handled by user code |                              |
| Overridable              | Yes                                                       | No                           |
| Use of (con-)/destructor | Yes                                                       | No                           |

Technically memory allocated by new comes from the 'Free Store' while memory allocated by malloc comes from the 'Heap'. Whether these two areas are the same is an implementation details, which is another reason that malloc and new can not be mixed.

---

## 10 difference between stack and heap

In the context of Operating Systems, stack and heap are the two sections of the memory layout of a process. The stack is used to keep track of variables/parameters local to a function in a program. Whenever you call a new function, a new stack frame is pushed to the stack with parameters and variables local to that function. When that function returns, the stack frame is popped out and the context switches back to the previous function (the caller).

A heap is a kind of a global memory pool. A function can allocate memory on the heap if it wants the data to live longer than the function itself. Objects allocated on the heap are

Feel free to modify

accessible to all the functions, given they have the reference/address of the object to access it. In C, you can allocate memory on the heap using the malloc(3) family of functions.

Stack:

- Stored in computer RAM just like the heap.
- Variables created on the stack will go out of scope and automatically deallocate.
- Much faster to allocate in comparison to variables on the heap.
- Implemented with an actual stack data structure.
- Stores local data, return addresses, used for parameter passing
- Can have a stack overflow when too much of the stack is used. (mostly from infinite (or too much) recursion, very large allocations)
- Data created on the stack can be used without pointers.
- You would use the stack if you know exactly how much data you need to allocate before compile time and it is not too big.
- Usually has a maximum size already determined when your program starts

Heap:

- Stored in computer RAM just like the stack.
- In C, variables on the heap must be destroyed manually and never fall out of scope. The data is freed with delete, delete[], or free
- Slower to allocate in comparison to variables on the stack.
- Used on demand to allocate a block of data for use by the program.
- Can have fragmentation when there are a lot of allocations and deallocations
- In C++ data created on the heap will be pointed to by pointers and allocated with new or malloc
- Can have allocation failures if too big of a buffer is requested to be allocated.
- You would use the heap if you don't know exactly how much data you will need at runtime or if you need to allocate a lot of data.
- Responsible for memory leaks

What is fork system call?

The purpose of fork() is to create a new process, which becomes the child process of the caller. After a new child process is created, both processes will execute the next instruction following the fork() system call. Therefore, we have to distinguish the parent from the child.

- 
1. What is virtual memory, explain
  2. Compiled language vs interpreted language, explain
  3. Override vs overload, explain
  4. What is cache
  5. What is mutex
  6. 智力题：三个basket，一个全apple，一个全orange，一个是apple和orange的混合，然后三个label分别写着apple，orange，apple & orange然后三个标签都贴错了，你只能取出一个水果然后

Feel free to modify

后如何把三个标签都贴到正确的那个basket上。注意关键点是三个标签全部都贴错，然后只能取出一个basket中的水果。这个可以先自己想想，想知道答案可以自己搜Google或者留言。

7. 编程题三题:

1. Print all odd numbers between 1 to 100 - 一种方法说完就过了
2. Remove duplicate from unsorted array - 问了两种方法
3. Power of 2 - 让我说as much as you can 的方法-google 1point3acres
8. (之前面经没见过的题目之一) What is fork: 我现场开google搜的= =
9. Given 1 billion number, get the largest 1 million. Large dataset means you cannot store all of them and sort. 这个用MinHeap解决。

---

十月份做完OA告诉我坑满了，11月14收到邮件有open position

21号电面，隔天收到onsite 问题跟地里的其他帖子差不多 攒人品 再发一次~

面试过程30分钟，不用写代码，口头解释就行

1. 讲一个project, why challenging

这个一定要提前准备好

2. stack与heap的区别

3. 算法题 median of datastream

follow up 1: 如果数字都在1~100 (这两个follow up在其他帖子没见到过)

follow up 2: 如果大部分数字在1~100, 小部分在此范围之外

每道题解释算法和复杂度

3. malloc与new的区别

4. mutex是什么，如何设计mutex, 有两个thread同时[申请](#)mutex怎么办

都是面经里的问题，heap vs stack，这两个哪个快？mutex是什么 data stream 里面找medium，算法复杂度

链接: <https://instant.1point3acres.com/thread/203359>

来源: 一亩三分地

30分钟电面，面试官迟了3分钟多，早1分钟走，因为他说有人要用他正用着的conference room。=.= 开始叫我描述下自选的一个project: project details, challenges等。问了下malloc和new的不同点。然后问了一道median data stream。我说用max heap + min heap。他说正确，但请用另一个STL data container。我讲了deque和stack，他都不满意。最后他说用set，我就跟他随便说了下找set里的median值，他就说good。=.= 然后问mutex是什么，怎样设计mutex。我乱答，然后他说如果有race condition呢？我说用queue存waiting threads。之后他说我的设计会有race condition，问我为什么会有race condition。我说了下因为hold and wait的原因会有deadlock。边答时边想，race condition和deadlock不一样啊。我在答非所问。。之后他就叫我问他问题。我问了junior developer和quant developer如何合作解决问题。他跟我介绍了公司里有什么position。=.= 这是我人生第一个面试分不出面试官是华人还是印度人。。。听名字像华人，但slang像印度人。

链接: <https://instant.1point3acres.com/thread/192012>

Feel free to modify

来源: 一亩三分地

发个电面攒人品。他家的电面说是半小时，那美国小哥硬是和我聊了一小时。由于OA我用的是java，所以被分到visualization组。他家问题感觉就是概念，编程就是说思路。先小哥自我介绍，然后被问到简历项目。概念题：1. Map, Queue, Stack, Vector 2. Overloading vs. Overriding 3. static vs. non-static 4. Deadlock 5. Cache 6. System call 7. Virtual Memory 8. python vs. Java 编程：（说思路）1. power of 2 2. remove duplicates 后来就闲聊问问他们公司技术部分组然后就拜拜了。

链接: <https://instant.1point3acres.com/thread/136739>

来源: 一亩三分地

Python的问题：1. Given a parent class named automobile, subclass named bus,这两个class都有一个叫做accelerate的方法，问，`automobile().accelerate()` 会调用automobile的方法还是bus的方法。。等等类似的问题三个（另外两个记不清了）

2. 给一个class named bus, 一个instance叫mybus, 一个membervariable 叫啥啥(忘了) = 42. 问在python里，直接call mybus.啥啥合不合法。（大概这个意思。。具体我也给忘了）

概念题：

1. What is Mutex?

2. Difference between compiled language and interpreted language, and their adv and disadv.

3. `fork()`, child process, 这个打开谷歌搜了照着读吧。

4. Difference among list, dict and set in python

5. `set()`的time complexity（这个楼主竟然很自以为是地答错了，答案是 $O(1)$ , $O(1)$ , $O(1)$ 重要的事儿说三遍。）

6. Python memory allocation是怎样的（大家大概了解一下就好了）

7. What is Python decorator?（然后问之前有没有用过）

Brain test：

三个baskets, 一个里面装满oranges,一个里面装满apples,一个里面装的是oranges+apples。三个baskets外面都贴有label, 但是label都是错的。让你只从一个篮子里面拿一个水果，怎么判断三个baskets里面装的是什么。

答案：从贴着apple+orange的篮子拿出来一个，因为是错的，所以里面要不然全是apple，要不然全是orange。比如拿出来的是apple，则这个篮子装的是apple，而那个贴apple标签的肯定是orange，然后贴orange的是混合的。

Feel free to modify

其他题：1. Print all add numbers between 1 to 100. (way1 : Iteration从1开始每次+2, way2 : mod 2)

2. Remove duplicated from unsorted array. (way1: python set method, way2: hasmap,前两种给的sorted的结果。 way3: add element to temp array, if this element has already existed in the temp array) 这题方法很多，大家随便虐

3. 如何判断integer是Power of 2. (way1 : 位运算, way2 : iteration mod 2 ) leetcode原题

4. 给你一大堆unsorted的数，找最大的前k个。要求不能store all of them, 也不能sort (用minheap)

. From 1point 3acres bbs

如果python用的不是很熟的话，建议大家面试之前先看看python的OOP，还有一些类、对象等等调用的语法啥的。

What is virtual memory, explain

virtual memory is a [memory management](#) technique that is implemented using both hardware and software. It maps [memory addresses](#) used by a program, called [virtual addresses](#), into physical addresses in computer memory. [Main storage](#) as seen by a process or task appears as a contiguous [address space](#) or collection of contiguous [segments](#). The operating [system](#) manages virtual address spaces and the assignment of real memory to virtual memory. Address translation hardware in the CPU, often referred to as a [memory management unit](#) or MMU, automatically translates virtual addresses to physical addresses.

Compiled language vs interpreted language, explain

Override vs overload, explain

Override:

is a type of function which occurs in a class which inherits from another class. An override function "replaces" a function inherited from the base class, but does so in such a way that it is called even when an instance of its class is pretending to be a different type through polymorphism.

Overload:

is the action of defining multiple methods with the same name, but with different parameters.

-google

What is cache

What is mutex

The point of a mutex is to synchronize two threads. When you have two threads attempting to access a single resource, the general pattern is to have the first block of code attempting access to set the mutex before entering the code. When the second code block attempts access, it sees the mutex is set and waits until the first block of code is complete (and un-sets the mutex), then continues.



Feel free to modify

Fork:

fork is an operation whereby a [process](#) creates a copy of itself. It is usually a [system call](#), . visit [1point3acres.com](#) for more.

System call:

is the programmatic way in which a [computer program](#) requests a service from the [kernel](#) of the [operating system](#) it is executed on. This may include hardware-related services (for example, accessing a [hard disk drive](#)), creation and execution of new [processes](#), and communication with integral kernel services such as [process scheduling](#). System calls provide an essential interface between a process and the operating system.

---

编程题三题:

Print all odd numbers between 1 to 100 - 一种方法说完就过了

Remove duplicate from unsorted array - 问了两种方法

Power of 2 - 让我说as much as you can 的方法 (之前面经没见过的题目之一)

an unsigned integer which is a power of 2 has only one of its bits as 1. So a simple solution would be to loop through the bits and count the number of 1s.

What is fork: 我现场开google搜的= =

Given 1 billion number, get the largest 1 million. Large dataset means you cannot store all of them and sort. 这个用MinHeap解决。

#### 1. Deadlock

a deadlock is a situation in which two or more competing actions are each waiting for the other to finish, and thus neither ever does.

#### 2. Mutex

#### 3. Stack and Heap

|              |                                                                                                                                                                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| voteaccepted | The stack is the memory set aside as scratch space for a thread of execution. When a function is called, a block is reserved on the top of the stack for local variables and some bookkeeping data. When that function returns, the block becomes unused and can be used the next time a function is called. |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

The heap is memory set aside for dynamic allocation. Unlike the stack, there's no enforced pattern to the allocation and deallocation of blocks from the heap; you can allocate a block at any time and free it at any time.

#### 4. Differences between python and C++

Feel free to modify

## 5. Differences between overloading和overriding

面试大概55分钟。1. 简历说一个project, 面试官提问。(大概10分钟内)

2. 问概念题!!! 概念题! 概念题! 概念题! 重要的事情要说三遍-

virtual memory 是什么 - (回答不好)

- interpreter 和 compile language的区别

- system call 是什么 (回答不好)

### - override 和 overload 的区别

## - arraylist, hashmap, queue, stack的区别

### - static 和 non-static的区别

- cache 概念 - finally是什么 - ... 有几题忘记了。。。大家好好准备！

3. 编程题-口述思路。3题很简单，每题要提供两种解法（秒杀）

- remove duplicates from unsorted array

- print all odd numbers between 1 - 100 - power of 2

4. 还有一条逻辑智力题（回答不好）有三个麻布袋，里面分别有 apple, orange, apple & orange, 然而麻布袋里的 labels 可能贴错了。分别从3个袋子里面取出一个 fruit, 问如何把 label 重新贴对。

海投、Quant Data Analysis OA：两小时四道题，鏖战汉滨，滑€滨+竿錄啃噬

1. 给两个int array: boxes and lemons, 一个代表箱子的大小, 一个代表西瓜的大小, 每个箱子最多只能装一个小于等于它size的西瓜。在西瓜size array选择起点i, 放到任意可放的箱子位置, i之后连续的西瓜i+1, i+2...都要找到箱子放进去且箱子中西瓜的相对顺序需要保持原来的顺序。求最多可放在箱子里的西瓜序列的长度 (西瓜数)

2. Given training data, whose features are 3-dimensional numerical features, 在不允许调用机器学习包（如sklearn）的情况下，写出为目标3 classes的classification方法

3.输入文件，输出unique words与26个字母的counts到文件里。invalid word（比如其它字符出现在了space separated words里面），不能被考虑

4.类似于给定2个快递员送货到每家的策略，一个快递员deliver的time slot另一个不能有action只能在原地等，求最短送完货需要的时间。简单但是写起来比较烦。

电面：

1小时 介绍个人背景20min 考Python三道题：

## 1. Generator如何使用

## 2.python都有哪些data analysis相关的包

### 3.在不让用乘法operator的情况下如何进行乘法

脑筋急转弯一道题：有1个完整的金条，需要每天付给员工 $1/7$ 并且连续7天都要付。在只能切两刀的条件下如何每天都给 $1/7$ 。

本人CS & Data Science硕士，无金融背景/经历。已挂。特来地里发面经攒人品

Feel free to modify

通过OA，给了一个45分钟的面试。面试官是个quant，来了先自己介绍了下背景，然后问我的研究。聊了会研究，聊蒙特卡洛模拟，然后又聊了股票价格随机过程。之后考了一个 2Sum。。。。我写了一个 $O(n \lg n)$ 的算法，问我能不能优化，把hash table给忘了。面试官提醒了下，说hash table也可以，能做到 $O(n)$ 。之后让我问问题，问了三个问题结束电话面试，说一周内给我消息。

---

刚刚面完Akuna Capital Shanghai Full time C++ developer职位，HR是Lucy Luo。预约晚上9点面试，9点10分发邮件询问面试，9点15分才进行面试。先问了，毕业留美还是回国？为什么放弃美国3年opt的机会，选择回国工作？期间，Lucy透露了上海公司给Jr. C++ developer工资是240k-250k RMB，芝加哥公司给80k-100k USD。你了解Akuna Capital的产品以及工作职位吗？Akuna Capital Shanghai主要是招聘廉价码农维护在美的投资系统平台，其工作码农不会参与到核心的交易业务。接着HR给了一段代码，让我计算x, y, text的值。主要是你要知道C++俩面强制转换的概念，知道unsigned int, long int都是4 Bytes，char是1 Byte。计算0x0F 0x00 0x00 0x00 0x05 0x00 0x00 等等。楼主跪在计算上了，直接口算。接着HR问了技术问题，第一个是你知道几种IPC in Linux。第二个是你知道Fork()和VFork()有什么区别。第三个，你是否能通过preorder重建binary tree？第四，你知道什么是system call？第五，你知道什么是fork？最后，她让我等两天后的结果。Lucy是中国人，英文不是很标准，电话听不太清楚。

关于Jr#3 OA题目一共是六道，110分钟，第一道只能用python3写，给一系列数，计算奇位上之和以及平均数。第二道是一道逻辑题，就是让你判断给定四点是不是能够不交叉的连接。第五道是找一组数中第二大的数字。第六道是计算最大乘积。第三，四都是比较基础的题目，10分钟都能解决。HR问的技术问题都比较偏重于Linux系统相关的。工作接触的东西应该也是问题所涉及的，感觉挺boring。希望接下来有好运！加油 $\wedge(\omega\wedge)^{\wedge}$ 。另外，求大米！！

---

电话面试：1个小时左右，why Akuna Capital? why not google, fb? why not Bay Area?

data stream, compute median and mean?

C++ basic questions: inheritance, polymorphism etc

what is your most challenging project? why challenging?

---

10月27号Akuna Capital电面新鲜面经

因为本人只会python 和 js所以问了一大堆python的问题，大约面试花了45分钟的样子

最开始问你python的使用程度怎样，做过哪些项目啥的。。。[1point3acres.com/bbs](http://1point3acres.com/bbs)

Python的问题：

1. Given a parent class named automobile, subclass named bus,这两个class都有一个叫做accelerate的方法，问，automobile().accelerate() 会调用automobile的方法还是bus的方法。。等等类似的问题三个（另外两个记不清了）

Feel free to modify

2. 给一个class named bus, 一个instance叫mybus, 一个membervariable 叫啥啥(忘了) = 42. 问在python里, 直接call mybus.啥啥合不合法。(大概这个意思。。具体我也给忘了)

概念题:

1. What is Mutex?
2. Difference between compiled language and interpreted language, and their adv and disadv.
3. fork(), child process, 这个打开谷歌搜了照着读吧。
4. Difference among list, dict and set in python.
5. set()的time complexity (这个楼主竟然很自以为是地答错了, 答案是 $O(1)$ ,  $O(1)$ ,  $O(1)$ 重要的事儿说三遍。) . 1point 3acres 聪哄懵
6. Python memory allocation是怎样的 (大家大概了解一下就好了) .
7. What is Python decorator? (然后问之前有没有用过)

Brain test :

三个baskets, 一个里面装满oranges,一个里面装满apples,一个里面装的是oranges+apples。三个baskets外面都贴有label, 但是label都是错的。让你只从一个篮子里面拿一个水果, 怎么判断三个baskets里面装的是什么。

答案: 从贴着apple+orange的篮子拿出来一个, 因为是错的, 所以里面要不然全是apple, 要不然全是orange。比如拿出来的是apple, 则这个篮子装的是apple, 而那个贴apple标签的肯定是orange, 然后贴orange的是混合的。

. from: [1point3acres.com/bbs](http://1point3acres.com/bbs)

其他题:

1. Print all add numbers between 1 to 100. (way1: Iteration从1开始每次+2, way2: mod 2) .  
From 1point 3acres bbs
  2. Remove duplicated from unsorted array. (way1: python set method, way2: hasmap,前两种给的sorted的结果。 way3: add element to temp array, if this element has already existed in the temp array) 这题方法很多, 大家随便虐
  3. 如何判断integer是Power of 2. (way1: 位运算, way2: iteration mod 2) leetcode原题
  4. 给你一大堆unsorted的数, 找最大的前k个。要求不能store all of them, 也不能sort (用minheap做)
- google 1point3acres

总体感觉面试官蛮nice的。如果python用的不是很熟的话, 建议大家面试之前先看看python的OOP, 还有一些类、对象等等调用的语法啥的。。老印语速巨快。。根本就只听得懂关键词然后来猜他问的是啥。。。之前那个automobile (啊都毛biao。。) accelerate (啊哥仨拉里)。。我后面听到bus, 我特么的才知道是啥。。不过名字并不重要, 你拿只笔记下来哪个是parent那个是child就好。。不确定的词可以原模原样的读给他听: 啊哥仨拉里...is the child class.. 反正挺简单的。。希望能有onsite。

另外在此非常感谢前辈们提供的OA和电面, 也祝个位[找工作](#)的小盆友们顺利~~~

---

接下来电面。。。挂了，好心赛。

1. Deadlock
2. Mutex

3. Stack and Heap. [1point3acres.com/bbs](http://1point3acres.com/bbs)

4. Differences between python and C++

5. Differences between overloading and overriding

还有好些类似的记不清了。。。鏈构鋤增塏鑷◆1point3acres聰哄漣

然后有几道很简单code题，说思路就好，不用写。 鍬ㄣ簪涓€浜€涓€涸鑼拌鎷◆.

面试人不好%>\_<%，最后问我什么问题要问，我问你有什么有趣的项目分享一下。。。他居然说好像是秘密，不能透露太多！！

活活打死！！！！

---

握草，居然没有问任何mutex和thread，我感觉我要挂我听力不好....让白人小哥重复了几次，小哥最后说他是leader，日了仙人

第一个问题：你关于python的项目

第 \* 个问题：解释下OOP的优势？because it's data oriented

第 \* 个问题：为啥要用OOP？module和class的区别是什么？还是data oriented，class的method对应自己的instance。moudule没有，跟java的mah一样

第二个问题：如何实现一个类，能用和list一样的[]，\_\_getitem\_\_，\_\_setitem\_\_，magic method

第三个问题：python 如何garbage collection，reference counter。然后追问，给出一个situation，ref counter不是0，但还是garbage。两个instance互相指，circle，deadlock，握草，这个地方考（要挂就在这里挂）

第四个问题：为啥要用generator？use on demand

第五个问题：解释下python里面的function是first class object？（解释下python里面的functional programming），closure啊。visit [1point3acres.com](http://1point3acres.com) for more.

问完了之后我问了下小哥 junior developer难道天天写前端？他说，我特么是infrastructure leader，（要挂就挂这里）

顺便问了下，如果performance需要，你们还用python？（要挂就挂这里）

python里面有个GIL我没搞明白，想问问他们的解决方案。（反正问了前面两个傻逼的问题，我豁出去了）

---

小哥是个美国人，和小哥聊得很开心。完全没有准备。问了好多问题 就记得几道，觉得问题本身有点奇怪：1) 如何debug race condition 其实我觉得race condition直接删了重写比较快，当然没敢什么说

2) Malloc. 1point 3acres 聰哄漣

Feel free to modify

3) design一个lock, followup: 如何atomic look up那些value (read my last write什么的)

4) 你遇到过最难的项目是什么-google 1point3acres

5) 你实习做了什么, 然后我说我做了什么, 他问我几周complete? (有点奇怪)

总之都是OS底层的东西, 在加上一点算法。最后让我问问题我实在想不出来就问芝加哥天气怎么样, 他就呵呵了说冷死了

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=202092&highlight=akuna>

---

选择题忘记了, coding 3题

第一题就是判断括号是不是平衡, leetcode原题

第二题似乎没有人贴过, 就是找一个array的第二大的元素

第三题就是出现过很多次的筛子游戏的题目了

. more info on [1point3acres.com](http://www.1point3acres.com)

求大米阿。。。

---

9月底参加的学校career fair, 不记得和Akuna Capital的recruiter聊了什么给的面试。后来查才发现Akuna Capital是一家在芝加哥发展很好的startup, 当然bar应该很高了, 自我安慰一下, 那么挂了也是情理之中的了。

=====

首先发了online codeing (题目比较简单, 时间很充裕)

1. Hardware caching:(多选)

typical computers have several layers of caches.

in general, caches with smaller latency are larger than those with higher latency.

data is moved from one cache to another one byte at a time.

(哎呀看都看不懂, 麻烦资深人士解释一下)

2. software engineering lifecycle(就是发生先后排序, 有design, unit test之类的)

. visit [1point3acres.com](http://www.1point3acres.com) for more.

3. 不是用来 control execution flow in a typical debugging tool (选的step up)

4. recursion (给了一段简单地程序, 问你它干嘛了)

5. trace a program (给了一段简单地程序, 问你输出什么)

def traverse(seq\_len):

    idx = 1

    traversed = []

    while len(traversed) < seq\_len:

        traversed.append(idx). From 1point 3acres bbs



Feel free to modify

```
traversed.append(2*len(traversed) + 1). 涓€浜-涓€攷珞-鎰帮紉鐢€鎴�  
idx += 2  
if idx > seq_len:  
    idx -= seq_len
```

```
sum = 0  
for idx in traversed:  
    i = traversed.pop(). Waral 錦氫鏈�洽澶�杩涗, 涓€  
    sum += (i * idx)  
  
print(sum)  
-google 1point3acres  
traverse(7)
```

.1point3acres 緬

6. rank the data structure, (look-up time): 给了 linkedlist, balance binary tree, hash table

7. 给一个int array和target, 像是2sum, 不同的是要求两个数各乘上100后等于target。 .

[1point3acres.com/bbs](http://1point3acres.com/bbs)

下面是我的code。

```
import java.io.*;  
import java.util.*;. 鎰帮紉鐢€鎴�, 涓€+鎰帮紉  
import java.text.*;. 鎰帮紉鐢€鎴�-涓€-涓€攷珞鎰帮br>  
import java.math.*;  
import java.util.regex.*;
```

```
public class Solution {  
    /*  
    * Complete the function below.  
    */.1point3acres 緬  
  
    static boolean any_two_integers(int[] array, int target) {  
        if(array == null)  
            return false;  
        if(target % 100 > 0)  
            return false;  
        target /= 100;
```

```
        int len = array.length; .1point3acres 緬  
        if(len <= 0). from: 1point3acres.com/bbs  
            return false;  
        // Store remain as key, indice as value
```

Feel free to modify

```
HashMap<Integer, Integer> remains = new HashMap<Integer, Integer>();
```

```
// Iterate every num. 1point 3acres 聰哄漣
for(int i = 0; i < len; i++) {
    int value = array[i];.鐸權聰哄漣-涓€€汧-涓€攷垆銑
    if(target % 2 == 0 && value == target / 2)
        return true;.鏈构鋤增垵鑹1point3acres聰哄漣
    if(remains.containsKey(value)) {
        return true;
    }

    int remain = target - value;
    //Notice: need to put the indice as value
    remains.put(remain, i);
}

// Cannot find these two nums
return false;

}

public static void main(String[] args) throws IOException{
    Scanner in = new Scanner(System.in);
    final String fileName = System.getenv("OUTPUT_PATH");
    BufferedWriter bw = new BufferedWriter(new FileWriter(fileName));
    boolean res;

    int _array_size = 0;
    _array_size = Integer.parseInt(in.nextLine());
    int[] _array = new int[_array_size];
    int _array_item;
    for(int _array_i = 0; _array_i < _array_size; _array_i++) {
        _array_item = Integer.parseInt(in.nextLine());. Waral 銑氫鏈�攡澶冨熸杩綯
        _array[_array_i] = _array_item;
    }

    int _target;
    _target = Integer.parseInt(in.nextLine());

    long startTime = System.currentTimeMillis();

    res = any_two_integers(_array, _target);

    long endTime = System.currentTimeMillis();
```

```

        bw.write(String.valueOf(res ? 1 : 0));
        bw.newLine();
        bw.write(String.valueOf(endTime - startTime < 1000 ? 1 : 0));. 涓€浜◆-涓攷咯-鰻幫紆鐓鑒賡
    }
}

```

9. remove duplicate in an array, and return an array in reverse sort.. 鋸磋鋤或滑@1point 3 acres

找到这样的pair, x和y。他们的sum和绝对值减都是pentagonal。并且绝对值减最大化。  
我的code参考：

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*; from: 1point3acres.com/bbs
import java.util.regex.*; more info on 1point3acres.com
. Waral 錦氫鏈爰洿澶氣构綉◆,
public class Solution {
    /*
```

```
public static long pentagonal[] = new long[10000];
```

```
return bruteForce();
}
```

```
public static int[] bruteForce() {
    int[] rst = new int[2];
```

Feel free to modify

```
{. 鐫櫨鑒宠聰哄漣-涓€浜+竺錄哨逆
for(int j = k - 1; j > 0; j--)
{
    if(isPentagonal(pentagonal(k) + pentagonal(j))
        && isPentagonal(pentagonal(k) - pentagonal(j)))
    {
        rst[0] = pentagonal(j);. Waral 銚氫鏈夋洸澶勭粯,
        rst[1] = pentagonal(k);
        return rst;. From 1point 3acres bbs
    }
}
}. visit 1point3acres.com for more.
}
. 鐫櫨鑒宠聰哄漣@1point 3 acres

public static int pentagonal(int n)
{
    return n * (3*n - 1) / 2;. 鐫櫨鑒宠聰哄漣@1point 3 acres
}

public static boolean isPentagonal(long num)
{
    if(Arrays.binarySearch(pentagonal, num) >= 0) 鐫櫨鑒宠聰哄漣.涓€浜+竺錄哨逆.
        return true;

    return false;
}. 鐫櫨鑒宠聰哄漣@1point3acres聰哄漣
    public static void main(String[] args) throws IOException{.1point3acres緬
        Scanner in = new Scanner(System.in);. 鐫櫨鑒宠聰哄漣@1point3acres聰哄漣
        final String fileName = System.getenv("OUTPUT_PATH");
        BufferedWriter bw = new BufferedWriter(new FileWriter(fileName));
        int[] res;-google 1point3acres
        res = find_pentagonal();. 1point3acres.com/bbs
        for(int res_i=0; res_i < res.length; res_i++) {
            bw.write(String.valueOf(res[res_i]));
            bw.newLine();
        }. more info on 1point3acres.com

        bw.close();
    }
}
.1point3acres緬
. 鐫櫨鑒宠聰哄漣@1point 3 acres
```

Feel free to modify

11. count bits. From 1point 3acres bbs

long int as input, return long (representing the number of non-zero bits in the binary representation of the input)

12. given two input strings, return a string consisting of the common chars in sorted order.

---

## OA

---

C++

要求在72 hrs完成一道题, 题目是matching engine. 就是模拟线上交易stock  
简单来说就是你要提供buy, sell, cancel, modify and print 的功能.

INPUT EXAMPLE

SELL 1000 10 orderA. 卖/买 价格 数量 ordername

BUY 1000 10 orderB. 卖/买 价格 数量 ordername. 1point 3acres 聰哄漕

CANCEL ordername. order name -google 1point3acres

PRINT

MODIFY SELL/BUY 1000 10 ordername 把原本的order 更改. 鏑況汉浜戢況,涓€浜†竺錄喟泣

不难, 但test cases过不了, 无心做了.....

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=204399&highlight=akuna>

---

C++

超大一题

时间三天. 涓€浜†-涓€浜†-鏑幫紝鐢€鎶€賁

做一个股票交易系统的一部分, 就是类似显示买卖十档的, 时间优先交易, 但要满足价格条件。.

[1point3acres.com/bbs](http://www.1point3acres.com/bbs)

可 BUY SELL MODIFY CANCEL PRINT order

做了一下午, 调了一晚上, 27个case 初始有9个case不过, 仔细阅读了一下描述, 然后当他保证  
的输入合法是屁话 做了输入验证后有7个case不过。

然后就再也找不到哪里出问题。自己编的case都是能过的 除非我理解有偏差。发了邮件要几个不  
过的case看看到底是哪里有问题

还没回复我。

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=207252&highlight=akuna>

Feel free to modify

---

1 remove\_char\_from\_string, 就是  
new\_str = re.sub(the\_character, "", the\_string)  
这样一个意思, 热身. 鏢鎚鎚或滑@1point 3 acres

2 smallest\_binary\_complement  
给你一个正整数, 求它的smallest binary complement, 比如说13 smallest的二进制表示方法就是1101, 所以结果是0010 => 2, 而3的结果就是0.  
. visit [1point3acres.com](http://1point3acres.com) for more.

3 find\_two\_closest\_and\_sum  
给一个数组, 和一个target\_num, 求数组中最接近target\_num的两个数的和. From 1point 3acres bbs

4 largest\_mono\_subsequence  
跟lis (最长递增子序列) 差不多, 不过可以skip数字, 返回的是最长的subsequence.然后如果有长度一样的subsequence, 就返回sum最大的那个

5 word\_distance  
给一个字符串中, 给两个word, 和一个数字n, 求这两个word在字符串中距离(<=n)的情形有多少组

---

第一次发面筋, 90分钟OA, 共5题,

1 删除字符串中的某个字符

2 忘记了, easy

3 在一个数组中找2个值与给定值最接近的数

4 . [1point3acres.com/bbs](http://1point3acres.com/bbs)

5 求一个长字符串用空格分开单词, 找寻 单词 a 和单词 b 最大间隔 n 的次数, 这个题目我觉得超坑爹, 有4个输入, input, a, b, n, 前3个输入时 字符串类型, 后面是int类型, 正常函数有输入变量, 变量a,b,n应该有东西, 但是我后面在hackerrank 里面cout 所有的输入, 发现input 里面包含后面的变量的值, 后面3个变量为空。要自己赋值, 但是要自己赋值的话, 我用它的测试例子在自己电脑用 java跑, 没有问题, 但是在hackerrank里面3个例子只过了一个。之前看地理的人说可以用java, 我发现只能用 c++ 或者 python3。而且OA 也变了

. From 1point 3acres bbs

第4题如图, 没有时间做完。

不熟c++, 花了不少时间google 基本的操作

find code in this link

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=199867&page=1&authorid=131002>

---



Feel free to modify

Python

刚做完他家号称180min的coding challenge。

在地里看面经基本都是 90min 五六道算法题这样 也比较简单的那种，但是都是有编号的#1 #2 #3...

默默看了眼我的邮件，眼睛望穿了也没有任何数字。。。鐔權鑿宠璁哄漕-涓€浜+竺錄喟泣  
后来想想一定是障眼法，只是写着180min而已，实际还是那些题。

直到我点进去，发现就1题。。。。。。倒计时真的从02:59开始。。。

当时开始蒙圈了，还能不能像个正常OA一样啦阿！！！！

点进去了，题目就拖不到底，密密麻麻的description，只有平复心情静静读着。。。。

读到第二段，非常好，脑里突然闪过，第一段刚刚讲了些什么！

就这样反反复复，我终于读懂了题，类似是要设计个stock trading [system](#)??但是也不是功能那么完善的。可是中间状态略多。

new, order\_acknowledge, order\_reject, cancel, cancel\_acknowledge, cancel\_reject, fill

。。。。。。鏗磋鎏貳滑@1point 3 acres

让你计算每一个不同order中间状态的position，至于这个Position怎么定义的呢，接近于当前有的share of stock，可以是负数，意思是你没有股票也照样可以卖 鏗ㄣ簪涓€浜+涓€涸鎵拌鎵+。  
然后这个position还要possible smallest，也就是只要你说我要sell，position立马减少你sell quantity那么多，直到后来你的order被reject，又或者你自己cancel并且cancel成功，还或者最后没有被fill那么多，这几种状态下position才会变，可是对于buy呢，一开始你说我要buy的时候是position完全不变的，对于一开始是buy的order，只有你最后真的被fill了（也就是被别人买了），那时候你才是真正买到，才会增加position。。。

anaways, 我是觉得略复杂，给的180min真不是盖的。。。。

对于这个oa我也是服气的。最后还友情提示：Since it's too complex, you don't necessarily need to get correct answer or pass all test cases. It's your thoughts really matters, so please add comment for further face-to-face interview communication if possible.....

最后test case有两个hidden case没过。我也是不想再想了。。。做过时间最长的oa，真的没有之一。。。。。

权当在攒经验活动脑子了。

---

投的Jr Dev#1。一直看着地里面经没变，稍微练习了一下Python就去做了OA,。结果题目全变了。。。虽然比较简单，但是Python只在实习中用过，做起算法不太熟练，耽误了不少时间，导致有些case没过。Anyway,我记性还不错，和大家分享一下。

依旧是五个题。鏗磋鎏貳滑@1point 3 acres

. more info on [1point3acres.com](http://1point3acres.com)

1.去掉String中的指定character

2.把int换成binary string, 然后把0换成1, 1换成0, 再变回int

3.找到int数组中离target最近的两个数（绝对值），返回他们的和。一开始看成了Sum离target最近，跪了。。。有一个case没过，不清楚，但是考虑了edge的情况

4.一组牌(2-10?), Arrays of Integers, 在不排序的情况下，找到最大连续，返回数组。根据test case, 相等可以在其中，如2, 3, 4, 5, 6, 7, 7

5.String input, String a, String b, int n 找到n个b连着a出现的情况，如a=Brown, b=The, n=1, s = The Brown is Brown of The Brown 就返回2. 应该a和b的前后顺序没关系

分享完毕，也算是攒点人品。祝在[找工作](#)的小伙伴们好运

Feel free to modify

. visit [1point3acres.com](http://1point3acres.com) for more.

. [1point3acres.com/bbs](http://1point3acres.com/bbs)

补充内容 (2016-9-21 06:03):. from: [1point3acres.com/bbs](http://1point3acres.com/bbs)

忘记说了，第四题可以弃牌，限制n张，和LC300略有区别。

补充内容 (2016-9-27 05:50):

再补充一下，第四题不只是subarray,是subsequence。连续指间隔为1。比如2, 3, 4, 5, 6. 如果2, 3, 5, 6就不行了

<http://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=202565&page=1&authorid=108105>

- 
- 1, LRU原则和cache的速度的简单问题. 涓€浜€-涓€攷恪-鎰帮紝鐢€鐔€戅
  - 2, software engineering cycle有哪几步
  - 3, step into, step out, step over, step up哪一个不是debugging的有效操作
  - 4, 给了一段代码问你代码干什么的，直接敲了一遍. 鏈€杩€鐔€戅1point3acres璁€哄€
  - 5, 一段Python代码，问最后输出什么，直接敲了一遍
  - 6, 选择题，链表，哈希表，二叉树哪个搜索最快
  - 7, two sum变形，给定一个数组，如果任意两个数的和的100倍等于target，返回true  
这题它给的输入有问题，是不能修改部分的代码，总是有一个exception，说  
numberformacionException
  - 8, 给定一个数组，找到两个数，a,a[j],满足i < j && a < a[j],使得a \* a[j]最大，返回这个乘积  
挺难的，一直超时，暴力解肯定是超时的，我自己想了一个说起来有点儿复杂的办法，如果有朋友有兴趣的话我再解释。不过结果也是超时，我想的办法我觉得应该是小于O(N^2)的，但是还是超时。不知道O(N)怎么解
  - 9, 数组去重，并逆序排序后输出
  - 10, pentagonal数，这个数啥意思请自己Google一下，给定  $P_n = n(3n - 1) / 2$ . 题目是要找到两个数  $P_x, P_y$ ，使得  $P_x - P_y$  和  $P_x + P_y$  都是pentagonal数，并且  $|y - x|$  最小  
参考了<http://www.mathblog.dk/project-e-pentagonal-numbers/>  
直接贴我的代码吧

/\*

\* Complete the function below.

\*/

```
static int[] find_pentagonal() {  
  
    int[] ret = new int[2];  
    boolean NOTfind = true;  
    int i = 1;  
    while(NOTfind){  
        i++;  
        int n = i * (3 * i - 1) / 2;  
        for(int j = i - 1; j > 0; j--){  
            int m = j * (3 * j - 1) / 2;
```

Feel free to modify

```
        if(isPen(n - m) && isPen(n + m)){
            ret[0] = Math.min(m,n);
            ret[1] = Math.max(m,n);
            NOTfind = false;
            break;
        }
    }
    }. 1point 3acres 聰哄漣
    return ret;
}

static boolean isPen(int num){
    double test = (Math.sqrt(1 + 24 * num) + 1) / 6;. From 1point 3acres bbs
    return test == (int)(test);
}
```

11,给一个long整数, 输出二进制码里1的个数

12,给两个string, 输出经过排序的他们共同的characters, 注意要支持重复的字符, 例如  
aeec,beef,输出ee而不是e

<http://www.1point3acres.com/bbs/thread-144391-1-1.html>

刚刚做了Akuna Capital的OA, 公司官网直接申的 software development intern, 他家OA居然设了deadline, 楼主上周特别忙没空做周五早上点开发现invitation过期了尼玛。。。又给HR发邮件求情说能不能再invite一次LOL。。。HR回邮件回的还挺快, 说再给一次机会但得这周日之前做完。。。LOL在湾区上学的楼主表示并没有想去芝加哥实习的打算, , 全当练习了哈哈, 看了两眼地里的帖子感觉不难于是直接点开做了。。

---

90分钟12道题 前面7个multiple choice, 后面5个coding。题目都不难, 时间很充裕, 楼主中间跑去泡了杯咖啡, 上了个卫生间, 热了个包子, 开了包薯片, 附带跟室友扯了半天废话还是提前二十分钟做完了。。

1. [optICAL](#) 多少种排列组合方式让所有vowel都在一起。 . 鏑浣汉浜戣浣,涓€浣+竺録啣泣
2. 找规律的那个题, 地里的面经貌似都有提到。 (前面这两道题怎么有种回到小学奥数感觉。。。)
3. 给一段code, recursive的, 问你这个code在干嘛。。
4. 一个问complexity的。一秒出答案。。 . visit [1point3acres.com](http://www.1point3acres.com) for more.
5. 这个貌似地里没人说过。。用linked list implement 一个queue, 问最少需要多少个pointer完成enqueue和dequeue。
6. 给一段函数问输出。。code很短, 愿意local敲一遍再跑也完全可以哈哈。。
7. 呃。。。忘了。。。

Feel free to modify

后面的code也都是原题貌似。。

maximum product, remove duplicates, string intersection, 2sum (\*100), count bits in binary expression

maximum product那道naive解会超时，不过中间加一些if 判断不是纯 $O(n^2)$ 的话就没事。。。或者貌似看地理好多人说用heap？表示并不知道怎么用，，，望有大神能解释一下哈哈～

嗯呐就是这些，，不知道他们家选人什么标准。。希望能进下一轮哈，，还从来没面过金融类公司呢。。

这周末还有两个coding challenge下周还有三个interview，攒人品！！

## Fulltime OA

akuna capital

Campus: C++ Jr Dev #3- 2017

Find the second most frequent number

Given a list of integers, return the number that appears second most frequently.

As an example, the function would receive:  
[0, 1, 2, 2, 2, 2, 1, 0, 5, 1]  
And return:  
1

YOUR ANSWER

Find and sum 2 closest numbers in list to a target number

You are given a list of unique numbers and a target number. Return the sum of the two numbers that have the smallest difference to the target number. You can assume the target number will not be in the list.

For example if you are given a list:  
[-15, -14, -9, -28, -17, 0, 6, 7, -6, -29]  
And a target number of 5.  
The two numbers in the list with the smallest difference to the target number would be 6 and 7 so the function would return 13.

Or of the list was:  
[21, 6, 27, 18]  
and the target number was 15.  
The two numbers in the list with the smallest difference to the target number would be 18 and 21 so the function would return 39

Connect the two paths

Overview  
Given an  $n \times n$  grid and two pair of points (A, B) and (P, Q), determine if it is possible to connect the A to B and P to Q without intersecting paths. Valid paths are those composed only with straight vertical and/or horizontal lines -- diagonal li

Definitions

- each point (e.g. A, B, P, Q) will have an x and y component
- the x axis is horizontal
- the y axis is vertical
- the point (0, 0) is located at the top-left corner of the grid
- the point (n, n) is located at the bottom-right corner of the grid

Feel free to modify

**Remove N Duplicates**

Write a function to remove all values from an integer array which appear exactly n times and returns an array of the remaining values sorted in ascending order.

Example  
Given: n = 3, integer array = [4, 4, 4, 3, 2, 1, 1]  
Returns [1, 1, 2, 3]

YOUR ANSWER

Campus: C++ Jr Dev #3- 2017 🕒 53m : 11s to test end

**Sum and Average from a List**

Given a list of integers, write a method that returns the sum and average of only the 1st, 3rd, 5th, 7th etc, element.  
For example, [1, 2, 3] should return 4 and 2.  
The average returned should always be an integer number, rounded to the floor. (3.6 becomes 3.)

## Intern OA

**Map Sum**

Create a class named MapSum with the following interface:

```
class MapSum {
public:
    MapSum(std::unordered_map<std::string, int> data);
    int sum(std::string prefix);
};
```

Where the sum method return the sum of all the map values whose key starts with the prefix. If the map contained {"apple":3, "apricot":5, "pear":7}, then sum("ap") would return 8 (the sum of the apple and apricot entries.)

**copy-constructible types**

In C++14, which of the following types is not copy-constructible?

PICK ONE OF THE CHOICES

- ☐ std::string
- ☐ std::shared\_ptr<std::string>
- ☐ std::unique\_ptr<std::string>
- ☐ std::weak\_ptr<std::string>
- ☐ All of the above are copy-constructible

[Clear selection](#)

[Submit answer & continue](#)

You can change your submission later.

Feel free to modify

?

1

2

3

4

5

6

7

8

9

☆ Concurrency

When the same location in memory is accessed by one or more threads, and at least one access is a write, a \_\_\_\_\_ occurs.

PICK ONE OF THE CHOICES

☐ deadlock

☐ race condition

☐ segmentation fault

☒ priority inversion

[Clear selection](#)

Submitted: D

Submit answer & continue

You can change your submission later.

?

1

2

3

4

5

6

7

8

9

10

11

12

13

14

☆ Optimization

```
void foo(std::list<int> &counts) {
    for(auto &c : counts) {
        c++;
    }
}
```

Counts contains over 4 million elements. Through benchmarking, we've determined that this function is unacceptably slow. Which of the following changes would yield the greatest performance improvement?

PICK ONE OF THE CHOICES

☐ make foo an inline function

☐ iterate over the list backwards

☐ use a prefix increment instead of a postfix increment in the loop body

☒ refactor your code to use std::vector instead of std::list

☐ use a classic for loop instead of the C++11 range-based for loop

[Clear selection](#)

Submitted: D

Submit answer & continue

You can change your submission later.



## Feel free to modify

☰

?

1

2

3

4

5

6

7

8

☆ vtables (virtual lookup tables)

Why do virtual function calls have a reputation for being slow?

PICK ONE OF THE CHOICES

☐ virtual functions typically require the object's vtable pointer be dereferenced

☐ virtual functions typically require copying the "this" pointer into the vtable

☐ virtual functions typically require copying the vtable pointer into a hidden std::function object

☒ virtual functions typically require "slicing", which can cause a vtable pointer reassignment

[Clear selection](#)

Submitted: D

Submit answer & continue

You can change your submission later.

9

10

11

12

13

14

☆ std::map

Internally, std::map is implemented as a

☰

?

1

2

3

4

☆ CRTP

In this problem, you will implement a non-virtual base class template using the [Curiously Recurring Template Pattern](#), or CRTP.

Your task is to define the Comparable template, implementing the following functions in terms of the Derived class `operator<`:

- `operator==`
- `operator!=`
- `operator>`
- `operator<=`
- `operator>=`

Additionally, each function should be noexcept and const-qualified. Instantiations of the Comparable template must be eligible for the empty base class optimization (i.e. stateless).

☰

?

1

2

3

4

5

6

7

8

9

10

☆ Reference to tuple element

```
auto a = 0;
auto b = '\0';
auto c = "0";
auto t = std::make_tuple(a, b, c);
auto &foo = std::get<2>(t);
```

What is the type of foo? (Hint: it's automatically deduced by the compiler)

PICK ONE OF THE CHOICES

☐ int &

☐ char const &

☐ char const \*&

☐ char &

☐ auto &

[Clear selection](#)