作者：policedog  
链接：[https://www.nowcoder.com/discuss/77928](https://www.nowcoder.com/discuss/77928" \t "_blank)  
来源：牛客网

携程是我春招第一次现场面试，之前面了好几家都是电面或者视频面，所以还略有一点紧张，去北京的高铁上还一直在看自己的笔记，结果看的都没问

到了酒店之后等了二十分钟左右就开始面试了，面试官是个总监模样的中年大叔，果然他自我介绍也是某个部门的技术总监，先让我先自我介绍，然后就开始面试了。

1. 先问项目，让我画项目架构图，问项目做压测了吗，吞吐量有多少，是否优化过

2.秒杀系统场景下怎么防止超卖，redis和数据库数据不一致怎么办，以什么为准。

**方案：如何解决秒杀的性能问题和超卖的讨论**

**抢订单环节一般会带来2个问题：**

**1、高并发**

**比较火热的秒杀在线人数都是10w起的，如此之高的在线人数对于网站架构从前到后都是一种考验。**

**2、超卖**

**任何商品都会有数量上限，如何避免成功下订单买到商品的人数不超过商品数量的上限，这是每个抢购活动都要面临的难题。**

1. ****前端****

****B：静态化****

****将活动页面上的所有可以静态的元素全部静态化，并尽量减少动态元素。通过CDN来抗峰值。****

1. ****后端****

****将存库从MySQL前移到Redis中，所有的写操作放到内存中，由于Redis中不存在锁故不会出现互相等待，并且由于Redis的写性能和读性能都远高于MySQL，这就解决了高并发下的性能问题。****

****利用redis中基于cas的乐观锁****

****秒杀开始之前，将库存量放到Redis当中，然后，对待每个请求，先判断是否大于0，是的话，就去进行库存量减一操作，如果成功，则商品信息加入购物车当中，如果失败，则不能加入到购物车，返回秒杀失败；小于0则直接返回秒杀失败。****

1. ****数据库****

****（使用缓存、使用索引、批量读取、延迟修改、分布式数据库、分表分区）****

1. ****服务器****

****（集群、分布式、CDN、反向代理）****

3.分布式session怎么处理的。

4.堆内存和栈内存，详细说说。

5.怎么判断一个对象是否需要垃圾回收

**（引用计数器和根搜索算法）**

1. 解释一下引用计数法和gc roots可达性分析法
2. 垃圾回收算法，介绍几个垃圾回收器

8.遇到过full gc频繁的情况吗？怎么定位排查呢

9.简单的代码题

插入排序，数组最大子序列和

题本身很简单，但是自己很紧张，脑袋一团糊，虽然写出来了，但是有一个边界条件没有判断好，面试官有点不满，意思大概说我太毛躁了，写代码应该先想好再下笔。

然后就开始玄学聊天了，让我问问题

我就问了携程的实习生培养情况，实习能不能接触到实际的开发项目，有没有人带，部门内部的技术栈之类的，面试官很随和，聊了很久，说了携程的实习生和应届生的培养什么的，然后给了我一些建议。

出来之后HR说只有一轮面试，大概一周后出结果

作者：春招小能手  
链接：[https://www.nowcoder.com/discuss/77478](https://www.nowcoder.com/discuss/77478" \t "_blank)  
来源：牛客网

三条算法：

1. 走台阶1，2，3步，请问到100步可以有多少种走法。我用的fn=fn-1+fn-2+fn-3递归做的，他说可以用动态规划减小复杂度。然后问动态规划和分治的区别是和前一个状态一个有关一个无关

1. m个数里取n个最大数，我用最小堆解决，复杂度mlogn

3.正负数组里面连续子数组和的最大值。我用暴力法，其实可以优化。

 其他：

string和stringbuilder

1. String类型的引用可以变、但是内容不可变，属于线程安全的；
2. SB内容是可变的，是属于线程非安全的，执行效率比较高；
3. 当进行字符串大量拼接操作的时候，应尽量避免使用String，尽可能使用SB）

jvm内存模型，和内存分配

（

1、工作内存（JVM栈、本地方法栈、程序计数器）主内存（Java堆 新生代 老年代、方法区 永久代）

2、JMM Java内存模型，是抽象概念、定义了一组规范，用于解决多线程环境下的线程安全问题，并提供了内置的解决方案和外部的可使用的同步工具类，从而保证程序在多线程环境下的执行的原子性、可见性、有序性；

3、JVM垃圾回收

）

gc回收机制，

（

1. 垃圾回收作用：减少内存泄漏、减少了程序员工作量、提高了系统安全
2. 什么时候进行垃圾回收：1）当应用程序空闲或者系统中没有程序运行的时候；2）当Java堆内存不够的时候
3. 哪些是gc roots：JVM栈中引用的对象，本地方法栈中引用对象，方法区中静态变量或者常量所饮用对象
4. 如何确定哪些对象是垃圾对象？：引用计数器算法、根搜索算法
5. 垃圾回收算法：复制算法、标记清除算法、标记整理算法、分代回收算法
6. CMS concurrent mark sweep并发的标记清除算法的垃圾回收器，其主要目的是：减少垃圾回收时间，降低应用程序等待暂停的时间；其垃圾回收分为六个步骤，包括初始标记阶段、并发标记阶段、并发与清理阶段、重新标记阶段、并发清理阶段、并发，主要用于老年代的垃圾回收，但是会产生 内存碎片

）

hashmap和hashtable区别。

（

1、hashtable线程安全，get put方法都是有synchronized修饰，还是马屁线程非安全

2、hashmap允许null ，还是他变了不允许null

3、hashmap初始table数组大小16，而hashtable初始13

4、扩容方式，hashmap 2倍 hashtable 2倍+1

5、数组+链表实现

）

设计模式

（

1、单例模式（饿汉式、懒汉式 private static）

2、工厂模式（beanFactory ）

3、策略模式（定义一组算法，并封装、相互之间可以替换）

4、模板方法模式（定义了一个算法框架，将具体的一些步骤推迟到子类中去实现）

5、代理模式（提供了目标类的另外一种访问方式，通过代理模式，在不改变目标类代码的前提下，可以通过代理对象扩充目标类的功能）

6、观察者模式：定义了对象之间的一对多的依赖关系，当一个对象状态发生变化的时候，依赖这个对象的其他对象将会收到通知并进行更新；如**侦听器**

）

作者：奇零可草  
链接：[https://www.nowcoder.com/discuss/76465](https://www.nowcoder.com/discuss/76465" \t "_blank)  
来源：牛客网

1.单例模式和多线程有关系吗？

首先写个单例：

1. **public** **class** SingleDemo {
2. **private** **static** SingleDemo s = **null**;
3. **private** SingleDemo(){}
4. **public** **static**  SingleDemo getInstance(){
5. **if**(s == **null**){
6. s = **new** SingleDemo();
7. }
8. **return** s;
9. }
10. }

改方法在多线程环境下，会出现多个对象实例

下面改进单例：

1. **public** **class** SingleDemo {
2. **private** **static** SingleDemo s = **null**;
3. **private** SingleDemo(){}
4. **public** **static** **synchronized** SingleDemo getInstance(){
5. **if**(s == **null**){
6. s = **new** SingleDemo();
7. }
8. **return** s;
9. }
10. }

加入同步函数后线程安全问题解决了

运行多次都是获取同一个实例，不会出现2个实例的情况了

**但是在多线程并发访问的情况下，每个线程每次获取实例都要判断下锁，效率比较低**，为了提高效率，我**加入了双重判断的方法，解决了效率的问题**

1. **public** **class** SingleDemo {
2. **private** **static** SingleDemo s = **null**;
3. **private** SingleDemo(){}
4. **public** **static**  SingleDemo getInstance(){
5. /\*如果第一个线程获取到了单例的实例对象，
6. \* 后面的线程再获取实例的时候不需要进入同步代码块中了\*/
7. **if**(s == **null**){
8. //同步代码块用的锁是单例的字节码文件对象，且只能用这个锁
9. **synchronized**(SingleDemo.**class**){
10. **if**(s == **null**){
11. s = **new** SingleDemo();
12. }
13. }
14. }
15. **return** s;
16. }
17. }

**用这种方式解决了懒汉式的线程安全问题，也提高了效率，**

2.面试提问也很累的，说说你擅长什么吧？

我说**集合框架和JVM还挺熟的**，让我说一下容器中有几个接口，Hashtable的父接口是谁，然后说我这个回答的还行

3.我说还要解释JVM内存中其他的部分吗，面试官说不用了，劝诫我把技术搞的深一点，说他们想找那些技术做的比较深的人，暗示我已经凉凉，，，，

**总结：**携程只有一面，感觉面试官是个技术大佬，我感觉我没get到他的点，估计是挂了

作者：ktao  
链接：[https://www.nowcoder.com/discuss/75323](https://www.nowcoder.com/discuss/75323" \t "_blank)  
来源：牛客网

自我介绍  
10亿个数，找出最大的10个（**最小堆**）  
两个数组的交集  
ArrayList Linkedlist hashset  
说一下hashmap  
线程池  
系统线程数量上限是多少？  
排序算法中，哪一个最快，说说原理  
用过哪些数据库？数据库索引？  
感觉要凉，面试官说我只知道用，不懂得研究实现原理。。

1. 自我介绍  
   2.问简历中的项目  
   3.http协议  
   4.设计模式  
   5.排序 冒泡 快排  
   6.算法 kmp  
   7.手写代码

[1~100自然数，已随机放入一个有98个元素的数组a[98]。写出尽量简单的方案，找出剩下的2个数并输出。](http://blog.chinaunix.net/uid-26382417-id-3269675.html)

**1、 **解方程****

   求出 x+y=？

   求出 x\*y=?

  然后解方程，但是一个问题100！将是个很大的数，无法做到

**2、 **堪称经典**----**--------------一数组值做另一数组下表

  //a[98]中存放着原数组；  
int i=0;  
unsigned char b[100];  
for(i=0;i<100;i++)  
  b[i]=0;  
  
//依次填入b[100]，b[0]=1,b[1]=2,b[2]=3,.......  
for(i=0;i<98;i++)  
  b[a[i]-1]=a[i];

//没有被填入数字的应该还是0，b[n]=0  
for(i=0;i<100;i++)  
  if(0==b[i]) printf("num:%d \n",(i+1));

作者：倾听潮汐  
链接：[https://www.nowcoder.com/discuss/42299](https://www.nowcoder.com/discuss/42299" \t "_blank)  
来源：牛客网

今天刚拿到携程，发一波面经，直接上干货（牛客网上视频面）

1.链表的定义

2.怎么实现链表翻转

3.怎么判断链表是否有环

4.二叉平衡树，怎么用一维数组存储

5.讲讲jvm**分区**

6.讲讲jvm gc

7.怎么求数组的最大子序列和

8.**final关键字4种用法**

9.sleep用法

10.await关键字

11.线程池

12.spring ioc，aop以及其优点

13.栈和堆的定义

现在记得的只有这么多了。

作者：神蜗牛  
链接：[https://www.nowcoder.com/discuss/36821](https://www.nowcoder.com/discuss/36821" \t "_blank)  
来源：牛客网

hashmap底层实现原理，负载因子，冲突处理，什么时候扩容

临界区

线程实现方法

线程池实现原理

java虚拟机分区

堆分区

tomcat配置文件

单链表判环

怎么统计北京市自行车数量

有没有女朋友

为什么之前去小公司实习

作者：愿時光温柔待你  
链接：[https://www.nowcoder.com/discuss/15340](https://www.nowcoder.com/discuss/15340" \t "_blank)  
来源：牛客网

【电面】

接到电话是在上午9点，因为有事就推迟到了下午，差不多一点半的时候再次接到电话~~港真，电面小哥声音着实好听~~

没有让自我介绍，直接提问：

1、简历上的项目是怎么来的？是兴趣还是接的？

2、项目里面用到了那些框架与技术，简述一下。都是自己做的么？

3、你常用的数据结构有哪些？

4、HashMap是如何实现的？

5、进程和线程的区别？

6、如何实现线程同步？

结束后让等后续通知。

【现场面】

9月30号邮件通知面试，自行预约面试时间。

于是约了面试当天下午2点半，早早去了，然后大概3点被点名面试。

面试官很年轻，大概30不到。问了一下有没有最新的简历，然后让自我介绍一番。

提问如下：

1、介绍一下你认为最突出的项目。

2、项目里用到的技术，为什么用这个技术。

3、**讲述一下HTTP的原理**

4、HashMap是怎么实现的？

**5、设置负载因子的目的是什么？**

所以会进入resize()方法，到达这段代码：

//取消多余的代码，我直接贴上进入的这个分支的代码else {

newCap = DEFAULT\_INITIAL\_CAPACITY;

newThr = (int)(DEFAULT\_LOAD\_FACTOR \* DEFAULT\_INITIAL\_CAPACITY);

}

newCap的意思是：即将用于hash的桶数组的长度，这里的默认值是16   
newThr的意思是：当桶中的键值对的个数超过这个值时就进行扩容，这时候上面提到的**负载因子就起作用了，所以这里的newThr为0.75×16 = 12**

所以经过这个默认构造方法，并且进行第一次put后，桶数组建立了，桶容量为16，**极限值为12。**

6、如何实现线程同步？

7、HashMap如何实现线程安全？

8、如何让HashMap保证在多线程写数据的同时，仍然可以被其他线程读数据？怎么做？

9、最后一个代码题：实现链表反转。

面试官检查一下代码后，问我还有没有问题要问他。问了一个业务相关的问题。之后他说今天就到这里了，让我回去等通知。

话说，同去的其他同学都面了三面.. 我又找HR小妹确认了一下只一面就让回去的事情，HR小妹说不同部门流程不同，我报的部门只有1面。最后说20号以后会给通知。