题目是那道love message的题  
提醒大家注意的是，可能不是传统意义的top k。。  
上来先是输出most loved，然后变成most k  
首先怪楼主忘记跟面试官讨论如果多个并列怎么办。。直接说了min heap的想法，小哥同意以后就做了  
然后小哥问如果只有dataset两个，且frequency都是1怎么办，最后我改成了max heap输出top rank k。。  
唉。。做题太心急了，发出来帮忙给大家提个醒，不要像我一样，还是要多跟面试官讨论清楚所有情况。。

coding  
- top k love message（再次感谢这位楼主的整理 [https://www.1point3acres.com/bbs/thread-542585-1-1.html](https://www.1point3acres.com/bbs/thread-542585-1-1.html" \t "_blank)）  
- follow up：如何处理 spam？数据结构是否能更优化？  
- 时间空间复杂度？

coding topk followup：streaming 输入，输出最近10分钟的topk

然后我的基本流程是： 一个印度小哥 人很nice，上来聊了5分钟他们的项目，然后聊了10分钟我的项目。接着做题，我选到的题目是那个 Love message。。简单来说就是TOP K （利口 刘舅儿）。然后我吧啦吧啦讲了我的两种方法，一种是O(nlogn) ，一种是O(nlogk)。然后我就坦白说，因为我比较熟悉O(nlogn)的办法，但是我可以继续优化，然后小哥说 都可以。  一切都很美好除了我自作主张在方法的开头加了一个判空的代码，但是一步小心把 == -> != 导致 打印不出来。然后我就找了很久的时间。  然后被印度小哥 在线抓虫。但是也没有问什么fellow up。  很绝望， 怎么就管不住手呢。。。。所以同志们，要注意检查题目。希望有onsite吧。。。。

coding一轮，love msg.  follow up有怎么处理spam:用set记录发送者，排序用size排。又问:如果spam在内容里面咋办，答把关键字存数据库里面，或者存tire里面，过滤。后来小哥实在想不出啥问题了，问可能有哪些spam内容😂

面试的流程是一开始面试官讲了他是data string组的，然后花了2到3分钟做自我介绍。面试官直接就贴题，就是一个top k的题，是统计网址的访问次数。楼主花了10分钟讲了一下思路。面试官在我提出想要使用maxheap的时候直接否决了我。他直接让我阐述最优的解。我说明了用minimum heap的解法。然后present了所有的case。面试官表示可以开始写code了。  
通过这次面试发现自己的面试节奏太慢了一些。在简单的题型下，面试官主要考察的还是思路的顺畅性。因为楼主讲的太慢了，导致面试官没有时间提出follow up。希望大家如果能够面对相同难度的题时，尽量把时间控制在20分钟内。

top K love message. 题目如下  
给你一个 Love Class  
class Love {  
    String sender;  
    String receiver;  
    String message;  
    public Log(String s, String r, String m) {  
        sender = s;  
        receiver = r;  
        message = m;  
    }  
}  
  
  
input = new Love[] {new Love("aaa", "bbbb", "message1"), new Love("aaa","dddd", "message2"), new Love("aaa","cccc", "message2"), new Love("cccc",“bbbb", "message2")}  
  
让你求收到love message Top k的 比如过这个例子就是就是 bbbb， 2次， dddd 1次， cccc 一次， 如果收到次数相同不需要二次排序，任意顺序返回即可。  
  
解法： 用min heap，维护一个大小为k的priorityqueue，最后把所有的节点都poll出来，Collections reveres即可。  
  
一些需要注意的细节：  
1. 楼主一开始写了一个 class Node {String receive, int frequency} 以及一个comparator去排序，貌似小哥觉得这样太麻烦了，问了一下why you want to use Node class。 楼主说一下，小哥说ok ok你用把。  
后来写完了才发现有一个更简单的写法（小哥说的）， **就是直接用一个HashMap<String(receiver), Integer(出现的次数)>**， 然后写comparator的时候，调用这个hashmap比较就可以了。  
  
2. 一些edge case看你有没有写，比如过input == null， 比如说 k 小于 0， 比如说 k 等于 0  
  
follow up：  
如  
new Love("aaa", "bbbb", "message1")  
new Love("aaa", "bbbb", "message2")  
同一个 sender send 到同一个 reveiver 多次，算作spam，这个时候只能算一次，怎么求top K。  
  
答案：  
用一个hashmap<String, HashSet<String>> 存sender send 了哪些receiver，如果这个receiver已经有了， continue就好  
  
各位看官如果都看到这里了，求点米哈～我想攒点米看面筋，大家都互帮互助一下呗～这个面筋已经很详细了，应该可以解决你对这个题目的疑惑～给别人米不扣分的哈，谢谢～

A:

follow up我觉得不要用map<sender , set>。一个sender发给一个receiver一次，也就是一个receiver 只能收到同一个sender一次，这样用map<reveiver , set>是一样的，而且和之前用的map都是同样的key, 所以用同样的思想稍微改一下就好。这样可能比用map<sender , set>更好一些。:)

考点 top k，（题目： love message，给一个list， list 的element包含： 发送方，接收方和message. 求收到message频率最高的k个receiver）follow up 比较多，但是就这题目也没什么花样，譬如，换成sender，boundary case譬如两人频率都是10， 怎么处理（我感觉是按需求来处理）谁能说说这个答案是啥。聊天聊得很愉快。 因为他说的所有知识我都学习过，所以聊的时候比较有底气。

题目top k....不说了, yelp貌似只会问这一题..  
  
然后, follow up, 他让我想这个系统怎么作弊(刷love), 然后怎么防作弊  
我想了两个办法  
1. A给自己刷love. 这个好做, 直接禁止自己给自己发msg就行了.  
2. A叫自己所有的朋友给自己刷love. 这个需要设定一个window, 限制A给某个人在window期间发love的数量. 如果超过阈值, 就不在计入  
然后他让我实现一下第二个, 楼主想着是个stream, 然后要用window来弄, 搞了一个hashtable+双链表, 然后给他一顿画图, 讲完之后, 他说不用这么麻烦, 咱们就一周算一个chunk, 每周清零就行....楼主一脸懵逼, 一看时间只有10几分钟了  
赶紧写代码, 还好思路简单, 就存一个X发给Y的msg数量的字典, 然后存进去的时候直接查有没有超过limit即可.   
然后run了一个小的test case 一个大的test case.  
然后让分析复杂度, 最后问了几个问题, 谈笑风生一波. 

刚结束的电面。挺常规的top k问题。一堆log，返回频数最高的k个url。讨论规则讨论了半天。写的有点慢。经典的follow up面试官都没有提。是不是凉了?  
用的他们指定的online ide。面试官还稍微迟到了5分钟，而且，只有图像，没有声音。面完后，感觉不是很好。期待有后续吧。

A: 一堆log，里面有url，timestamp。让你统计访问频次最高的k个页面。

然后就直接出题目了  
给一堆log, 里面有url和timeStamp  
  
第一问:

return top K most visited url, 这也是之前许多面经中提到的map + sort(or pq) 的top k题, yelp真的很喜欢这个Time: O(nlogn) -> O(nlogk)  
Space: O(n + k)

follow up:

加了两个method, 且变成了实时系统:  
第一个: getTopKMostVisitedUrlInPast10Min(int k)  
第二个: processNewLog(Log log)  
明显, 第二个函数在实现时要照顾第一个  
第二个: 函数需要用到双向Queue, 也就是Deque, 实时维护deque里的ts全部都是10分钟内的, 挺好想的, 关键是怎么告知对方, 建议多练练, 模拟模拟  
第一个: 根据Deque的size, 把map里面的entry放到不同的bucket里面, 在getTopK的时候从大的bucket往小的收割, 割够了就直接返回  
  
由于时间紧张, 我只写了第二个的pseudo code, 第一个只阐述了思想, 画了一点简图, 对方一直是poker face, 说着perfect, cool, 但愿他是真这么想

https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=544119&extra=page%3D4%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3046%5D%5Bvalue%5D%3D36%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline

全程没问项目背景和经验，一来就问你想做什么类型的工作，是不是任何组都可以加入？然后稍微聊了两句就准备开始coding。题目还是地里原题，Love message那道，follow up也一样，可以自行搜索一下。既然是找最大嘛又要nlongk,那就用一个min heap就行了，当时code的很顺，只不过最后一行我脑子秀逗了，因为最小堆poll出来结果顺序是反的，要再反过来一下，我用了Collections.sort(result, Collections.reverseOrder());因为排序的是String，就用的字母顺序sort了，导致结果不对。其实直接用Collections.reverse（）就行了，不过我倒是反秀了一波debug能力。然后follow up啥的就是去除spam，那直接把map的value改成一个set就行了，最后看set的size。不过这个小哥问的非常的细，让你解释各种你这么写的原因，包括你哪些地方你觉得可以加comment啊，你这里用了Map<> = new HashMap<>，为啥这么声明啊. Heap里面你为啥自己建了一个新的类不直接用Entry啊，非常很多联系到一些基础的小问题不时的穿插，但我感觉我都答得挺好因为我能看到他满意的表情2333. 总之最后面的时间超了15分钟，全程聊得很愉快，我心想应该没啥问题。

算法 topk from data stream + follow up 从[系统设计](http://https:/www.educative.io/courses/grokking-the-system-design-interview?affiliate_id=5749180081373184/" \t "_blank)角度考虑怎么设计api

coding ： 常见的发送love，follow up 有 如何防止spam，比如说同一个人不能连续对某人发送. 还有如果输入是data stream怎么处理.

A: 输入是连续的string的input，分别将他们三个一组的分为三个variable的Love class， 然后存在list中，然后再继续process

一道love message的题 给list of hashmap, 每个hashmap里面有sender的邮件，receiver的邮件，love message的内容  
求top n个收到love message的人

高频top k love那道题，followup1: 如何优化空间？max heap-> min heap, followup 2: 如果是real application，该怎么improve code？前后端怎么分？ followup3: 感觉小哥实在想不出问题了，问如果production code该怎么办？我加了10分钟comment，改varaiable name....嗯 没问前端...

上周五刚面完，就是设计一个love message的题，比如给一个list，形式诸如[{"sender":'\*\*, "email":\*\*,"receiver":\*\*},{"sender":'\*\*, "email":\*\*,"receiver":\*\*}....]，设计一个算法获取拿到最多message的人，后续问如果要写另一个设法获取发出最多love message的人，怎么抽象一个公共的方法，然后问如果有duplicate怎么处理。希望有Onsite, 求点大米，祝各位求职顺利~

给了一个点赞的list，每个元素有以下三个成员：  
接收者  
信息  
发送者  
  
然后这个list比如就是这样：

Comment[] comments = new Comment[]{

               new Comment("A", "great", "X"),

               new Comment("A", "great1", "X"),

               new Comment("A", "great2", "X"),

               new Comment("B", "great3", "X"),

               new Comment("B", "great4", "Y"),

               new Comment("B", "great5", "Z"),

               new Comment("C", "great5", "X"),

               new Comment("C", "great5", "Y"),  
};  
要求返回被点赞次数最多的K个人，并列的话任意哪个都行。比如如果K是2， 就返回 【A，B】  
之后follow up是，如果同一个人给的赞只能算一次，防止灌水的情况。答案就变成了 【B，C】