



[View in Evernote](#)

Last updated: Dec 27, 2018





>> Recent update:

Filter: linkedin + 在职跳槽 + general.

Page 6

View in Evernote

>> Average Package:

Package: 150k + 250k/4 + 50k + 10%

First year total salary: 277K

Green Card: < half year

>> Positions:

1. Infra : 3 system design, hard onsite interview.
2. App: 1 system design.
3. tools developer experience. easy coding problem. Some system design.

>> System Design:

1. Store email using what database. noSql vs SQL?
2. topk shared URL in 24 hours
3. 设计linkedin的news feed, 基本要求是对所有的news, 每次用户登陆都显示根据某个排序条件（可能是很复杂的多个条件综合算出一个分数后的, 这个计算过程不一定可以被SQL表达）后的top K的新闻, 然后会问怎么做过去1分钟/5分钟/一天的top K, 跨机房啊, 一致性啊这些问题。
4. design calendar. 用户调动自己日历, 调取别人日历, 增加会议。Outlook
5. log process
6. 设计graph题, 给定一个 int[] getFridend(int user), O(1) complexity, 求两个users 是不是一级联系, 二级联系和三级联系。先在local解, 后来问图很大, 怎么scale到多个machine上
7. 实现线程池/ExecutorService, 跑需要延时的任务
8. publish and query. 存event id和内容, 内容大小不定。然后query API是 getEventSince(id, int bufferSize), 求怎么存储数据可以更利于high concurrency。这题答得不太好, 要了好多hint。最后面试官还说到想问lock, 但是没时间了。
9. tiny url. 最终挂了, 反馈是系统设计短网址不会架设多个服务器来处理请求,
10. 设计一个文本数据库, 对于query 高效的返回某个key在数据库中出现的所有位置, 出现次数是否大于k
11. 设计一个传感器网络, 有大量的传感器实时发送数据, access point 向这些传感器发送query,如何快速得到结果
12. <https://www.1point3acres.com/bbs/thread-294114-1-1.html>
13. <https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=175538&extra=&highlight=linkedin%2B%CF%B5%CD%B3&page=1>
14. <https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=446923&extra=page%3D3%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D6%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D2%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>
15. 闲扯单机的 再cross node.
16. append only datastore like HDFS
17. 设计. 三哥. 设计metrics collection and monitor svstem. 收集host的1second. 1minute. 1hour



metrics。如何scale, 如果是10000 hosts怎么收集。提供各种方案, kafka, HDFS, pub-sub model。各种分析优缺点, 然后go over workflow。这轮还算比较顺利。

[View in Evernote](#)

18. Design search index.

19. distributed hash table

20. Design 一个message streaming system. 每个message 可以有 4Bytes ~ 16 KB.需要考虑的有retention, work load, memory overflow, race condition。只需要考虑一个machine。不说 high level idea, 说具体的components, data structure

21. 求平均值 + follow up

22. Linkedin Infra 3 system design: 就是KV Store (~自制db), 找朋友, build invert index(/solr), 设计类kafka系统。。

23. <http://massivetechinterview.blogspot.com/2015/06/algorithm-how-to-count-number-of.html>

24. 倒排索引很多怎么存可以快速查找. Hot to make it distributed

25. design instagram

26. 设计一个系统可以统计出过去24小时里面top k exception。具体要聊到数据怎么从数据中心发到这个server端, 然后怎么存数据, 怎么生成top k exception

27. . 设计一个系统, client会发1B-16M的message到server, server要把message存在memory里并返回一个id给client, client可以用id fetch message。很细致的讨论了存哪些metadata, 每个metadata到底占多少空间, 为什么每个message都存metadata会非常不efficient, 怎么样优化存metadata的方式等等。

28. 假设已有如下两个function Output Compute(Input in) Output Merge(Output o1, Output o2) 要求在最多可以有K个thread的情况下实现 Output MergeAll(List<Input> inputs)。实际上是个multi-thread programming的问题, 需要写出比较detail的代码。

29. 然后看时间还挺多, 就出了道system design的题, 就是如何autocomplete, 先用trie树, 然后再聊到用cache来记录大部分结果。这轮问题不大。

30. <https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=427788&extra=page%3D5%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D6%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D2%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

31. , 输入是文件, 要求query文件里面的内容, 这轮死了。

system design, 要求就是特别简单, client 存文件, 然后user query 内容, 文件是1MB。这个我也不知道怎么设计比较好。

当时有几个问题就是如果client 同时往一个database里面写的话需要lock, 怎么解决, 我就想说多分几个database 存, 但是呢就需要merge, 多久merge 一次。

面试官说的很清楚就是存文件, 大小确定, query里面的content, 返回文件。这些文件得存啊, 怎么存呢, 怎么存才能更快地query, 我一个数据库存文件。

32. Coding design: m blocks, each block has m bytes, support files read/write, design metadata and data, need to allocate memory, free, update and get

33. , 问一个design LinkedIn homepage, 需要有发一条状态share出去, 和显示feeds的功能, 主要问了很多database的问题, 感觉是个要求特别严格的manager, 有点不确定答的是否满足他的要求

34. ip blacklist service

35. 一个精心包装的 KV 缓存设计, 要能保存 500M 个网页, 查询时间不得多余 1s, QPS 等需要自己问。

36.



[https://www.1point3acres.com/bbs/forum.php?](https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=429962&extra=page%3D5%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D6%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D2%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline)

[mod=viewthread&tid=429962&extra=page%3D5%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D6%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D2%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline](https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=429962&extra=page%3D5%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D6%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D2%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline)

领英有很多host上面跑各种不同的services。作为码农需要各种ssh到不同host上查log特别麻烦。能否设计一个系统解决这个问题

基本就是设计一个centralized log service。可以参考splunk设计。先分成几个部分

1. 各个host上的log collector
2. 中间录入log的processing server和data store
3. 前端查log的UI和REST API server

然后每个部分依次dive in。这里赞下面面试官，过程当中有很多引导性的问题，不会出现stuck或者完全没方向的情况

1. collector

- 最straight forward的办法service写log file, 另一个process边读边发, 会有什么问题? 如何改进?
- log一条条发还是攒一波一起发? 怎么确定上次发到哪?
- 如果一个host上跑多个service, 如何避免单个service太noisy占I/O等资源导致其他service发不了log?
- logging process crashed了怎么办? 如何避免data loss?

3. UI/API

- 有哪些基本的use case? 各自的REST API怎么写?
- 如果要支持根据app name/hostname等来查询log, 后端需要做什么改动? 还有哪些可能的query criteria?
- 出现一次查询返回1B log的情况怎么办?
- 除了看log之外, 这个系统还能提供其他一些什么功能来improve quality of life? 后端需要做什么改动?

2. processing service + data store

- 数据库用sql还是nosql, 为什么?
- schema design
- 数据库怎么扩展 (提示: 这里是write >> read的情况), 以及如何避免单点failure?
- 为什么需要一个processing service而不是直接让collector写数据库?
- processing service用sync还是async?
- 如何scale processing service? 扩展后如何避免同一个log写入多次?

>> CS knowledge:

1. HTTP request
2. virtual memory. Why use virtual memory.
3. stack vs heap
4. multi-thread/process communication



4. multi thread/process communication.

5. why thread switch faster than process switch? difference between thread and process.

[View in Evernote](#)

6. LRU, 多线程怎么办, 能不能用读写锁

7. check if a graph is a tree

8. java: final, finally, finalize

9. mute semaphore vs binary semaphore

10. what is memory map file

11. what is db transaction? How to ensure durability? how to restore if crash before commit.

12. TCP vs UDP. Which layer.

13. mmap是什么, 读和写shared memory object操作系统内部会发生什么? -ba

14. Database Transaction

15. **Write-back vs Write-Through.**

16. hadoop vs spark

17. java里的final, finally, finalize

18. How does processes communicate with each other? IPC: different ways, RPC, shared memory, message passing, pipes, socket

19. 多个application 同时运行, memory 不够用了, OS 会怎么办。OS 如何 free, allocate memory的, namely, memory management. 比如page share, compress, 或者swap

20. 给一个文件, 问你用程序读这个文件的时候, OS都发生了什么操作。怎么读大于内存的文件, 怎样优化等等。。这个我答得不是很好, 稍微讲了下思路, 但是说我平时不怎么操作文件, 就过了

21. paging, page demand, page fault, Thrashing,

22. Java exception。问的非常细, 详细描述什么是exception, 有哪些分类, 为什么要用exception, 而不是直接打印出错误 etc.

>> BQ:

1. talk about your weakness, how you overcome it.
2. behavior question: 你在项目中遇到过最难的问题是什么
3. 跟经理意见不一致怎么办
4. how to trouble shooting, how to improve your current project
5. How to improve linkedin website
6. how to code review
7. 簡歷解說, 叫我把他當一個新人, 把現在的部門在做的事情介紹給他聽
8. why cassandra
9. how much linked open source you know about?
10. Project you proud of
11. largest challenge.
12. why come linked?
13. 聊工作, 怎么提高developer 的 productivity
14. your strongest and your weakness
15. 选一个你觉得最自豪的项目, 然后标准是把面试官当作new hire on board, 给他讲这个project, 关注大框架, 如何design, 同时也会问detail, 基本45分钟吧, 因为前面各自自我介绍5分钟, 后面留10分钟问问题。我的面试官没有问很多, 但之前看到别人也被问类似如果让你重新设计, 你会improve哪里之类的问题。我的面试官是说这轮就是看你有没有technical communication的能力, 就是跟组里人交流你做的东西, 所以既有technical detail, 又关注communication
- 16.



>> LeetCode

[View in Evernote](#)

[5, 12, 20, 30, 33, 34, 35, 40, 46, 50, 56, 57(支持addInterval()和getWeightedSum) 61, 65(don't think about e), 68, 76, 79]

100, 104, 133, 146, 151, 152, 153, 156, 160, s170 (优化find, getSum o(1)), 187,
 200, 200, 205(muti-word), 212, 215, 235, 236, 240, 243, 244, 245, 254, 265, 270, 272, 282, 297
 314, 339, 352, 364, 366, 373(smallest product, without sort)
 432, 464, 474,
 516, 535,
 605, 611, 612, 671(find 3rd minimum or kth minimum value), 698,
 716, 730, 722, 726

>> Other Coding:

flip tree upside and down.

BST top K closest neighbour

实现hashmap, 全部都实现,

given a filter and a iterator to implement a validIterator 也出了解法

第二题是自己实现一个stack, 可以O(1)实现push, pull, getMiddle, 讨论了几分钟写完又跑了一下test.

coding, rotation string

two sum to k sum...

topology sort

N-Queue I, II

Fizz Buzz

word search,

word ladder,

paint house,

list intersection and union,

serialize and deserialize binary tree. follow up: if tree is deep.

implement max heap.

tree level by level

罗马数字转普通数字,

LRU,

find triplet,

是否一个图的所有店可以标记为2个不同的颜色。

Nested List Weight Sum 1&2

interval problem

经典的岛屿 (0, 1二维矩阵) 个数, 改成不返回岛屿个数, 返回每个岛屿的一个点。在白板上写的很认真, 走了几个例子, 然后follow up了一下, 如input矩阵太大了 只能一块一块load怎么办

101. Symmetric Tree

<http://www.1point3acres.com/bbs/thread-279704-1-1.html-baidu>

1point3acres

有一个优化的地方就是, 当无法在cache里面找到key时, 需要从data base 里面去取, 这个时候如果新取到的key的rank 要小于cache里面的最小rank 时, 不需要更新cache

1. 实现一个二叉搜索树, 问我都能实现啥功能, 我说insert, find, findMin/findMax, 然后让我实现了insert.

2. 按行打印出来这个二叉树 (利口妖灵儿)。

3. 这俩题没啥说的, 然后还剩不到半小时了, 问我给一堆点的集合, 找到其中距离指定Point最近的k个点, 这题比较常见

做法是用na 时间复杂度O(nlogk) 然后他问复杂度用的方法 直接问我有没有更好的 我说可以用快排的思路来



做。

平均期望 $O(n)$, worst case $O(n^2)$, 他让我用这种方法写。所以小伙伴们注意这题要会这两种方法。

[View in Evernote](#)

code部分就是让实现一个 hashtable, 给正常的Map interface, 让实现 add, remove, get等。做完以后要求写成 thread safe的, 不能synchronize method或者synchronize this

implement a special stack, peek(), push(), pop(),

pop current max element in the stack, may not the top one popMax(),

retrieve current max element in the stack, same, may not the top one peekMax()

as fast as possible. popmax 和 pop 我刚开始想的都是 $O(N)$, 后来把其中一个优化成 $O(1)$

push(), peek(), pop(),

popMax(), peekMax()

。都是原题。max stack要写 $\log n$ 的优化版本。我用的double linkedlist + priority queue

public class RetainBestCache<K, T extends Rankable> {
Waral 博客有更多文章,. From 1point 3acres bbs

private Map<K, T> cache;. visit 1point3acres for more.

private Map<Long, Set<K>> rankingOfObject;

private DataSource<K, T> dataSource;. 留学论坛-一亩-三分地

private int maxSizeOfCache; 来源一亩.三分地论坛.

. from: 1point3acres

/* Constructor with a data source (assumed to be slow) and a cache size */

/* Gets some data. If possible, retrieves it from cache to be fast. If the data is not cached,

* retrieves it from the data source. If the cache is full, attempt to cache the returned data,

* evicting the T with lowest rank among the ones that it has available

* If there is a tie, the cache may choose any T with lowest rank to evict.

*/

具体参考这个帖子第二道: <http://www.1point3acres.com/bbs/thread-279704-1-1.html>

- : BST找Top K Closet node, 用inorder和heap做的, 面试官貌似又不太了解Java (?), 他问我 queue的offer是什么, 然后又出了道利口160
- 215. Kth Largest Element in an Array变种, find k closest point to (0,0) in 2d plane.
- 然后写代码。给你List<String> list1, List<Character> list2. 让你用list2里面的字母, 组成字符串, 但字符串必须在 list1里面。如何并行计算?
- find and replace in string. 有点像 利口霸散散, 但是不给starting inde, 只给s, source_str 和 target_str. 要注意如何handle corner case, 比如原词是"acc", 替换词是"accacc", 讨论应该输出什么, 如何处理。
- <https://www.geeksforgeeks.org/flip-binary-tree/>,
- <https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=433813&extra=page%3D5%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D6%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D2%26searchoption%5B310>



9%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline

IncreaseFrequency(int) //让某个int的freq + 1, 如果是一个没出现过的数字, freq设为1 View in Evernote
DecreaseFrequency(int) //让某个int的freq - 1, 最低为0, freq=0的数字不会再继续降, 尝试decrease没出现过的数字则报错

int GetFreq(int) // 返回一个freq=input的数字, 有多个数字有那个freq的话返回任意一个, 没有那个freq的数字存在则报错

这真是我自己作死, 三哥先问了个easy, 给两个sorted list, 然后merge成一个, 要求是merge后的list里不能有重复

之后的follow up是merge k个sorted list, 当时太累, 忘了表演推理过程了, 直接说用queue做, 三哥装傻让我给他走个case, 我走完之后三哥直接换题了.....

还是k个sorted list, 可能有重复, 现在让你按序找出所有list中都必须存在的元素, 比如

[1, 3, 5, 7]

[1, 1, 3, 5, 7] => return [1, 7]

[1, 4, 7, 9]

Not under linked tag

course schedule ii	https://leetcode.com/problems/course-schedule-ii/description/
rotate list	https://leetcode.com/problems/rotate-list/
Reverse Words in a String	https://leetcode.com/problems/reverse-words-in-a-string/description/
Longest Palindromic Substring	https://leetcode.com/problems/longest-palindromic-substring/
Minimum Window Substring	https://leetcode.com/problems/minimum-window-substring/description/
Edit Distance	https://leetcode.com/problems/edit-distance/description/



Lowest Common Ancestor of a Binary Tree	https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-tree/description/
Regular Expression Matching	https://leetcode.com/problems/regular-expression-matching/description/
Validate Binary Search Tree	https://leetcode.com/problems/validate-binary-search-tree/description/
Find Minimum in Rotated Sorted Array	https://leetcode.com/problems/find-minimum-in-rotated-sorted-array/description/

View in Evernote

Interval problem

```
public interface Intervals {
```

```
    /** * Adds an interval [from, to] into internal structure. */
    void addInterval(int from, int to);
```

```
    /** * Returns a total length covered by intervals. * If several intervals
    intersect, intersection should be counted only once. * Example: * * addInterval(3,
    6) * addInterval(8, 9) * addInterval(1, 5) * * getTotalCoveredLength() -> 6 * i.e.
    [1,5] and [3,6] intersect and give a total covered interval [1,6] * [1,6] and [8,9]
    don't intersect so total covered length is a sum for both intervals, that is 6. * *
    ----- * ____ * ----- * * 0 1 2 3 4 5 6 7 8 9 10 * */
```

```
    int getTotalCoveredLength();
```

```
}
```

```
public class MyIntervals implements Intervals {
```

```
    List<Length> l = new LinkedList<Length>();
```

```
    @Override
```

```
    public void addInterval(int from, int to) {
```

```
        l.add(new Length(x,y));
```

```
    }
```

```
    @Override
```

```
    public int getTotalCoveredLength {
```

[View in Evernote](#)

```

Collections.sort(l);
int retLength = 0;
Length lastone = new Length(0,0);
for(Length len : l) {
    if(len.x > lastLen.y) {//locate apart
        totalLen += len.y - len.x;
        lastLen = len;
    } else if(len.y > lastLen.y) { //overlapping
        totalLen += len.y - lastLen.y;
        lastLen = len;
    }
}

```

//注意这里不需要考虑如果后一个在前一个里面会怎么样，因为lastLen会维持一样，写一次仍然跟前一个做比较

```

    }
    return totalLen;
}

}

```

```

public class Length implements Comparable<Length> {
    public int x, y;
    public Length(int x, int y) {
        this.x = x;
        this.y = y;
    }
    @Override
    public compareTo(Length o) {
        if(o.x > this.x) return 1;
        if(o.x < this.x) return -1;
        if(o.x == this.x) {
            if(o.y > this.y) return 1;
            if(o.y < this.y) return -1;
            return 0;
        }
    }
}

```

[Terms of Service](#)[Privacy Policy](#)[Report Spam](#)