

九月腾讯，创新工场，淘宝等公司最新面试三十题（第 171-200 题）

引言

曾记否，去年的 10 月份也同此刻一样，是找工作的高峰期，本博客便是最初由整理微软等公司面试题而发展而来的。如今，又即将迈入求职高峰期--10 月份，所以，也不免关注了网上和我个人建的算法群 Algorithms1-12 群内朋友发布和讨论的最新面试题。特此整理，以飨诸位。至于答案，望诸位共同讨论与思考。

最新面试十三题

好久没有好好享受思考了。ok，任何人有任何意见或问题，欢迎不吝指导：

1. 五只猴子分桃。半夜，第一只猴子先起来，它把桃分成了相等的五堆，多出一只。于是，它吃掉了一个，拿走了一堆；第二只猴子起来一看，只有四堆桃。于是把四堆合在一起，分成相等的五堆，又多出一个。于是，它也吃掉了一个，拿走了一堆；.....其他几只猴子也都是这样分的。问：这堆桃至少有多少个？（朋友说，这是小学奥数题）。

参考答案：先给这堆桃子加上 4 个，设此时共有 X 个桃子，最后剩下 a 个桃子。这样：

第一只猴子分完后还剩： $(1-1/5)X=(4/5)X$ ；

第二只猴子分完后还剩： $(1-1/5)2X$ ；

第三只猴子分完后还剩： $(1-1/5)3X$ ；

第四只猴子分完后还剩： $(1-1/5)4X$ ；

第五只猴子分完后还剩： $(1-1/5)5X=(1024/3125)X$ ；

得： $a=(1024/3125)X$ ；

要使 a 为整数， X 最小取 3125。

减去加上的 4 个，所以，这堆桃子最少有 3121 个。

2. 已知有个 `rand7()` 的函数，返回 1 到 7 随机自然数，让利用这个 `rand7()` 构造 `rand10()` 随机 1~10。

（参考答案：这题主要考的是对概率的理解。程序关键是要算出 `rand10`，1 到 10，十个数字出现的考虑都为 10%。根据排列组合，连续算两次 `rand7` 出现的组合数是 $7*7=49$ ，这 49 种组合每一种出现考虑是相同的。怎么从 49 平均概率的转换为 1 到 10 呢？方法是：

1. `rand7` 执行两次，出来的数为 $a1=rand7()-1$ ， $a2=rand7()-1$ 。

2. 如果 $a1*7+a2<40$ ， $b=(a1*7+a2)/4+1$ ；如果 $a1*7+a2\geq 40$ ，重复第一步。参考代码如下所示：

```
1. int rand7()
```

```

2. {
3.     return rand()%7+1;
4. }
5.
6. int rand10()
7. {
8.     int a71,a72,a10;
9.     do
10.    {
11.        a71= rand7()-1;
12.        a72 = rand7()-1;
13.        a10 = a71 *7 + a72;
14.    } while (a10>= 40);
15.    return (a71*7+a72)/4+1;
16. }

```

3. 如果两个字符串的字符一样，但是顺序不一样，被认为是兄弟字符串，问如何在迅速匹配兄弟字符串（如，bad 和 adb 就是兄弟字符串）。思路：判断各自素数乘积是否相等。更多方法请参见：http://blog.csdn.net/v_JULY_v/article/details/6347454。
4. 要求设计一个 DNS 的 Cache 结构，要求能够满足每秒 5000 以上的查询，满足 IP 数据的快速插入，查询的速度要快。
5. 一个未排序整数数组，有正负数，重新排列使负数排在正数前面，并且要求不改变原来的正负数之间相对顺序 比如： input: 1,7,-5,9,-12,15 ans: -5,-12,1,7,9,15 要求时间复杂度 $O(N)$,空间 $O(1)$ 。（此题一直没看到令我满意的答案，一般达不到题目所要求的：时间复杂度 $O(N)$ ，空间 $O(1)$ ，且保证原来正负数之间的相对位置不变）。

updated: 设置一个起始点 j, 一个翻转点 k,一个终止点 L

从右侧起

起始点在第一个出现的负数，翻转点在起始点后第一个出现的正数,终止点在翻转点后出现的第一个负数(或结束)

如果无翻转点，则不操作

如果有翻转点，则待终止点出现后，做翻转，即 $ab \Rightarrow ba$ 这样的操作

翻转后，负数串一定在左侧，然后从负数串的右侧开始记录起始点，继续往下找下一个翻转点

例子中的就是

1, 7, -5, 9, -12, 15

第一次翻转: 1, 7, -5, -12, 9, 15 => 1, -12, -5, 7, 9, 15

第二次翻转: -5, -12, 1, 7, 9, 15

N 维翻转空间占用为 $O(1)$ 复杂度是 $2N$ ；在有一个负数的情况下，复杂度最大是 $2N$ ；在有 i 个负数的情况下，复杂度最大是 $2N+2i$ ，但是不会超过 $2N+N$ 实际的复杂度在 $O(3N)$ 以内

但从最终时间复杂度分析，此方法是否真能达到 $O(N)$ 的时间复杂度，还待后续考证。感谢 John_Lv, MikovChain。2012.02.25。

1, 7, -5, -6, 9, -12, 15（后续：此种情况未能处理）

1 7 -5 -6 -12 9 15

1 -12 -5 -6 7 9 15

-6 -12 -5 1 7 9 15

更多请参考此文，程序员编程艺术第二十七章：重新排列数组（不改变相对顺序&时间 $O(N)$ &空间 $O(1)$ ，半年未被 KO）

http://blog.csdn.net/v_july_v/article/details/7329314。

6. 淘宝面试题：有一个一亿节点的树，现在已知两个点，找这两个点的共同的祖先。
7. 海量数据分布在 100 台电脑中，想个办法高效统计出这批数据的 TOP10。（此题请参考本博客内其它文章）。
8. 某服务器流量统计器，每天有 1000 亿的访问记录数据，包括时间、url、ip。设计系统实现记录数据的

保存、管理、查询。要求能实现一下功能：

（1）计算在某一时间段（精确到分）时间内的，某 url 的所有访问量。

（2）计算在某一时间段（精确到分）时间内的，某 ip 的所有访问量。

9.

假设某个网站每天有超过 10 亿次的页面访问量，出于安全考虑，网站会记录访问客户端访问的 ip 地址和对应的时间，如果现在已经记录了 1000 亿条数据，想统计一个指定时间段内的区域 ip 地址访问量，那么这些数据应该按照何种方式来组织，才能尽快满足上面的统计需求呢，

设计完方案后，并指出该方案的优缺点，比如在什么情况下，可能会非常慢？（参

考答案：用 B+ 树来组织，非叶子节点存储（某个时间点，页面访问量），叶子节点是访问的 IP 地址。这个方案的优点是查询某个时间段内的 IP 访问量很快，但是要统计某个 IP 的访问次数或是上次访问时间就不得不遍历整个树的叶子节点。或者可以建立二级索引，分别是时间和地点来建立索引。）

10.

腾讯 1. 服务器内存 1G，有一个 2G 的文件，里面每行存着一个 QQ 号（5-10 位数），怎么最快找出出现过最多次的 QQ 号。（此题与稍后下文的第 14 题重复，思路参考请见下文第 14 题）。

腾讯 2. 如何求根号 2 的值，并且按照我的需要列出指定小数位，比如根号 2 是 1.41421356237... 我要列出 1 位小数就是 1.4 2 位就是 1.41，1000 位就是 1.41421356237... 等。。

11.

给定一个字符串的集合，格式如：{aaa bbb ccc}，{bbb ddd}，{eee fff}，{ggg}，{ddd hhh} 要求将其中交集不为空的集合合并，要求合并完成后的集合之间无交集，例如上例应输出 {aaa bbb ccc ddd hhh}，{eee fff}，{ggg}。

12.

创新工场面试题：abcde 五人打渔，打完睡觉，a 先醒来，扔掉 1 条鱼，把剩下的分成 5 份，拿一份走了；b 再醒来，也扔掉 1 条，把剩下的分成 5 份，拿一份走了；然后 cde 都按上面的方法取鱼。问他们一共打了多少条鱼，写程序和算法实现。提示：共打了多少条鱼的结果有很多。但求最少打的鱼的结果是 3121 条鱼（应该找这 5 个人问问，用什么工具打了这么多条鱼）。

（<http://blog.csdn.net/nokiaguy/article/details/6800209>）。

13. 我们有很多瓶无色的液体，其中有一瓶是毒药，其它都是蒸馏水，实验的小白鼠喝了以后会在 5 分钟后死亡，而喝到蒸馏水的小白鼠则一切正常。现在有 5 只小白鼠，请问一下，我们用这五只小白鼠，5 分钟的时间，能够检测多少瓶液体的成分？

淘宝 2012 笔试（研发类）：

<http://topic.csdn.net/u/20110922/10/e4f3641a-1f31-4d35-80da-7268605d2d51.html>（一参考答案）。

ok, 这 13 道题加上此前本博客陆陆续续整理的微软面试 187 题: [重启开源, 分享无限--诚邀你加入微软面试 187 题的解题中](#), 至此, 本博客内已经整理了整整 200 道面试题。

后续整理

以下是后续整理的最新面试题, 不断更新中 (2011.09.26)

14、腾讯最新面试题: 服务器内存 1G, 有一个 2G 的文件, 里面每行存着一个 QQ 号 (5-10 位数), 怎么最快找出出现过多次的 QQ 号。

以下是个人所建第 Algorithms_12 群内朋友的聊天记录:

首先你要注意到, 数据存在服务器, 存储不了 (内存存不了), 要想办法统计每一个 qq 出现的次数。

比如, 因为内存是 1g, 首先 你用 hash 的方法, 把 qq 分配到 10 个 (这个数字可以变动, 比较) 文件 (在硬盘中)。

相同的 qq 肯定在同一个文件中, 然后对每一个文件, 只要保证每一个文件少于 1g 的内存, 统计每个 qq 的次数, 可以使用 hash_map(qq, qq_count)实现。然后, 记录每个文件的最大访问次数的 qq, 最后, 从 10 个文件中找出一个最大, 即为所有的最大。更多读者可以参见此文: [海量数据处理面试题集锦与 Bit-map 详解](#)。

那若面试官问有没有更高效率的解法之类的?这时, 你可以优化一下, 但是这个速度很快, hash 函数, 速度很快, 他肯定会问, 你如何设计, 用 bitmap 也行。

15、百度今天的笔试题: 在一维坐标轴上有 n 个区间段, 求重合区间最长的两个区间段。

16、华为社招现场面试 1: 请使用代码计算

1234567891011121314151617181920*2019181716151413121110987654321 。

华为面试 2: 1 分 2 分 5 分的硬币, 组成 1 角, 共有多少种组合。

17、百度笔试题:

一、系统有很多任务, 任务之间有依赖, 比如 B 依赖于 A, 则 A 执行完后 B 才能执行

(1) 不考虑系统并行性, 设计一个函数 (Task *Ptask,int Task_num) 不考虑并行度, 最快的方法完成所有任务。

(2) 考虑并行度, 怎么设计

```
typedef struct{
    int ID;
    int * child;
    int child_num;
}Task;
```

提供的函数:

`bool doTask(int taskID);`无阻塞的运行一个任务;

`int waitTask(int timeout);`返回运行完成的任务 id, 如果没有则返回-1;

`bool killTask(int taskID);`杀死进程

二、必答题 (各种 const)

1、解释下面 ptr 含义和不同

`double* ptr = &value;`

//ptr 是一个指向 double 类型的指针, ptr 的值可以改变, ptr 所指向的 value 的值也可以改变

`const double* ptr = &value`

//ptr 是一个指向 const double 类型的指针, ptr 的值可以改变, ptr 所指向的 value 的值不可以改变

`double* const ptr=&value`

//ptr 是一个指向 double 类型的指针, ptr 的值不可以改变, ptr 所指向的 value 的值可以改变

`const double* const ptr=&value`

//ptr 是一个指向 const double 类型的指针, ptr 的值不可以改变, ptr 所指向的 value 的值也不可以改变

2、去掉 const 属性, 例: `const double value = 0.0f; double* ptr = NULL;`怎么才能让 ptr 指向 value?

强制类型转换, 去掉 const 属性, 如 `ptr = <const_cast double *>(&value);`

http://topic.csdn.net/u/20110925/16/e6248e53-1145-4815-8d24-9c9019d24bd8.html?see_d=1665205011&r=75709169#r_75709169

18、如果用一个循环数组 `q[0..m-1]` 表示队列时, 该队列只有一个队列头指针 `front`, 不设队列尾指针 `rear`, 求这个队列中从队列头到队列尾的元素个数 (包含队列头、队列尾) (华赛面试题、腾讯笔试题)。

19、昨晚淘宝笔试题:

1. 设计相应的数据结构和算法，尽量高效的统计一片英文文章（总单词数目）里出现的所有英文单词，按照在文章中首次出现的顺序打印输出该单词和它的出现次数。

2、有一棵树（树上结点为字符串或者整数），请写代码将树的结构和数据写到一个文件中，并能通过读取该文件恢复树结构。

20、13 个球一个天平，现知道只有一个和它的重量不同，问怎样称才能用三次就找到那个球？（<http://zhidao.baidu.com/question/66024735.html>）。

21、搜狗笔试题：一个长度为 n 的数组 $a[0], a[1], \dots, a[n-1]$ 。现在更新数组的每个元素，即 $a[0]$ 变为 $a[1]$ 到 $a[n-1]$ 的积， $a[1]$ 变为 $a[0]$ 和 $a[2]$ 到 $a[n-1]$ 的积，...， $a[n-1]$ 为 $a[0]$ 到 $a[n-2]$ 的积（就是除掉当前元素，其他所有元素的积）。程序要求：具有线性复杂度，且不能使用除法运算符。

思路： $left[i]$ 标示着 $a[i]$ 之前的乘积， $right[i]$ 标示着 $a[i]$ 之后的乘积，但不申请空间，那么 $a[i] = left[i] * right[i]$ 。不过， $left$ 的计算从左往右扫的时候得出， $right$ 是从右往左扫得出。因为就是当中某个元素 $a[i]$ 的左边所有元素与右边所有元素的乘积，就这么简单。所以计算 $a[i] = left[i] * right[i]$ 时，直接出结果。

22、后 2012 年 4 月 6 日的腾讯暑期实习生招聘笔试中，出了一道与上述 21 题类似的题，原题大致如下：

两个数组 $a[N]$ ， $b[N]$ ，其中 $A[N]$ 的各个元素值已知，现给 $b[i]$ 赋值， $b[i] = a[0] * a[1] * a[2] \dots * a[N-1] / a[i]$ ；

要求：

1. 不准用除法运算
2. 除了循环计数值， $a[N]$ ， $b[N]$ 外，不准再用其他任何变量（包括局部变量，全局变量等）
3. 满足时间复杂度 $O(n)$ ，空间复杂度 $O(1)$ 。

说白了，你要我求 $b = a[0] * a \dots a[i-1] * a[i+1] \dots a[N-1] / a$ ，就是求： $a[0] * a[1] \dots a[i-1] * a[i+1] \dots a[N-1]$ 。只是我把 $a[i]$ 左边部分标示为 $left[i]$ ， $b[i]$ 右边部分标示为 $right[i]$ ，而实际上完全不申请 $left[i]$ ，与 $right[i]$ 变量，之所以那样标示，无非就是为了说明：除掉当前元素 $a[i]$ ，其他所有元素 ($a[i]$ 左边部分，和 $a[i]$ 右边部分) 的积。读者你明白了么？

下面是此 TX 笔试题的两段参考代码，如下：

```

void array_multiplication(int A[], int OUTPUT[], int n) {
    int left = 1;
    int right = 1;
    for (int i = 0; i < n; i++)
        OUTPUT[i] = 1;
    for (int i = 0; i < n; i++) {
        OUTPUT[i] *= left;
        OUTPUT[n - 1 - i] *= right;
        left *= A[i];
        right *= A[n - 1 - i];
    }
}

```

```

1. //ncicc
2. b[0] = 1;
3. for (int i = 1; i < N; i++)
4. {
5.     b[0] *= a[i-1];
6.     b[i] = b[0];
7. }
8. b[0] = 1;
9. for (i = N-2; i > 0; i--)
10. {
11.     b[0] *= a[i+1];
12.     b[i] *= b[0];
13. }
14. b[0] *= a[1];

```

from wasd6081058 上面第二段代码最后一行的意义是：我们看第二个循环，从 N-2 到 1；再看 for 循环中 b[0] 的赋值，从 N-1 到 2，而根据题目要求 $b[i] = a[0]*a[1]*a[2]*\dots*a[N-1]/a[i]$ ，b[0] 应等于 $a[1]*a[2]*\dots*a[N-1]$ ，所以这里手动添加 a[1]。

此外，也有朋友在微博上：<http://weibo.com/1761944724/ydwuMt9bH> 给出了另外一种解法，

如下图所示：

$B_i = A_1 \times A_2 \times \dots \times A_{i-1} \times A_{i+1} \times \dots \times A_n$ 。把 B_i 分成两段，即 $C_i = A_1 \times A_2 \times \dots \times A_{i-1}$ ， $D_i = A_{i+1} \times \dots \times A_{i+2} \times \dots \times A_n$ ，即 $B_i = C_i \times D_i$ 。
对于 C_i ，有 $C_i = C_{i-1} \times A_{i-1}$ ；对于 D_i ，有 $D_i = D_{i+1} \times A_{i+1}$ 。我们可以想象有如下矩形：



在矩形中， B_i 为矩形中第 i 行所有数字的乘积。每一个的数字被对角线上的数字分成两段，用粉红色标注的为 C_i 中的数字，用淡绿色标注的为 D_i 中的数字。我们可以从上到下计算 C_i ，从下到上计算 D_i ，并最终得到 B_i 。

```
public static void multiply(double array1[], double array2[]){
    if(array1.length == array2.length && array1.length > 0){

        // 此时array2中的值相当于分析中的数组c中的值，从上往下计算
        array2[0] = 1;
        for(int i = 1; i < array1.length; ++i){
            array2[i] = array2[i - 1] * array1[i - 1];
        }

        // temp的值为分析中的数组d中的元素的值，从下往上计算
        int temp = 1;
        for(int i = array1.length - 2; i >= 0; --i){
            temp *= array1[i + 1];
            array2[i] *= temp;
        }
    }
}
```

@何海涛Harry
weibo.com/zhedahht

23、腾讯高水平复试题：

- 1.有不同的手机终端，如 iPhone，安卓，Symbian，不同的终端处理不一样，设计一种服务器和算法实现对不同的终端的处理。
- 2.设计一种内存管理算法。
- 3.A 向 B 发邮件，B 收到后读取并发送收到，但是中间可能丢失了该邮件，怎么设计一种最节省的方法，来处理丢失问题。
- 4.设计一种算法求出算法复杂度。

24、人人笔试 1：一个人上台阶可以一次上 1 个，2 个，或者 3 个，问这个人上 n 层的台阶，总共有几种走法？

人人笔试 2：在人人好友里，A 和 B 是好友，B 和 C 是好友，如果 A 和 C 不是好友，那么 C 是 A 的二度好友，在一个有 1 0 万人的数据库里，如何在时间 $O(n)$ 里，找到某个人的十度好友。

25、淘宝算法面试题：两个用户之间可能互相认识，也可能是单向的认识，用什么数据结构来表示？如果一个用户不认识别人，而且别人也不认识他，那么他就是无效节点，如何找出这些无效节点？自定义数据接口并实现之，要求尽可能节约内存和空间复杂度。

26、淘宝笔试题：对于一颗完全二叉树，要求给所有节点加上一个 pNext 指针，指向同一层的相邻节点；如果当前节点已经是该层的最后一个节点，则将 pNext 指针指向 NULL；给出程序实现，并分析时间复杂度和空间复杂度。

27、腾讯面试题：给你 5 个球，每个球被抽到的可能性为 30、50、20、40、10，设计一个随机算法，该算法的输出结果为本次执行的结果。输出 A，B，C，D，E 即可。

28、搜狐笔试题：给定一个实数数组，按序排列（从小到大），从数组中找出若干个数，使得这若干个数的和与 M 最为接近，描述一个算法，并给出算法的复杂度。

29、阿里巴巴研究院（2009）：

1. 有无序的实数列 V[N]，要求求里面大小相邻的实数的差的最大值，关键是要求线性空间和线性时间

2. 25 匹赛马，5 个跑道，也就是说每次有 5 匹马可以同时比赛。问最少比赛多少次可以知道跑得最快的 5 匹马

3. 有一个函数 int getNum()，每运行一次可以从一个数组 V[N] 里面取出一个数，N 未知，当数取完的时候，函数返回 NULL。现在要求写一个函数 int get()，这个函数运行一次可以从 V[N] 里随机取出一个数，而这个数必须是符合 1/N 平均分布的，也就是说 V[N] 里面任意一个数都有 1/N 的机会被取出，要求空间复杂度为 O(1)

30、微软面试题：Given a head pointer pointing to a linked list ,please write a function that sort the list

in increasing order. You are not allowed to use temporary array or memory copy

struct

{

int data;

struct S_Node *next;

}Node;

Node * sort_link_list_increasing_order (Node *pheader):