**2021**年腾讯精选面试题及答案

1. 删除字符串S1中在字符串S2中岀现的字符。

基本思路:把si的字符存到一个曙七里面，然后遍历麗,看是否出现过，出现过就erase 掉。但是直接输出set的元素这样会改变顺序，要想顺序不变，就顺序遍历一下si看 是否出现，出现就输出。

#include <iostream>

#include <cstdio>

#include "string,

#include <cmath>

#include 〈algorithm〉

#include 〈vector,

#include

<set>

#include

<queued

#include

〈Map,

using namespace std; typedef long long LL;

const int maxn=1005; set<char>s;

int main。

string si,s2;

cin»sl»s2;

int len=sl. length();

for (int i=0;i<len;i++)

s. insert (si [i]);

len=s2. length();

for (int i=0;i<len;i++)

[

if (s. count(s2[i]))

s・ erase(s・find(s2[i]));

1

len=sl. length();

for (int i=0;i<len;i++)

[

if (s. count(si[i])) cout«sl[i];

cout<<endl;

return 0;

1. 求一个论坛的在线人数，假设有一个论坛，其注册ID有 两亿个，每个ID从登陆到退岀会向一个日志文件中记下登陆 时间和退岀时间，要求写一个算法统计一天中论坛的用户在 线分布，取样粒度为秒。

—天总共有3600\*24=86400秒。

定义一个长度为86400的整数数组intdelta [86400],每个整数对应这一秒的人数变化 值，可能为正也可能为负。开始时将数组元素都初始化为0。

然后依次读入每个用户的登录时间和退出时间，将与登录时间对应的整数值加1,将与 退出时间对应的整数值减1。

这样处理一遍后数组中存储了每秒中的人数变化情况。

定义另外一个长度为86400的整数数组intonline.num [86400],每个整数对应这一秒 的论坛在线人数。

假设一天开始时论坛在线人数为0,则第1秒的人数online\_num[0]=delta[0] o第n+1 秒的人数 online\_num [n] =online\_num [n~l] +de 11 a [n]。

这样我们就获得亍一天中任意时间的在线人数。

1. 有序链表合并.

/林

* Definition for singly-linked list.
* struct ListNode {
* int val;
* ListNode \*next;
* ListNode(int x) : val(x), next(NULL) {}
* }；

\*/

class Solution {

public：

ListNode\* mergeTwoLists(ListNode\* 11, ListNode\* 12) {

if (11 == NULL) {

return 12;

} else if (12 == NULL) {

return 11;

} else {

if (ll->val <= 12->val) { ll->next = mergeTwoLists(ll->next, 12); return 11;

se {

12~>next = mergeTwoLists(11, 12->next); return 12;

1. 有n种硬币，面额分别为Ln,每种硬币都有无限个，假 设要付款的金额为mo

m/n+J J(m%n)

1. 一个数列:-12-34-56…询问q次，每次询问区间[l,r] 的区间和，输岀每个询问的答案。

第1个和第2个加起来为1,第3, 4个加起来也为1……

所以前i项和为：

i/2+(i&l)\*i;

区间和可以用前i项和算出来了

1. 牛妹有剪刀，石头，布(以0, 1, 2表示)三种卡片无限 张。现在牛妹拿岀n张排成一排。然后你也拿岀n张牌一一 对应比对。若赢一局则获得一分。若你想得k分。现在输入 n, k和牛妹的n张牌分别是什么，你想要恰好得k分，有多 少种方法。

很容易想到答案跟牛妹每一张牌是什么'没有关系。没一张牌只需要考虑嬴、不嬴。

嬴k分，那就是从n张牌中篁出k张嬴，

其他输，所以组合数c (n, k).对于嬴了答k张，只有一种方法，但是对于剩下的n-k张,

都有平局和输掉两种情况，所以是2的n-k次方。

两者相乘就是答案。

结果很大对mod=le9+7职余，用到同余定理。

求2的幕直接暴力求(当然也可以快速嘉)

求组合数的时候用到除法，

又要职余,所以用到逆元。所以用到逆元公式(当然还有其他求法)：Pow (x, mod-2)%mod;

但是mod=le9+7,所以暴力求幕会超时,

方法是用快速求幕法压缩时间(快速幕就不贴代码了)

typedef long long 11;

11 fast (11 a, 11 n) // 快速幕 pow(a,n)

11 inv(ll x, 11 mod)

[

return fast (x, mod~2);

7. const的含义及实现机制,比如:const int 1,是怎么做到i只 可读的？

const用来说明所定义的变量是只读的。

这些在编译期间完成，編译器可能使用常数直接替换掉对此变量的引用。

8.有一个射击游戏有m种颜色的气球，颜色分别为此m现 在一个人开了n枪，告诉你一个数列，表示打爆的气球颜色 分别是多少。(注意，0表示这一枪没有打中，mmp这里害 得我debug T好久)求一个最小区间[l,r],在区间内包含了 所有l~m颜色。输岀区间长度。

这个題是XUPT 2019寒假训练最后一场比呑的原題的强化版。刚好我做了并且在 biiibix±给学弟学妹们讲了，很奈斯。

用一个变量维护当前区间里有多少种颜色，用book数组表示第i种颜色在当前区间内 出现了多少次。

然后尺取。

9.到商店里买200的商品返还100优惠券（可以在本商店代 替现金）。请问实际上折扣是多少？

由于优惠券可以代替现金,所以可以使用200元优惠券买东西，然后还可以获得100元的 优惠券。

假设开始时花了 x元,那么可以买到xM/2枝/4+.的东西。所以实际上折扣是50%（当然， 大部分

时候很难一直兑换下去，所以50%是折扣的上限）

如果使用优惠券买东西不能获得新的优惠券，那么总过花去了 200元，可以买到200+100

元的商品，所以实际折扣为200/300=67%。

10. TCP三次握手的过程,accept发生在三次握手哪个阶段？

accept发生在三次握手之后。

第一次握手:客户端发送syn包（syn=j）到服务器。

第二次握手:服务器收到syn包，必须确认客户的sY（ack=j+l）,同时自己也发送一个ASK 包（ask=k）*。*

第三次握手:客户端收到服务器的SYN+ACK包,向服务器发送确认包ACK（ack=k+l）。 握手完成后，客户端和服务器就建立了 tcp连接。这时可以调用accept函数获得此连 接。

1. 用UDP协议道讯时怎样得知目标机是否获得了数据包？

可以在每个数据包中插入一个唯一的ID,比如timestamp或者逢増的into 发送方在发送数据时将此ID和发送时间记录在本地。

接收方在收到数据后将ID再发给发送方作为回应。

发送方如果收到回应，则知道接收方已经收到相应的数据包；如果在指定时间内没有收 到回应，则数据包可能丢失，需要重复上面的过程重新发送一次，直到确定对方收到。

1. 求一个论坛的在线人数，假设有一个论坛,其注册ID有两 亿个，每个ID从登陆到退岀会向一个日志文件中记下登陆时 间和退岀时间，要求写一个算法统计一天中论坛的用户在线 分布，取样粒度为秒。

—天总共有3600\*24=8600秒。

定义一个长度为86400的整数数组int delta[86400],每个整数对应这一秒的人数变化 值，可能为正也可能为负。开始时将数组元素都初始化为0。

然后依次读入每个用户的登录时间和退出时间，将与登录时间对应的整数值加L将与 退出时间对应的整数值减1。

这样处理一迷后数组中存储了每秒中的人数变化情况。

定义另外一个长度为86400的整数数组mt online num [86400, §个整数对应这一秒的 论坛在线人数。

假设一天幵始时论坛在线人数为。，则第：秒的人数online num[0]= delta [0] o第n+1 秒的人数

line num[n]= online num[n"l]+delta[n] o

这样我们就获得了一天中任意时间的在线人数。

1. 从10G个数中找到中数在一个文件中有10G个整数，乱序

排列，要求找岀中位数。内存限制为2G.

不妨假设10G个整数是64bit的。

2G内存可以存放256M个64bit整数。

我们可以将64bit的整数空间平均分成256M个取值范围，用2G的内存对每个取值范围 内出现整数个数进行统计。这样遍历一边10G整数后，我们便知道中数在那个范围内出 现,以及这个范围内总共出现了多少个整数。

如果中数所在范围出现的整数比较少,我们就可以对这个范围内的整数进行排序，找到 中数。如果这个范围还可以采用同样的天法将此范围再次分成多个更小的范围 （256M-228,所以最多需要3次就可以将此范围缩小到1,也就找到了中数）

1. 两个整数集合A和B,求其交集。
2. 读取整数集合A中的整数,将读到的整数插入到叫中，并将对应的值设为1。
3. 读取整数集合B中的整数,如果该整数在map中并且值为1,则将此数加入到交集当中, 并将在map中的对应值改为2。

通过更改map中的值，避免了将同样的值谕出两次。

15.找岀1到10w中没有出现的两个数字有1到10w这10w 个数，去除2个并打乱次序，如何找出那两个数？

申请10w个bit的空间，每个bit代表一个数字是否出现过。

幵始时将这10个bit都初始化为0,表示所有数字都没有出现过。

然后依次读入已经打乱循序的数字，并将対应的bit设为1。

处理完所有数字后,根据为0的bi得出没有出现的数字。

首先计算1到10W的和，平方和。

然后计算给定数字的和，平方和。

两次的到的数字相减，可以得到这两个数字的和，平方和。

所以我们有

X + y = n

x"2 + y"2 = m

解方程可以得到X和y的值。

16.有1000瓶水，其中有一瓶有毒，小白鼠只要尝一点带奇的 水24小时后就会死亡,至少要多少只小白鼠才能在24小时时 鉴别岀那瓶水有毒？

最容易想到的就是用1000只小白鼠,每只喝一瓶。但显然这不是最好答案。

既然每只小白鼠喝一瓶不是最好答案，那就应该每只小白鼠喝多瓶。那每只应该喝多少 瓶呢？

首先让我们换种问法，如果有X只小白鼠，那么24小时内可以从多少瓶水中找出那瓶有 毒的？

由于每只小白鼠都只有死或者活这两种结果，所以x只小白鼠最大可以表示2 “x种结果。 如果让每种结果都对应到某瓶水有毒，那么也就可以从2“x瓶水中找到有毒的那瓶水。 那如何来实现这种对应关系呢？

第一只小白鼠喝第1到2“ (x-1)瓶,第二只小白鼠喝第1到第2“ (x-2)和第2“ (x-l)+l 到第2\* (x-l)+2伝-2)施...以此类推。

回到此題，总过1000瓶水,所以需要最少10只小白鼠。

17.根据上排的数填写下排的数，并满足要求.

根据上排给出十个数，在其下排填出对应的十个数,要求下排每个数都是上排对应位置 的数在下排出现的次数。上排的数:0,爲2, 3,4, 5,6, 7,8,9。

1. 给40亿个不重复的unsigned int的整数，没排过序的，然

后再给几个数，如何快速判断这几个数是否在那10亿个数当

中？

unsigned int的取值范围是。到2“32-1°我们可以申请连续的232/8-512M的内存，用 每一个bit对应个unsigned int数字。首先将51M内存都初始化为0,然后每处理一个 数字就将其对应的bit设置为1。当需要查询时，直接找到对应bit,看其值是。还是1 即可。

1. 1-20的两个数把和告诉A积告诉B,A说不知道是多少,B 也说不知道，这时A说我知道了,B接着说我也知道了，问这两 个数是多少？

2和3

1. 爸爸妈妈妹妹小强，至少两个人同一生肖的概率是多少？

1-12\*11\*10\*9/12\*12212\*12=1-5 596=4196

1. 计算 ab«2.

运算符优先级:括号,下标,-> 和（•成员）最高；

单目的比双目的高；

算术双目的比其他双目的高；

位运算高于美糸运算；

关系运算高于按位运算（与，或,异或）*；*

按位运算高于逻辑运算；

目的只有一个条件运算，低于逻辑运算；

贼值运算仅比，（顺序运算）高；

在此題中，位左移。”优先级高于按位异或"所以b先左移两位（相当于乘以4）再与a 异或。

例如:当 a=6,b=4 时;则 a\*b«2=22

1. 如何输出源文件的标题和目前执行行的行数?

printf("The file name:%dn^, \_FILE\_);

printf ("The current line No： %d\n”, \_LINE\_);

ANSI C标准预定义宏；

\_LINE\_

\_FILE\_

\_DATE\_

\_STDC\_当要求程序严格遵循ANSC标准时该标识符被赋值为1 \_cpluspluS\_当编写C催序时该标识符被定义

1. a[3]4 哪个不能表示 a[l]l: \*(&a[0][0]+5) \*((a+l)+l) \*(&a[l]+l)\*(&a[0][0]+4)\*(&a[l]+l).

a是数组的首地址,a[1]就表示a[l] [0]地址了，不用再取地.

23. fun((expl,exp2),(exp3,exp4,exp5))几个实参?

两个.

形式参数:在声明和定义函数时，写在函数名后的括号中的参数. 实参是调用参数中的变里，行参是被调用函数中的变量.

1. 希尔，冒泡，快速，插入哪个平均速度最快？

快速排序

快速排序、归并排序和基数排序在不同情况下都是最快最有用的

25. enum的声明方式

enum枚举类型名｛

枚举常量L

枚举常里2,

枚举常里n

For example：

enum weekday i Sunday, monday, tuesday, Wednesday, thursday, friday, Saturday}; enum weekday week-day; //week\_day就是一个枚举类型变量

1. 频繁的插入刪除操作使用什么结构比较合适，链表还是

数组？

链表

1. \*p=NULL; \*p= new charl[100]; sizeof(p)各为多少？

都为4。因为都是指针类型,所占存储空间必然为4

1. 顺序查找的平均时间？

(1+2+3+... -hi)/n= (n+1) /2

1. for(i=0,sum=0;i<10;++i,sum+=i)的运行结果？

sum=55

1. 不能做switch()的参数类型是？

switch的参数不能为浮点型

1. 不使用其他变里,交换两个整型a,b的值？

x=x+y; y=x\_y; x=x\_y

32.写岀foatx与“零值“比较的if语句。

if (x>=0.000001 && x<=-0.000001) (x 不为 0 的比较) float: 6位精度

double： 16位精度

33.腾讯服务器每秒有2W个QQ号同时上线，找岀5min内重

新登入的qq号并打印岀来。

如果空间足够大，可以定义一个大的数组a[qq号],初始为零然后.这个qq号登陆了 就 a[qq 号]++

最后统计大于等于2的QQ号

这个用空间来代替时间

不成熟的想法

2w x 300s

所以用6000.000个桶。刪除超时的算法后面说,所以平均捅的大小是1 假设qq号码一共有10'10个，所以每个補装的q号码是10“10/（6\*10~6）个， 这个是插入时候的最坏效率（插入同个桶的时候是顺序查找插入位置的） qq的节点结构和上面大家讨论的基本一样，増加一个指针指

向输出列表,后面说

struct QQstruct {

num\_type qqnum,

timestamp 1as t\_lo g on\_t ime,

QQstruct \*pre,

QQstruct \*next,

OutPutlist \*out //用于free节点的时候,顺便更新下输出列表

I

另外増加两个指针列表

第一个大小300的循环链表,自带一个指向QQStruct的域,循环存300秒内的qq指针。 时间一过就fee掉，所以保证所有捅占用的空闾在2wX30以内.

第二个是输出列表，就是存放题目需要输出的节点。

如果登陆的用户,5分钟内完全没有重复的话，每秒free2w个节点

不辿在free的时候，要判断一下时间是不是真的起时，因为把节点入桶的时候,遇到重 复的，

会更新一下最后登陆的时间。当然啦，这个时候，要把这个Q号码放到需要输出的列表里 面

34.给一个奇数阶N幻方，填入数字123.NAN,使得橫竖斜方

向上的和都相同.

#include^iostream>

*9* include<iomanip>

*9* include<cmath>

usingnamespace std;

int main()

[

int n;

cin>>n;

int i;

int \*\*Matr = new int \*[n]; //动态分配二维数组

for(i=0;i<n;++i)

Katr[i]=new int[n]： //动态分配二维

数组

//j=n/2代表首行中间数作为起点,艮]1所在位置

int j=n/2,num=l： //初始值

i=0；

while(num!=n\*n+l) {

//往右上角延升，若超出则用嚥专移到左下角

Katr [ (i%n+n) %n] [ (j%n+n) %n] =nujn;

//斜行的长度和n是相等的，超出则转至下一写信.

if (num%n==0) {

i++；

} else {

}

for (i=0;i<n;i++) {

for (j=O;j<n;++j) {

cout « setw ((int) loglO:n\*n) +4) «Matr [i] [j ]; //格式控制 cout « endl << endl; //格式控制

}

I

for (i=0; i<n; ++i) {

delete [] Matr [i];

return 1;

1. IP地址的编码分为哪俩部分？

网络号和主机号。不过是要和子网掩码按位与上之后才能区分哪些是网络位哪些是主机 位

1. 描述实时系统的基本特性.

在特定时间内完成特定的任务，实时性与可靠性

1. Internet釆用哪种网络协议?该协议的主要层次结构？

TCPP协议。应用层、传输层、网络层、数据链路层和物理层

1. Internet物理地址和| P地址转换釆用什么协议？

地址解析协议 ARP address resolution protocol

1. 请描述C++的内存管理方式.

在C卄中内存主要分为5个存储区：

栈（Stack）:局部变量，函数参数等存储在该区，由编译器白动分配和释放.栈属于计 算机系统的数据结构，进栈出栈有相应的计算机指令支持，而且分配专门的寄存器存储 栈的地址，效率分高，内存空间是连续的，但栈的内存空间有限。

堆（Heap）:需要程序员手动分配和释放（new, delete）,属于动态分配方式。内存空间 几乎没有限制，内存空间不连续，因此会产生内存碎片。操作系统有一个记录空间内存 的谴表，当收到内存申请时遍历演表，找到第一个空问大于申请空问的堆节点，将该节 点分配给程序，并将该节点从链表中删除。一般，系统会在该内存空间的首地址处记录 本次分配的内存大小，用于delete释放该内存空间。

全局滯态存储区:全局变量，静态变量分配到该区，到程序结束时自动释放，包括DATA 段（全局初始化区）与BSS段（全局未初始化段）。其中，初始化的全局变量和静态变 量存放在DATA段，未初始化的全局变量詛静态变量存放在BSS段。BSS段特点：在程 序执行前BSS段自动清零，所以未初始化的全局变量和静态变量在程序执行前已经成为 0.

文字常量区：存放常量，而且不允许修改。程序结束后由系统释放。

程序代码区：存放程序的二进制代码

40. hash表的实现，包括STL中的哈希桶长度常数。

hash表的实现主要涉及两个问题：散列函数和碰撞处理。

1） hash function （散列函数）*。*最常见的散列函数:f（x） = x % TableSize .

2） 碰撞问题（不同元素的散列值相同）•解决碰撞问题的方法有许多种，包括线性探 测、二次探测、开链等做法。SGL版本使用开链法，使用一个链表保持相同散列值的元 素。

虽然开链法并不要求表格大小必须为质数，但SGISII.仍然以质数来设计表格大小，并 且将28个质数（逐渐呈现大约两倍的关系）计算好，以备随时访问，同时提供一个函 数，用来查询在这28个质数之中，“最接近某数并大于某数”的质数。

41. hash表如何rehash,怎么处理其中保存的资源.

先想想为什么需要rehash：

因为，当1 oadFactor （负载因子）hash表查找的期望复杂度为0（1）.因此， 每次往hash表中添加元素时，我们必须保证是在loadFactor <1的情况下，才能够添 加。

模仿C++的vector扩容方式,Hash表中毎次发现loadFactor==l时,就开辟一个原来 捅数组的两倍空间（称为新捅数组）*，*然活把原来的捅数组中元素全部转移过来到新的 捅数组中。注意这里转移是需要元素一个个重新哈希到新捅中的。

42. redis的主从复制怎么做的？

Redis旧版复制功能只有同步和命令传播。新版复制功能加入了部分同步的功能。

1） 同步：

2） 命令传播：

当主服务器会将自己执行的写命令，也艮睡造成主从服务器不一致的那条写命令，发送 给从服务器执行，当从服务器执行了相同的写命令之后，主从服务器将再次回到一致状 态。

3部分同步：（断线后重复制）

复制偏移量：通过对比主从服务器的复制偏移量，程序可以很容易地知道主从服务器是 否处于一致状态。

复制积压緩冲区：主服务保存最近的写命令到复制积压緩冲区，是一个先进先出队列 服务器运行ID:从服务器记录上次同步的主服务器的Id。

43. ubuntu开机的时候系统做了什么？

1） 加载BIOS

BIOS程序首先检查，计算机硬件能否满足运行的基本条件，这叫做”硬件自检”*。*硬 件自检完成后，BIOS把控制权转交给下一阶段的启动程序。

2） 读取MBR

计算机读职该设备的第一个扇区，也就是读取最前面的512个字节。如果这512个字节 的最后两个字节是0x55和OxAA,表明这个设备可以用于启动：如果不是，表明设备不 能用于启动，控制权于是被转交给”启动顺序”中的下一个设备。

3） Bootloader

在这种情况下，计算机读职"主引导记录"前面446字节的机器码之后，不再把控制权 转交给某一个分区，而是运行事先安装的”启动管理器"（boot loader）,由用户选 择启动哪一个操作系统。

Boot Loader就是在操作系统内核运行之前运行的一段小程序。通过这段小程序，我们 可以初始化硬件设备、建立内存空间的映M图，从而将系统的软硬件环境带到一个合适 的状态，以便为最终调用操作系统内核做好一切准备。

Boot Loader有若干种，其中Grub、Lilo和spfdisk是常见的Loadero Linux环境中, 目前最流行的启动管理器是Grub。

4） 加载内核

内核的加载，内核加载后，接开始操作系统初始化，根据进程的优先级启动进程。

1. 程序什么时候应该使用线程，什么时候单线程效率高。

1耗时的操作使用线程，提高应用程序响应

2并行操作时使用线程，如C/S架构的服务器端并发线程响应用户的请求。

3多CPU系统中，使用线程提高CPU利用率

4改善程序结构。一个既长又复杂的进程可以考虑分为多个线程，成为几个独立或半独 立的运行部分，这样的程序会利于理解和修改。

其他情况都使用单线程。

1. 介绍一下模板和容器。如何实现?（也许会让你当场举例 实现）

模板可以说比较古老了，但是当前的泛型編程实质上就是模板编程。它体现了一种通 用和泛化的思想*。*STL有7种主要容器：

vector, list, deque, map, multimap, set, nultiset.

1. C语言同意一些令人震惊的结构,下面的结构是合法的

吗，如果是它做些什么？

inta= 5, b=7, c;c= a+++b;

这个问题将做为这个测验的一个愉快的结尾。不管你相不相信，上面的例子是完全合乎 语法的。问题是编译器如何处理它?水平不高的编译作者实际上会争论这个问题，根据 最处理原则，编译器应当能处理尽可能防有合法的用法。因此，上面的代码被处理成: c = a++ + b;因此，这段代码持行后a = 6, b = 7, c = 12。如果你知道答案，或 猜出正确答案，做得好。如果你不知道答案，我也不把这个当作问题。我发现这个问题 的最大好处是:这是一个关于代码编写风格，代码的可读性，代码的可修改性的好的话 题

1. #include 与#include “file.h” 的区别?

前者是从Standard Library的路径寻找和引用file, h,而后者是从当前工作路径搜寻 并引用file, ho

1. 内存的分配方式有几种？

1） 从静态存储区域分配。内存在程序编译的时候就已经分配好，这块内存在程序的整个 运行期间都存在。例如全局变量。

2） 在栈上创建。在执行函数时，函数内局部变量的存储单元都可以在栈上创建，函数执 行结束时这些存储单元白动被释放。栈内存分配运算内置于处理器的指令集中，效率很 高，但是分配的内存容量有限。

3） 从堆上分配，亦称动态内存分配。程序在运行的时候用malloc或new申请任意多少 的内存，程序员自己负责在何时用free或delete释放内存。动态内存的生存期由我们 决定，使用非常灵活，但问题也最多。

1. 如何让局部变量具有全局生命期。

具体的生命期的概念我觉得我还要好好深入的学习一下，但是这个题目还算比较简单, 即用static修饰就可以了，但是只是生命期延长，范围并没有扩大，除非把这个变量 定义在函数体外的静态区，不过那样就变成全局变量了，仿佛不符合题目要求。

50. strtok函数在使用上要注意什么问题。

这个问题我不知道能不能回答全面，因为实在是用的很少。这个函数的作用是分割字符 串，但是要分割的字符串不能是常量，这是要注意的。比如先定义一个字符串：char array []=,, parti, part2v strtok 的原形是 char \*strtok (char \*string, char

\*delim);,我们将作为分隔符，先用pt=strtok (array, ”，”)；，得到的结果print 出来就是” pmtl〃 ,那后面的呢，要写成pt=strtok (NULL, ”，”)；，注意，要用NULL, 如果被分割的字符串会被分成N段，那从第二次开始就一直要用NULL。总结起来，需 要注意的是：被分割的字符串和分隔符都要使用变量;除第一次使用指向字符串的指针 外，之后的都要使用NULL;注意使用这个函数的时候千万别把指针跟丢了，不然就全乱 了。

51.用预处理指^define声明一个常数，用以表明1年中有 多少秒(忽略闰年问题)

#define SECONDS\_PER\_YEAR(60 \* 60 \* 24 \* 365)UL 我在这想看到几件事情：

1. . #define语長的星本知识(例如：不能以分号结束，括号的使用，等等)
2. .懂得预处理器将为你计算常数表达式的值，因此，直接写出你是如何计算一年中有 多少秒而不是计算出实际的值，是更清毗而没有代价的。
3. .意识到这个表达式将使一个16位机的整型数溢出-因此要用到长整型符号L,告诉 编译器这个常数是的长整型数。
4. .如果你在你的表达式中用到UL (表示无符号长整型)，那么你有了一个好的起点。 记住，第一印象很重要。

52.有A、B、C、D四个人，要在夜里过一座桥。他们通过 这座桥分别需要耗时1、2、5、10分钟，只有一支手电，并 且同时最多只能两个人一起过桥。请问如何安排，能够在17 分钟内这四个人都过桥？

A & B —>2 mins

1. mins <-- A

C & D —>10 mins

1. mins <— B

A & B —> 2 mins

—共 2 + 1 + 10 + 2 + 2= 17 mins

1. 1-20的两个数把和告诉A,积告诉B, A说不知道是多少， B也说不知道，这时A说我知道了，B接着说我也知道了， 问这两个数是多少？

2和3

1. 从300万字符串中找到最热门的10条搜索的输入信息是 一个字符串，统计300万输入信息中的最热门的前10条， 我们每次输入的一个字符串为不超过255byte,内存使用只 有1G。请描述思想，写岀算法（c语言），空间和时间复杂 度。

300万个字符串最多（假设没有重复，都是最大长度）占用内存3M\*1K/4=O. 75G。所以 可以将所有字符串都存放在内存中进行处理。

可以使用key为字符串（事实上是字符串的hash值），值为字符串出现次数的hash 来统计每个每个字符串出现的次数。并用一个长度为10的数组/链表来存储目前出现次数最 多的10个字符串。

这样空间和时间的复杂度都是0 （n）。

1. 如何找岀字典中的兄弟单词。给定一个单词a,如果通 过交换单词中字母的顺序可以得到另外的单词b,那么定义 b是a的兄弟单词。现在给定一个字典，用户输入一个单词, 如何根据字典找岀这个单词有多少个兄弟单词？

使用hash\_map和链表。

首先定义一个key,使得兄弟单词有相同的key,不是兄弟的单词有不同的key。例如, 将单词按字母从小到大重新排序后作为其key,比如bad的key为abd, good的key为dgooo 使用链表将所有兄弟单词串在一起，hash\_map的key为单词的key, value为链表的起 始地址。

开始时，先遍历字典，将每个单词都按照key加入到对应的链表当中。当需要找兄弟单 词时，只需求取这个单词的key,然后到hash\_map中找到对应的链表即可。

这样创建hash.map时时间复杂度为0 (n),查找兄弟单词时时间复杂度是0 (1)*。*

1. 找出数组中出现次数超过一半的数，现在有一个数组，已知一个数出现的次数超过了一 半，请用

0(n)的复杂度的算法找出这个数。

答案1：

创建一个hash\_map, key为数组中的数，value为此数出现的次数。遍历一遍数组，用 hash-map统计每个数出现的次数，并用两个值存储目前出现次数最多的数和对应出现的次 数。

这样可以做到*0* (n)的时间复杂度和*0* (n)的空间复杂度，满足题目的要求。

但是没有利用“一个数出现的次数超过了一半”这个特点。也许算法还有提高的空间。 答案2:

值用两个变量A和B,其中A存储某个数组中的数，B用来计数。开始时将B初始化为

0。

遍历数组，如果B=0，则令A等于当前数，令B等于1;如果当前数与A相同，则B=B+1: 如果当前数与A不同，则令B=B-1。遍历结束时,A中的数就是要找的数。

这个算法的时间复杂度是0 (n),空间复杂度为0 (1)*。*

1. n个空间(其中n<lM),存放a到a+n-1的数，位置随 机且数字不重复，a为正且未知。现在第一个空间的数被误 设置为-1。已经知道被修改的数不是最小的。请找出被修改 的数字是多少。

算法的时间复杂度是。(

57.找出被修改过的数字n个空间(其中n<lM),存放a 到a+n-1的数，位置随机且数字不重复，a为正且未知。现 在第一个空间的数被误设置为-L己经知道被修改的数不是 最小的。请找岀被修改的数字是多少。

例如:n=6, a=2,原始的串为5, 3, 7,6, *2,4.*现在被别人修改为T, 3, 7,6, *2,4.*现在希 望找到5。

由于修改的数不是最小的，所以遍历第二个空间到最后一个空间可以得到a的值。

a 到 a±n~l 这 n 个数的和是 total=na+ (nT) n/2*。*

将第二个至最后一个空间的数累加获得sub-total o

那么被修改的数就是total-subtotal o

58.设计DNS服务器中cache的数据结构。

要求设计一个DNS的Cache结构，要求育駒多满足每秒5000以上的查询，满足IP数据的 快速插入，查询的速度要快。（题目还给出了一系列的数据，比如：站点数总共为5000 万，IP地址有1。0。万，等等）

DNS服务器实现域名到IP地址的转换。

每个域名的平均长度为25个字节（估计值），每个IP为4个字节，所以Cache的每个 条目需要大概30个字节。

总共50M个条目，所以需要1.5G个字节的空间。可以放置在内存中。（考虑到每秒5000 次操作的限制，也只能放在内存中。）

可以考虑的数据结构包括hash.map,字典树，红黑树等等。

59. 序 列 seq司a,b, ••• z,aa,ab ••• az,ba,bb, ••• bzz ••• ,za,zb, ••• zz,aaa,…]类似与excel的排列，任意给岀一个字符串 s=[a-z]+（由a-z字符组成的任意长度字符串），请问s是序列 seq的第几个。

注意到每满26个就会向前进一位，类似一个26进制的问题。

比如ab,则位置为26\*1+2;

比如za,则位置为26\*26+1;

比如 abc,则位置为 26\*26\*1+26\*2+3;

60.找出第k大的数字所在的位置。写一段程序，找岀数组 中第k大小的数，输出数所在的位置。例如｛2, 4, 3, 4, 7） 中，第一大的数是7,位置在4。第二大、第三大的数都是4, 位置在1、3随便输出哪一个均可。

先找到第k大的数字，然后再遍历一遍数组找到它的位置。所以题目的难点在于如何最 高效的找到第k大的数。

我们可以通过快速排序，堆排序等高效啊非序算法对数组进行排序，然后找到第k大的 数字。这样总体复杂度为*0* （NlogN）。

我们还可以通过二分的思想，找到第k大的数字，而不必对整个数组排序。从数组中随 机选一个数L通过让这个数和其它数比较，我们可以将整个数组分成了两部分并且满足， {x, xx,..., t} < {y, yy,...} o

在将数组分成两个数组的过程中，我们还可以记录每个子数组的大小。这样我们就可以 确定第k大的数字在哪个子数组中。

然后我们继续对包含第k大数字的子数组进行同样的划分，直到找到第k大的数字为止*。* 平均来说，由于每次划分都会使子数组缩小到原来1/2,所以整个过程的复杂度为0（N）*。*

202碩阿里精选 202何百度精选 202碩腾汛精选  
面涙题.pdf 题.pdf 面涙题.pdf

Q面试分享.mp4

S TCPI呦议栈，一次课开启你的网络之门.mp4 碍&面

Q高校能服务器为什么房要内存池.mp4

Q手踌W线密&.mp4

Q reactorigtHQ线®现高并澎騁.mp4

Q ngi n>®§—线密&的实现.m p4

Q MySQL的块應提作.mp4

Q高并发tcpip网塔io.mp4

D 去中心化，p2p，网络承§一^1走.mp4

Q 服家能職化\_糸璀遊.mp4

Q 区块链的底昙，去中心化网络的设计.mp4

Q 深入注出UDP传箍原理及踌分片方法.mp4

Q线程月陛事.mp4

Q后台服务进暨了怎么办.mp4