复试上机大题的考查，一般为：**文件的输入、输入；排序；对坐标点的处理；**

**对数据的处理（如:求素数、求正反序列都是素数）；对单词、字母的统计**。今年的上机题是：有 20000 个数存储于二进制文件中，读取出来，然后求一个最大子集，其中两两互相不为倍数，不为约数，最大公约数为 1，然后满足的数据输出到指定文件中。这次上机的给的数据很多，但考查的还是数据处理、文件处理等。把历年真题好好做几遍就行，然后一定要每个都调试出来！

2018真题

有20000个数存储于二进制文件中，读取出来，然后求一个最大子集，其中两两互相不为倍数，不为约数，最大公约数为1，然后满足的数据输出到指定文件中。

读取⼀一个⼆二进制⽂文件，⾥里里⾯面有两万个数字，求⼀一个最⼤大⼦子集将

其写⼊入到⼀一个⽂文件中：

1.最⼤大⼦子集任意两个数字之间不不成倍数，且最⼤大公约数为 1；

2.按格式写⼊入到⽂文件中

格式：(第⼀一⾏行行为总个数)

12

2 3 5 7 13

17 23 29 31 37

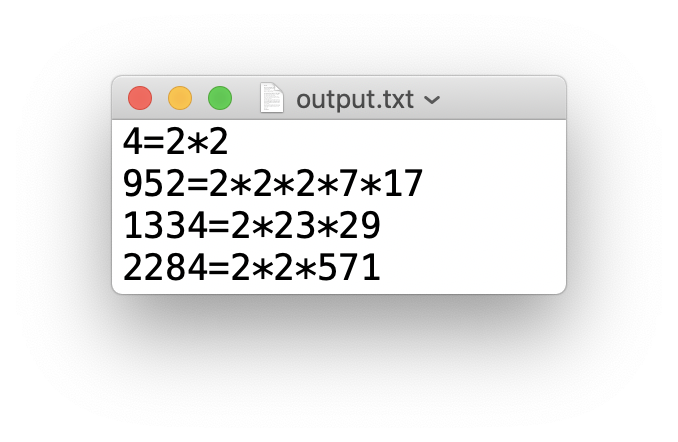
41 43

请各位考生从考试信息发布网站下载数据文件data.txt，然后将这个数据文件**手动**保存在**D盘根目录**中。

**已知：**数据文件data.txt是一个文本文件，其中存放了100个不超过32768的非负整数。

**请编写程序完成如下功能：**

1. 编写函数read\_file从文件中读取数据，将所有的整数按照其在文件中出现的顺序依次存储到数组arr中；
2. 编写函数print将数组arr显示在屏幕上，每行显示n个数，每个整数占6列；
3. 编写函数count统计数字0至9在数组arr所有整数中的出现次数，将结果放入数组res中（即res[0]存储数字0的出现次数，res[1]存储数字1的出现次数，其余以此类推）；
4. 编写函数print\_res将数组res显示在屏幕上，每行显示5个数，可以复用步骤(2)中print函数；
5. 编写函数sort\_array将数组arr中的整数按照**因子和**从小到大排序，如果两个整数的因子和相等，则按照它们的自然大小排序（注意：计算一个整数的因子和时**包括1和其本身**）；
6. 编写函数filter\_array对数组arr中的整数进行筛选，结果继续保存在arr中，筛选规则如下：保留所有的偶数，同时保证这些偶数按照从小到大排序。说明：完成筛选之后，数组arr中的元素可以分成两部分，前半部分是有效内容，即所有的偶数，后半部分则是无效内容，参数size记录了数组arr中有效内容的长度（**注意**：筛选要求在原数组上进行，**如使用新的辅助数组来完成筛选，扣10分**）；
7. 编写函数write\_file对数组arr中的有效内容（即所有偶数）进行质因数分解，并将结果输出到屏幕和文本文件output.txt中。输出要求：每一个整数的质因数分解结果占一行，具体显示格式如下图所示：



#include <stdio.h>

//老师的写法

void print(int \*arr, int size, int num){

int i;

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

printf("%-6d", arr[i]);

if((i+1)%num==0){

printf("\n");

}

}

}

void print\_res(int res[], int num){

print(res, num, 5);

}

//void read\_file(const char filename[100], int \*p){

void read\_file(const char\* filename, int \*p){ //上一行和这一行的区别？？？字符数组/字符串

int num;

FILE \*fp1;

fp1=fopen(filename,"r");

fscanf(fp1,"%d",&num);

while(!feof(fp1)){ //检查有没有到文件尾

\*p=num;

p++;

fscanf(fp1,"%d",&num);

}

fclose(fp1);

}

void count(int arr[], int size, int res[10]){

for(int i=0;i<size;i++){

int num=arr[i];

while(num>0){

int k=num%10;num/=10;

res[k]++;

}

}

}

int split(int num){ //将整数拆成因子和

int sum=0;

for(int i=1; i<=num; i++){

if(num%i==0){

sum=sum+i;

}

}

return sum;

}

void sort\_array(int arr[], int size){

for(int i=0; i<size; i++){

for(int j=0; j<size-1-i; j++){

int n1=split(arr[j]);

int n2=split(arr[j+1]);

if(n1>n2){

int bigger=arr[j];

arr[j]=arr[j+1];

arr[j+1]=bigger;

}

}

}

}

void filter\_array(int arr[], int \*size){

int i=0,j=\*size-1;

while(1){

while(arr[i]%2==0)i++;

while(arr[j]%2==1)j--;

if(i<j)

{

int t=arr[i];

arr[i]=arr[j];

arr[j]=t;

}

else break;

}

\*size=i;

for(i=0;i<\*size-1;i++){

for(j=0;j<\*size-1-i;j++){

if(arr[j]>arr[j+1]){

int bigger=arr[j];

arr[j]=arr[j+1];

arr[j+1]=bigger;

}

}

}

}

void write\_file(const char\*s, int \*arr, int size){

FILE \*fp;

fp=fopen(s,"w");

int i;

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

printf("%d=",arr[i]);

fprintf(fp,"%d=",arr[i]);

int n=arr[i];

int j=2;

while(1){

if(n%j==0){

n=n/j;

if(n==1){ //如果是最后一个数了

printf("%d",j);

fprintf(fp,"%d",j);

printf("\n");

fprintf(fp,"\n",j);

break;

}

printf("%d\*",j);

fprintf(fp,"%d\*",j);

}

else{

j++;

}

}

}

}

int main(int argc, const char \* argv[]) {

int arr[100], size = 100, res[10] = {0};

read\_file("D:\\data.txt", arr);

print(arr, size, 10); //每行显示10个数，所以第三个参数为10

count(arr, size, res);

printf("数字0-9的出现次数依次是\n");

print\_res(res, 10); //res数组的长度为10，所以第二个参数为10

sort\_array(arr, size);

printf("排序之后\n");

print(arr, size, 10); //每行显示10个数，所以第三个参数为10

filter\_array(arr, &size);

printf("筛选之后\n");

print(arr, size, 10); //每行显示10个数，所以第三个参数为10

write\_file("D:\\output.txt", arr, size);

return 0;

}