**D3568. 质数的孤独**

**时间限制：1.0s   内存限制：256.0MB**

总提交次数：**[470](http://oj.tsinsen.com/AllSubmits.page?type=a&gpid=D3568,D3581,D3643,D5325,D5352,D6577,P1377)**   AC次数：**277**   平均分：**71.45**

将本题分享到：

[查看未格式化的试题](http://oj.tsinsen.com/D3568###)   [提交](http://oj.tsinsen.com/D3568###)   [试题讨论](http://oj.tsinsen.com/D3568###)

　　质数的孤独　　题目难度：易  
　　占分比例：30%

**问题描述**

　　质数是孤独的，除了二三之外，身边都挤着其他合数，阻隔它们的亲密接触。  
　　现在，定义一个质数的孤独值I为在数轴上离它最近的一个质数和它的差的绝对值。如：  
　　I(2)=1  
　　I(3)=1  
　　I(5)=2  
　　I(7)=2  
　　I(11)=2  
　　I(13)=2  
　　I(17)=2  
　　I(19)=2  
　　I(23)=4  
　　I(29)=2  
　　给出一个质数p，求距离它最近的质数，以及它的孤独值。如果有两个质数和p一样近，取值较小的。

**输入格式**

　　一行，一个质数p

**输出格式**

　　一行，空格分开的两个值，第一个表示与p距离最近的质数，第二个表示p的孤独值

**样例输入**

5

**样例输出**

3 2

**数据规模和约定**

　　输入质数2<p<=20000

**D3569. 那些年，你们系的女孩**

**时间限制：1.0s   内存限制：256.0MB**

总提交次数：**[349](http://oj.tsinsen.com/AllSubmits.page?type=a&gpid=D3569,D3582,D3644,D5113,P1376)**   AC次数：**120**   平均分：**34.38**

将本题分享到：

[查看未格式化的试题](http://oj.tsinsen.com/D3569###)   [提交](http://oj.tsinsen.com/D3569###)   [试题讨论](http://oj.tsinsen.com/D3569###)

　　那些年，你们系的女孩　　题目难度：易  
　　题目占分比例：30%

**问题描述**

　　那些年，你们系的女孩很少。为了逃避这个事实，在一份记录那些年女生数量的文本文件中采用了暗码。在这份文件中记录了一些小于10000的正整数。实际的女生数量=正整数十进制表示下每一位上的数字之和。  
　　比如，在这份文件的中，某届7字班女生数量暗码的那一行的整数值为5723，则该届女生的数量为5+7+2+3=17人。  
　　现在给你这份文本文件，计算出各届女生的实际数量，分行输出到另一个文件中。

**输入格式**

　　若干个正整数，以换行分隔。文件记录的那些年份数不超过一万年，也就是文件中的正整数个数不超过10000个。

**输出格式**

　　若干行，依照输入顺序，逐行输出暗码对应的女生数量。

**样例输入**

　　9809  
　　7034  
　　4404  
　　7714  
　　4821  
　　2738  
　　6497  
　　7758  
　　3677  
　　768

**样例输出**

　　26  
　　14  
　　12  
　　19  
　　15  
　　20  
　　26  
　　27  
　　23  
　　21

**说明**

　　此题使用文件输入输出，在所给代码的函数thoseyears的参数中，file1为输入文件名，file2为输出文件名。

**输入方法提示**

　　请使用以下方式输入判断文件末尾：  
　　ifstream fin(文件名);  
　　int n;  
　　while (fin >> n) {  
　　}  
　　也就是说fin >> n如果为真，说明输入成功；如果为假说明输入失败，文件中已没有数字。不要使用fin.eof()判断，因为文件末尾可能会有换行符。

*这是一道完善程序的试题，你只需要在下面程序标注的"*@你的代码*"的位置补充适当的语句或语句段使程序能正确运行即可，在提交的时候，你要提交的内容只包括补充的内容，不包括其他的代码。*

1. #include <iostream>
2. #include <fstream>
3. #include <cstdlib>
4. #include <cstdio>
5. #include <cstring>
7. using namespace std;
9. void thoseyears(char \*file1, char\* file2)
10. {
11. @你的代码
12. }

**D3570. 丢失的数**

**时间限制：1.0s   内存限制：256.0MB**

总提交次数：**[294](http://oj.tsinsen.com/AllSubmits.page?type=a&gpid=D3570,D3580,D3642,D5353,P1378)**   AC次数：**109**   平均分：**68.37**

将本题分享到：

[查看未格式化的试题](http://oj.tsinsen.com/D3570###)   [提交](http://oj.tsinsen.com/D3570###)   [试题讨论](http://oj.tsinsen.com/D3570###)

　　丢失的数　　本题难度：易  
　　占分比例：25%

**问题描述**

　　你有n个数1，2，…，n，你不慎丢失若干，你只记得还拥有哪些，请程序帮你自己找回丢失了的那些数。

**输入格式**

　　第一行两个正整数，用空格隔开，第一个为n，表示你曾拥有数的数量，第二个为m，表示没有丢的数的数量。  
　　接下来m行，每行一个正整数，表示没有丢的数，保证没有重复

**输出格式**

　　共n-m行，按从小到大的顺序分行输出丢失的数

**样例输入**

10 7  
1  
4  
5  
6  
8  
9  
10

**样例输出**

2  
3  
7

**数据规模和约定**

　　70%的数据，m<=n<=1000  
　　100%的数据，m<=n<=100000

**D3571. 后缀数组**

**时间限制：1.0s   内存限制：256.0MB**

总提交次数：**[271](http://oj.tsinsen.com/AllSubmits.page?type=a&gpid=D3571,D3579,D3641,D5354,P1379)**   AC次数：**98**   平均分：**44.65**

将本题分享到：

[查看未格式化的试题](http://oj.tsinsen.com/D3571###)   [提交](http://oj.tsinsen.com/D3571###)   [试题讨论](http://oj.tsinsen.com/D3571###)

　　后缀数组　　本题难度：中  
　　本题占分比例：10%

**问题描述**

　　在模版串检索问题中，有一种非常有用的数据结构：后缀数组。对于一个串S=s[1]s[2]…s[n]（注意，下标从1开始），它的第i个后缀记为SF[i]=s[i]…s[n]，也就是从原串第i个字符开始到末尾的部分。比如串S=“liuyuchen”的后缀们为：  
　　SF[1]=”liuyuchen”  
　　SF[2]=”iuyuchen”  
　　SF[3]=”uyuchen”  
　　SF[4]=“yuchen”  
　　SF[5]=“uchen”  
　　SF[6]=“chen”  
　　SF[7]=“hen”  
　　SF[8]=“en”  
　　SF[9]=“n”  
　　后缀数组是指将所有后缀按字典序从小到大排序后得到的数组，例子S的后缀数组为：  
　　{“chen”, “en”,”hen”, “iuyuchen”, “liuyuchen”, “n”,“uchen”, “uyuchen”, “yuchen”}  
　　上述数组的元素依次为SF[6], SF[8], SF[7], SF[2], SF[1], SF[9], SF[5], SF[3], SF[4]。  
　　一般来说，由于有了原串，不需要把每个后缀都完全存储。我们只需要记下后缀的起始位置，也就是SF[i]的下标i。于是之前例子的后缀数组就可以记录为各后缀的起始字符在原串S中的下标组成的序列：  
　　{6, 8, 7, 2, 1, 9, 5, 3, 4}  
　　请编程对输入的字符串，输出它的后缀数组对应的下标序列。  
　　注意：下标从1开始计数。

**输入格式**

　　一行，一个字符串，只包含小写英文字母

**输出格式**

　　输出一行，为输入字符串的后缀数组的下标序列，两个下标之间用一个空格分隔。

**样例输入**

liuyuchen

**样例输出**

6 8 7 2 1 9 5 3 4

**数据规模和约定**

　　对于100%的数据，L<=100

**D3572. 金陵十三钗**

**时间限制：1.0s   内存限制：256.0MB**

总提交次数：**[1162](http://oj.tsinsen.com/AllSubmits.page?type=a&gpid=D3572,D3578,D3640,D5328,D6576,D6581,P1380)**   AC次数：**140**   平均分：**56.28**

将本题分享到：

[查看未格式化的试题](http://oj.tsinsen.com/D3572###)   [提交](http://oj.tsinsen.com/D3572###)   [试题讨论](http://oj.tsinsen.com/D3572###)

　　金陵十三钗  
　　本题难度：难  
　　本题占分比例：5%

**问题描述**

　　在电影《金陵十三钗》中有十二个秦淮河的女人要自我牺牲代替十二个女学生去赴日本人的死亡宴会。为了不让日本人发现，自然需要一番乔装打扮。但由于天生材质的原因，每个人和每个人之间的相似度是不同的。由于我们这是编程题，因此情况就变成了金陵n钗。给出n个女人和n个学生的相似度矩阵，求她们之间的匹配所能获得的最大相似度。  
　　所谓相似度矩阵是一个n\*n的二维数组like[i][j]。其中i,j分别为女人的编号和学生的编号，皆从0到n-1编号。like[i][j]是一个0到100的整数值，表示第i个女人和第j个学生的相似度，值越大相似度越大，比如0表示完全不相似，100表示百分之百一样。每个女人都需要找一个自己代替的女学生。  
　　最终要使两边一一配对，形成一个匹配。请编程找到一种匹配方案，使各对女人和女学生之间的相似度之和最大。

**输入格式**

　　第一行一个正整数n表示有n个秦淮河女人和n个女学生  
　　接下来n行给出相似度，每行n个0到100的整数，依次对应二维矩阵的n行n列。

**输出格式**

　　仅一行，一个整数，表示可获得的最大相似度。

**样例输入**

4  
97 91 68 14  
8 33 27 92  
36 32 98 53  
73 7 17 82

**样例输出**

354

**数据规模和约定**

　　对于70%的数据，n<=10  
　　对于100%的数据，n<=13

**样例说明**

　　最大相似度为91+92+93+73=354