**1355.开学第一考**

**问题描述**

开学的第一个周末，像往届一样，新生都会参加英语分级考试。所谓英语分级考试就是为了测试一下新生的英语水平，并将他们分成1～4,4个等级。分级规则如下：95分（含95分）以上为level4,只需修一个学期的大学英语;85分（含85分）至95分为level3,需修大学英语两个学期;75分（含75分）至85分为level2,需修3个学期的大学英语;剩下的为level1,对不起，就只能修4个学期的大学英语了。小明是刚12级计算机系的新生，他不小心知道了自己的分级考试成绩，于是乎他就迫切的想知道自己会被分在哪一个level，这不，就麻烦一下新生中大神小强，写一个小程序来判定一下小明会被分在哪一个level吧！

**输入格式**

输入一个整数代表小明的分数

**输出格式**

输出小明被分在哪一个英语等级，level1、level2、level3、level4，如果分数不在[0,100]区间内，输出"Error!"（注意：不输出引号）

**样例输入**

60

101

**样例输出**

level1

Error!

#include<stdio.h>

int main()

{

int score;

scanf("%d",&score);

if(score>=95&&score<=100)

printf("level4\n");

else if(score>=85&&score<95)

printf("level3\n");

else if(score>=75&&score<85)

printf("level2\n");

else if(score>=0&&score<75)

printf("level1\n");

else

printf("Error!");

return 0;

}

**1356.小明的考题1——分段函数**

**问题描述**

小明是一个数学狂，自吹自擂说自己的数学是如何如何的好，但是他的风头却被大神小强抢光了，小明就很郁闷，自言自语了一句：不就是编程牛逼点嘛，有本事编几个程序出来解解数学题啊。运气挺背的是，小强就在他后面，不小心就听到了……小强倒是很大方的说：要不你就出几个题考考我？小明这下就得瑟了，呵，小样，看我不整死你！因为小明不知大神的底，于是乎第一题就出了一个简单的分段函数题：

http://59.77.8.111/upload/image/1356.jpg

**输入格式**

输入仅包含一个整数x,并且-100 <= x <= 100

**输出格式**

输出仅一个整数，表示函数值f（x）

**样例输入**

-10

**样例输出**

-440

#include<stdio.h>

int main()

{

int g,f,x;

scanf("%d",&x);

if(x>=-100&&x<=100)

g=x\*x-550;

{

if(x>=g)

f=g-x;

else

f=g+x+4;

}

printf("%d\n",f);

return 0;

}

**1357.小明的考题2——数与单词**

**问题描述**

上一道分段函数的题目被小强轻松秒杀了，这一次，小明知道该出有深度一点的题目了。“数学是科学之王”，小明对此坚信不疑！数学的强悍之处在于它能通过一种映射的关系，把两种不相关的东西联系在一起。现在，小明想考小强的是，小强需要建立起阿拉伯数字与英文字母的对应关系，小明不想过于为难小强，就限制了一下定义域为0—9的10个数字，映射规则为：0—zero、1—one、…、9—nine，现在小明随便说一个0—9的数字，小强的程序能后输出该数字对应的英文单词。

**输入格式**

输入数据为一个[0,9]区间的整数

**输出格式**

输出该数字对应的英文单词

**样例输入**

8

**样例输出**

eight

#include<stdio.h>

int main()

{

int x;

scanf("%d",&x);

if(x==0)

printf("zero");

if(x==1)

printf("one");

if(x==2)

printf("two");

if(x==3)

printf("three");

if(x==4)

printf("four");

if(x==5)

printf("five");

if(x==6)

printf("six");

if(x==7)

printf("seven");

if(x==8)

printf("eight");

if(x==9)

printf("nine");

return 0;

}

**1358.小明的考题3——必杀技**

**问题描述**

    小明为了这最后一题煞费苦心，不过他还是没有什么头绪。这天，他去宿舍楼下的邮局给高中的那群哥们姐妹寄几张咱厦大魅力风光的明信片。抬头的时候瞥了一眼墙上挂着的信函资费说明，详情如下：

平信分本埠和外埠，按首重和续重计收资费，首重100克以内，每重20克（不足20克按20克计算）本埠0.80元，外埠1.20元。续重101～2000克，每重100克（不足100克按100克计算）本埠1.20元，外埠2.00元。

这下，小明可就有主意了。他就用这个信函的资费考考小强。

**输入格式**

输入两个整数k w，以空格隔开。（k=1 or k=2，k=1代表本埠，k=2代表外埠。w代表信函重量，单位g ，已知0<w<=2000）

**输出格式**

输出资费，结果保留两位小数。

**样例输入**

1 95

**样例输出**

4.00

#include<stdio.h>

int main()

{

int k,w;

float s;

scanf("%d%d",&k,&w);

if(w>0&&w<=2000)

{

if(k==1)

{

if(w<=100)

if(w%20!=0)

s=0.80\*(w/20+1);

else

s=0.80\*(w/20);

else

{

if(w%100!=0)

s=4.00+1.20\*(w/100);

else

s=4.00+1.20\*(w/100-1);

}

printf("%.2f\n",s);

}

else

{

if(k==2)

{

if(w<=100)

if(w%20!=0)

s=1.20\*(w/20+1);

else

s=1.20\*(w/20);

else

{

if(w%100!=0)

s=6.00+2.00\*(w/100);

else

s=6.00+2.00\*(w/100-1);

}

printf("%.2f\n",s);

}

else

printf("error!\n");

}

}

else

printf("error!\n");

return 0;

}

**1359.四则运算器**

**问题描述**

三题下来，小明败得彻彻底底的，看着小强写的那些代码，自己瞬间压力山大了。不过，小明就是小明，灵光忽地乍现：自己何不拜小强为师，这样以后的编程不就有师傅做靠山，那不就变得easy啦？小明说到做到，当即就拜小强为师。小强秉着助人为乐的原则，也就收下了小明这个徒弟。不过呢，拜师得有考验才行了，这不，就换小强考考小明啦！不过，小强也不打算怎么为难小强！既然小明数学不错，那就让他去做做简单的数学方面的题吧。给出一个计算式，求结果。

ps：计算式只包含“+”、“-”、“\*”、“/”其中一种运算

**输入格式**

输入一个计算式，格式为：a # b = 。#为运算符。注意，运算符与数之间有空格！当#为/，即计算除法时，要判断除数是否为0，若除数为0，输出“Error!”

**输出格式**

输出结果，保留4位小数。当#为/，即计算除法时，要判断除数是否为0，若除数为0，输出“Error!”（输出无引号）

**样例输入**

15 + 23 =

**样例输出**

38.0000

#include<stdio.h>

int main()

{

float a,b,d;

char c;

scanf("%f %c %f =",&a,&c,&b);

if(c==43)

{

d=a+b;

printf("%.4f\n",d);

}

else if(c==45)

{

d=a-b;

printf("%.4f\n",d);

}

else if(c==42)

{

d=a\*b;

printf("%.4f\n",d);

}

else if(c==47)

{

if(b==0)

printf("Error!\n");

else

{

d=a/b;

printf("%.4f\n",d);

}

}

else

printf("输入错误\n");

return 0;

}

**1360.算日期**

**问题描述**

    自从收了小明这个徒弟之后，小强的生活就没平静过，小明发扬勤奋好问的精神，总是缠着小强问这问那的。这天，小强在去食堂的路上不小心看见小明迎面走来，正打算避开他的时候，不过小明也正好看见小强像他走去，就喊着“师傅师傅”，小强这下没得躲了。小明问了小强非常意外的问题：师傅，今天几号啊。小强随口答道：XX号。不过就在这时，小强却想到了一个可以打发小明的方法，他对小明说：小明啊，你不是想让我给你出出题做嘛，我正好想到一个，今天是XXXX年XX月XX日，你给我算算今天是今年的第多少天吧！小明还想问问怎么做来着，只见小强已经开溜了……

**输入格式**

    输入一个日期，格式为yyyy/mm/dd

**输出格式**

输出一个整数，表示该日期在当年中为第几天

.

**样例输入**

2000/03/01

**样例输出**

61

#include<stdio.h>

int main()

{

int a,b,c,d,e;

scanf("%d/%d/%d",&a,&b,&c);

{

if(a%4==0)

{

if(a%100==0)

{

if(a%400==0)

d=1;

else

d=0;

}

else

d=1;

}

else

d=0;

}

{

if(d)

{

switch(b)

{

case 1:

e=c;

break;

case 2:

e=31+c;

break;

case 3:

e=60+c;

break;

case 4:

e=91+c;

break;

case 5:

e=121+c;

break;

case 6:

e=152+c;

break;

case 7:

e=182+c;

break;

case 8:

e=213+c;

break;

case 9:

e=244+c;

break;

case 10:

e=274+c;

break;

case 11:

e=305+c;

break;

case 12:

e=335+c;

break;

}

}

else

{

switch(b)

{

case 1:

e=c;

break;

case 2:

e=31+c;

break;

case 3:

e=59+c;

break;

case 4:

e=90+c;

break;

case 5:

e=120+c;

break;

case 6:

e=151+c;

break;

case 7:

e=181+c;

break;

case 8:

e=212+c;

break;

case 9:

e=243+c;

break;

case 10:

e=273+c;

break;

case 11:

e=304+c;

break;

case 12:

e=334+c;

break;

}

}

}

printf("%d\n",e);

return 0;

}

**1361.分数约简**

**问题描述**

自从小明拜小强为师后，自知和师傅之间有着无法逾越的差距！不过他并不气馁，他决定通过的努力一点一点缩短与师傅之间的距离！这不才刚学完循环结构，他就迫不及待的想练习了。于是他自己去找了一些题来练手！

这第一个题很简单，就是给出一个分数，把它化为最简分式。

**输入格式**

第一行输入一个整数T，代表有T个测试数据。

接下来的T行，每行输入两个非负整数m、n，m是分子，n是分母。

**输出格式**

输出最简分式，格式a/b。若b=1，只输出a；若分母n=0，则输出"Error!"（输出无引号）

**样例输入**

3

12 16

15 3

20 0

**样例输出**

3/4

5

Error!

#include <stdio.h>

int main()

{

int T,a[200],t,i,max,min;

scanf("%d",&T);

for(i=0; i<2\*T; i+=2)

scanf("%d%d",&a[i],&a[i+1]);

for(i=0; i<2\*T; i+=2)

{

if(a[i]>=0&&a[i]<100000&&a[i+1]>0&&a[i+1]<100000)

{

if(a[i]!=a[i+1])

{

if(a[i]>a[i+1])

{

max=a[i];

min=a[i+1];

if(a[i]%a[i+1]==0) printf("%d\n",a[i]/a[i+1]);

else

{

while(min!=0)

{

t=min;

min=max%min;

max=t;

}

printf("%d/%d\n",a[i]/t,a[i+1]/t);

}

}

else

{

if(a[i]==0) printf("%d\n",a[i]);

else

{

max=a[i+1];

min=a[i];

while(min!=0)

{

t=min;

min=max%min;

max=t;

}

printf("%d/%d\n",a[i]/t,a[i+1]/t);

}

}

}

else printf("%d\n",a[i]/a[i+1]);

}

else printf("Error!\n");

}

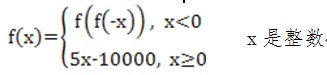
return 0;

}

**1362.再做分段函数**

**问题描述**

这次小明给自己找了一个分段函数的题：



**输入格式**

    第一行输入一个n，代表有n个测试数据，接下来的n行，每行输入变量x的值

**输出格式**

    输入对应变量x的函数值f(x)，每个结果一行

**样例输入**

2

-1500

500

**样例输出**

2500

-7500

#include <stdio.h>

int main()

{

int n,a[100],y,i;

scanf("%d",&n);

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

scanf("%d",&a[i]);

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

{

while(a[i]<0)

{

a[i]=-5\*a[i]-10000;

if(a[i]>=0) break;

}

y=5\*a[i]-10000;

printf("%d\n",y);

}

return 0;

}

**1363.简单的数列求和**

**问题描述**

再来，有通项公式

http://59.77.8.111/upload/image/1363.jpg

求前n项和Sn

**输入格式**

输入n

**输出格式**

输出前n项和Sn(保留4位小数)

**样例输入**

5

**样例输出**

7.5000

#include <stdio.h>

int main()

{

float n,a,fenzi,i,j,sum=0;

scanf("%f",&n);

for(i=1; i<=n; i++)

{

fenzi=0;

for(j=1; j<=i; j++)

fenzi=fenzi+j;

a=fenzi/(i+1);

sum=sum+a;

}

printf("%.4f",sum);

return 0;

}

**1364.找闰年**

    再来，小明回想到，自己不是曾经出了一题闰年的题来考自己的师傅吗？那我何不就改改，然后自己做？他想：要不我就输出给定的任一一个时间段之间的闰年吧！

**输入格式**

输入两个整数m，n，代表两个年份。保证m<n。1000<m<n<3000

**输出格式**

输出[m,n]之间的所有闰年，每行5个

**样例输入**

1998 2015

**样例输出**

2000 2004 2008 2012

#include <stdio.h>

int main()

{

int m,n,year,leap,i=0;

scanf("%d%d",&m,&n);

if(m>1000&&m<n&&n<3000)

for(year=m; year<=n; year++)

{

if(year%4==0)

{

if(year%100==0)

{

if(year%400==0) leap=1;

else leap=0;

}

else leap=1;

}

else leap=0;

if(leap==1)

{

printf("%d ",year);

i++;

if(i%5==0) printf("\n");

}

}

return 0;

}

**1365.小明的自娱自乐**

**问题描述**

练习完前面四个题，小明有点累了，就上上网，找找c语言的资料。碰巧看到这样一道题：输入一个正整数，然后从个位开始一次输出每一位数字对应的英文字母。例如：输入1532，输出two three five one。

**输入格式**

第一行输入一个T，代表有T组测试数据，T<100

对于每一组测试数据，输入一个整数x（0<x<100000000）

**输出格式**

对于每一组测试数据，输出对应结果，每个结果占一行

**样例输入**

3

24

1993

562431

**样例输出**

four two

three nine nine one

one three four two six five

#include <stdio.h>

int main()

{

int T,m,a[100],i;

scanf("%d",&T);

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

scanf("%d",&a[i]);

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

{

while(a[i]!=0)

{

m=a[i]%10;

a[i]/=10;

switch(m)

{

case 0:

printf("zero ");

break;

case 1:

printf("one ");

break;

case 2:

printf("two ");

break;

case 3:

printf("three ");

break;

case 4:

printf("four ");

break;

case 5:

printf("five ");

break;

case 6:

printf("six ");

break;

case 7:

printf("seven ");

break;

case 8:

printf("eight ");

break;

case 9:

printf("nine ");

}

}

printf("\n");

}

return 0;

}

**1366.Simple a + b**

**问题描述**

普通的a + b太没有意思了，小明决定玩点花样：如果加法没有进位，结果会是怎样？

**输入格式**

输入两个正整数a和b(0<=a, b<100000000)

**输出格式**

输出没有进位情况下加法的结果。

PS：类似125 +932的情况，输出的数的首位不能为0，即125 + 932 =57；类似5555 + 5555的情况输出只有一个 0.

**样例输入**

45 79

**样例输出**

14

#include <stdio.h>

int main()

{

int a,b,x,y,s[8],i,j;

scanf("%d%d",&a,&b);

for(i=0; i<8; i++)

{

x=a%10;

y=b%10;

a/=10;

b/=10;

s[i]=(x+y)%10;

}

for(i=7; i>0; i--)

if(s[i]!=0) break;

for(j=i; j>=0; j--)

printf("%d",s[j]);

return 0;

}

**1367.几何级数的计算机验证法**

**问题描述**

从前有棵树，叫高数，上面挂了许多人！这个传说一直言传至今！不过在小明看来，他肯定不会是那个会挂的人，他比其他提前预习了后面的章节，其中，他看了几何级数http://59.77.8.111/upload/image/1367.jpg|x|<1 从理论上证明，他已经看懂了。但是，他想到，既然自己是学计算机的，何不借用计算机来进行验证？于是他开始编程……当等式左右两项之差的绝对值小于10^-6时，认定等式左右两项相等，请输出此时等式右边的项数n是多少！

**输入格式**

第一行输入一个整数T，代表有T组测试数据

对于每组测试数据，输入一个(-1,1)间的浮点数x

**输出格式**

输出每组测试数据的结果，各占一行

**样例输入**

9

0.1

0.2

0.3

0.4

0.5

-0.6

-0.7

-0.8

-0.9

**样例输出**

6

8

11

15

20

26

37

59

125

#include <stdio.h>

#include <math.h>

int main()

{

int T,n,i;

double x,y,l,r;

scanf("%d",&T);

for(i=1; i<=T; i++)

{

scanf("%lf",&x);

l=1/(1-x);

n=0;

r=0;

while((l-r)>=1e-6||(r-l)>=1e-6)

{

y=pow(x,n);

r=r+y;

n++;

}

printf("%d\n",n-1);

}

return 0;

}

**1368.解铃还须系铃人1**

**问题描述**

小明喜欢看谍战片，特别是喜欢看破译电报的桥段。现在他学了C语言，他想借助电脑，将自己的一些文档进行加密。不过有一次在整理硬盘的时候，他不小心把源文档给删掉了。只留下了加密后的文档。当然之间看加密的文档是看不出他写的是什么，不过小明还记得自己是怎么加密的，规律如下：  
即第i个字母变成第（26-i+1）个字母。简单归纳该方法为对称转换加密。  
按照上述规律转换，非字母字符不变。例如“China!”转换为“Xsrmz!”

A→Z,  a→z  
B→Y,  b→y  
C→X,  c→x  
…  
Z→A，z→a

**输入格式**

输入一行加密后的字符串

**输出格式**

输出破译的字符串，注意请不要遗漏任何字符和输出多余的字符

**样例输入**

R droo erhrg Xsrmz mvcg dvvp.

**样例输出**

I will visit China next week.

#include <stdio.h>

int main()

{

char c;

c=getchar();

while(c!='\n')

{

if(c>=65&&c<=90)

c=155-c;

if(c>=97&&c<=122)

c=219-c;

printf("%c",c);

c=getchar();

}

printf("%c",c);

return 0;

}

**1370.谁是下一个十佳歌手**

**问题描述**

一年一度的校园十佳歌手如期而至。和往届一样，每一位参与选手准备一首参赛曲目，选手唱完之后，评委们会给出他们的分数。从这些分数中去掉一个最高分和一个最低分，然后计算平均分所得结果即为该选手的最终得分。由于参赛选手众多，一个一个的手动计算太慢，于是主委会找到计算系求助，希望咱们系能帮忙写一个计算得分的程序！

**输入格式**

第一行输入一个T，代表有T组测试数据，T<=100

对于每一组测试数据，输入有两行，第一行为一个正整数n，n<=100，代表有n个裁判。第二行输入n个正整数，代表n个裁判打的分数。分数<=100.

**输出格式**

对于每组数据，输出结果，保留2位小数

**样例输入**

1

5

88 94 87 83 90

**样例输出**

88.33

#include <stdio.h>

int main()

{

int T,n,i,j,k;

float a[100],t,sum;

scanf("%d",&T);

for(k=0; k<T; k++)

{

scanf("%d",&n);

sum=0;

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

scanf("%f",&a[i]);

for(j=0; j<n-1; j++)

for(i=0; i<n-1-j; i++)

if(a[i]>a[i+1])

{

t=a[i];

a[i]=a[i+1];

a[i+1]=t;

}

for(i=1; i<n-1; i++)

sum=sum+a[i];

printf("%.2f\n",sum/(n-2));

}

return 0;

}

**1371.九九乘法表**

**问题描述**

大家都知道九九乘法表吧，小明因为前面的很多题目都不会，实在无聊，就手写了一个nn乘法表（如九九乘法表），但是没自信的他不知道写的对不对，你能也写一个程序帮他检验一样吗。

**输入格式**

输入一个整数n。（n<=20）

**输出格式**

输出nn乘法表，第i行有i个表达式，第i行第j个表达式为j\*i=k，其中k为i和j的乘积。

**样例输入**

   3

**样例输出**

1\*1=1

1\*2=2 2\*2=4

1\*3=3 2\*3=6 3\*3=9

#include <stdio.h>

int main()

{

int n,a[20][20],i,j,k;

scanf("%d",&n);

for(i=1; i<=n; i++)

{

for(j=1; j<=i; j++)

{

k=i\*j;

printf("%d\*%d=%d ",j,i,k);

}

printf("\n");

}

return 0;

}

**1372.奇偶求和计算**

**问题描述**

给一个长度为n的整数数列，要求分别输出奇数项的和与偶数项的和。

**输入格式**

第一行一个整数n,表示数列有n个数。（N<=1000）

第二行n个整数，表示每个数的数值。（0<任意整数<1000）

**输出格式**

输出两行，第一行为奇数项的和，第二行为偶数项的和。

**样例输入**

5

1 3 2 2 8

**样例输出**

11

5

#include <stdio.h>

int main()

{

int n,a[1000],i,s1=0,s2=0;

scanf("%d",&n);

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

scanf("%d",&a[i]);

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

{

if(i%2==0) s1=s1+a[i];

else s2=s2+a[i];

}

printf("%d\n%d\n",s1,s2);

return 0;

}

**1374.进制转换**

**问题描述**

小强给小明布置了一个小任务。写一个二进制转十进制的程序。

**输入格式**

第一行输入一个T，代表有T组测试数据

对于每一组测试数据，输入一个二进制串 ，长度不超过31

**输出格式**

对于每一组测试数据，输出转换后的结果

**样例输入**

2

10

101010111

**样例输出**

2

343

#include <stdio.h>

#include <math.h>

int main()

{

char a[31],c;

int T,i,n,b,s,j;

scanf("%d%c",&T,&c);

for(j=0; j<T; j++)

{

i=0;

n=0;

s=0;

a[i]=getchar();

while(a[i]!='\n')

{

n++;

i++;

a[i]=getchar();

}

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

{

b=(a[i]-48)\*pow(2,n-i-1);

s=s+b;

}

printf("%d\n",s);

}

return 0;

}

**1377.寻书之旅1**

**问题描述**

小华和小明小强是好朋友，但是总是夹在一个数学高手和一个编程高手中间，搞得他压力山大。于是小华决定去图书馆借一些书来好好的给自己充充电！海韵的图书馆虽小，但就好比小麻雀一样，五脏俱全。大家在开学初都已经通过了图书馆考试，知道通过索书号找书是最快的。小华已经通过查询找到了自己想要的书的索书号（一个正整数）并定位到了某一个书架。这个书架最多能放n本书，即每一个放书位有一个编号，从左至右1—n的编号。书架并不一定放满了书，现在书架上方有L（L<=n）本书。小明现在想知道他所想找的那本书的编号是多少！存在这个书架没有那本书的可能。

**输入格式**

第一行输入书籍数量L和小华所要找的书的索书号k, 0<L、k<10000

第二行输入L个数ai，0<ai<100000，每一个整数ai代表一本书的索书号。

数据保证没有重复索书号。

**输出格式**

输出小华所要找的书在书架上的放书位编号，如果没有这本书，则输出“Error！”（注意：不要输出引号）

**样例输入**

6 9

1 6 9 5 3 10

**样例输出**

3

#include <stdio.h>

#include <math.h>

int main()

{

int L,k,a[10000],i,n=0,leap;

scanf("%d%d",&L,&k);

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

scanf("%d",&a[i]);

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

{

if(a[i]!=k)

{

n++;

leap=0;

}

else

{

n++;

leap=1;

break;

}

}

if(leap==1) printf("%d",n);

if(leap==0) printf("Error!");

return 0;

}

**1378.寻书之旅2**

**问题描述**

看完一本书之后，小华决定再去借一本书！同样的，他事先查询好了自己想要的书的索书号并找到了对的书架。不过他发现这次书籍是有经过整理的，索书号是按照非递减的顺序从左至右的摆放的。现在小明要从这摆放的L本书中找到他要的那本书，同样存在书籍不在书架上的可能！

**输入格式**

第一行输入书籍数量L和小华所要找的书的索书号k, 0<L、k<100000

第二行输入L个整数ai，0<ai<100000，每一个正数ai代表一本书的索书号。

数据保证没有重复索书号。

**输出格式**

输出小华所要找的书在书架上的放书位编号，若不在书架上，则输出“Error！”（注意：不要输出引号）

**样例输入**

6 9

1 3 5 6 9 10

**样例输出**

5

答案同上

**1379.合并队伍**

**问题描述**

小华虽然在数学上不及小明，编程上肯定不及小强。但是在体育方面却是无人能及的。小华可是在千军万马中千辛万苦地杀出一条血路，终于选上了一门体育课！第一次体育课，小华向老师毛遂自荐当体育委员，当然老师也爽快的答应了！在整理队伍的时候，小明发现女生站两排有点短了，于是想把两排女生合为一排。在合并之前，每一排女生都是从左至右按身高从高到低的顺序排好的。为了保持这个顺序，小华要怎么指导女生，才能保证在合并之后，女生仍然是按照从左至右按身高从高到低的顺序排列的？

**输入格式**

第一行输入两个正整数m，n，m为第一排女生的人数，n为第二排女生人数。0<m,n<10000

第二行输入m个正整数ai，从高到低排列，代表m个女生的身高

第三行输入n个正整数bi，从高到低排列，代表n个女生的身高

**输出格式**

输出合并后的身高，从高到低排列

**样例输入**

5 6

165 163 159 155 150

169 164 160 158 155 151

**样例输出**

169 165 164 163 160 159 158 155 155 151 150

#include <stdio.h>

int main()

{

int m,n,a[20000],i,j,t;

scanf("%d%d",&m,&n);

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

scanf("%d",&a[i]);

for(j=0;j<m+n-1;j++)

for(i=0;i<m+n-1-j;i++)

if(a[i]<a[i+1])

{t=a[i];a[i]=a[i+1];a[i+1]=t;}

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

printf("%d ",a[i]);

return 0;

}

**1381.不重叠子串数**

**问题描述**

小明又找到一道难一点的题：给出两个字符串t和s，求出t中能找出几个不重叠的s串。比如t="tobeornottobe"，s="to"，则t中有2个不重叠的"to"串。

**输入格式**

第一行是一个正整数n,n<=10,表示接下来有几个测试数据

接下来的每组数据有两个字符串t和s，都只包含字母，且长度不超过100000

**输出格式**

对每组数据输出对应的答案

**样例输入**

2

vvvvvvvvvvu

vvvu

tobeornottobe

to

**样例输出**

1

2

#include<stdio.h>

#include<string.h>

int main()

{

char t[100001],s[100001],temp[100001];

long int n,i,j,l,q,count=0;

long int lent,lens;

scanf("%ld",&n);

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

{

scanf("%s",t);

scanf("%s",s);

lent=strlen(t);

lens=strlen(s);

for(j=0; j<lent; j++)

if(t[j]==s[0])

{

for(l=0,q=j; l<lens; l++,q++)

temp[l]=t[q];

temp[lens]='\0';

if(strcmp(temp,s)==0)

{

count++;

j+=lens-1;

}

}

printf("%ld\n",count);

count=0;

}

return 0;

}

**1382.小明的生词本**

**问题描述**

    小明知道自己的单词量远远不及他人，于是他想编程给自己建一个生词本，每次遇到新的单词的时候就加入到自己的生词本中。不过由于有时记性不好，本来已经存在生词本中的单词又被小明当成生词加入到了生词本中，对于这种情况，并不需要再给这个单词在分配一块内存来存放数据，也就是说，保证生词本中的单词之间的相异性。再加入一系列单词后，小明想知道自己的生词本中有多少个单词。

**输入格式**

第一行为一个正整数N，0 < N <= 1000，N代表小明加入的单词总数

接下来N行，每一行都是由小写字母组成的单词，单词长度不超过10

**输出格式**

输出生词本中的生词总数M

**样例输入**

9  
the  
quick  
brown  
fox  
jumps  
over  
the  
lazy  
dog

**样例输出**

8

#include <stdio.h>

#include <string.h>

int main()

{

char str[1000][11];

int n,i,j,m;

scanf("%d",&n);

m=n;

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

scanf("%s",str[i]);

for(i=0;i<n-1;i++)

for(j=i+1;j<n;j++)

if(strcmp(str[i],str[j])==0&&strcmp(str[i],"\0")!=0&&strcmp(str[j],"\0")!=0)

{m=m-1;strcpy(str[j],"\0");}

printf("%d",m);

return 0;

}

**1383.回文号码**

**问题描述**

每一个新生，在录取通知书里面，都会包含两张电话卡，一张电信，一张移动。至于小明选了哪张卡，小编我不知道。不过在小强收小明为徒的那天，小强问小明的电话号码是多少，小明回答道：18046164081。小强惊讶地说到：不错嘛，回文号码啊！这时小明才发现，原来自己的号码不管顺着念还是倒着念都是一样的！现在他想通过程序来判断他的通讯录里面有多少个回文号码！

**输入格式**

第一行输入一个整数N，代表电话号码总数 0<N<10000

接下来N行，每一行输入一个电话号码

**输出格式**

N个号码中的回文号码数量

**样例输入**

5

18016164081

18746164781

15059689555

18046164081

13255622543

**样例输出**

2

#include <stdio.h>

int main()

{

int N,i,j,m=0;

char a[10000][11],c;

scanf("%d",&N);

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

scanf("%s",a[i]);

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

if(a[i][0]==a[i][10])

if(a[i][1]==a[i][9])

if(a[i][2]==a[i][8])

if(a[i][3]==a[i][7])

if(a[i][4]==a[i][6])

m++;

printf("%d",m);

return 0;

}

**1384.卡片游戏**

**问题描述**

这一天，小明他们玩了一个游戏。每个人手中有n张卡片，每一张卡片上有一个整数，接着裁判给出一个整数k，然后每一个人需要选择两张卡片使得两张卡片上的数字之和为k，如果可以出卡片，那么打出这两张卡片。如果不行，那么就该下家出卡片。谁先出完卡片谁就是赢家。现在轮到小明出卡片了，不过由于卡片较多且卡片上数据较大，小明尽可能要在最短时间内决定能不能出卡片，因为其他玩家已经在嚷嚷“快点啊，我等得花儿都谢了！”

**输入格式**

第一行输入两个整数n、k，2<=N<=1000，0<K<=2,000,000,000。

第二行输入n个整数ai (0<ai<=1,000,000,000)，代表n张卡片上的数。

**输出格式**

为了简单起见，（因为两张卡片数字之和为k的组合可能有多种）如果可以出卡片，则输出“Yes”；如果不能出卡片，则输出“No”（无引号）

**样例输入**

    5 10

4 5 6 2 1

**样例输出**

Yes

#include <stdio.h>

int main()

{

int n,k,a[1000],i,j,leap=0;

scanf("%d%d",&n,&k);

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

scanf("%d",&a[i]);

for(i=0; i<n-1; i++)

{

for(j=i+1; j<n; j++)

if(a[i]+a[j]==k)

{

leap=1;

break;

}

}

if(leap==1) printf("Yes");

else printf("No");

return 0;

}

**1385.石子消耗游戏**

**问题描述**

小明和小强在路上看到了一堆石子，数了数发现有N颗。无聊的他们决定玩一个游戏，由小明先取K颗石子，取完后换小强取，取2K颗，接着再换小明取，取3K颗，以此类推，他们轮流取，每次都比上一次多取K颗，问最后谁把石子取完（包括取的时候石子不够了），以及他们一共取了多少轮。请用函数嵌套来实现。

**输入格式**

输入只有一行，包括2个整数，N，K，如题目所述。（0<N，K<2^31）

**输出格式**

输出两行。

第一行若小明先取完，则输出”MING!”，否则输出”QIANG!”。（不包括引号）

第二行输出一个整数，代表取的轮次数。

**样例输入**

20 3

**样例输出**

QIANG!

4

**提示**

1、小明取3颗

2、小强取6颗

3、小明取9颗

4、小强取2颗（取完）

#include <stdio.h>

int main()

{

int wuliao(int n,int k);

int n,k;

scanf("%d%d",&n,&k);

if(wuliao(n,k)%2==0)

printf("QIANG!\n%d\n",wuliao(n,k));

else

printf("MING!\n%d\n",wuliao(n,k));

return 0;

}

int wuliao(int n,int k)

{

int i;

for(i=1; i\*k<=n; i++)

n=n-i\*k;

return i;

}

**1386.颠倒的数字**

**问题描述**

写一个函数，输出给定的十进制数的转置数，即将给定的数字的最后一位和第一位对调，倒数第二位和第二位对调…（负号不变），例如120->021,123->321,-135->-531。

**输入格式**

第一行输入一个n，代表有n个数需要“转置”。（n<=100）

接下来输入n行，每行一个整数，范围在[-1000000,1000000]。

**输出格式**

对于每一个数，输出“转置”后的结果。

**样例输入**

3

0

123

-531

**样例输出**

0

321

-135

#include <stdio.h>

int main()

{

void transpose(int a[100],int n);

int n,i,a[100];

scanf("%d",&n);

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

scanf("%d",&a[i]);

transpose(a,n);

return 0;

}

void transpose(int a[100],int n)

{

int i,num;

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

{

if(a[i]<0)

{printf("-");a[i]=-a[i];}

else if(a[i]==0)

{printf("%d\n",a[i]);break;}

while(a[i]!=0)

{

num=a[i]%10;

a[i]/=10;

printf("%d",num);

}

printf("\n");

}

}

**1387.进制转换进阶版**

**问题描述**

    给若干个十进制，要求把每个数转成M（2<=M<=16）进制，用递归实现，和一般进制表示法相同，A代表10，B代表11，C代表12，D代表13，E代表14，F代表15。

**输入格式**

    第一行输入一个整数T，表示需要转换的次数。

    接下来T行每行两个十进制数n和m，表示要把n转成m进制。(0<n<2^31,2<=m<=16)

**输出格式**

对每个数输出一行，表示转换结果。

**样例输入**

3

157 2

157 8

157 16

**样例输出**

   10011101

   235

   9D

#include <stdio.h>

int main()

{

void convert(int k,int m);

int n,i,k,m;

scanf("%d",&n);

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

{

scanf("%d%d",&k,&m);

convert(k,m);

printf("\n");

}

return 0;

}

void convert(int k,int m)

{

if(k<m)

{

if(k>=0&&k<10) printf("%d",k);

else if(k==10) printf("A");

else if(k==11) printf("B");

else if(k==12) printf("C");

else if(k==13) printf("D");

else if(k==14) printf("E");

else if(k==15) printf("F");

}

else

{

convert(k/m,m);

if(k%m>=0&&k%m<10) printf("%d",k%m);

else if(k%m==10) printf("A");

else if(k%m==11) printf("B");

else if(k%m==12) printf("C");

else if(k%m==13) printf("D");

else if(k%m==14) printf("E");

else if(k%m==15) printf("F");

}

}

**1388.高精度乘法**

**问题描述**

之前做过了一道关于高精度加法的，想必难不到大家，那么请大家现在尝试一下高精度乘法吧！

**输入格式**

输入只包含一行，包括两个整数A和B，A、B用空格隔开。(0<=A,B<=10^1000）

**输出格式**

输出一行，A和B的乘积。

**样例输入**

111111111111 111111111111

**样例输出**

12345679012320987654321

#include <stdio.h>

#include <string.h>

void zhuan(char a[],int b[])

{

int i,j;

for(i=0,j=strlen(a); i<strlen(a); i++,j--)

b[j]=a[i]-'0';

b[0]=i;

}

void cheng(int a[],int b[],int c[])

{

int i,j;

c[0]=a[0]+b[0];

for(i=1;i<=a[0]+b[0];i++) c[i]=0;

for(i=1;i<=a[0];i++)

for(j=1;j<=b[0];j++)

{

c[i+j-1]+=a[i]\*b[j];

c[i+j]+=c[i+j-1]/10;

c[i+j-1]%=10;

}

while(!c[c[0]]&&c[0]>1) c[0]--;

}

void print(int a[])

{

int i;

for(i=a[0];i>0;i--)

printf("%d",a[i]);

printf("\n");

}

int main()

{

char a[1001], b[1001];

int c[1001],d[1001],e[2002];

scanf("%s",a);

scanf("%s",b);

zhuan(a, c);

zhuan(b, d);

cheng(c, d, e);

print(e);

return 0;

}

**1389.简单的统计**

**问题描述**

写一个函数，判断整型数组中值为K的数有多少个。

**输入格式**

第一行输入一个整数n,代表数组有元素的个数。（n<=10000）

第二行输入n个数的数组。(数值在[1,100])

第三行输入一个整数K，如题所述。(数值在[1,100])

**输出格式**

输出整型数组中值为K的个数。

**样例输入**

6

1 3 5 3 5 7

3

**样例输出**

2

#include <stdio.h>

#include <math.h>

int main()

{

int number(int a[],int n,int k);

int n,a[10000],k,i;

scanf("%d",&n);

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

scanf("%d",&a[i]);

scanf("%d",&k);

printf("%d",number(a,n,k));

return 0;

}

int number(int a[],int n,int k)

{

int num=0,i;

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

if(a[i]==k) num++;

return num;

}

**1390.伪ackerman函数**

**问题描述**

小明在看数学书的时候看到了一个叫ackerman的数学函数，感觉非常神奇，于是他写了几个小数据算了一下，发现怎么也算不出来。无奈的他小小的修改了一下函数，发现变的简单许多。可是他现在想要得到的结果有很多，你能帮他吗？该函数如下：

        {n+1;            m=0,n>0  
A(m,n)=     {A(m-1,1);       n=0,m>0  
            {A(m-1,n-1)+n;   n>0,m>0  
            {0;              n=0,m=0

**输入格式**

输入第一行一个整数T，表示小明询问你的次数。(T<=100)

接下来的T行每行两个整数M,N。（0<=M,N<=1000）

**输出格式**

对每个M,N输出相应的A(M,N)的值，每个值输出一行。

**样例输入**

3

1 1

22 15

23 21

**样例输出**

1

125

232

#include <stdio.h>

int main()

{

int ackerman(int,int);

int t,i,m,n;

scanf("%d",&t);

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

{

scanf("%d%d",&m,&n);

printf("%d\n",ackerman(m,n));

}

return 0;

}

int ackerman(int m,int n)

{

int a;

if(m==0)

{

if(n==0) a=0;

else if(n>0) a=n+1;

}

else if(m>0)

{

if(n==0) a=ackerman(m-1,1);

else if(n>0) a=ackerman(m-1,n-1)+n;

}

return a;

}

**1391.简单的几何问题**

**问题描述**

写一个函数，给定半径r，输出对应圆的周长，圆的面积，球的表面积，以及球的体积，四舍五入保留两位小数。

**输入格式**

输入只包含一行，一个实数r，代表半径，注意过程中所有变量应使用double类型。(0<r<100)

**输出格式**

输出4行，每行分别代表对应圆的周长，圆的面积，球的表面积，以及球的体积，保留两位小数。（π取3.14159）

**样例输入**

1.1

**样例输出**

6.91

3.80

15.21

5.58

#include <stdio.h>

#define PI 3.14159

int main()

{

void circle(double);

double r;

scanf("%lf",&r);

circle(r);

return 0;

}

void circle(double r)

{

double c,s1,s2,v;

c=2\*PI\*r;

s1=PI\*r\*r;

s2=4\*PI\*r\*r;

v=4\*PI\*r\*r\*r/3;

printf("%.2lf\n%.2lf\n%.2lf\n%.2lf\n",c,s1,s2,v);

}

**1392.数组的平衡**

**问题描述**

写一个函数，计算一个整形数组A的中心分割点。假设某个分割点的下标为k，那么S1= A[1]+A[2]+…+A[k], S2=A[k+1]+A[k+2]+…+A[n],让S1和S2差距最小的下标K即为数组A的中心分割点，若有多个中心分割点，选择值小的。

**输入格式**

    第一行输入一个整数n,代表数组元素个数。（n<=10000）

第二行输入n个整数，每个数的范围[0,1000]。

**输出格式**

    输出题目中所述的中心分割点的值，若有多个中心分割点，选择值小的。

**样例输入**

   5

   1 1 1 2 3

**样例输出**

3

#include <stdio.h>

int main()

{

int spot(int a[],int n);

int n,a[10000],i;

scanf("%d",&n);

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

scanf("%d",&a[i]);

printf("%d",spot(a,n));

return 0;

}

int spot(int a[10000],int n)

{

int k,s1,s2,i,j,s[10000],min,num=0;

for(k=0; k<n-1; k++)

{

s1=0;

s2=0;

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

s1=s1+a[i];

for(j=k+1; j<n; j++)

s2=s2+a[j];

if(s1>s2)

s[k]=s1-s2;

else s[k]=s2-s1;

}

min=s[0];

for(i=0; i<n-1; i++)

if(s[i]<min)

{

min=s[i];

num=i;

}

return num+1;

}

**1393.哥德巴赫猜想的验证**

**问题描述**

著名的哥德巴赫猜想是这样的。1.“任何一个大于二的偶数都可以分解成两个素数之和”。2.“任何一个大于5的奇数都可以分解成三个素数之和”。现在请你写两个函数验证一下这个著名的猜想。

**输入格式**

输入**只有一个整数N**，代表要验证的数，5<=N<=10000，当N为偶数时，用验证哥德巴赫猜想1，当N为奇数时，验证哥德巴赫猜想2。

**输出格式**

对于输入的N，输出相应的表达式，要求加数构成的序列的字典序最小，即确保第一位最小，再确保第二位最小（再确保第三位最小）。

例如N=13，13=3+3+7=7+3+3。

则应输出

13=3+3+7

**样例输入**

8

**样例输出**

   8=3+5

#include <stdio.h>

int prime(int k)

{

int i,j,count;

if(k<=2) return(2);

else if(!(k%2)) k++;

for(i=k;; i+=2)

{

count=1;

for(j=i-2; j>=2; j-=2)

if(!(i%j))

{

count--;

break;

}

if(count) return(i);

}

}

int main()

{

int n,t,s;

scanf("%d",&n);

if(n%2)

{

for(t=1; t<n/2; t+=2)

for(s=t; s<n/2; s+=2)

if(n-prime(t)-prime(s)==prime(n-prime(t)-prime(s)))

{

printf("%d=%d+%d+%d\n",n,prime(t),prime(s),n-prime(t)-prime(s));

return 0;

}

}

else

{

for(t=3; t<n/2; t+=2)

if(n-prime(t)==prime(n-prime(t)))

{

printf("%d=%d+%d\n",n,prime(t),n-prime(t));

return 0;

}

}

return 0;

}

**1394.抛硬币**

**问题描述**

其实，小明有一个小秘密，那就是暗恋计算机系的一个女生很久了。但是又不知道那个女生是怎么想的。3.7女生节就要到了，小明一直在纠结要不要跟那个女生表白。思来想去没有结果。于是乎他觉得采用最原始的办法——抛硬币来决定。正面表白，反面继续沉默观望。不过就像《爱情公寓3》结尾曾小贤说的那样“当面对两个选择时，抛硬币总能奏效，并不是因为它总能给出对的答案，而是在你把它抛在空中的那一秒里，你突然知道你希望它是什么！”呃呃呃，扯远了，其实当硬币还停留在空中的时候，小明想知道的却是：如果我连续抛N次，那么不连续出现正面的可能情况有多少种（即任何相邻两次都不全是正面）？哎，也许这就是作为工科生的思维吧！

**输入格式**

输入只有一行一个整数N，表示抛N次硬币。（N<=15）

**输出格式**

输出只有一行，表示不连续出现正面的可能情况的种数。

**样例输入**

4

**样例输出**

8

**提示**

有16种可能，1表示正面，可行的有0000,0001,0010,0100,0101,1000,1001,1010，共8种。

#include <stdio.h>

int main()

{

int pao(int);

int n;

scanf("%d",&n);

printf("%d",pao(n));

return 0;

}

int pao(int n)

{

int type;

if(n==1) type=2;

else if(n==2) type=3;

else type=pao(n-1)+pao(n-2);

return type;

}

**1395.组合**

**问题描述**

给1到N，N个数，要你从中选取M个出来，请输出每一种的选取情况（根据序列字典序输出，即两个序列比大小，第一位小的小，若相等第二位小的小，若相等第三位小的小……）。

**输入格式**

输入一行，包括两个整数N,M。（2<M<N<10）

**输出格式**

输出若干行，每行表示一个组合方式，按字典序输出。

**样例输入**

4 3

**样例输出**

1 2 3

1 2 4

1 3 4

2 3 4

#include<stdio.h>

int main()

{

void output(int a[],int i,int m,int n,int k,int b[]);

int a[10],i=0,n,m;

int b[10],k=0;

scanf("%d %d",&n,&m);

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

a[i]=i+1;

i=0;

output(a,i,m,n,k,b);

return 0;

}

void output(int a[],int i,int m,int n,int k,int b[])

{

int j=i;

for(; j<n-m+1; j++)

{

b[k]=a[j];

if(m!=1)

output(a,j+1,m-1,n,k+1,b);

else

{

int d=k;

for(; d>=0; d--)

printf("%d ",b[k-d]);

putchar('\n');

}

}

}

**1397.数据排序**

**问题描述**

按一定顺序给n个学生数据“学号语文成绩 数学成绩 英语成绩 物理成绩 化学成绩 生物成绩”，现在让你按其中某一成绩从大到小排序（询问1按语文成绩排序，询问2按数学成绩排序，询问3按英语成绩排序，询问4按物理成绩排序，询问5按化学成绩排序，询问6按生物成绩排序），若成绩相同学号小的排在前面。输出排序后每个同学之前在原数据中的位置（如第一个读入数据的同学为1，最后一个读入数据的同学为N）。

**输入格式**

输入的第一行为一个整数N，代表有N个学生数据。（n<=1000）

接下来每一行一次输入学号、语文成绩、数学成绩、英语成绩、物理成绩、化学成绩、生物成绩，其中学号是一个十四位整数**(该数值超过int类型表示的最大范围，请使用字符数组比较或者long long类型)**，成绩是一个范围[0，100]的整数。

再一行输入一个询问整数K∈{1,2,3,4,5,6}，表示按第K种成绩排序，如题目所述。

**输出格式**

输出N行，代表排序后每个同学在原数据中所处的位置，即这个同学在原数据中是第几行。

**样例输入**

3

23020102203458 70 50 30 60 80 90

23020102203457 50 60 70 80 90 100

23020102203456 70 60 50 30 20 80

2

**样例输出**

3

2

1

#include<stdio.h>

struct stu

{

long long num;

int score[7];

int wz;

};

int main()

{

int n,i,j,k;

struct stu a[10001],t;

scanf("%d",&n);

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

{

scanf("%lld %d %d %d %d %d %d",&a[i].num,&a[i].score[0],&a[i].score[1],&a[i].score[2],&a[i].score[3],&a[i].score[4],&a[i].score[5]);

a[i].wz=i+1;

}

scanf("%d",&k);

for (i=0; i<n-1; i++)

{

for (j=0; j<n-1-i; j++)

{

if (a[j].score[k-1]<a[j+1].score[k-1])

{

t=a[j];

a[j]=a[j+1];

a[j+1]=t;

}

if (a[j].score[k-1]==a[j+1].score[k-1])

{

if (a[j].num>a[j+1].num)

{

t=a[j];

a[j]=a[j+1];

a[j+1]=t;

}

}

}

}

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

printf("%d\n",a[i].wz);

return 0;

}

**1400.出队统计**

**问题描述**

有N个人按照顺时针围成一圈，编号1到N。他们从1开始按顺时针报数，每报到M的人出列，并从下一个开始报数，报到M的人再出列，直到全部出列，请问你能给出这N个人的出队顺序吗？

**输入格式**

输入只有一行，包括两个整数N,M（2<=N,M<=1000）

**输出格式**

输出一行N个数，代表出队序列，每两个数之间用一个空格隔开。

**样例输入**

7 3

**样例输出**

3 6 2 7 5 1 4

#include<stdio.h>

int main()

{

int n,m,k,i,j,a[1000],\*p;

scanf("%d %d",&n,&j);

p=a;

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

{

\*(p+i)=i+1;

}

i=0;

k=0;

m=0;

while(m<n-1)

{

if(\*(p+i)!=0) k++;

if(k==j)

{

printf("%d ",\*(p+i));

\*(p+i)=0;

k=0;

m++;

}

i++;

if(i==n)i=0;

}

i=0;

while(a[i]==0)i++;

printf("%d",a[i]);

return 0;

}

**1401.寻找最大内角**

**问题描述**

请定义Point结构体，Point结构体有两个int域代表点的坐标。以结构体变量作为参数，编写GetMaxAngle(Point A, Point B, Point C)函数，表示获得三个点构成的三角形中最大的内角角度。现在给你N个点的坐标，问在这些点中任意选取3个点作为三角形，有可能得到的最大内角是多少。

**输入格式**

第一行输入一个整数N,代表点的个数。（N<=50）

接下来N行每行两个整数X Y，代表每个点的坐标，（0<=X,Y<=1000），保证没有重叠点以及三点共线。

**输出格式**

输出一个实数，表示最大内角的度数，弧度制，保留6位小数。

**样例输入**

4

0 0

5 5

3 5

4 6

**样例输出**

2.896614

#include<stdio.h>

#include<math.h>

struct point

{

int x;

int y;

};

double getmaxangle(struct point a,struct point b,struct point c)

{

double m,n,p,j,k,l;

m=sqrt((a.x-b.x)\*(a.x-b.x)+(a.y-b.y)\*(a.y-b.y));

n=sqrt((c.x-b.x)\*(c.x-b.x)+(c.y-b.y)\*(c.y-b.y));

p=sqrt((c.x-a.x)\*(c.x-a.x)+(c.y-a.y)\*(c.y-a.y));

j=acos((m\*m+n\*n-p\*p)/(2\*m\*n));

k=acos((m\*m+p\*p-n\*n)/(2\*m\*p));

l=acos((n\*n+p\*p-m\*m)/(2\*n\*p));

if(j>=k&&j>=l) return(j);

else if(k>=j&&k>=l) return(k);

else return(l);

}

int main()

{

int n,i,j,p;

double t=0.0,m;

struct point a[51];

double getmaxangle(struct point,struct point,struct point);

scanf("%d",&n);

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

{

scanf("%d%d",&a[i].x,&a[i].y);

}

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

{

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

{

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

{

if(i!=j&&i!=p&&j!=p)

{

m=getmaxangle(a[i],a[j],a[p]);

if(m>t) t=m;

}

}

}

}

printf("%.6f\n",t);

return 0;

}

**1402.第K早之日**

**问题描述**

写一个结构体变量（包括年月日），输入N个年/月/日信息，用链表保证其有序，在输入一个k,最后输出第k天，即给定输入的时间最早的第k天。

**输入格式**

第一行输入一个整数N，表示给定的天数。

接下来N行，每行给出形如YYYY/MM/DD的格式，代表YYYY年MM月DD日。（例如：1991/06/21，0123/02/01）最后一行一个整数K(1<=K<=N)，如题所述。

**输出格式**

仅一行，输出给定的日子里面第K早之日，输出格式与输入格式相同。

**样例输入**

4

1937/05/23

1991/02/15

1946/10/30

1999/06/09

2

**样例输出**

1946/10/30

#include<stdio.h>

#include<string.h>

int main()

{

char a[1000][12],str[12];

int n,i,k,j;

scanf("%d",&n);

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

scanf("%s",a[i]);

scanf("%d",&k);

for(i=0;i<n-1;i++)

{

for(j=i+1;j<n;j++)

{

if(strcmp(a[i],a[j])>0)

{

strcpy(str,a[j]);

strcpy(a[j],a[i]);

strcpy(a[i],str);

}

}

}

printf("%s",a[k-1]);

return 0;

}

**1411.查找**

**问题描述**

编写下列函数  
void find\_two\_largest(int a[],int n,int \*largest,int \*second\_largest);  
当传递长度为n的数组a时，函数将在数组a中搜寻最大元素和第二大元素，把它们分别存储在largest

**输入格式**

第一行一个整数N  
第二行N个整数，整数间有一个空格的间隔

**输出格式** 输出N个数中的最大元素和第二大元素，用一个空格隔开

**样例输入**

5  
1 2 3 4 5

**样例输出**

5 4

#include<stdio.h>

int main()

{

void find\_two\_largest(int a[],int n,int \*largest,int \*second\_largest);

int n,i,a[100];

scanf("%d",&n);

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

scanf("%d",&a[i]);

int largest,\*p1=&largest,second\_largest,\*p2=&second\_largest;

find\_two\_largest(a,n,p1,p2);

printf("%d %d\n",largest,second\_largest);

return 0;

}

void find\_two\_largest(int a[],int n,int \*largest,int \*second\_largest)

{

int m,i;

m=0;

for(i=1; i<n; i++)

if(a[m]<a[i]) m=i;

\*largest=a[m];

int k;

if(m==0)

k=1;

else

k=0;

for(i=k+1; i<n; i++)

if(a[k]<a[i]&&a[i]<a[m]) k=i;

\*second\_largest=a[k];

}

**1413.模式匹配**

**问题描述**

　　以字符串的形式给出文本text和模式串p1,p2...pn。请使用指针方法编写程序判断有多少个模式串出现在文本中。

**输入格式**

　　第一行一个整数N(N<=20)，表示模式串的个数。  
　　第二行，一个字符串，表示给定的文本。文本长度<=20。  
　　第三行至第N+2行，每行一个字符串，表示模式串。模式串长度<=10。

**输出格式**

　　输出一个整数，即出现在文本中的模式串的个数。

**样例输入**

2  
Abc  
Ab  
bc

**样例输出**

2

#include <stdio.h>

char text[21],pattern[21];

int n,count=0;

int main()

{

int i;

char \*p,\*q,\*r;

scanf("%d",&n);

scanf("%s",text);

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

{

scanf("%s",pattern);

for (p=text; \*p!='\0'; p++)

{

r=p;

for (q=pattern; \*r!='\0' && \*q!='\0' && \*r==\*q; q++,r++)

;

if (\*q=='\0')

{

count++;

break;

}

}

}

printf("%d",count);

return 0;

}

**1417.A+B**

**问题描述**

计算 a+b

**输入格式**

两个整数 a,b (0<=a,b<=10)

**输出格式**

输出 a+b 之和

**样例输入**

1 2

**样例输出**

3

#include <stdio.h>

int main()

{

int a,b;

scanf("%d%d",&a,&b);

printf("%d",a+b);

return 0;

}

**1483.归一法则**

**问题描述**

一个偶然的机会，街边算命的老人跟小明打赌，只要小明给他一个正整数n，他按照下面的规则，总是能在有限的次数内把这个给定的正整数n变成1。规则很简单：如果n是奇数，则将n变为3n+1，否则变为n的一半。小明一开始不相信，但是试了很多次，老人都做到了，例如3->10->5->16->8->4->2->1。于是小明输了10块钱。现在小明学会了C语言，他要让计算机帮他检验一下老人说的到底是不是真的。

**输入格式**

输入n，n<=109

**输出格式**

输出变换的次数。

**样例输入**

3

**样例输出**

7

#include <stdio.h>

int main()

{

int n,i=0;

scanf("%d",&n);

while(n!=1)

{

if(n%2==0) n=n/2;

else n=3\*n+1;

i++;

}

printf("%d",i);

return 0;

}

**1484.成绩统计**

**问题描述**

C语言期中考试的成绩出来了，老师想了解一下班上的整体情况，于是让小明编写一个程序，算一算班上的平均分，并统计一下90分以上（含90），80分以上（含80），70分以上（含70），60分以上（含60），不及格的人数分别为多少。

**输入格式**

第一行输入班级人数n（0<=n<100）

第二行输入n个成绩[0,100]

**输出格式**

输出班级的平均成绩（四舍五入保留一位小数），以及90分以上（含90），80分以上（含80），70分以上（含70），60分以上（含60），不及格的人数。

**样例输入**

6

86 65 80 90 56 68

**样例输出**

74.2 1 2 0 2 1

#include <stdio.h>

int main()

{

int n,i,a=0,b=0,c=0,d=0,e=0;

float m[100],sum=0;

scanf("%d",&n);

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

{

scanf("%f",&m[i]);

sum=sum+m[i];

}

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

{

if(m[i]<=100&&m[i]>=90) a++;

if(m[i]<90&&m[i]>=80) b++;

if(m[i]<80&&m[i]>=70) c++;

if(m[i]<70&&m[i]>=60) d++;

if(m[i]<60&&m[i]>=0) e++;

}

printf("%.1f %d %d %d %d %d",sum/n,a,b,c,d,e);

return 0;

}

**1485.成绩排序**

**问题描述**

老师让小明把班上的成绩排个序，这样每个学生都可以知道自己在班级的排名。

**输入格式**

第一行输入班级人数n（0<=n<100）

第二行输入n个成绩[0,100]

**输出格式**

从高到低输出n个成绩

**样例输入**

6

86 65 80 90 56 68

**样例输出**

90 86 80 68 65 56

#include <stdio.h>

int main()

{

int n,i,j,a[100],t;

scanf("%d",&n);

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

scanf("%d",&a[i]);

for(j=0; j<n-1; j++)

for(i=0; i<n-1-j; i++)

if(a[i]<a[i+1])

{

t=a[i];

a[i]=a[i+1];

a[i+1]=t;

}

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

printf("%d ",a[i]);

return 0;

}

**1486.第K高的人**

**问题描述**

对于男生来说，身高是一个比较“严肃”的话题：谁都希望自己能长高一点。很不幸，小明连170cm的门槛都没能迈进。也不知道小明从哪弄来一份全院男生的资料，不过小明最关心的还是身高！现在小明想从全院n个男生中找出第k高的男生的身高是多少，不过由于信息学院人口众多，小明想尽可能用快的方法找到他想知道的第k高的身高！

**输入格式**

第一行输入总男生人数n和需查询的第k高的身高（0<k<=n<100,50<n<100）

第二行输入n个身高值[150,250]，保证n个身高值互不相同。

**输出格式**

输出第k高的身高值。

**样例输入**

10 6

160 175 180 165 170 171 169 185 172 168

**样例输出**

170

#include <stdio.h>

int main()

{

int n,k,a[100],i,j,t;

scanf("%d%d",&n,&k);

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

scanf("%d",&a[i]);

for(j=0; j<n-1; j++)

for(i=0; i<n-1-j; i++)

if(a[i]<a[i+1])

{

t=a[i];

a[i]=a[i+1];

a[i+1]=t;

}

printf("%d",a[k-1]);

return 0;

}

**1487.单词个数统计**

**问题描述**

做腻了数的题目，小明决定做做字符串处理的题目。这不，小明找到了这样一道题：输入一行字符，统计其中有多少个单词，单词之间用空格，逗号，或句号分隔开。

**输入格式**

长度不超过100000的一行字符，由空格,逗号，句号和字母组成。

**输出格式**

包含的单词数。

**样例输入**

Life is a journey, not a destination.

**样例输出**

7

#include<stdio.h>

int main()

{

char str[10000]= {0};

int i,num=0,word=0;

char c;

gets(str);

for(i=0; (c=str[i])!='\0'; i++)

if(c==' '||c==','||c=='.')

word=0;

else if(word==0)

{

word=1;

num++;

}

printf("%d",num);

return 0;

}

**1488.判断素数**

**问题描述**

写一个函数，判断一个数是不是素数。在主函数中输入多个整数，输出素数的个数。

**输入格式**

输入一个整数n,代表数组有元素的个数。（n<=10000）

第二行输入n个数的数组。(数值在[1,10000])

**输出格式**

输出整型数组中素数的个数。

**样例输入**

6

4 33 53 3 5 100

**样例输出**

3

#include <stdio.h>

#include <math.h>

int main()

{

int number(int a[],int n);

int n,a[10000],i;

scanf("%d",&n);

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

scanf("%d",&a[i]);

printf("%d",number(a,n));

return 0;

}

int number(int a[],int n)

{

int i,leap,j,num=0;

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

{

if(a[i]!=1)

{

leap=0;

for(j=2; j<=sqrt(a[i]); j++)

if(a[i]%j==0) leap=1;

if(leap==0) num++;

}

}

return num;

}

**1489.变与不变**

编写下列函数

void change(int \*a,int \*b,int flag);

根据flag的值对a、b进行处理。当flag为正数时，对a、b指向的变量求绝对值; 当flag为0时， a、b指向变量值不变; 当flag为负数时，交换a、b指向的变量的值。

**输入格式**

第一行为数据的组数n

第二行为三个整数,分别为a b flag

**输出格式**

输出a、b指向变量的值，用一个空格隔开

**样例输入**

3

-1 2 3

-1 2 -3

-1 2 0

**样例输出**

1 2

2 -1

-1 2

#include <stdio.h>

int main()

{

void change(int \*a,int \*b,int flag);

int \*p1,\*p2,flag,n,i,a[1000],b[1000];

p1=a;

p2=b;

scanf("%d",&n);

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

{

scanf("%d%d%d",p1+i,p2+i,&flag);

change(p1+i,p2+i,flag);

}

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

printf("%d %d\n",\*(p1+i),\*(p2+i));

return 0;

}

void change(int \*a,int \*b,int flag)

{

int temp;

if(flag>0)

{

if(\*a<0) \*a=-\*a;

if(\*b<0) \*b=-\*b;

}

else if(flag<0)

{

temp=\*a;

\*a=\*b;

\*b=temp;

}

}

**1490.高级模式匹配**

**问题描述**

给定两个字符串s和t，判断s是否是t的子序列，即从t中删除若干个（包括0个）字符，将剩余的字符连接起来，即可获得s。字符串s和t长度不超过1000。

**输入格式**

第一行，字符串t，文本长度<=1000。

第二行，字符串s，文本长度<=1000。

**输出格式**

若s是t的子序列，输出1，否则输出0

**样例输入**

ATBhCde His

This

ATBhCde His

That

**样例输出**

1

0

#include <stdio.h>

int main()

{

char t[1000],s[1000];

int i,j,m=0,leap=1;

gets(t);

gets(s);

for(i=0;s[i]!='\0';i++)

{

for(j=m;t[j]!='\0';j++)

if(t[j]==s[i])

{

m=j+1;

break;

}

if(t[j]=='\0') leap=0;

}

printf("%d",leap);

return 0;

}

**1491.子串删除**

**问题描述**

给定两个字符串s和t，若s是t的子串，将t中的子串s删除，若存在多个子串，则全部删除；若s不是t的子串，对字符串t不做处理。字符串s和t长度不超过1000。

**输入格式**

第一行，字符串t，文本长度<=1000。

第二行，字符串s，文本长度<=1000。

**输出格式**

处理后的字符串t

**样例输入**

Hello World!

Hello

Hello World!

Ho

No pain, no gain

ain

**样例输出**

World!

Hello World!

No p, no g

#include <stdio.h>

#include <string.h>

int main()

{

char t[1000],s[1000];

int i,j,len,leap,k;

gets(t);

gets(s);

len=strlen(s);

for(i=0;t[i]!='\0';i++)

{

leap=1;

if(t[i]==s[0])

{

for(j=1;j<len;j++)

if(t[i+j]!=s[j]) leap=0;

if(leap==1)

{

for(k=0;t[i+k]!='\0';k++)

t[i+k]=t[i+k+len];

i--;

}

}

}

printf("%s",t);

return 0;

}

**1492.破译时间密码**

**问题描述**

截获了一串敌军的密码字符串，其中包含2013年中某个重要的时间（某月某日），需对其破解翻译。字符串中包含数字和其他字符，删除其中的非数字字符，得到一个数字字符串，记为整数N，并按照如下公式计算：Q =（N\*3） MOD 1131 + 101 ，M = Q /100，D = Q MOD 38，其中MOD表示取余运算。计算得到的M、D均为整数，M表示月份，D表示日。判断翻译后的日期（某月某日）是否正确，若翻译结果正确，则显示翻译后的日期（某月某日），若翻译结果不对，则可能是截获的密码串有误，给出提示信息“input error!”。例如：若计算后的M = 9，D = 12，则输出9月12日，若计算后的M = 10，D = 32，则输出“input error!”（因为不存在10月32日）。

**输入格式**

一行一个字符串，文本长度<=1000。

**输出格式**

第一行为：整数N、Q、M、D（分别以一个空格间隔）

第二行为：翻译后的日期

**样例输入**

I like 3this gam4e and this i12s just a test.

http1://ww1w.google.c4om.hk7

**样例输出**

3412 158 1 6

1月6日

1147 149 1 35

input error!

#include<stdio.h>

#include<string.h>

int main()

{

char a[1001];

int N=0,Q,M,D,i,j,l,k=1;

gets(a);

for(i=j=0; a[i]!='\0'; i++)

if(a[i]>='0'&&a[i]<='9')

a[j++]=a[i];

a[j]='\0';

l=strlen(a);

for(i=l-1; i>=0; i--)

{

N+=(a[i]-'0')\*k;

k\*=10;

}

Q=(N\*3)%1131+101;

M=Q/100;

D=Q%38;

printf("%d %d %d %d\n",N,Q,M,D);

if(M>=1&&M<=12&&D>=1&&D<=31)

{

if(M==1||M==3||M==5||M==7||M==8||M==10||M==12)

{

printf("%d月%d日\n",M,D);

}

else if(M==4||M==6||M==9||M==11)

{

if(D<31)

printf("%d月%d日",M,D);

else printf("input error!");

}

else if(M==2)

{

if(D<29)

printf("%d月%d日",M,D);

else printf("input error!");

}

}

else printf("input error!");

return 0;

}

我的答案

#include <stdio.h>

#include <math.h>

int main()

{

char str[1000];

int a[100];

float N=0;

int Q,M,D,i,k=0;

gets(str);

for(i=0; str[i]!='\0'; i++)

if(str[i]>='0'&&str[i]<='9')

{

a[k]=str[i]-'0';

k++;

}

for(i=0; k>0; i++,k--)

N=N+a[i]\*pow(10.0,1.0\*(k-1));

int NN=(int)N;

Q=(NN\*3)%1131+101;

M=Q/100;

D=Q%38;

printf("%d %d %d %d\n",NN,Q,M,D);

if((M==1||M==3||M==5||M==7||M==8||M==10||M==12)&&(D>=1&&D<=31))

printf("%d月%d日",M,D);

else if((M==4||M==6||M==9||M==11)&&(D>=1&&D<=30))

printf("%d月%d日",M,D);

else if(M==2&&D>=1&&D<=29)

printf("%d月%d日",M,D);

else printf("input error!");

return 0;

}

**1493.数据查找**

**问题描述**

按一定条件对n个货物数据“货号 库存 价格 体积”查询，现在按其中某一条件查找大于等于某数值的数据（询问1按库存查找，询问2按价格查找，询问3按体积查找）。输出符合要求的数据信息。

**输入格式**

输入的第一行为一个整数N，代表有N个货物数据。（n<=1000）

接下来每一行依次输入货号、库存、价格、体积，其中货号是一个六位整数，其余数据是范围[0，10000]的整数。

再一行输入2个询问整数K∈{1,2,3}，表示按第K种条件查找，如题目所述, 0<=S<=10000 表示查找大于等于该数值的数据。

**输出格式**

按原顺序输出符合条件的货物数据（包括货号、库存、价格、体积）；若无符合条件的数据，输出“nothing”。

**样例输入**

3

122034 70 50 30

230201 50 60 70

102203 56 90 63

2 60

**样例输出**

230201 50 60 70

102203 56 90 63

#include <stdio.h>

struct goods

{

int num;

int stock;

int price;

int volume;

}a[1000];

int main()

{

int n,i,k,kn,leap=0;

scanf("%d",&n);

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

scanf("%d%d%d%d",&a[i].num,&a[i].stock,&a[i].price,&a[i].volume);

scanf("%d%d",&k,&kn);

if(k==1)

{

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

if(a[i].stock>=kn)

{

leap=1;

printf("%d %d %d %d",a[i].num,a[i].stock,a[i].price,a[i].volume);

printf("\n");

}

if(leap==0) printf("nothing");

}

if(k==2)

{

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

if(a[i].price>=kn)

{

leap=1;

printf("%d %d %d %d",a[i].num,a[i].stock,a[i].price,a[i].volume);

printf("\n");

}

if(leap==0) printf("nothing");

}

if(k==3)

{

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

if(a[i].volume>=kn)

{

leap=1;

printf("%d %d %d %d",a[i].num,a[i].stock,a[i].price,a[i].volume);

printf("\n");

}

if(leap==0) printf("nothing");

}

return 0;

}

**1494.简单链表**

**问题描述**

给定一个空链表，请按进行如下操作。

**输入格式**

每行输入一个整数，表示依次将这些整数插入链表；直到输入0为止，表示输入结束；

**输出格式**

按照插入的顺序依次输出链表中的偶数，若无偶数，输出字符串“Empty”，整数之间以一个空格分开。

**样例输入**

4

3

5

18

122

0

**样例输出**

4 18 122

#include <stdio.h>

#include <stdlib.h>

#define LEN sizeof(struct list)

struct list

{

int num;

struct list \* next;

};

int n;

struct list \* creat(void)

{

struct list \* head;

struct list \* p1,\* p2;

n=0;

p1=p2=(struct list \*)malloc(LEN);

scanf("%d",&p1->num);

head=NULL;

while(p1->num!=0)

{

n=n+1;

if(n==1)

head=p1;

else

p2->next=p1;

p2=p1;

p1=(struct list \*)malloc(LEN);

scanf("%d",&p1->num);

}

p2->next=NULL;

return head;

}

void print(struct list \*head)

{

int leap=0;

struct list \* p;

p=head;

while(p!=NULL)

{

if(p->num%2==0)

{

printf("%d ",p->num);

leap=1;

}

p=p->next;

}

if(leap==0) printf("Empty");

}

int main()

{

struct list \* head;

head=creat();

print(head);

return 0;

}

**1496.文件存数**

**问题描述**

从控制台读取一个整数K，表示将斐波那契数列的前K个数以ASCII方式存入文本文件testin.txt中，各个数之间以一个空格分隔开。

**输入格式**

第一行一个整数K(1<=K<=500)

**输出格式**

无控制台输出。

**样例输入**

10

**样例输出**

无控制台输出。

**1497.简单编辑**

**问题描述**

从控制台读取一个整数K,从文件test.in中读取N个整数，删除其中的第K个整数，并将剩余的N-1个数按初始顺序输出到test.out中。

**输入格式**

第一行一个整数K(1<=K<=N)

接下来从文件test.in中读取数据。

test.in的内容结构为：

第一行一个整数N

第二至第N+1行，每行一个整数。

**输出格式**

无控制台输出。

**样例输入**

无

**样例输出**

无

**1498.文件复制**

**问题描述**

利用文件读写的方法，实现文件复制。

**输入格式**

输入共两行

第一行为字符串src。表示待复制文件的路径名。

第二行为字符串dest。表示文件将被复制到的位置的路径名。

**输出格式**

本题无输出。

**样例输入**

test.in

test.out

**样例输出**

    无

**提示**

给定的文件不一定是文本文件，请用二进制流的方式读入和输出文件

**1499.fc.exe(Simplified)**

**问题描述**

编写一个程序，比较两个以字符形式存储的文件并打印它们第一个不相同的行。

**输入格式**

第一行为字符串filename1。表示第一个文件的路径名。

第二行为字符串filename2。表示第二个文件的路径名。

**输出格式**

打印两个文件第一个不相同的行，先输出第一个文件的，再输出第二个文件的，各占一行。输入数据保证一定存在输出。

**样例输入**

Test1.txt

Test2.txt

**样例输出**

I love Amoy.

I love Beijing.

**提示**

Test1.txt的内容为：

I love China.

I love Amoy.

Test2.txt的内容为：

I love China.

I love Beijing.

程序输出如上面所示。

**1500.归并排序**

**问题描述**

给定两个文件。每个文件内都记录着一组按从小到大顺序排列的整数。要求合并这两组文件的内容，将所有数按从小到大的顺序输出到第三个文件中。

**输入格式**

第一行，字符串filename1，表示第一个文件的路径名。

第二行，字符串filename2，表示第二个文件的路径名。

第三行，字符串filename3，表示第三个文件的路径名。

所有文件的结构为：文件内每行一个整数。

**输出格式**

无控制台输出。

**样例输入**

File1.txt

File2.txt

File3.txt

**样例输出**

无控制台输出。

**提示**

不应忽略特殊情况。

**1501.整理成绩**

**问题描述**

以散乱顺序给定学生每一门课程的成绩，要求你编写程序将其按照一定的格式整理输出到指定的文件中。

**输入格式**

第一行两个整数N,M。分别表示学生总人数，课程总数。

第二行一个字符串filename,表示输出文件的路径名。

接下来M行，每行一个字符串，表示课程名。

接下来NM行，每行由ID COURSE\_NAME SCORE三个数据组成

ID表示学生编号（1<=ID<=N)

COURSE\_NAME为课程名

SCORE 为该学生该门课程的成绩(成绩为整数)

**输出格式**

按照学生ID从小到大，课程名按字典序从小到大的顺序输出每个学生的成绩，输出格式要求如下：

对于每个学生，第一行输出学生ID

接下来M行，每行输出COURSE\_NAME SCORE

COURSE\_NAME为课程名

SCORE 为该学生该门课程的成绩

最后输出一个空行

**样例输入**

2 2

score.txt

Math

Chinese

1 Math 10

2 Chinese 3

2 Math 99

1 Chinese 98

**样例输出**

(在文件score.txt中)

1

Chinese 98

Math 10

2

Chinese 3

Math 99