**【1003】输出：只有一行，“This is a C program.”。**

**请注意不需要输出引号，行尾输出换行。**

【代码】：

#include<stdio.h>

int main()

{

printf("This is a C program.\n");

return 0;

}

**【1004】给定变量a的初始值，计算进行如下赋值表达式计算之后a的值：  
a+=a-=a\*=a**

【代码】：

#include<stdio.h>

int main()

{

int a;

scanf("%d",&a);

a=a+=a-=a\*=a;

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

}

**【1005】：设x和y是单精度浮点类型（float），其中x=111111.111，y=222222.222，输出x和y的和**

【代码】：

#include<stdio.h>

int main()

{

float x,y;

x=111111.111;

y=222222.222;

printf("%.6f\n",x+y);

return 0;

}

**【1007】：请参照本章例题，编写一个C程序，输出以下信息：**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
         Very    Good!  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

数＊号可看出，Very前面9空格，Good前面……

\*也是输出的一部分，别光打印Very Good!

【代码】：

#include<stdio.h>

int main()

{

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf(" Very Good!\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

**【1008】：要将"China"译成密码，译码规律是：用原来字母后面的第4个字母代替原来的字母．例如，字母"A"后面第4个字母是"E"．"E"代替"A"。请编一程序，用赋初值的方法使cl、c2、c3、c4、c5五个变量的值分别为，'C’、'h’、'i’、'n’、'a’，经过运算，输出变换后的密码。**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char c1,c2,c3,c4,c5;

c1='C';

c2='h';

c3='i';

c4='n';

c5='a';

int num=4;

// scanf("%c%c%c%c%c",&c1,&c2,&c3,&c4,&c5);

printf("%c%c%c%c%c\n",c1+num,c2+num,c3+num,c4+num,c5+num);

return 0;

}

【1009】**求Sn=1!+2!+3!+4!+5!+…+n!之值，其中n是一个数字。**

【代码】：

#include<stdio.h>

int main()

{

int n,i,sum;

scanf("%d",&n);

int j=1;

sum=0;

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

{

j=i\*j;

sum = sum+j;

}

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

return 0;

}

【1010】：**求以下三数的和,保留2位小数 1~a之和 1~b的平方和 1~c的倒数和**

【代码】：

#include<stdio.h>

int main()

{

int a, b,c;

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

float sum=0;

for(int i=1;i<=a;i++)

{

sum=sum+i;

}

for(int i=1;i<=b;i++)

{

sum = sum+i\*i;

}

for(int i=1;i<=c;i++)

{

sum = sum +1.0/i;

}

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

return 0;

}

【1013】：**设圆半径r，圆柱高h 求圆周长C1、圆面积Sa、圆球表面积Sb、圆球体积Va、圆柱体积Vb。** 用scanf输入数据，输出计算结果，输出时要求文字说明，取小数点后两位数字。请编程序。 PI＝3.14

【代码】：

#include<stdio.h>

int main()

{

float r,h;

float PI=3.14;

scanf("%f%f",&r,&h);

float C1,Sa,Sb,Va,Vb;

C1=2\*PI\*r;

Sa=PI\*r\*r;

Sb=4\*PI\*r\*r;

Va=(4.0/3.0)\*PI\*r\*r\*r;

Vb=PI\*r\*r\*h;

printf("C1=%.2f\n",C1);

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

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

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

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

return 0;

}

【1014】**输入一个华氏温度，要求输出摄氏温度。公式为 c=5(F-32)/9 输出要求有文字说明，取位2小数**。

【代码】：

#include<stdio.h>

int main()

{

float f,c;

scanf("%f",&f);

c=5\*(f-32)/9.0;

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

return 0;

}

【1015】**有三个整数a b c,由键盘输入，输出其中的最大的数**。

【代码】：

#include<stdio.h>

int main()

{

int a,b,c,max;

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

max=a>b?a:b;

max=max>c?max:c;

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

return 0;

}

【1016】：**有一个函数  
y={ x      x<1  
    | 2x-1   1<=x<10  
    \ 3x-11  x>=10**

【代码】：

#include<stdio.h>

int main()

{

int x,y;

scanf("%d",&x);

if(x<1) y=x;

else if(x>=1&&x<10) y=2\*x-1;

else y=3\*x-11;

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

return 0;

}

【1017】：**给出一百分制成绩，要求输出成绩等级‘A’、‘B’、‘C’、‘D’、‘E’。 90分以上为A 80-89分为B 70-79分为C 60-69分为D 60分以下为E**

【代码】：

#include<stdio.h>

int main()

{

int s;

scanf("%d",&s);

if(s>=90) printf("A\n",s);

else if(s>=80) printf("B\n");

else if(s>=70) printf("C\n");

else if(s>=60) printf("D\n");

else printf("E\n");

return 0;

}

【1018】：**给出一个不多于5位的整数，要求 1、求出它是几位数 2、分别输出每一位数字 3、按逆序输出各位数字，例如原数为321,应输出123**

【代码】：

#include<stdio.h>

#include<math.h>

int main()

{

int a,b,c,d,e,x;

scanf("%d",&x);

a=x/10000;

b=x%10000/1000;

c=x%1000/100;

d=x%100/10;

e=x%10;

if(a!=0)

{

printf("5\n%d %d %d %d %d\n",a,b,c,d,e);

printf("%d%d%d%d%d\n",e,d,c,b,a);

}

else if(b!=0)

{

printf("4\n%d %d %d %d\n",b,c,d,e);

printf("%d%d%d%d",e,d,c,b);

}

else if(c!=0)

{

printf("3\n%d %d %d\n",c,d,e);

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

}

else if(d!=0)

{

printf("2\n%d %d\n",d,e);

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

}

else

{

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

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

}

}

【1019】：**企业发放的奖金根据利润提成。利润低于或等于100000元的，奖金可提10%;  
利润高于100000元，低于200000元（100000<I≤200000）时，低于100000元的部分按10％提成，高于100000元的部分，可提成 7.5%;  
200000<I≤400000时，低于200000元部分仍按上述办法提成，（下同），高于200000元的部分按5％提成；  
400000<I≤600000元时，高于400000元的部分按3％提成；600000<I≤1000000时，高于600000元的部分按1.5%提成；  
I>1000000时，超过1000000元的部分按1%提成。从键盘输入当月利润I,求应发奖金总数。**

【代码】：

#include<stdio.h>

int main()

{

int a,b,s;

scanf("%d",&a);

s=a/100000;

b=0;

if(s/10>10){s=10;}

switch(s)

{

case 10:b=b+(a-1000000)/100;

a=1000000;

case 9:

case 8:

case 7:

case 6:b=b+(a-600000)\*15/1000;

a=600000;

case 5:

case 4:b=b+(a-400000)\*3/100;

a=400000;

case 3:

case 2:b=b+(a-200000)\*5/100;

a=200000;

case 1:b=b+(a-100000)\*75/1000;

a=100000;

case 0:b=b+a\*10/100;

}

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

return 0;

}

【1020】：**输入两个正整数m和n，求其最大公约数和最小公倍数。**

【代码】：

#include<stdio.h>

int main()

{

int m,n,t,sum;

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

sum=m\*n;

if(m<n)

{

t=m;

m=n;

n=t;

}

t=m%n;

while(t)

{

m=n;

n=t;

t=m%n;

}

printf("%d %d\n",n,sum/n);

}

【1021】：**输入一行字符，分别统计出其中英文字母、数字、空格和其他字符的个数。**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char s[100];

int e=0,d=0,k=0,o=0;

gets(s);

int i=0,c;

while((c=s[i++])!='\0')

{

if(((c>='a')&&(c<='z'))||((c>='A')&&(c<='Z'))) e++;

else if((c>='0')&&(c<='9')) d++;

else if(c==' ') k++;

else o++;

}

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

return 0;

}

【1022】**求Sn=a+aa+aaa+…+aa…aaa（有n个a）之值，其中a是一个数字。 求2+22+222+2222+22222（n=5），给定数字2，n由键盘输入，求相应的和。**

【代码】：

#include<stdio.h>

int main()

{

int i,n,sum,t;

scanf("%d",&n);

sum =0;

t=0;

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

{

t=t\*10+2;

sum=sum+t;

}

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

return 0;

}

【1023】：**打印出所有"水仙花数"，**所谓"水仙花数"是指一个三位数，其各位数字立方和等于该本身。 例如：153是一个水仙花数，因为153=1^3+5^3+3^3。

【代码】：

#include<stdio.h>

int main()

{

int i,sum,a,b,c;

for(i=100;i<1000;i++)

{

a=i/100;

b=(i-a\*100)/10;

c=i%10;

if(i==a\*a\*a+b\*b\*b+c\*c\*c)

{

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

}

}

return 0;

}

【1024】**一个数如果恰好等于它的因子之和，这个数就称为"完数"。** 例如，6的因子为1、2、3，而6=1+2+3，因此6是"完数"。 编程序找出N之内的所有完数，并按下面格式输出其因子：

【代码】：

#include<stdio.h>

int main()

{

int n,i,sum,j;

scanf("%d",&n);

sum=0;

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

{

sum=0;

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

{

if(i%j==0)

sum=sum+j;

}

if(sum==i)

{

printf("%d its factors are ",i);

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

{

if(i%j==0)

printf("%d ",j);

}

printf("\n");

}

}

}

【1025】**有一分数序列： 2/1 3/2 5/3 8/5 13/8 21/13...... 求出这个数列的前N项之和，保留两位小数。**

【代码】：

#include<stdio.h>

int main()

{

int n,i;

float sum=0,a,b,t;

scanf("%d",&n);

a=1;b=2;

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

{

sum = sum+b/a;

t=b;

b=a+b;

a=t;

}

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

return 0;

}

【1026】：**一球从M米高度自由下落，每次落地后返回原高度的一半，再落下。 它在第N次落地时反弹多高？共经过多少米？ 保留两位小数**

【代码】：

#include<stdio.h>

int main()

{

int n,i;

float sum,h;

scanf("%f%d",&h,&n);

sum = h;

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

{

h=h/2;

sum=sum+2\*h;

}

sum=sum-h\*2;

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

}

【1027】：

**猴子吃桃问题。**猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个。 第二天早上又将剩下的桃子吃掉一半，又多吃一个。以后每天早上都吃了前一天剩下的一半零一个。 到第N天早上想再吃时，见只剩下一个桃子了。求第一天共摘多少桃子。

【代码】：

#include<stdio.h>

int main()

{

int n,sum,i;

scanf("%d",&n);

sum=1;

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

{

sum=(sum+1)\*2;

}

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

return 0;

}

【1029】：**用筛法求之N内的素数**。

【代码】

#include<stdio.h>

int main()

{

int n,i,j;

scanf("%d",&n);

for(i=2;i<=n;i++)

{

for(j=2;j<=i;j++)

{

if(i%j==0) break;

}

if(i==j)

{

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

}

}

}

【1030】：**用选择法对10个整数从小到大排序。**

【代码】：

#include<stdio.h>

int main(){

int a[11],i,j,k,x;

for(i=1;i<11;i++){

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

}

for(i=1;i<11;i++){

for(j=i;j<11;j++)

{

if(a[i]>a[j])

{

x=a[j];

a[j]=a[i];

a[i]=x;

}

}

}

for(i=1;i<11;i++){

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

}

}

【1031】求一个3×3矩阵对角线元素之和

【代码】：

#include<stdio.h>

int main()

{

int a[3][3],i,j,sum1,sum2;

sum1=0;

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

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

}

}

sum2=0;

for(i=0;i<3;i++)

{

j=i;

sum1=sum1+a[i][j];

j=2-i;

sum2=sum2+a[i][j];

}

printf("%d %d",sum1,sum2);

}

【1032】：**已有一个已递增有序的9个元素的数组，今输入一个数，要求按原来排序的规律将它插入数组中。**

【代码】：

#include<stdio.h>

int main()

{

int a[10],i,t,r,x;

for(i=0;i<9;i++)

{

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

}

scanf("%d",&t);

for(i=0;i<9;i++)

{

if(a[i]>t)

{

break;

}

}

for(;i<10;i++)

{

x=a[i];

a[i]=t;

t=x;

}

for(i=0;i<10;i++)

{

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

}

}

【1033】**输入10个数字，然后逆序输出**。

【代码】

#include<stdio.h>

int main()

{

int a[10],i;

for(i=0;i<10;i++)

{

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

}

for(i=9;i>=0;i--)

{

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

}

}

【1034】：**写两个函数，分别求两个整数的最大公约数和最小公倍数，用主函数调用这两个函数，并输出结果两个整数由键盘输入。**

【代码】：

#include<stdio.h>

int gys(int a,int b)

{

int r;

if(a<b)

{

r=a;

a=b;

b=r;

}

r=a%b;

while(r)

{

a=b;

b=r;

r=a%b;

}

return b;

}

int gbs(int a,int b,int t)

{

return(a\*b/t);

}

int main()

{

int a,b,t,x;

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

t=gys(a,b);

x=gbs(a,b,t);

printf("%d %d",t,x);

}

【1035】：**求方程 的根，用三个函数分别求当b^2-4ac大于0、等于0、和小于0时的根，并输出结果。从主函数输入a、b、c的值。**

【代码】：

#include<math.h>

jie1(float a,float b,float c)

{

float x1,x2,d;

d=b\*b-4\*a\*c;

x1=(-b+sqrt(d))/(2.0\*a);

x2=(-b-sqrt(d))/(2.0\*a);

printf("x1=%.3f+%.3fi x2=%.3f-%.3fi",x1,0.0,x2,0.0);

}

jie2(float a,float b,float c)

{

float x1,x2,d;

d=b\*b-4\*a\*c;

x1=(-b)/(2.0\*a);

x2=(-b)/(2.0\*a);

printf("x1=%.3f+%.3fi x2=%.3f-%.3fi",x1,0.0,x2,0.0);

}

jie3(float a,float b,float c)

{

float x1,x2,d;

d=b\*b-4\*a\*c;

x1=-b/(2.0\*a);

x2=(sqrt(-d))/(2.0\*a);

printf("x1=%.3f+%.3fi x2=%.3f-%.3fi",x1,x2,x1,x2);

}

int main()

{

float a,b,c,d;

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

d=b\*b-4\*a\*c;

if(d>0) jie1(a,b,c);

else if(d=0) jie2(a,b,c);

else jie3(a,b,c);

}

【1037】**写一个函数，使给定的一个二维数组（３×３）转置，即行列互换。**

【代码】：

#include<stdio.h>

int zhuanzhi(int a[3][3])

{

int i,j,t;

for(i=0;i<3;i++)

{

for(j=i;j<3;j++)

{

t=a[i][j];

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

a[j][i]=t;

}

}

}

int main()

{

int a[3][3],i,j;

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

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

}

}

zhuanzhi(a);

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

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

}

printf("\n");

}

}

【1038】：**写一函数，使输入的一个字符串按反序存放，在主函数中输入输出反序后的字符串**

【代码】：

#include<stdio.h>

#include<string.h>

int fanxu(char str[])

{

int n=strlen(str),i;

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

{

printf("%c",str[i]);

}

}

int main()

{

char str[100];

gets(str);

fanxu(str);

}

【1039】**写一函数，将两个字符串连接**

【代码】

#include<stdio.h>

#include<string.h>

int addstr(char a[],char b[])

{

int n=strlen(a);

int t=strlen(b);

int i;

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

{

a[n+i]=b[i];

}

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

{

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

}

}

int main()

{

char a[100],b[100];

gets(a);

gets(b);

addstr(a,b);

}

【1040】**写一函数，将一个字符串中的元音字母复制到另一个字符串，然后输出。**

【代码】：

#include<stdio.h>

#include<string.h>

int copy(char a[],char b[])

{

int i=0,j=0;

int n=strlen(a);

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

{

if((a[i]=='a')||(a[i]=='e')||(a[i]=='i')||(a[i]=='u')||(a[i]=='o'))

{

b[j]=a[i];

j++;

}

}

}

int main()

{

char a[300],b[300];

scanf("%s",a);

copy(a,b);

printf("%s",b);

}

【1042】：

**编写一函数，由实参传来一个字符串，统计此字符串中字母、数字、空格和其它字符的个数，在主函数中输入字符串以及输出上述结果。 只要结果，别输出什么提示信息。**

【代码】：

#include<stdio.h>

#include<string.h>

int e,d,k,o;

int tongji(char str[])

{

int i;

int n=strlen(str);

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

{

if(((str[i]>='a')&&(str[i]<='z'))||((str[i]>='A')&&(str[i]<='Z'))) e++;

else if((str[i]>='0')&&str[i]<='9') d++;

else if((str[i]==' ')) k++;

else o++;

}

}

int main()

{

char str[300];

gets(str);

tongji(str);

printf("%d %d %d %d",e,d,k,o);

}

【1043】定义一个带参的宏，使两个参数的值互换，并写出程序，输入两个数作为使用宏时的实参。输出已交换后的两个值。

【代码】：

#include<stdio.h>

#define CHANGE(a,b,t) t=a;a=b;b=t;

int main()

{

int a,b,c;

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

CHANGE(a,b,c);

printf("%d %d\n",a,b);

}

【1044】**输入两个整数，求他们相除的余数。用带参的宏来实现，编程序**。

【代码】

#include<stdio.h>

#define YUSHU(a,b) a%b;

int main()

{

int a,b,c;

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

c=YUSHU(a,b);

printf("%d",c);

}

【1046】**给年份year，定义一个宏，以判别该年份是否闰年。提示：宏名可以定义为LEAP\_YEAR，形参为y，既定义宏的形式为 #define LEAP\_YEAR(y) （读者设计的字符串**）

【代码】：

#include<stdio.h>

#define LEAP\_YEAR(y) (((y%4==0)&&(y%100!=0))||(y%400==0))

int main()

{

int y,t;

scanf("%d",&y);

t=LEAP\_YEAR(y);

if(t) printf("L");

else printf("N");

}

【1047】**请设计输出实数的格式，包括：⑴一行输出一个实数；⑵一行内输出两个实数；⑶一行内输出三个实数。实数用"6.2f"格式输出**。

【代码】

#include<stdio.h>

int main()

{

float a;

scanf("%f",&a);

printf("%6.2f\n",a);

printf("%6.2f %6.2f\n",a,a);

printf("%6.2f %6.2f %6.2f\n",a,a,a);

}

【1048】：分别用函数和带参的宏，从三个数中找出最大的数。

【代码】：

#include<stdio.h>

#define MAX(a,b,c) ((a>b)?a:b)>c?((a>b)?a:b):c;

int max(int a,int b,int c)

{

float t;

t = a>b?a:b;

t = t>c?t:c;

return t;

}

int main()

{

int a,b,c;

float t;

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

t=max(a,b,c);

printf("%.3f\n",t);

t=MAX(a,b,c);

printf("%.3f",t);

}

【1049】：**输入一行电报文字，将字母变成其下一字母（如’a’变成’b’……’z’变成’ａ’其它字符不变**）。

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char str[100];\

gets(str);

int i;

int n=strlen(str);

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

{

if(((str[i]>='a')&&(str[i]<='y'))||((str[i]>='A')&&(str[i]<='Y'))) str[i]=str[i]+1;

else if(str[i]=='z') str[i]='a';

else if(str[i]=='Z') str[i]='A';

}

puts(str);

return 0;

}

【1050】**输入三个整数，按由小到大的顺序输出**。

【代码】：

#include<stdio.h>

int main()

{

int a,b,c,t;

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

if(a>b) {t=a;a=b;b=t;}

if(b>c) {t=b;b=c;c=t;}

if(a>b) {t=a;a=b;b=t;}

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

}

【1051】：**输入三个字符串，按由小到大的顺序输出**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char a[100],b[100],c[100],t[100];

gets(a);gets(b);gets(c);

if(strcmp(a,b)>0)

{

strcpy(t,a);

strcpy(a,b);

strcpy(b,t);

}

if(strcmp(b,c)>0)

{

strcpy(t,b);

strcpy(b,c);

strcpy(c,t);

}

if(strcmp(a,b)>0)

{

strcpy(t,a);

strcpy(a,b);

strcpy(b,t);

}

puts(a);

puts(b);

puts(c);

}

【1052】：**输入10个整数，将其中最小的数与第一个数对换，把最大的数与最后一个数对换。写三个函数； ①输入10个数；②进行处理；③输出10个数。**

【代码】：

#include<stdio.h>

int main()

{

int a[10],i,t;

for(i=0;i<10;i++)

{

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

}

for(i=0;i<10;i++)

{

if(a[i]<a[0])

{

t=a[0];

a[0]=a[i];

a[i]=t;

}

if(a[i]>a[9])

{

t=a[9];

a[9]=a[i];

a[i]=t;

}

}

for(i=0;i<10;i++)

{

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

}

}

【1053】：**有n个整数，使前面各数顺序向后移m个位置，最后m个数变成前面m个数，见图。写一函数：实现以上功能，在主函数中输入n个数和输出调整后的n个数。**

【代码】：

#include<stdio.h>

int main()

{

int a[100];

int n,m,i;

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

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

{

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

}

scanf("%d",&m);

for(i=n+m-1;i>=m;i--)

{

a[i]=a[i-m];

}

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

{

a[i]=a[n+i];

}

for(i=0;i<10;i++)

{

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

}

}

【1055】：**有一字符串，包含n个字符。写一函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char a[100],b[100];

int i,n,m;

scanf("%d",&n);

getchar();

gets(a);

scanf("%d",&m);

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

{

b[i]=a[i+m-1];

}

puts(b);

}

**【1056】定义一个结构体变量（包括年、月、日）。计算该日在本年中是第几天，注意闰年问题。**

【代码】：

#include<stdio.h>

struct data

{

int year;

int month;

int data;

};

int main()

{

struct data da;

int year,month,data;

int d1[12] = {31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

int d2[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

scanf("%d%d%d",&da.year,&da.month,&da.data);

year=da.year;

int n=0,i;

if((year%4==0&&year%100!=0)||(year%400==0))

{

for(i=0;i<da.month-1;i++)

{

n=n+d1[i];

}

n=n+da.data;

}

else

{

for(i=0;i<da.month-1;i++)

{

n=d2[i]+n;

}

n=n+da.data;

}

printf("%d",n);

}

【1058】：

有N个学生，每个学生的数据包括学号、姓名、3门课的成绩，从键盘输入N个学生的数据，要求打印出3门课的总平均成绩，以及最高分的学生的数据（包括学号、姓名、3门课成绩）

【代码】：

**【1059】：打印菱形**

**从键盘输入一个整数n(1≤n≤9)，打印出指定的菱形.**

【代码】：

#include<stdio.h>

int main()

{

int n,i,j;

scanf("%d",&n);

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

{

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

{

printf(" ");

}

for(j=0;j<2\*i+1;j++)

{

printf("\*");

}

printf("\n");

}

for(i=n-2;i>=0;i--)

{

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

{

printf(" ");

}

for(j=0;j<2\*i+1;j++)

{

printf("\*");

}

printf("\n");

}

}

**【1060】：[三角形判定](https://www.ieeacm.top/problem.php?id=1060)**

**给定三条边，请你判断一下能不能组成一个三角形**

【代码】：

#include<stdio.h>

int main()

{

int a,b,c;

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

if(a+b>c&&a+c>b&&b+c>a)

{

printf("YES");

}

else

{

printf("NO");

}

}

【1064】：**[整除的尾数](https://www.ieeacm.top/problem.php?id=1064)**

**一个整数，只知道前几位，不知道末二位，被另一个整数除尽了，那么该数的末二位该是什么呢？**

【代码】：

#include<stdio.h>

int main()

{

int a,b,i;

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

if(a==0&&b==0) return 0;

else if((0<a<10000)&&(10<b<100))

{

for(i=0;i<100;i++)

{

if((a+i)%b==0)

{

if(i<10)

{

printf("0%d",i);

}

else

{

printf(" %d%d",i/10,i%10);

}

}

}

}

}

**【1065】：[青年歌手大奖赛\_评委会打分](https://www.ieeacm.top/problem.php?id=1065)**

**青年歌手大奖赛中，10个评委会给参赛选手打分。选手得分规则为去掉一个最高分和去掉一个最低分，然后计算平均得分，编程计算某选手的得分，保留两位小数**

【代码】：

#include<stdio.h>

#include<string.h>

#include<math.h>

int main()

{

int a[10],sum=0,i,t;

float ave;

for(i=0;i<10;i++)

{

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

}

for(i=0;i<10;i++)

{

if(a[0]<a[i]){

t=a[0];

a[0]=a[i];

a[i]=t;

}

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

{

t=a[1];

a[1]=a[i];

a[i]=t;

}

}

for(i=2;i<10;i++)

{

sum=sum+a[i];

}

ave=sum/(8.0);

printf("%.2f",ave);

}

【**1066】：合并后的新字符串S**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char a[100],b[100];

gets(a);

gets(b);

int m,i;

if(strlen(a)>strlen(b)) m=strlen(b);

else m=strlen(a);

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

{

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

printf("%c",b[strlen(b)-1-i]);

}

if(strlen(a)>strlen(b))

{

for(i=m;i<=strlen(a)-1;i++){

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

}

}else{

for(i=strlen(b)-1-m;i>=0;i--)

{

printf("%c",b[i]);

}

}

return 0;

}

**【1067】：八进制转换为十进制**

**ps:八进制是一种以8为基数的计数法，采用0，1，2，3，4，5，6，7八个数字，逢八进1。**

【代码】

#include<stdio.h>

#include<string.h>

int main()

{

int n,t=1,x,sum;

scanf("%d",&x);

sum=0;

n=x;

while(n)

{

sum+=(n%10)\*t;

t=t\*8;

n=n/10;

}

printf("%d",sum);

}

**【1068】：[网购打折商品](https://www.ieeacm.top/problem.php?id=1068)**

**某网上购物网站对用户实行优惠，买家购物货款p越多，则折扣越多。今天正值该网站优惠折扣日，买家可以获得8%的折扣。请编程从键盘输入买家购物货款p，计算并输出买家折扣后实际应付出的价钱。**

【代码】：

#include<stdio.h>

int main()

{

float p,price;

scanf("Input payment p:%f",&p);

price=p\*0.92;

printf("price = %.1f\n",price);

}

**【1070】：[学分绩点计算](https://www.ieeacm.top/problem.php?id=1070)**

**已知某大学期末考试学分绩点的计算公式为：学分绩点 =（工科数学 \* 5 + 英语 \* 1.5 + 线性代数 \* 3.5）/ 10**

**请编程从键盘按顺序输入某学生的工科数学、英语和线性代数成绩，计算并输出其学分绩点。**

【代码】：

#include<stdio.h>

int main()

{

int math1,English,math2;

scanf("Input math1, English and math2:%d,%d,%d",&math1,&English,&math2);

printf("Final score = %.2f\n",(math1\*5+English\*1.5+math2\*3.5)/10);

return 0;

}

**【1071】：[字符串比对](https://www.ieeacm.top/problem.php?id=1071)**

**你会读到两个字符串，每个字符串占据一行，每个字符串的长度均小于10000字符，而且第一个字符串的长度小于第二个字符串的。你的程序要找出第一个字符串在第二个字符串中出现的位置，输出这些位置，如果找不到，则输出-1。**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

char a[10000],b[10000];

gets(a);

gets(b);

int sign=0,n=strlen(a),t=strlen(b);

if(n<t)

{

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

{

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

{

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

{

if(j==n-1) printf("%d ",i);

}

else break;

}

}

}

else

printf("-1");

}

**【1072】：[逆序输出](https://www.ieeacm.top/problem.php?id=1072)**

**读入一系列的正整数，预先不知道正整数的数量，一旦读到-1，就表示输入结束。然后，按照和输入相反的顺序输出所读到的数字，不包括最后标识结束的-1。**

【代码】：

#include<stdio.h>

#include<stdlib.h>

#include<malloc.h>

typedef struct node

{

int num;

struct node \*next;

}Node;

int main()

{

int a;

Node \*head=NULL;

do{

scanf("%d",&a);

if(a!=-1)

{

Node \*p=(Node\*)malloc(sizeof(Node));

p->num=a;

p->next=head;

head=p;

}

}while(a!=-1);

while(head)

{

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

head =head->next;

}

return 0;

}

【1073】：**[计算时间差](https://www.ieeacm.top/problem.php?id=1073)**

**编程从键盘任意输入两个时间（例如4时55分和1时25分），计算并输出这两个时间之间的间隔。要求不输出时间差的负号。**

【代码】:

#include<stdio.h>

#include<string.h>

#include<math.h>

typedef struct clock{

int hour;

int minute;

}CLOCK;

CLOCK Calculate(CLOCK t1,CLOCK t2)

{

CLOCK c3;

int x;

x=((t1.hour)\*60+(t1.minute)-((t2.hour)\*60+(t2.minute)));

x=abs(x);

c3.hour=x/60;

c3.minute=x%60;

return c3;

}

int main()

{

CLOCK c1,c2,c3;

printf("Input time one(hour, minute):");

scanf("%d,%d",&c1.hour,&c1.minute);

printf("Input time two(hour, minute):");

scanf("%d,%d",&c2.hour,&c2.minute);

c3=Calculate(c1,c2);

printf("%d hour %d minute\n",c3.hour,c3.minute);

}

【1074】**[日期显示](https://www.ieeacm.top/problem.php?id=1074)**

**编写一个程序, 接收用户录入的日期信息并且将其显示出来. 其中, 输入日期的形式为月/日/年(mm/dd/yy), 输出日期的形式为年月日(yy.mm.dd)。**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

int a,b,c;

printf("Enter a date (mm/dd/yy):\n");

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

printf("You entered the date: %04d.%02d.%02d\n",c,a,b);

return 0;

}

**【1075】：[逆序数的拆分计算](https://www.ieeacm.top/problem.php?id=1075)**

**从键盘输入一个4位数的整数，编程计算并输出它的逆序数（忽略整数前的正负号）。例如，输入-1234，忽略负号，由1234分离出其千位1、百位2、十位3、个位4，然后计算4\*1000+3\*100+2\*10+1 = 4321，并输出4321。再将得到的逆序数4321拆分为两个2位数的正整数43和21，计算并输出拆分后的两个数的平方和的结果。**

【代码】：

#include<stdio.h>

#include<math.h>

int main()

{

int x,a,b,c,d,sum,e,f,t;

scanf("Input x:\n%d",&x);

x=abs(x);

a=x/1000;

b=(x-a\*1000)/100;

c=(x-a\*1000-b\*100)/10;

d=x%10;

sum=d\*1000+c\*100+b\*10+a;

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

e=d\*10+c;

f=b\*10+a;

printf("a=%d,b=%d\n",e,f);

t=e\*e+f\*f;

printf("result=%d\n",t);

}

【1077】：**[水手分椰子](https://www.ieeacm.top/problem.php?id=1077)**

**n(1<n<=5)个水手在岛上发现一堆椰子，先由第1个水手把椰子分为等量的n堆，还剩下1个给了猴子，自己藏起1堆。然后，第2个水手把剩下的n-1堆混合后重新分为等量的n堆，还剩下1个给**了猴子，自己藏起1堆。以后第3、4个水手依次按此方法处理。最后，第n个水手把剩下的椰子分为等量的n堆后，同样剩下1个给了猴子。请用迭代法编程计算并输出原来这堆椰子至少有多少个,n的值要求从键盘输入。若输入的n值超出要求的范围，程序输出"Error!"。

【代码】：

#include<stdio.h>

int divide(int n,int total,int times);

int main(){

int n;

int total=1;

scanf("%d",&n);

if(n<=1||n>5){

printf("Error!\n");

return 0;

}

while(divide(n,total,0)!=1){

total++;

}

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

return 0;

}

int divide(int n,int total,int times){

if(times==n){

return 1;

}

return !((total-1)%n)&&divide(n,(total-1)\*(n-1)/n,++times);

}

**【1078】：[出租车计价](https://www.ieeacm.top/problem.php?id=1078)**

**已知某城市普通出租车收费标准为：起步里程为3公里，起步费为8元，10公里以内超过起步里程的部分，每公里加收2元，超过10公里以上的部分加收50%的回空补贴费，即每公里3元。出租车营运过程中，因堵车和乘客要求临时停车等客的，按每5分钟加收2元计算，不足5分钟的不计费。从键盘任意输入行驶里程（精确到0.1公里）和等待时间（精确到分钟），请编程计算并输出乘客应支付的车费，对结果进行四舍五入，精确到元。**

【代码】：

#include<stdio.h>

int main()

{

int t;

float d,f;

scanf("Input distance and time:%f,%d",&d,&t);

if(d<3){

f=8;

}

else if(d<=10)

{

f=8+(d-3)\*2;

}

else

{

f=8+7\*2+(d-10)\*3;

}

f=f+(t/5)\*2;

printf("fee = %.0f\n",f);

}

**【1079】：[博弈论](https://www.ieeacm.top/problem.php?id=1079)**

**假设一个班级中的学生分为A、B两类，A类同学学习热情很高，乐于做题，采用compete模式可以获得成就感并且在期末拿到高分，compete模式可以让他们有10分的收益；采用standard模式他们也可以在期末拿到高分，**但不能满足他们的求知欲，standard模式可以让他们有8分的收益。B类同学仅仅希望期末拿高分，如果采用compete模式，他们竞争不过A类同学，期末成绩不理想，因此compete模式能给他们6分的收益；如果采用standard模式，他们可以完成规定任务并拿到高分，因此standard模式可以让他们有10分的收益。  
 编程输入A类和B类同学分别占班级总人数的百分比，分别计算并输出采用compete和standard两种刷题模式下的全班总收益，并输出这个班级在这场博弈中的Best Response是哪种模式。注： 程序中使用的数据类型为float

【代码】：

#include<stdio.h>

int main()

{

float a,b,c,s;

scanf("Input percent of A and B:%f%f",&a,&b);

c=a\*10+b\*6;

s=a\*8+b\*10;

printf("compete = %.4f\nstandard = %.4f\n",c,s);

if(c>s){

printf("The Best Response is compete!\n");

} else{

printf("The Best Response is standard!\n");

}

}

【1081】**魔术师猜数**

**在一种室内互动游戏中，魔术师要每位观众心里想一个三位数abc（a、b、c分别是百位、十位和个位数字），然后魔术师让观众心中记下acb、bac、bca、cab、cba五个数以及这5个数的和值。只要观众说出这个和是多少**，则魔术师一定能猜出观众心里想的原数abc是多少。例如，观众甲说他计算的和值是1999，则魔术师立即说出他想的数是443，而观众乙说他计算的和值是1998，则魔术师说：“你算错了！”。请编程模拟这个数字魔术游戏。要求用函数实现，函数原型为：int Magic(int m);  
其中形参m代表观众计算的和值。

【代码】

#include<stdio.h>

int Magic(int n)

{

int a,b,c;

int sum;

for(a=0;a<=9;a++){

for(b=0;b<=9;b++){

for(c=0;c<=9;c++){

if(a\*100+c\*10+b+b\*100+a\*10+c+b\*100+c\*10+a+c\*100+a\*10+b+c\*100+b\*10+a==n){

return a\*100+b\*10+c;

}

}

}

}

}

int main()

{

int n;

scanf("The number is %d",&n);

if(Magic(n)!=4995){ //用4995确定是否error原因是如果迭代到最后如果仍没有找到合适

printf("The number is %d\n",Magic(n)); //的值，出错后的最终结果始终为4995(尝试出的结果

}else{

printf("The sum you calculated is wrong!\n");

}

}

**【1083】：Homework**

**临近开学了，大家都忙着收拾行李准备返校，但I\_Love\_C却不为此担心! 因为他的心思全在暑假作业上：目前为止还未开动(-\_-!!还以为他有多冷静呢)。  
暑假作业是很多张试卷，我们这些从试卷里爬出来的人都知道，卷子上的题目有选择题、填空题、简答题、证明题等。 而做选择题的好处就在于工作量很少，但又因为选择题题目都普遍很长。 如果有5张试卷，其中4张是选择题，最后一张是填空题，很明显做最后一张所花的时间要比前4张长很多。 但如果你只做了选择题，虽然工作量很少，但表面上看起来也已经做了4/5的作业了。 I\_Love\_C决定就用这样的方法来蒙混过关。  
他统计出了做完每一张试卷所需的时间以及它做完后能得到的价值（按上面的原理，选择题越多价值当然就越高咯）。 现在就请你帮他安排一下，用他仅剩的一点时间来做最有价值的作业。需要注意的是，I\_Love\_C剩下的暑假时间真的很少，有可能不能完成整张卷子，而每张试卷的价值与其完成度成正比。**

【代码】：

#include<stdio.h>

#include<algorithm>

#include<string.h>

using namespace std;

struct wk{

double tt,v,ff;

};

bool comp(wk a,wk b){

return a.ff>b.ff;

}

int main(){

wk my[12000];

int m,n,i,j;

double k;

while(~scanf("%d%d",&n,&m)&&(n||m)){

for(i=0,j=0,k=0;i<n;i++){

scanf("%lf%lf",&my[i].tt,&my[i].v);

my[i].ff=my[i].v/my[i].tt;

}

sort(my,my+n,comp);

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

{

if(j+my[i].tt<=m)j+=my[i].tt,k+=my[i].v;

else

{

if(j!=m)

{

j=m-j;

k+=j\*my[i].ff;

}

break;

}

}

printf("%.2lf\n",k);

}

}

**【1084】：Redraiment的遭遇**

**Redraiment的老家住在工业区，日耗电量非常大。是政府的眼中钉肉中刺，但又没办法，这里头住的可都是纳税大户呀。 今年7月，又传来了不幸的消息，政府要在7、8月对该区进行拉闸限电。但迫于压力，限电制度规则不会太抠门，政府决定从7月1日停电，然后隔一天到7月3日再停电，再隔两天到7月6日停电，一次下去，每次都比上一次晚一天。 Redraiment可是软件专业的学生，怎么离得开计算机。如果停电，就“英雄无用武之地”了。呵呵。 所以他开始盘算起自己回家的日子了，他想知道自己到家后到底要经历多少天倒霉的停电。你能帮他算一算吗？**

【代码】

#include<stdio.h>

int main()

{

int m,d;

int day[63];

int i=1,t=1,j;

for(j=1;j<=62;j++) //七八月每一天统一置为1

{

day[j]=1;

}

while(i<=62){ //将停电当天置为0

day[i]=0;

t++;

i=i+t;

}

while(scanf("%d/%d\n",&m,&d)){

if(m==0&&d==0) break;

int td=0; //过去已经停电的天数

if(m==7){

int k;

for(k=1;k<d;k++){

if(day[k]==0){

td++;

}

}

}else if(m==8){

td=7; //七月一共停电7天

int k;

for(k=1;k<d;k++){

if(day[k+31]==0){ //八月下标需加上31

td++;

}

}

}else{ //除七八月其他月份停电情况不考虑

td=10;

}

printf("%d\n",10-td);

}

}

**【1085】1、2、3、4、5...**

**为了能让自己的好朋友吃的饱饱的，redraiment决定统计一下有多少只鸽子和有多少只兔子，以便带来足够的食物。一、二、三、四、五...他开始数了。  
现在，他已经知道有这些鸽子和兔子一共有n个头和m只脚。请你帮他写个程序计算一下一共有多少只鸽子和兔子。**

【代码】

#include<stdio.h>

int main()

{

int n,m;

while(scanf("%d%d\n",&n,&m)&&n!=0&&m!=0){

if(m%2!=0||m<2\*n||m>4\*n)

{

printf("Error\n");

}

else{

printf("%d %d\n",(4\*n-m)/2,(m/2-n));

}

}

return 0;

}

**【1088】：Redraiment猜想**

**redraiment在家极度无聊，于是找了张纸开始统计素数的个数。 设函数f(n)返回从1->n之间素数的个数。 redraiment发现: f(1) = 0 f(10) = 4 f(100) = 25 ... 满足g(m) = 17 \* m2 / 3 - 22 \* m / 3 + 5 / 3 其中m为n的位数。 他很激动，是不是自己发现了素数分布的规律了！ 请你设计一个程序，求出1->n范围内素数的个数，**

【代码】：

#include<stdio.h>

#include<math.h>

int main(){

int i,j,n,flag=-1,count=0;

while(scanf("%d\n",&n),n){

count=0;

if(n==1){

printf("0\n");

}

else{

for(i=2;i<=n;i++){

flag=0;

for(j=2;j<=sqrt(i);j++){

if(i%j==0){

flag=1;break;

}

}

if(flag==0){

count++;

}

}

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

}

}

}

**【1089】：童年生活二三事**

**Redraiment小时候走路喜欢蹦蹦跳跳，他最喜欢在楼梯上跳来跳去。 但年幼的他一次只能走上一阶或者一下子蹦上两阶。 现在一共有N阶台阶，请你计算一下Redraiment从第0阶到第N阶共有几种走法。**

【代码】：

#include<stdio.h>

int main(){

int a[41];

int i,n;

a[1]=1,a[2]=2;

for(i=3;i<=40;i++){

a[i]=a[i-2]+a[i-1];

}

while((scanf("%d",&n)==1)&&(n!=0))

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

return 0;

}

**【1090】University**

**在大学里，很多单词都是一词多义，偶尔在文章里还要用引申义。这困扰Redraiment很长的时间。**

**他开始搜集那些单词的所有意义。他发现了一些规律，例如**

**“a”能用“e”来代替, “c”能用“f”来代替……**

**现在他给出了字母的替换规则，如下所示，A被E替换，B被C替换，依次类推。**

**A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**

**E C F A J K L B D G H I V W Z Y M N O P Q R S T U X**

**a b c d e f g h i j k l m n o p q r s t u v w x y z**

**e r w q t y g h b n u i o p s j k d l f a z x c v m**

【代码】：（C++）

#include<iostream>

#include<cstdio>

using namespace std;

int main()

{

char B[]={"ECFAJKLBDGHIVWZYMNOPQRSTUX"};

char C[]={"erwqtyghbnuiopsjkdlfazxcvm"};

char a;

int num;

while((a=getchar())!='#')

{

if(a=='\n')

cout<<'\n';

else if(a==' ')

cout<<' ';

else if(a>='A'&&a<='Z')

{

num=a-65;

cout<<B[num];

}

else{

num=a-97;

cout<<C[num];

}

}

return 0;

}

**【1091】：苏军知道德军用的加密方法是这样的： 1.获得一段文字后，求出它的长度(包括空格)len。 2.进入加密运算的第1步:把所有下标是1倍数的字符做顺时针旋转。 3.进入加密运算的第2步:把所有下标是2倍数的字符做逆时针旋转。 4.进入加密运算的第3步:把所有下标是3倍数的字符做顺时针旋转。 5.按上面的规则，第奇数步按顺时针旋转，偶数步按逆时针旋转，一直到第len步为止。  
比如原文是:abcde 1.获得长度len = 5 2.1的倍数有1、2、3、4、5，所以把这5个字符按顺时针旋转，得到eabcd。 3.2的倍数有2、4，所以把这2个字符按逆时针旋转，得到ecbad。 4.3的倍数有3，所以把这1个字符按顺时针旋转，得到ecbad。 5.4的倍数有4，所以把这1个字符按逆时针旋转，得到ecbad。 6.5的倍数有5，所以把这1个字符按顺时针旋转，得到ecbad。 最后的结果是ecbad。  
现在给你加密后的文章，让你还原成原来的文章。**

【代码】：

#include<stdio.h>

#include<string.h>

#define LEN 1000+10

int turn\_right(char \*c,int \*a,int k) //顺时针旋转

{

int i;

char temp;

temp=c[a[k-1]];

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

c[a[k-1-i]]=c[a[k-2-i]];

c[a[0]]=temp;

}

int turn\_left(char \*c,int \*a,int k) //逆时针旋转

{

int i;

char temp;

temp=c[a[0]];

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

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

c[a[k-1]]=temp;

}

char c[LEN];

int a[LEN][LEN]={0};

int main()

{

int len;

int i,j,k=0;

for(i=1;i<=1000;i++){ // 这里把1-n的所有倍数都处理出来，不用每次循环都求，节省时间

k=0;

for(j=0;j<1000;j++){

if((j+1)%i==0)

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

}

}

while(gets(c))

{

len=strlen(c);

for(i=len;i>=1;i--)

{

k=len/i;

if(k==1)

{

continue;

}

if(i%2==0) turn\_right(c,a[i],k);

else turn\_left(c,a[i],k);

}

puts(c);

}

return 0;

}

**【1130】简易版最长序列**

**给你一组数(未排序)，请你写设计一个程序：求出里面个数最多的数（可以不连续出现），并输出这个数的长度。例如：给你的数是：1、 2、 3、 3、 4、 4、 5、 5、 5 、6, 其中只有6组数：1, 2, 3-3, 4-4, 5-5-5 and 6. 最长的是5那组，长度为3。所以输出3。**

**第一行为整数t((1 ≤ t ≤ 10))，表示有n组测试数据。每组测试数据包括两行，第一行位数组的长度n (1 ≤ n ≤ 10000)。第二行为n个整数，所有整数Mi的范围都是(1 ≤ Mi ≤ 232)**

【代码】

#include<stdio.h>

#include<string.h>

int main(){

int m,n,i,j,t,len,lenmax,a[10000];

scanf("%d",&m);

while(m--){

scanf("%d",&n);

len=1;lenmax=1;

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

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

}

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

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

if(a[i]>a[j]){

t=a[i];

a[i]=a[j];

a[j]=t;

}

}

}

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

if(a[i]==a[i-1]) len++;

else len=1;

if(len>lenmax) lenmax=len;

}

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

}

}

# 【1133】**A × B problem**

**输入：数据的第一行是整数T(1 ≤ T ≤ 20)，代表测试数据的组数。 接着有T组数据，每组数据只有一行，包括两个非负数A和B。 但A和B非常大，Redraiment能保证这些数用long来保存一定会溢出。 但A和B的位数最大不会超过100位。**

**输出：对应每组测试数据，你都要输出两行： 第一行为："Case #:", # 代表这是第几组测试数据。 第二行是一个等式："A \* B = Sum", Sum 代表 A × B 的结果。 你要注意这个等式里包含了几个空格。 要求每组数据之间都需要保留一个空行。**

【代码】：

#include <stdio.h>

#include <string.h>

int main()

{

char a[101],b[101];

int i,j,k,cas,count=1,num2,num3;

scanf("%d",&cas);

getchar();

while(cas--)

{

int c[100000]={0};

scanf("%s %s",a,b);

num2=strlen(a);

num3=strlen(b);

getchar();

if(strcmp(a,"0")==0||strcmp(b,"0")==0)

{

printf("Case %d:\n",count);

printf("%s \* %s = 0\n",a,b);

count++;

}

else {

for(i=0;i<num2;i++)

{

for(j=0;j<num3;j++){

c[i+j]+=(a[num2-i-1]-'0')\*(b[num3-j-1]-'0');//从个位数开始乘以第二个数字的每一位

k=i+j;

while(c[k]>=10) //要进位

{c[k+1]+=c[k]/10;c[k]%=10;k++;}

}

}

i=9999;num2=i;//从得数最高位是否为0开始判断数字总共有几位

while(c[i]==0)

{

i--;num2 = i;

}

printf("Case %d:\n",count);

count++;

printf("%s \* %s = ",a,b);

for(i=num2;i>=0;i--)

{printf("%d",c[i]);}

printf("\n");}

if(cas>=1)

{printf("\n");}

}

return 0;

}

# 【1138】**英文金曲大赛**

**当时有7个评委，每个评委都要给选手打分，现在要求去掉一个最高分和去掉一个最低分，再算出平均分。结果精确到小数点后两位。**

**输入**

**测试数据包括多个实例。 每组数据包括7个实数，代表评委们对该选手的评分。紧接着是选手的名字，名字的长度不超过30个字符。 输入直到文件结束。**

**输出**

**算出每位选手名字和最终得分，结果保留两位有效数字。**

【代码】：

#include<stdio.h>

struct ss{

double score[7];

char name[30];

}opp;

int main(){

int i,j;

double ave,sum=0,temp;

while(~scanf("%lf",&opp.score[0])){ //当读到文件结尾时终止循环

sum=0;

for(i=1;i<=6;i++){

scanf("%lf",&opp.score[i]);

}

scanf("%s",opp.name);

for(i=0;i<6;i++){ //排序

for(j=i+1;j<7;j++){

if(opp.score[i]>opp.score[j]){

temp=opp.score[i];

opp.score[i]=opp.score[j];

opp.score[j]=temp;

}

}

}

for(i=1;i<=5;i++){

sum=sum+opp.score[i];

}

ave=sum/5.0;

printf("%s %.2lf\n",opp.name,ave);

}

}

# **【1139】 有假币**

**现在知道假币的重量比真币的质量要轻。给你一个天平，请用最快的时间把那个可恶的假币找出来**

**输入**

**1≤n≤230,输入0结束程序。**

**输出**

**最少要称几次一定能把那个假币找出**

【代码】

#include<stdio.h>

#include<math.h>

int main (){

long long int n;

int m;

while(scanf("%lld",&n)&&n) //先输入n,再判断是否为0，当输入n为0时则不进行处理

{

m=ceil(log(n)/log(3));//ceil返回大于或者等于指定表达式的最小整数 log(n)/log(3为计算找规律得出

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

}

return 0;

}

# **【1140】 寄居蟹与海葵**

1. **其中A代表0，B代表1，依次类推。我们称两者相加等于25的就算是吻合，比如A和Z相吻合，B与Y吻合，依次类推。海葵的褶皱是"ABCDEFG"，寄居蟹是"ZYXWVUT"。这样，它们就可以完全吻合了。 2.海葵的褶皱是"AHBICJDKELFMGN"，寄居蟹是"ZYXWVUT"。这样，寄居蟹可以和海葵的部分序列"ABCDEFG"相吻合 (注意:部分序列不改变字符原来的先后顺序，比如"ACB"就不是它的部分序列)。 3.海葵的褶皱是"ABCD"，寄居蟹是"ZYXWVUT"。这样，虽然海葵可以和寄居蟹前面一段完全吻合，但它比寄居蟹要小，不能完全保护寄居蟹的安全，所有它们是不适合的。 4.海葵的褶皱是"HIJKLMNOPQ"，寄居蟹是"ZYXWVUT"。这样，它们就可以完全不吻合了。 现给你两段字符串S1、S2，分别代表海葵和寄居蟹的外壳，为了它们以后各都能快乐地生活，请你帮忙计算一下它们是不是吻合的。**

**输入**

**输入包括多组测试数据。 每组测试数据包括两个字符串H、J，分别代表海葵的外壳和寄居蟹的外壳。可以保证它们的长度都小于100000。 输入以0 0结束。**

**输出**

**如果寄居蟹和海葵的外壳能吻合，就输出"Yes"，否则输出"No"**。

【代码】：

#include<stdio.h>

#include<string.h>

#include<math.h>

int main(){

char b[100010],c[100010];

while(scanf("%s %s",b,c)){

if(strcmp(b,"0")==0&&strcmp(c,"0")==0) break;

int m=strlen(b);

int n=strlen(c);

int i=0,k=0;

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

if(b[i]-'A'+c[k]-'A'==25){

k++;

}

}

if(k>=n)

printf("Yes\n");

else

printf("No\n");

}

}

# **1141: 天仙配**

**记分规则是：男方的分数等于把从自己所站的位置开始一直累加到开头，女方的分数等于从自己所站位置开始一直累加到末尾。如果某一对新人的得分数是相同的，那你们就获胜。可以得到一份司仪精心准备的礼物。^\_^ 比如，有3对新人，地上的那一排数字为:3，6，2，4，5，2。 如果男方站在第三个位置(2)，他的得分为：3+6+2=11；女方站在第4个位置(4)，她的得分为4+5+2=11。两人得分相同，可以获胜。 或者男方站第6个位置(2)，女方站第1个位置(3)，他们的得分都等于22，也可以获胜。**

**输入**

**输入包括多组测试数据。 每组测试数据包括两行。 第一行为一个数据n，即新人的对数。 第二行有2n个数据，代表地上的数字。 输入以0结束，这一行不做处理。**

**输出**

**每组输出占一行。 输出共有几种站法。**

【代码】C++

#include <iostream>

#include <algorithm>

using namespace std;

long people[500000],total[100000];

int main()

{

int n,i,j,num,s1,s2;

while(scanf("%d",&n)!='0')

{

num=0;

s1=0;

s2=0;

j=0;

if(n==0)

break;

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

{

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

}

for(i=0; i<2\*n; i++) //将男方在各个位置的数字统计出来存在数组中

{

s1=s1+total[i];

people[j]=s1;

j++;

}

for(i=2\*n-1; i>=0; i--) //将女方在各个位置的分数统计在数组中 people（2n~4n)

{

s2=s2+total[i];

people[j]=s2;

j++;

}

sort(people,people+4\*n);//对各个分数进行排序 (0~4n)

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

{

if(people[i]==people[i+1])

num++;

}

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

}

return 0;

}

# **1142: 鹊桥相会**

**输入**

**第一行有两个数据w、n，分别代表天河的宽度(单位:km)和喜鹊的只数(1≤w≤1000, 1≤n≤10000)。 接下来从第二行到第n+1行每行都有两个数据t、v，分别代表1只喜鹊的初始位置(单位:m)和它的飞行速度(单位:m/s)(-1000≤t≤1000, -100≤v≤100)。 所有的数据范围都不会超过32位整数的表示范围(用int型数据不会溢出)。 输入以0 0结束。**

**输出**

**如果牛郎能到达对岸输出他到达对岸所花的总时间(结果精确到秒即可，小数部分舍去)；否则输出“Can't Solve”。**

【代码】：

#include<stdio.h>

int main(){

int i,n;

int w,v,t;

double min,time;

while(scanf("%d%d",&w,&n),w+n){

min=1<<30; //取最小值

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

scanf("%d%d",&t,&v);

if(t<=0&&v>0){

time=1.0\*(w\*1000-t)/v; //位置大于0的鸟是不可能换，只需要考虑位置小于0的鸟

if(time<min) min=time;

}

}

if(min==(1<<30)) printf("Can't Solve\n");

else printf("%d\n",(int)min);

}

}

# **1143: 杨辉三角**

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

**输入**

**输入数据包含多组测试数据。 每组测试数据的输入只有一个正整数n（1≤n≤30），表示将要输出的杨辉三角的层数。 输入以0结束**

**输出**

**对应于每一个输入，请输出相应层数的杨辉三角，每一层的整数之间用一个空格隔开，每一个杨辉三角后面加一个空行。**

【代码】

#include<stdio.h>

int main(){

int n,i,j;

int a[30][30];

while(scanf("%d",&n)!='0'){

if(n==0) break;

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

a[i][0]=1;

a[i][i]=1;//第一列全部数字为1，每行最后一个数字也是1

}

for(i=2;i<n;i++){

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

a[i][j]=a[i-1][j-1]+a[i-1][j];//中间值都等于上一行前一列+上一行此列

}

}

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

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

{

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

}

printf("\n");

}

printf("\n");

}

return 0;

}

# **1144: 蟠桃记**

**孙悟空在大闹蟠桃园的时候，第一天吃掉了所有桃子总数一半多一个，第二天又将剩下的桃子吃掉一半多一个，以后每天吃掉前一天剩下的一半多一个，到第n天准备吃的时候只剩下一个桃子。这下可把神仙们心疼坏了，请帮忙计算一下，第一天开始吃的时候桃子一共有多少个桃子。**

**输入：输入数据有多组，每组占一行，包含一个正整数n（1≤n≤30），表示只剩下一个桃子的时候是在第n天发生的。 输入以0结束。**

**输出：对于每组输入数据，输出第一天开始吃的时候桃子的总数，每个测试实例占一行。**

【代码】；

#include<stdio.h>

int main(){

int n,i,sum;

while(scanf("%d",&n)!=0){

if(n==0) break;

sum=1;

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

sum=(sum+1)\*2;

}

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

}

}

# **1145: 养兔子**

**题目描述:一只成熟的兔子每天能产下一胎兔子。每只小兔子的成熟期是一天。 某人领养了一只小兔子，请问第N天以后，他将会得到多少只兔子。**

**输入:测试数据包括多组，每组一行，为整数n(1≤n≤90)。 输入以0结束。**

**输出:对应输出第n天有几只兔子(假设没有兔子死亡现象)。**

【代码】：

#include<stdio.h>

#include<string.h>

int main()

{

int n,i;

long long int a[10000]={1,2};

while(scanf("%d",&n)!=0){

if(n==0) break;

for(i=2;i<n;i++)

{

a[i]=a[i-1]+a[i-2];

}

printf("%lld\n",a[n-1]);

}

}

# **1147: 完美数**

**题目描述:如6的所有真约数是1、2、3，而且6=1＋2＋3。像这样，一个数所有真约数的和正好等于这个数，通常把这个数叫做完美数。**

**输入:输入数据有多组，每组占一行，包括两个整数m和n（1≤m, n≤99999999）。 输入以0 0结束**

**输出:对于每个测试实例，要求输出所有在给定范围内的完美数，就是说，输出的完美数必须大于等于m，并且小于等于n，如果有多个，则要求从小到大排列在一行内输出，之间用一个空格隔开； 如果给定的范围内不存在完美数，则输出No； 每个测试实例的输出占一行。**

【代码】

#include <stdio.h>

int dict[] = { 6, 28, 496, 8128, 33550336 }; //不先计算出来会出现时间超限情况

int main()

{

int m, n;

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

{

if (m == 0 && n == 0)

{

break;

}

int flag = 0;

for (int i = 0; i < 5; i++)

{

if (dict[i] >= m && dict[i] <= n)

{

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

flag++;

}

}

if (flag == 0)

{

printf("No");

}

printf("\n");

}

return 0;

}

# **1150: 找零钱**

**题目描述：我们知道人民币有1、2、5、10、20、50、100这几种面值。 现在给你n(1≤n≤250)元，让你计算换成用上面这些面额表示且总数不超过100张，共有几种。 比如4元，能用4张1元、2张1元和1张2元、2张2元，三种表示方法。**

**输入：输入有多组，每组一行，为一个整合n。 输入以0结束。**

**输出：输出该面额有几种表示方法。**

【代码】：

#include <iostream>

#include <cstdio>

#include <cstring>

#include <algorithm>

#define ll long long int

#define MAX 17

#define N 255

using namespace std;

int a[N][N];

int b[N][N];

int ans[N];

int v[]= {1,2,5,10,20,50,100};

void init()///打表

{

int i,j,k,l;

a[0][0]=1;

for (i=0; i<7; i++)///面值

{

for (j=0; j\*v[i]<N; j++)///步长

{

for (k=0; k+j\*v[i]<N; k++)///从x^0到x^N遍历一遍

{

for(l=0; l+j<=100; l++)///钞票数量

{

b[j\*v[i]+k][l+j]+=a[k][l];

}

}

}

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

{

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

{

a[j][k]=b[j][k];

b[j][k]=0;

}

}

}

for(i=0;i<N;i++)///转换

{

for(j=0;j<=100;j++)

{

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

}

}

}

int main()

{

int n;

init();

while(scanf("%d",&n)!=EOF)

{

if(n==0)

{

break;

}

printf("%d\n",ans[n]);

}

return 0;

}

# **1151: 漂亮菱形**

输入：测试数据包括多行，每行1个整数h，为菱形的高度。 输入以0结束。

输出：输入对应的菱形。

【代码】：

#include<stdio.h>

int main()

{

int n,i,j,k;

while(scanf("%d",&n)!=0)

{

if(n==0) break;

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

{

for(j=0;j<(n+1)/2-i;j++) printf(" ");

for(k=1;k<=2\*i-1;k++) printf("\*");

printf("\n");

}

for(i=n/2;i>0;i--)

{

for(j=0;j<(n+1)/2-i;j++) printf(" ");

for(k=1;k<=2\*i-1;k++) printf("\*");

printf("\n");

}

}

}

# **1152: Least Common Multiple**

**题目描述**

**求n个数的最小公倍数。**

**输入**

**输入包含多个测试实例，每个测试实例的开始是一个正整数n，然后是n个正整数。**

**输出**

**为每组测试数据输出它们的最小公倍数，每个测试实例的输出占一行。你可以假设最后的输出是一个32位的整数。**

【代码】

#include<stdio.h>

int main()

{

int x\_y(int x,int y);

int i,n,a[1111],s;

while(scanf("%d",&n)!=EOF)

{

if(n==0) break;

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

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

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

else

{

s=x\_y(a[0],a[1]);

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

s=x\_y(s,a[i]);

}

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

}

return 0;

}

int x\_y(int x,int y)

{

int a,b,t,m;

a=x;b=y;

if(x>y)

{t=x;x=y;y=t;}

for(;x!=0;)

{

t=y%x;

y=x;

x=t;

}

m=(a/y)\*b;

return m;

}

# **1153: 第几天**

**给定一个日期，输出这个日期是该年的第几天。**

**输入:数据有多组，每组占一行，数据格式为YYYY-MM-DD组成，具体参见sample input ,另外，可以向你确保所有的输入数据是合法的。**

**输出:对于每组输入数据，输出一行，表示该日期是该年的第几天。**

【代码】：

#include<stdio.h>

struct data{

int year;

int month;

int day;

};

int main()

{

struct data da;

int year,month,day,n=0;

while (scanf("%d-%d-%d", &da.year, &da.month, &da.day) != EOF)

{

int d1[12] = {31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

int d2[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

year=da.year;

n=0;

if((year%4==0&&year%100!=0)||(year%400==0))

{

for(int i=0;i<da.month-1;i++)

{

n=n+d1[i];

}

n=n+da.day;

}

else

{

for(int i=0;i<da.month-1;i++)

{

n=n+d2[i];

}

n=n+da.day;

}

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

}

}

# **1154: 编辑距离**

假设字符串的基本操作仅为：删除一个字符、插入一个字符和将一个字符修改成另一个字符这三种操作。 我们把进行了一次上述三种操作的任意一种操作称为进行了一步字符基本操作。 下面我们定义两个字符串的编辑距离：对于两个字符串a和b，通过上述的基本操作，我们可以把a变成b或b变成a，那么字符串a变成字符串b需要的最少基本字符操作步数称为字符串a和字符串b的编辑距离。 例如：a="ABC",b="CBCD",则a与b的编辑距离为2。 你的任务就是：编一个快速的程序来计算任意两个字符串的编辑距离。

输入包含多组测试数据。每组测试数据一行，为字符串A和字符串B。 字符串的长度不大于1024，且全为字母。

输出编辑距离。

【代码】

# **1156: 等值数目**

**已知两个整数数组f[]和g[]，它们的元素都已经从小到大排列。例如f[]中可能有1，2，2，3，3，g[]中有1，2，2，2，3（注意这里的f[]、g[]与输入样例中的f[]、g[]正好颠倒过来了，但是输出结果都应该是4，由计科类1901班黄恩怡同学特此提醒）。 请写一个程序，算出这两个数组彼此之间有多少组相同的数据。就以上例而言: f[0]与g[0]是第一组； f[1]与g[1]是第二组； f[2]与g[2]是第三组； f[3]与g[4]是第四组。**

**输入：第一行为两个整数m, n(1≤m, n≤1000)，分别代表数组f[], g[]的长度。 第二行有m个元素，为数组f[]。 第三行有n个元素，为数组g[]。**

输出等值数目。

【代码】

#include<stdio.h>

#include<string.h>

int main()

{

int i,j,m,n,k;

int a[1000],b[1000];

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

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

{

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

}

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

{

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

}

i=j=k=0;

while(i<n&&j<m)

{

if(a[i]==b[j]) {

k++;i++;j++;}

else{

if(a[i]<b[j]) i++;

else j++;

}

}

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

}

# **1161: 输出入门**

**计算A+B**

**输入数据有多组。 每组一行，为两个整数A, B。 输入以0 0结束。**

**输出A+B的值，每组数据之间保留一个空行。**

【代码】

#include<stdio.h>

int main()

{

int a,b,sum;

while(scanf("%d%d",&a,&b)!='0 0'){

if(a==0&&b==0) break;

sum=0;

sum=a+b;

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

printf("\n");

}

}

# **1170: 求奇数的乘积**

**给你n个整数，求他们中所有奇数的乘积。**

**输入数据包含多个测试实例，每个测试实例占一行，每行的第一个数为n，表示本组数据一共有n个，接着是n个整数，你可以假设每组数据必定至少存在一个奇数。**

**输出每组数中的所有奇数的乘积，对于测试实例，输出一行。**

【代码】：

#include<stdio.h>

int main()

{

int n,sum,num,i;

while(scanf("%d",&n)!=EOF)

{

sum=1;

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

{

scanf("%d",&num);

if(num%2!=0) sum=sum\*num;

}

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

}

}

# **1171: 平方和与立方和**

**给定一段连续的整数，求出他们中所有偶数的平方和以及所有奇数的立方和。**

**输入数据包含多组测试实例，每组测试实例包含一行，由两个整数m和n组成。**

**输出对于每组输入数据，输出一行，应包括两个整数x和y，分别表示该段连续的整数中所有偶数的平方和以及所有奇数的立方和。 你可以认为32位整数足以保存结果。**

【代码】

#include<stdio.h>

int main()

{

int m,n,i,j,x,y;

while(scanf("%d%d",&m,&n)!=EOF)

{

x=0;

y=0;

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

{

if(i%2==0) x=x+i\*i;

}

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

{

if(i%2!=0) y=y+i\*i\*i;

}

printf("%d %d",x,y);

printf("\n");

}

}

# **1172: 绝对值排序**

**输入n(n<=100)个整数，按照绝对值从大到小排序后输出。题目保证对于每一个测试实例，所有的数的绝对值都不相等。**

**输入数据有多组，每组占一行，每行的第一个数字为n,接着是n个整数，n=0表示输入数据的结束，不做处理。**

**输出对于每个测试实例，输出排序后的结果，两个数之间用一个空格隔开。每个测试实例占一行。**

【代码】

#include<stdio.h>

#include<math.h>

int main()

{

int n,i,t,j;

int a[100];

while(scanf("%d",&n)!='0')

{

if(n==0) break;

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

{

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

}

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

{

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

if(abs(a[j])<abs(a[j+1]))

{

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

}

}

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

{

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

}

printf("\n");

}

}

# **1176: C语言考试练习题\_一元二次方程**

**解一元二次方程ax2+bx+c=0的解。**

**输入a,b,c的值。**

**输出两个根X1和X2，其中X1>=X2。。 结果保留两位小数。**

【代码】

#include<stdio.h>

#include<math.h>

int main()

{

double a,b,c,d,x1,x2;

scanf("%lf%lf%lf",&a,&b,&c);

d=b\*b-4\*a\*c;

if(d>=0)

{

x1=(-b+sqrt(d))/(2\*a);

x2=(-b-sqrt(d))/(2\*a);

if(x1>=x2) printf("%.2lf %.2lf\n",x1,x2);

else printf("%.2lf %.2lf\n",x2,x1);

}

else printf("error!\n");

}