### 加法的横式与竖式运算式子

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

    int a,b;

    cin>> a >>b ;

    cout <<a<<"+"<<b<<"="<< a+b <<endl;

    cout <<setw(7)<<a<< endl;

    cout << "+"<<setw(6) <<b<< endl;

    cout << "-------"<<endl;

    cout << setw(7)<<a+b<<endl;

    return 0;

}

### 分西瓜

#include <iostream>

//#include <stdio.h>

//#include <iomanip>

//#include <math.h>

using namespace std;

int main()

{

int weight,a,b;

cin>> weight;

a=weight%2;

b=weight-2;

if( a==0 && b%2==0 && weight!=2)

cout << "YES, you can divide the watermelon into two even parts." <<endl;

else

cout << "NO, you can't divide the watermelon into two even parts." <<endl;

//printf("%.3lf\n",avr);

return 0;

}

### 某年某月有几天

#include<stdio.h>

int main(){

int year, month, ok = 0;

scanf("%d %d", &year, &month);

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

{

ok = 1;

}

switch(month){

case 1:

case 3:

case 5:

case 7:

case 8:

case 10:

case 12:printf("31\n"); break;

case 4:

case 6:

case 9:

case 11:printf("30\n"); break;

case 2: ok ? printf("29\n") : printf("28\n"); break;

}

return 0;

}

### 寻找最大和最小值

#include <iostream>

#include <algorithm>

//#include <stdio.h>

using namespace std;

int main()

{

int n;

cin>>n ;

int \*p = new int[n];

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

{

cin >> p[i];

}

cout << (\*max\_element(p, p + n)) << endl << (\*min\_element(p, p + n))<<endl;

//printf("%.3lf\n",avr);

return 0;

}

### [菲菠哪切数列](http://115.236.49.52:83/problem/3553)

#include <iostream>

using namespace std;

int fib(int n);

int main()

{

int a;

while(cin>>a)

{

cout<<fib(a)<<endl;

}

return 0;

}

int fib(int n)

{

if(n==1||n==2)

return 1;

else

return fib(n-1)+fib(n-2); //递归

}

### [求数列的第N项](http://115.236.49.52:83/problem/1119)

#include <iostream>

#include <stdio.h>

using namespace std;

int main()

{

int n;

cin>>n ;

int \*p = new int[n];

\*p=0;\*(p+1)=1;\*(p+2)=1;

for (int i = 3; i <= n-1; i++)

{

\*(p+3)=\*p + (\*(p+1))\*2 + \*(p+2);

p++;

}

cout << \*(p+2) <<endl;

//printf("%.6lf\n",\*p);

return 0;

}

### 循环-5的倍数

#include <iostream>

//#include <stdio.h>

using namespace std;

int main()

{

    int a,b,c;int num,count=0;

    cin >> a >> b;

    for (int i=a ; i <= b ; i++  ) //之间包括了本身

    {

        c=i;num=0;

        while(c!=0)

        {

          num = num + c%10;

          c/=10;

        }

        if (num%5==0&&num!=0)

            count++;

    }

    cout<< count <<endl;

    //printf("%.2lf\n",DI);

    return 0;

}

### 循环-判断素数

//#include <iostream>

//#include <stdio.h>

//using namespace std;

//

//int isfun(int n)

//{ //判断参数n是否是素数

// int i=n;

// if(i<2) return 0;

// for(i=n-1;i>=2;i--)

// {

// if(n%i==0) return 0;

// }

//}

//

//int main()

//{

// int n,m;

// cin >> n;

// for(;n;n--)

// {

// cin>> m ;

// if(isfun(m))

// cout<<"Yes"<<endl;

// else

// cout<<"No"<<endl;

// }

//

//

//

// return 0;

//}

//

#include <iostream>

#include <stdio.h>

using namespace std;

int isfun(int n)

{ //判断参数n是否是素数

int i=n;

if(i<2) return 0;

for(i=n-1;i>=2;i--)

{

if(n%i==0) return 0;

}

}

int main()

{

int T,m,nn;

cin >> T;

nn=T;

int \*p= new int [T];

for(;T;T--)

{

cin>>m;

\*p=m;

p++;

}

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

{

p--;

}

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

{

if(isfun(\*p))

cout<<"Yes"<<endl;

else

cout<<"No"<<endl;

//cout<< \*p << endl;

p++;

}

return 0;

}

### 循环-判断字符

#include <iostream>

using namespace std;

#include <stdio.h>

#include <stdlib.h>

void fun(char \*p)

{

int k=0,j=0,g=0;

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

{

if((\*p >= 'a' && \*p <= 'z' )|| (\*p >= 'A' && \*p <= 'Z'))

j++;

else if(\*p >= '0' && \*p <= '9')

k++;

else if(\*p == '?')

{ }

else

g++;

}

cout<<"Letters="<<j<<endl;

cout<<"Digits="<<k<<endl;

cout<<"Others="<<g<<endl;

}

int main()

{

char \*a=new char[10000];

if( cin.getline(a,100) )

fun(a);

return 0;

}

### 循环—输出金字塔图形

//#include <iostream>

//using namespace std;

//int main( )

//{

// int i,j,n,m;

// while(1)

// {

// cin >>n;

// for(i=1;i<=n;++i) //一共要输出n行

// {

// //输出第i行

// for(j=1; j<=n-i; ++j) //输出n-i个空格

// cout<<" ";

// for(j=1; j<=2\*i-1; ++j) //输出2\*i-1个星号

// cout<<"\*";

// cout<<endl;

// }

// }

//

// return 0;

//}

#include <iostream>

using namespace std;

int main( )

{

int i,j,n;

while(1)

{

cin >> n;

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

{

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

cout<<" ";

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

cout<<"\*";

cout<<endl;

}

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

{

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

cout<<" ";

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

cout<<"\*";

cout<<endl;

}

}

return 0;

}

### 逆序输出

#include<iostream>

using namespace std;

int main()

{

int n,m,o ;

cin >> o;

for(;o;o--)

{

cin >> n;

int \*p=new int[n];

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

{

cin >> \*p;

p++;

}

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

{

p--;

cout << \*p << " ";

}

cout << endl;

delete p;

}

return 0;

}

### 读入字符数组

#include<bits/stdc++.h>

using namespace std;

int main()

{

int T;

cin>>T;

int t=0;

int j=0;

string s;

getline(cin,s);

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

{

getline(cin,s);

j=0;

if(s[0]==' '||s[s.size()-1]==' ')

{

cout<<"NO\n";

break;

}

else if(s.size()%5!=4)

{

cout<<"NO\n";

break;

}

else

{

for(int k=0; k<s.size()-4; k=k+5)

if(s[k]=='o'&&s[k+1]=='i'&&s[k+2]=='e'&&s[k+3]=='r'&&s[k+4]==' ')

t=0;

else

{

t=1;

cout<<"NO\n";

break;

}

if(t==0)

{

for(int k=s.size()-4; k<=s.size()-1; k++)

if(s[k]=='o'&&s[k+1]=='i'&&s[k+2]=='e'&&s[k+3]=='r')

cout<<"YES\n";

}

}

}

return 0;

}

#include<iostream>

#include<string>

using namespace std;

int main()

{

int T;

cin>>T;

int t=0;

int j=0;

string s;

getline(cin,s);

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

{

getline(cin,s);

j=0;

if(s[0]==' '||s[s.size()-1]==' ') //首尾都没有空格

{

cout<<"NO"<<endl;

//break;

}

else if(s.size()%5!=4) //中间只有一个空格

{

cout<<"NO"<<endl;

//break;

}

else

{

for(int k=0; k<s.size()-4; k=k+5)

if(s[k]=='o'&&s[k+1]=='i'&&s[k+2]=='e'&&s[k+3]=='r'&&s[k+4]==' ')

{

//t=0;

cout<<"YES\n";

break;

}

else

{

t=1;

cout<<"NO\n";

break;

}

}

}

return 0;

}

### 统计输入的字符数\*\*\*

#include<bits/stdc++.h>

using namespace std;

int a[27];

int main()

{

string s;

int i=0;

while(getline(cin,s))

{ i=0;

while(s!=""&&i<s.size())

{

if(s[i]>='a'&&s[i]<='z')

a[s[i]-'a'+1]++; //巧妙的利用a[x]++，数组++，里面内容+1

i++;

}

}

for(int j=1;j<=26;j++)

if(a[j])

printf("%c:%d\n",j+'a'-1,a[j]);

return 0;

}

### 筛法求素数\*\*\*

#include<bits/stdc++.h>

using namespace std;

//T个询问，每次询问一个整数N，每次输出前N个数中素数的个数。T<=1亿 N<=1000000

int c[1000001]; //刚开始，所有数都标记为质数

int a[1000001]; //表示前N个数总共有多少个素数

int m=0;

int main()

{

//

for(int i=2;i<=1000000;i++)

{

if(c[i]==0) //if(!a[i])

for(int j=2\*i;j<=1000000;j=j+i)

c[j]=1;

}

for(int i=2;i<=1000000;i++)

{

if(c[i]==0)

m++;

a[i]=m;

}

int T;

cin>>T;

int n;

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

{

cin>>n;

cout<<a[n]<<"\n";

}

return 0;

}

### 对称矩形

#include<iostream>

using namespace std;

int main()

{

int n,i;

cin >> n;

int a=0;

for( i=3 ; n > a ; i=i+2)

{

a=(i/2+1)\*(i/2+1)+(i/2)\*(i/2);

}

cout << i-2 ;

return 0;

}

### 数组-出现次数最多的数

循环比较a[0]与a[1]、a[2]…a[n]如果有相等的数，则b[0]++；

#include<iostream>

using namespace std;

int main()

{

int n,a[20],i,j,flag=0,max;

int b[20]={0};

scanf("%d",&n);

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])

b[i]++;

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

if(b[i]!=0)

{

flag=1;

break;

}

if(flag==0)

{

printf("%d",a[0]);

return 1;

}

max=b[0];

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

if(b[i]>max)

max=b[i];

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

if(max==b[i])

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

return 0;

}

#include<iostream>

#include<stdio.h>

using namespace std;

int maxa(int a[],int n)

{

int k,maxk=0,max=a[0];//maxk记录最大出现次数，max记录最大出现次数的值

for(int i=0;i<n;i++)//遍历数组a中的元素，对当前元素a[i]，让它与它之前的每个元素相比较，计算出现的次数k

{

k=0;

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

{

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

{

k++;

if(k>maxk)

{

maxk=k;max=a[j];

}

}

}

}

return max;

}

int main()

{

int n;

cin >> n;

int a[20];

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

{

cin >> a[i];

}

int num=maxa(a, n);

cout << num << endl;

return 0;

}