

## 素数筛法

小度机器人最近正在添加功能，Robin希望小度机器人可以告诉他从1到N（ $1 < N \leq 10000$ ）有多少个素数。

输入包括一行，仅一个数字N。

输出在(1, N]区间内的所有的素数。

提示：想像将(1, N]的数字看成沙子和小石头，视非素数为沙子，视素数为小石头。将沙子筛走，剩下的就是小石头了。考虑到N最大也就是10000，你可以开一个长度为10000的数组，让数组元素的值作为筛去与否的标志，比如筛去以后让元素值为1，然后依次输出就可以了。当然，如果你有更好的办法，也可以试试哦！

样例1

输入：

8

输出：

2

3

5

7

```
#include<stdio.h>
```

```
#include<math.h>
```

```
int main()
```

```
{
```

```
    int i,j,n;
```

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

```
    for (j = 2; j <= n; j++)
```

```
    {
```

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

//判断是否为一个素数

```
        {
```

```
            if(j % i == 0)
```

```
            {
```

```
                break;
```

```
            }
```

//剔除合数，当该数为合数时，不作任何处理。用数i去判

断，若均为素数，则2到根号j都除不了j，于是i就大于根号j

```
        }
```

```
        if (i > sqrt(j))
```

```
        {
```

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

```
        }
```

//当j为素数，，用于判断的i，就一定大于sqrt(j)，利用该

特性把素数j筛选出来

```
    }
```

```
}
```

-----纯粹按“筛选法”原理实现-----

```
#include<stdio.h>
void main()
{
    int i;
    int N,count,p=0;
    int r[1001];//限制数据量大小为1000,因为下面数组是从1开始的，所以这里写为1001。不哟啊0。

    printf("你想求多少以内的素数:");

    scanf("%d",&N);
    for(i=1;i<=N;i++)//为方便计，从1起
        r[i]=1;
    count=2;//筛选起点为2
    while(count<=N/2) //显然：count不会超过N/2,必能使留下的数全为素数。
    {
        for(i=count+1;i<=N;i++)
        {
            if(r[i]==1&&i%count==0)
                r[i]=0;
        }
        for(i=count+1;i<=N;i++)
        {
            if(r[i]==1)
            {
                count=i;
                break;
            }
        }
    }

    printf("%d以内的素数为:\n",N);
    for(i=2;i<=N;i++)
        if(r[i]==1)
        {
            p++;

            printf("%d ",i);
            if(p%10==0) //增设p为输出换行
                printf("\n");
        }
        printf("\n");
    }
    -----纯粹按“筛选法”原理实现-----
```

```

#include<stdio>

int main(int argc, char const *argv[])
{
    //if it is prime ,
    it's 0
    bool prime[1000];
    for (int i = 0; i < 1000; ++i)
    {
        prime[i]=0;
    }
    prime[0]=1;
    prime[1]=1;
    for(int i=2;i<1000;i++)
    {
        if(prime[i]==0)
        {
            for(int j=i*2;j<1000;j+=i)
            prime[j]=1;
        }
    }
    for (int i = 0; i < 1000; ++i)
    {
        if(prime[i]==0)printf("%d ",i);
    }
    return 0;
}

```

此筛选法遵循了C程序模块化的习惯，将筛选法独立为一个函数在[主函数](#)里调用，此代码在VC6.0中完全可以直接使用。

```

#include"stdio.h"
#include"string.h"
int a[10000]; /
/定义一个容器
int n,i,j,k,count=0,temp,cs=1;
int SXF(int n) /
/筛选法
. n为

```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52

```
范围
{
for(i=0;
i<n;i++
)//在定
义的范
围内循
环
{
if(a[i]<
=1) //判
断当前
数组存
放数值
是否大
于1
contin
ue; //为
真，结
束本次
循环
temp=
2*a[i]; /
/为假，
temp
赋值为
当前数
值的两
倍
while(t
emp<n
) //tem
p不能
超过范
围
{
a[temp
]=1; //
将通过
的数值
的倍数
全部赋
值为1
temp=t
emp+a
[i];
}
}
return
0;
}
void m
ain()
{
for(j=0;
j<1000
0;j++) //
容器赋
值
{
a[j]=j;
}

printf(
"请输
入你要
查找素
数的范
围（1~
10000
）：");

scanf(
"%d",&
n);
SXF(n)
;
printf(
"范围
内的素
数：\n
");
for(k=0
;k<n;k+
```

```
+)
{
if(a[k]<
=1) //因为合数
已经全部赋值为1,
所以通过都是
素数
continue;

printf(
"%d ",a
[k]);
count+
+;

while(
count>
=cs) //以下可
不加,
为
杨辉三
角格式
排列
{

printf(
"\n");
count=
0;
cs++;
}
}

printf(
"\n");
}
```

```

#include<stdio.h>
int main(int argc, _TCHAR* argv[])
{
    int num=100;
    int a[100];
    for(int i=0; i<num; i++)
    {
        a[i]=i+2;
    }

    for(int i=1; i<num-1; i++) //i+1作为除数
    {
        for(int j=i+1; j<num; j++) //a[j]作为被除数
        {
            if(a[j]!=0 && a[j]%(i+1)!=0)
            {
                a[j]=0;
                //非素数置零
            }
        }
    }

    //十个一组输出
    for(int i=1, n=0; i<num; i++)
    {
        if(a[i]!=0)
        {
            printf("%d\t", a[i]);
            if(++n%10==0)
            {
                printf("\n");
            }
        }
    }
    printf("\n");
    return 0;
}

```

