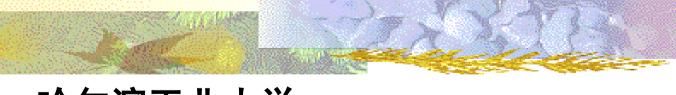


规格严格 功夫到家



# 第7章 数组

## ——数组的其他应用之筛法求素数



哈尔滨工业大学

苏小红 sxh@hit.edu.cn

#### 数组的其他应用

- 除了一维、二维数据表、矩阵运算等,还有其他应用吗?
- 问题:如何判断一个整数x是否是素数(Prime Number)?
  - \* 不能被1和x以外的其他数整除的正整数
  - \* 试商法,用2~x-1之间的整数去试商,看x能否其被整除
  - \* 用2~sqrt(x)之间的整数去试商,看x能否被其整除
  - \* x = a \* b
  - \* 若a>=sqrt(x), 则b<=sqrt(x)

如何求100 以内的所有 素数?

```
int IsPrime(int x)
{
   int i, squareRoot;
   if (x <= 1) return 0;
   squareRoot = (int)sqrt(x);
   for (i=2; i<=squareRoot; i++)
   {
      if (x%i == 0) return 0;
   }
   return 1;
}</pre>
```



```
#include <stdio.h>
#include <math.h>
int IsPrime(int x);
int main()
  int i;
  for (i=1; i<=100; i++)
        if (IsPrime(i)) printf("%d\t", i);
  printf("\n");
                                         int IsPrime(int x)
  return 0;
                                            int i, squareRoot;
                                            if (x <= 1) return 0;
                                            squareRoot = (int)sqrt(x);
                                            for (i=2; i<=squareRoot; i++)</pre>
                                                if (x%i == 0) return 0;
```

return 1;

int a[N+1]; (N值为100)

初始化: 令a[2]=2,a[3]=3,...,a[N]=N

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
筛2的	筛 <mark>2</mark> 的倍数														
2	3	0	5	0	7	0	9	0	11	0	13	0	15	0	17
筛3的	筛3的倍数														
2	3	0	5	0	7	0	0	0	11	0	13	0	0	0	17
筛5的	筛5的倍数														
2	3	0	5	0	7	0	0	0	11	0	13	0	0	0	17

依次从a中筛掉2的倍数,3的倍数,5的倍数,.....,sqrt(N)的倍数;

既筛掉所有素数的倍数,直到a中仅剩下素数为止(剩下的不为0的数不是任何数的倍数)

- 自顶向下、逐步求精设计算法
- step 1: 设计总体算法
  - \* 初始化数组a, 使a[2]=2, a[3]=3,..., a[N]=N
  - \* 对i=2,3,....,sqrt(N)分别做: "筛掉a中所有a[i]的倍数"
  - \* 输出数组中余下的a[i]!=0的数
- step 2: 对 "筛掉a中所有的a[i]的倍数" 求精
  - \* 对数组a中a[i]后面的所有数a[j]分别做:
  - \* 若 "a[j]是a[i]的倍数",则 "筛掉a[j]"

■ 自顶向下、逐步求精设计算法 ■ step 2: 对 "筛掉a中所有的a[i]的倍数" 求精 \* 对数组a中a[i]后面的所有数a[i]分别做: \* 若 "a[j]是a[i]的倍数",则 "筛掉a[j]" ■ step 3: 对 "若a[j]是a[i]的倍数。则筛掉a[j]" 求精 if (a[i]!=0 && a[j]!=0 && a[j]%a[i]==0)a[j] = 0;

```
void SiftPrime(int a[], int n)
  int i, j;
  for (i=2; i<=n; i++)
      a[i] = i;
  for (i=2; i<=sqrt(n); i++)
    for (j=i+1; j<=n; j++)
        if (a[i]!=0 && a[j]!=0 && a[j]%a[i]==0)
          a[i] = 0;
```

```
void PrintPrime(int a[], int n)
{
   int i;
   for (i=2; i<=n; i++)
   {
      if (a[i] != 0)
      {
        printf("%d\t", a[i]);
      }
      printf("\n");
}</pre>
```

```
#include<stdio.h>
#include<math.h>
#define N 100
void PrintPrime(int a[], int n);
void SiftPrime(int a[], int n);
int main()
{
   int a[N+1];
   SiftPrime(a, N);
   PrintPrime(a, N);
   return 0;
}
```

## 讨论

- 鲁智深吃馒头。
- 据说,鲁智深一天中午匆匆来到开封府大相国寺,想蹭顿饭吃,当时大相国寺 有99个和尚,只做了99个馒头
- 智清长老不愿得罪鲁智深,便把他安排在一个特定位置,之后对所有人说:
- 从我开始报数(围成一圈),第5个人可以吃到馒头(并退下)
- 按此方法,所有和尚都吃到了馒头,唯独鲁智深没有吃上
- 请问他在那个位置?
- 能否借鉴筛法求出剩下的最后一个人的位置?





