

# 数组1

泰州市第二中学附属初中 谢志锋

# 一、为什么要使用数组

通过前面的学习，我们可以很轻松地编程完成这样一个任务：输入3名同学的身高并逆序输出。如果将这个任务中同学的人数改为50，仅运用以前所学知识编写程序，那需要定义50个简单变量h0, h1, ..., h49保存每个同学的身高。如果读入50个同学的身高，就需要如下语句：

```
cin>>h0>>h1>>.....>>h49;
```

这样编程就不能用上我们所学的循环语句，如果再加上其它数据处理，程序会极其冗长，也就不能体现出编程的优势。

## 二、一维数组的定义

定义一维数组的一般格式为：

类型标识符 数组名[常量表达式]; //常量表达式的值即为一维数组元素的个数。

例如：int h[50]; //表示数组h有50个元素，它们的编号从0开始，为0~49，每个元素都是int型。

也可以用下面的方法定义：

①int h[5\*10];

②#define N 50

int h[N];

③int N=50;

int h[N];

## 二、一维数组的定义

也就是说定义一个数组必须要指出：存储在每个元素中的值的类型、数组名、数组中的元素个数。数组的类型不但可以是int型,还可以是float型、double型、char型等基本数据类型,也可以是我们自定义的数据类型,一旦声明了该数组的数据类型,那么该数组中的所有元素都是声明时的数据类型。

和定义变量一样,我们也可以一次定义多个同一个类型的数组,

如int h[50],g[100];

### 三、一维数组的引用

定义了数组后，就可以引用数组中的任意一个元素，引用格式为：

数组名[下标]

其中下标只能为整型常量或整型表达式，下标用一对中括号括起来，下标值应在数组定义的下标值范围内，如h[5]、h[5+1]。能逐个引用数组元素的值而不能一次引用整个数组中的全部元素的值。

```
int a[10], b[10];
```

```
.....
```

```
a=b;//不能引用整个数组
```

问题1、已知一维数组中A中，每个元素在存储时要占8个字节，共有100个元素，分配内存时是从地址SA开始连续分配的，请问第10个元素的存储地址是多少？

问题2、对定义“`int a[2];`”的正确描述是（ ）。

- A、定义一维数组a，包含a[1]和a[2]两个元素
- B、定义一维数组a，包含a[0]和a[1]两个元素
- C、定义一维数组a，包含a[0]、a[1]和a[2]三个元素
- D、定义一维数组a，包含a(0)、a(1)和a(2)三个元素



例3、引用数组元素时，其数组下标的数据类型允许是（ ）。

A. 整型常量

B. 整型表达式

C. 整型常量或整型表达式

D. 任何类型的表达式

## 例1、走楼梯(stairs, 1S, 256MB)

### 【问题描述】

已知一个楼梯有n级，从下往上走，一步可以走一级，也可以走两级，走到第N级楼梯有多少种走法？

### 【输入格式】

一行一个整数n。

### 【输出格式】

一行n个用空格隔开的整数，表示走到第1级、第2级、.....第n级分别有多少种走法。

### 【输入样例】

2

### 【输出样例】

1 2

### 【数据规模】

对100%的数据满足： $0 < n \leq 30$ 。



例2、幸运的划分(lucky, 1S, 256MB) (来源: codeforces)

【问题描述】

判断一个数 $n$ 是否能被一个幸运数整除, 一个幸运数是一个只包含4或7的数, 如7、47、477等是幸运数, 17、42则不是幸运数。

【输入格式】

一行一个正整数 $n$ 。

【输出格式】

一行,如果能被幸运数整除输出“YES” 否则输出“NO”。

【输入样例】 47

【输出样例】 YES

【数据规模】

对100%的数据满足:  $1 \leq n \leq 1000$ 。

### 例3、陶陶摘苹果(apple, 1S, 256MB)

#### 【问题描述】

陶陶家的院子里有一棵苹果树，每到秋天树上就会结出10个苹果。苹果成熟的时候，陶陶就会跑去摘苹果。陶陶有个30厘米高的板凳，当她不能直接用手摘到苹果的时候，就会踩到板凳上再试试。

现在已知10个苹果到地面的高度，以及陶陶把手伸直的时候能够达到的最大高度，请帮陶陶算一下她能够摘到的苹果的数目。假设她碰到苹果，苹果就会掉下来。

#### 【输入格式】

包括两行数据。

第一行包含10个100到200之间（包括100和200）的整数（以厘米为单位）分别表示10个苹果到地面的高度，两个相邻的整数之间用一个空格隔开。

第二行只包括一个100到120之间（包含100和120）的整数（以厘米为单位），表示陶陶把手伸直的时候能够达到的最大高度。

**【输出格式】**

一行，这一行只包含一个整数，表示陶陶能够摘到的苹果的数目。

**【输入样例】**

100 200 150 140 129 134 167 198 200 111  
110

**【输出样例】**

5

#### 例4、插队(jumpque, 1S, 256MB)

##### 【问题描述】

有 $n$ 个人（每个人有一个唯一的编号，用1到 $n$ 之间的整数表示）在一个水龙头前排队准备接水，现在第 $n$ 个人有特殊情况，经过协商，大家允许他插队到第 $x$ 个位置。求第 $n$ 个人插队后的排队情况。

##### 【输入格式】

包括三行数据。

第一行包含1个正整数 $n$ ，表示有 $n$ 个人。

第二行包含 $n$ 个用空格隔开的正整数，表示排在队伍中的第1个到第 $n$ 个人的编号。

第三行包含1个正整数 $x$ ，表示第 $n$ 个人插队的位置。

##### 【输出格式】

一行，包含 $n$ 个用空格隔开的正整数，表示第 $n$ 个人插队后的排队情况。

### 【输入样例】

7

7 2 3 4 5 6 1

3

### 【输出样例】

7 2 1 3 4 5 6

### 【数据规模】

对于100%的数据满足： $2 < n \leq 100$ ， $1 \leq x < n$ 。



## 例5、队伍调整(removenum, 1S, 256MB)

### 【问题描述】

有 $n$ 个人（每个人有一个唯一的编号，用1到 $n$ 之间的整数表示）在一个水龙头前排队准备接水，现在第 $x$ 个人有特殊情况离开了队伍，求第 $x$ 个人离开队伍后的排队情况。

### 【输入格式】

包括三行数据。

第一行包含1个正整数 $n$ ，表示有 $n$ 个人。

第二行包含 $n$ 个用空格隔开的正整数，表示排在队伍中的第1个到第 $n$ 个人的编号。

第三行包含1个正整数 $x$ ，表示第 $x$ 个人离开队伍。

### 【输出格式】

一行，包含 $n$ 个用空格隔开的正整数，表示第 $x$ 个人离开队伍后的排队情况。



### 【输入样例】

7

7 2 3 4 5 6 1

3

### 【输出样例】

7 2 4 5 6 1

### 【数据规模】

对于100%的数据满足： $2 < n \leq 100$ ， $1 \leq x \leq n$ 。

例6、校门外的树(tree, 1S, 256MB) (来源: noip2005普及组)

【问题描述】

某校大门外长度为 $L$ 的马路上有一排树，每两棵相邻的树之间的间隔都是1米。我们可以把马路看成一个数轴，马路的一端在数轴0的位置，另一端在 $L$ 的位置；数轴上的每个整数点，即0, 1, 2, ...,  $L$ ，都种有一棵树。

由于马路上有一些区域要用来建地铁。这些区域用它们在数轴上的起始点和终止点表示。已知任一区域的起始点和终止点的坐标都是整数，区域之间可能有重合的部分。现在要把这些区域中的树（包括区域端点处的两棵树）移走。你的任务是计算将这些树都移走后，马路上还有多少棵树。

### 【输入格式】

第一行有两个整数L ( $1 \leq L \leq 10000$ ) 和 M ( $1 \leq M \leq 100$ )，L代表马路的长度，M代表区域的数目，L和M之间用一个空格隔开。接下来的M行每行包含两个不同的整数，用一个空格隔开，表示一个区域的起始点和终止点的坐标。

### 【输出格式】

包括一行，这一行只包含一个整数，表示马路上剩余的树的数目。

### 【输入样例】

```
500 3
150 300
100 200
470 471
```

### 【输出样例】

```
298
```

### 【数据规模】

对于10%的数据，区域之间没有重合的部分；  
对于其它的数据，区域之间有重合的情况。

## 例7、超过平均身高同学的人数(num, 1S, 256MB)

### 【问题描述】

给定n个同学的身高（均为100到200之间的正整数），求超过平均身高的同学人数。

### 【输入格式】

包括两行数据。

第一行包含一个整数n。

第二行包含n个用空格隔开的正整数。

### 【输出格式】

一行，这一行只包含一个整数，表示超过平均身高的同学人数。。

### 【输入样例】

6

160 155 170 175 172 164

### 【输出样例】

3

## 例8、旗手(flagman, 1S, 256MB)

### 【问题描述】

导游往往喜欢从所带的旅游团中选一个身高最高的游客站在前面帮着拿旅行社的旗帜，现在给定n个游客的身高（均为正整数），将身高最高的游客和第一个游客调换位置，再依次输出他们的身高。

### 【输入格式】

包括两行数据。

第一行包含一个整数n，表示有n个游客。

第二行包含n个用空格隔开的正整数，表示n个游客的身高。

### 【输出格式】

一行，包含n个用空格隔开的正整数，表示调换位置后各个位置上游客的身高。



## 【输入样例】

6

160 155 170 175 172 164

## 【输出样例】

175 155 170 160 172 164