

1：一维数组

在C语言中使用数组必须先进行定义。一维数组的定义方式为：

类型说明符 数组名 [常量表达式];

类型说明符是任一种基本数据类型或构造数据类型。数组名是用户定义的数组标识符。方括号中的常量表达式表示数据元素的个数，也称为数组的长度。

对于数组说明应注意以下几点：

1：不能在方括号中用变量来表示元素的个数。

例如：

```
1. int n=5;  
2. int a[n];
```

解释：因为C语言的数组是真实数组，申明几个数据元素，就分配多少磁盘空间，n可变的话，系统就不知道该分配多少空间，又因为C语言的申明和定义可以分开（不分开的话就没必要用变量了），n的数值会变化，所以会导致申明和定义的数组大小不同。

一维数组的初始化：

给数组赋值的方法除了用赋值语句对数组元素逐个赋值外，还可采用初始化赋值和动态赋值的方法。

数组初始化赋值是指在数组定义时给数组元素赋予初值。数组初始化是在编译阶段进行的。这样将减少运行时间，提高效率。初始化赋值的一般形式为：

类型说明符 数组名[常量表达式] = { 值, 值……值 };

其中在{ }中的各数据值即为各元素的初值，各值之间用逗号间隔。例如：

```
1. int a[10]={ 0,1,2,3,4,5,6,7,8,9};
```

相当于

```
a[0]=0; a[1]=1 ... a[9]=9;
```

C语言对数组的初始化赋值还有以下几点规定：

1) 可以只给部分元素赋初值。当{ }中值的个数少于元素个数时，只给前面部分元素赋值。例如：

```
1. int a[10]={0,1,2,3,4};
```

表示只给a[0]~a[4] 5个元素赋值，而后5个元素自动赋0值。

2) 只能给元素逐个赋值，不能给数组整体赋值。例如给十个元素全部赋1值，只能写为：

```
1. int a[10]={1,1,1,1,1,1,1,1,1,1};
```

而不能写为：

```
1. int a[10]=1;
```

3) 如给全部元素赋值，则在数组说明中，可以不给出数组元素的个数。例如：

```
1. int a[5]={1,2,3,4,5};
```

可写为：

```
1. int a[]={1,2,3,4,5};
```

动态赋值是执行时可以键盘输入再赋值给数组元素。