
嵌入式开发

数组(一)

创客学院 小美老师

▶ 1 / 掌握一维数组的使用

2 / 总结与思考

数组概述

数组

- 构造数据类型之一
- 数组是具有一定顺序关系的若干个变量的集合，组成数组的各个变量称为数组的元素
- 数组中各元素的数据类型要求相同,用数组名和下标确定。数组可以是一维的，也可以是多维的。

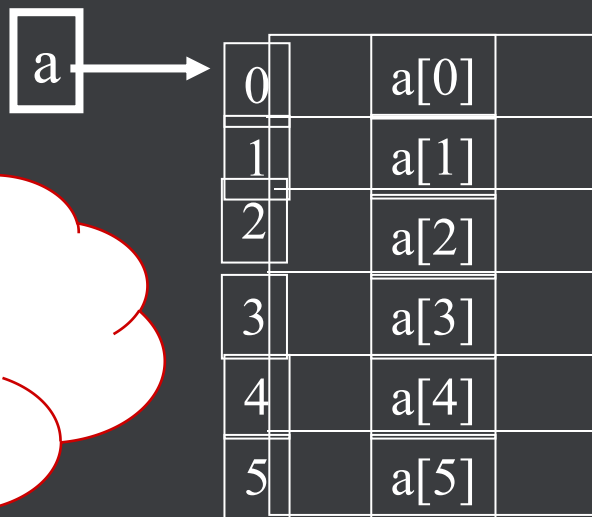
一维数组

一维数组的定义

- 所谓一维数组是指只有一个下标的数组。它在计算机的内存中是连续存储的。
- C语言中，一维数组的说明一般形式如下：
 <存储类型> <数据类型> <数组名>[<表达式>] ;

一维数组

例: `int a[6];`



数组名表示内存首地址，是地址常量
`sizeof(数组名)` 是数组占用的总内存空间

编译时分配连续内存
内存字节数=数组维数*
`sizeof(元素数据类型)`

一维数组

注意事项

- C语言对数组不作越界检查，使用时要注意

```
int a[5]; a[5] = 10
```

- 关于用变量定义数组维数

```
int i = 15; int a[i]
```

一维数组的引用

- 数组必须先定义，后使用
- 只能逐个引用数组元素，不能一次引用整个数组
- 数组元素表示形式：数组名[下标]
 - 其中：下标可以是常量或整型表达式

```
例 int a[10];  
    printf( "%d" , a); (×)  
  
    for(j=0;j<10;j++)  
        printf( "%d\t" , a[j]); (✓)
```

一维数组的初始化

- 初始化方式：在定义数组时，为数组元素赋初值
 - `int a[5]={1,2,3,4,5};`
- 说明
 - 数组不初始化，其元素值为随机数
 - 对static数组元素不赋初值，系统会自动赋以0值
 - 只给部分数组元素赋初值

一维数组的初始化

```
static int a[5];
```

等价于： a[0]=0; a[1]=0; a[2]=0; a[3]=0; a[4]=0;

```
int a[5]={6,2,3};
```

等价于： a[0]=6; a[1]=2; a[2]=3; a[3]=0; a[4]=0;

```
int a[3]={6,2,3,5,1};    (×)
```

```
int a[]={1,2,3,4,5,6}; //编译系统根据初值个数确定  
数组维数
```

程序举例

- 冒泡排序

- 它重复地走访过要排序的数列，一次比较两个元素，如果顺序错误就交换。走访数列的工作是重复地进行直到没有再需要交换，也就是说该数列已经排序完成。



程序举例

例：

| | | | | | |
|-------|-----|-----|-----|-----|-----|
| 38 | 38 | 38 | 13 | 13 | 13 |
| 49 | 49 | 13 | 27 | 27 | 27 |
| 76 | 13 | 27 | 30 | 30 | 30 |
| 13 | 27 | 30 | 38 | 38 | |
| 27 | 30 | 49 | 49 | | |
| 30 | 76 | 76 | | | |
| 97 | 97 | | | | |
| 初始关键字 | 第一趟 | 第二趟 | 第三趟 | 第四趟 | 第五趟 |

总结与思考

主要介绍了一维数组的概念、引用和初始化。演示了有关一维数组的程序的编写。

思考

- 如何改进冒泡排序的效率？

扫一扫，获取更多信息



THANK YOU