

C语言程序设计(第5章数组) PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/134/2021_2022_C_E8_AF_AD_E8_A8_80_E7_A8_8B_c97_134835.htm

数组是一个由若干同类型变量组成的集合，引用这些变量时可用同一名字。数组均由连续的存储单元组成，最低地址对应于数组的第一个元素，最高地址对应于最后一个元素，数组可以是一维的，也可以是多维的。

5.1 一维数组

一维数组的一般说明形式如下：

`type-specifier var_name [size].` 在C语言中，数组必须显示地说明，以便编译程序为它们分配内存空间。在上式中，类型说明符指明数组的类型，也就是数组中每一个元素个数，一维数组的总字节数可按下式计算： $\text{sizeof(类型)} \times \text{数组长度} = \text{总字节数}$

[例5-1] 将数字0到9装入一个整型数组。

```
main() { int x[10]. /* 定义包含10个整型数的数组，引用为x[0], x[1].. x[9] */ int t. for (t=0. t)
```

C语言并不检验数组边界，因此，数组的两端都有可能越界而使其它变量的数组甚至程序代码被破坏。在需要的时候，数组的边界检验便是程序员的职责。

例如，当使用`gets()`接收字符输入时，必须确认字符数组的长度足以存放最长的字符串。一维数组在本质上是由同类数据构成的表，例如，对下列数组`a: char a[7]`图5-1说明了数组a在内存中的情形，假定起始地址为1000。

5.1.1 向函数传递一维数组

将一维数组传递给函数时，把数组名作为参数直接调用函数即可，无需任何下标。这样，数组的第一个元素的地址将传递给该函数。C语言并不是将整个数组作为实参来传递，而是用指针来代替它。例如，下面的程序将数组i的第一个元素的地址传递给函数`func1()`。

```
main() { int i[10]. func1(i).
```

/*函数调用，实参是数组名*/...} 函数若要接收一维数组的传递，则可以用下面的二种方法之一来说明形式参数；1) 有界数组；2) 无界数组。例如，函数func1()要接收数组i可如下说明: func1(str) char str[10]. /* 有界数组，数组的下标只能小于或等于传递数组的大小。*/ {...} 也可说明为: func1(str) char str[]. /* 无界数组*/ {...} 这二种说明方法的效果是等价的，它们都通知编译程序建立一个字符指针。第一种说明使用的是标准的数组说明；后一种说明使用了改进型的数组说明，它只是说明函数将要接收一个具有一定长度的整型数组。细想就会发现，就函数而言，数组究竟有多长并无要紧，因为C语言并不进行数组的边界检验。事实上，就编译程序而言，下面的说明也是可行的。 func1 (str). int str[32]. {...} 因为编译程序只是产生代码使函数func1()接收一个指针，并非真正产生一个包含32个元素的数组。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com