

C语言中数组和指针的互操作 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_E4_B8_AD_c97_134914.htm C是一种怀旧的语言，因为它的历史很久远，然而自从各种面向对象的编程语言的相续出现让它的影响力日减。当然了，这是无可非议的，但是C的高效性是其他语言无妨比拟的，所以我们有必要把握其中的精华与奥妙，也就有必要知道其中的基本的数据结构的比如数组，稍微有点深度的堆栈、列表、结构体等的操作和实现。指针也是C语言中的一个很优秀灵活的结构，对它的了解也是必不可少的。我们一般都认为数组是一维的数据存储结构，因为二维数组或者说矩阵都可以看作是多个一维数组的组合结构，定义在其上的数据存储访问方式是一样的。所以一维数组是其中最基础的最重要的部分，只有理解了此类数据结构的本质才能触类旁通了。数组（array）是若干同类变量的聚合，允许通过统一的名字引用其中的变量。所以数组也就是一个同一类型的数据的有限集合。可以通过下表来访问数组中的某一/些数组元素。在C语言中数组都由连续的内存区域构成（有时候，不一定是这样），最低地址对应首元素，数组的下标是从0开始的，所以首元素也就是数组下标为0的元素，最高的地址对应最末的元素，即第N-1个元素（如果我们定义的数组为N元）。数组的定义方式：在C语言中允许在声明数组的时候同时对其进行初始化，也可以把声明和定义放在不同的位置，初始化的一般的类似于如下的表达式：`type_specifier array_name[size1]...[sizeN] = {value_list}`. 其中vlaue_list是由逗号(，)分隔的常量表，常量表

必须和type_specifier兼容。最后由分号与下一个语句分隔。由此可见一维数组的定义方式为：type_specifier array_name[size] = {value_list}。如下：char hello[12] = { ' H ' , ' e ' , ' l ' , ' l ' , ' o ' , ' ' , ' ' , ' ' , ' w ' , ' o ' , ' r ' , ' l ' , ' d ' , ' ' }。注意：字符数组是一" ' ' "收尾的，这是C标准的一部分。因为在操作字符数组的时候是以 ' ' 作为结束判断的标志。当然了，如果你定义的是一个字符串那就不用加这个 ' ' 了。因为有机制帮助你自动添加。上面的例子的串的生命方式为：string hello = "Hello, world"。(当然了，具体的实现中你必须把"string.h"头文件加入到你的文件中)，或者你也可以这样来声明：char *hello = "Hello, world"；或者char hello[] = "Hello, world"；切换为字符指针数组，其效果是一样的)。数槌踟蓟氛焙蚘箍?nbsp.以不标明最大小，即 char hello[] = { ' H ' , ' e ' , ' l ' , ' l ' , ' o ' , ' ' , ' ' , ' ' , ' w ' , ' o ' , ' r ' , ' l ' , ' d ' , ' ' }。这时候编译器会根据后边的赋值情况为数组分配合适的内存空间，这个你不用担忧，除非机器正处于内存缺状态。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