

c_c 深入探讨数组内存模型 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_c___E6_B7_B1_E5_85_c97_134649.htm 对于一个好学的人来书，怎么做不重要，关键是为什么要这样做，只有深入到这一步了，才敢说‘阿，我懂了’，于是，以后碰到类似的问题，就可以从我们知道的原理出发，悠然自得而又满怀信心地推导出正确的结果。然而，好学的人往往注定了困惑于一时的不解，并且，付出过多的时间去试图解决这种迷惑不解，其中的辛苦是可想而知的，因为，我就是这样走过来的。所以，一直都想写一个关于指针，数组，地址，左右值等等这些在语言的学习里面最让人困惑不解的问题。怎奈时间有限，一直没有能完成这样的想法，今天一样是太忙，可能不能完整的把我想说的写出来了，那么，就走个捷径，抛开c/C的语义，从系统底层的原理来就事论事的解决一下网友的问题，本人水平不高，全当抛砖引玉。[注意，下面的讲解全部以32位系统为例，也就是说，一个int占用4 bytes]的。比如：

```
void main(){int a=0.int array[10].a=3.//okarray[0]=4.//okarray={1,2,4}.//errorprintf("%d,%d,%d",a,array[0],array).//okreturn.}
```

对于上面的程序段，a代表了系统中一个4byte的内存区域，编译时候用a来代表内存的值[也就是所谓右值]，array来代表一段（4*10）byte内存区域的值，而具体到array这个变量的身上，编译器解释他的时候，把它看成是这段内存的首地址。所以上面的程序段中的那个printf打印出来的因该是a的值，array第一个元素的值,array的首地址。基于以上的一些事实和理论，int *p=array.int

`**p1=amp.array`//一个指向了‘指向数组的指针’得指针。语义上的不同，对于编译器来解释这个变量的行为的时候非常重要，但是对于系统的内部来说，基本的内存的结构一样的，我们如果能抓住这点的话，那么，指针对我来说将是透明的。下面举个例子：`int array[10].int array2d[3][10]`.这两个数组的区别和联系分别是什么，如果你能很清楚地认识到那么，你已经有很好的功力了。首先，不同点，最表面的语义上的不同就是一个是1维，一个是二维，`for(int i=0;i{for(int j=0.j{array2d[j][i]=array[i].}}`上面的程序将有3行数据的array2d数组每行都设成和array相同的值，从这一层来看，我们更能清晰地感觉到行和列的存在[也就是意识到了维的存在。但是在内存内部的实现又是什么样的情况呢[注意，现在来说相同点啦. 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