

C语言程序设计(第6章指针)5 PDF转换可能丢失图片或格式，  
建议阅读原文

[https://www.100test.com/kao\\_ti2020/135/2021\\_2022\\_C\\_E8\\_AF\\_AD\\_E8\\_A8\\_80\\_E7\\_A8\\_8B\\_c97\\_135148.htm](https://www.100test.com/kao_ti2020/135/2021_2022_C_E8_AF_AD_E8_A8_80_E7_A8_8B_c97_135148.htm)

## 6.5 指针的地址分配

我们可以定义指针变量指向任何类型的变量。在上述的处理过程中，指针变量指向的变量通过传递变量的地址来实现。指针变量的取值是内存的地址，这个地址应当是安全的，不可以是随意的，否则，写入内存单元的值将会使得已存放的数据或程序丢失。应使用编译系统提供的标准函数来实现地址分配。ANSI标准建议设置了两个最常用的动态分配内存的函数malloc()和free()，并包含在stdlib.h中，但有些C编译却使用malloc.h包含。使用时请参照具体的C编译版本。我们这里所指的动态内存分配其含义是指：当定义指针变量时，其变量的取值是随机的，可能指向内存的任一单元。若指针的指向是不安全的内存地址，在该地址空间上的数据交换就会产生意料不到的效果。为此，在程序的执行过程中，要保证指针操作的安全性，就要为指针变量分配安全地址。在程序执行时为指针变量所做的地址分配就称之为动态内存分配。当无需指针变量操作时，可以将其所分配的内存归还系统，此过程我们称之为内存单元的释放。malloc()用以向编译系统申请分配内存；free()用以在使用完毕释放掉所占内存。[例6-21] 两个字符串的交换。

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
main()
{
    char *ptr1,*ptr2,*temp;
    ptr1=malloc(30). /*动态为指针变量分配
    长度为30字节的存储空间*/
    ptr2 = malloc(20).
    temp =
    malloc(30).
    printf("input str1:"). gets(ptr1). /*输入字符串*/
    printf("input str2:"). gets(ptr2).
    printf("str1-----str2\n").
```

```
printf("%s.....%s\n", ptr1, ptr2). strcpy(temp,ptr1). /* 串复制*/  
strcpy(ptr1, ptr2). strcpy(ptr2,temp). printf("str1-----str2\n").  
printf("%s.....%s\n",ptr1,ptr2). free(ptr1). free(ptr2).} 为指针变量
```

分配的存储空间长度取决于存放字符的多少。在上述的程序中，两个串的交流可以通过标准函数strcpy() 来完成，也可以通过串指针交换指向完成，用temp=ptr1； ptr1 = ptr2； ptr2 = temp；三条赋值语句实现。但是，利用指针交换指向，其物理意义与串通过函数进行的复制完全不同。前者是存放串地址的指针变量数据交换，后者是串在内存物理空间的数据交换。指针变量用完后，将指针变量所占的存储空间释放。运行程序：

```
runinput str1: Chinainput str2:  
Chengdustr1-----str2China-----Chengdustr1-----  
-str2Chengdu----- China 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
```