# 第六章 指针变量的基础知识

本章学习目标：

* 掌握指针类型定义
* 掌握一重指针的声明和使用规则
* 掌握使用内存示意图的方式分析程序
* 掌握多重指针的声明和使用
* 理解const指针
* 了解空指针以及通用指针
* 了解指针变量的运算。

## 6.1 实践题

**一、认识指针变量和指针数据类型**

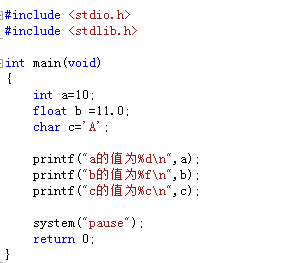
**实验目的**

1. 理解指针数据类型。
2. 掌握指针变量与普通变量的关系。

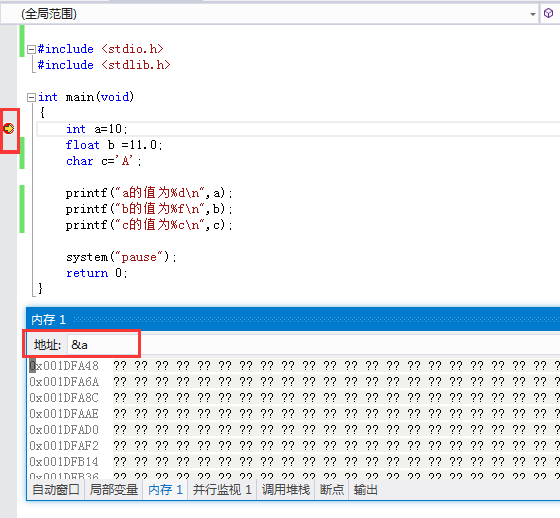
**实验步骤**

步骤1：定义int型的变量a，float型变量b，char型变量c，观察普通变量所占内存情况。

1.原始代码

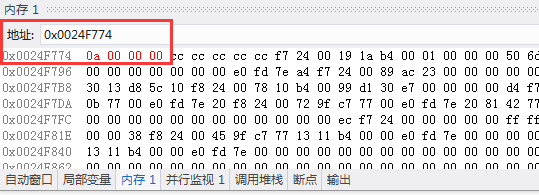


2.加断点，在内存窗口查看各变量的存储

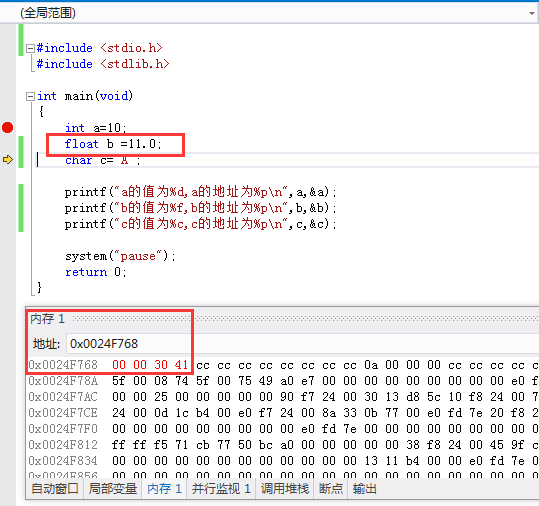


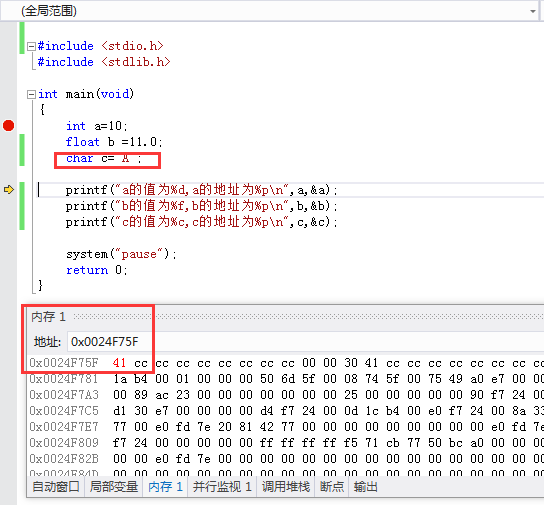
单击单步执行，查看到整型变量a的地址为0x0024F774，从此处开始的4个字节存放变量的值10，且采用“低位在低字节”的形式存储。



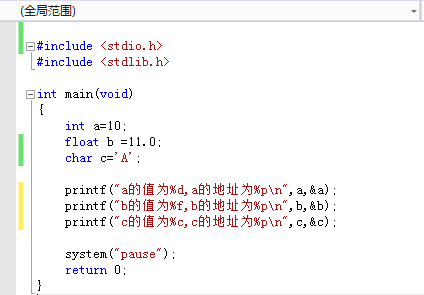


同理查看变量b和c的内存存储情况。

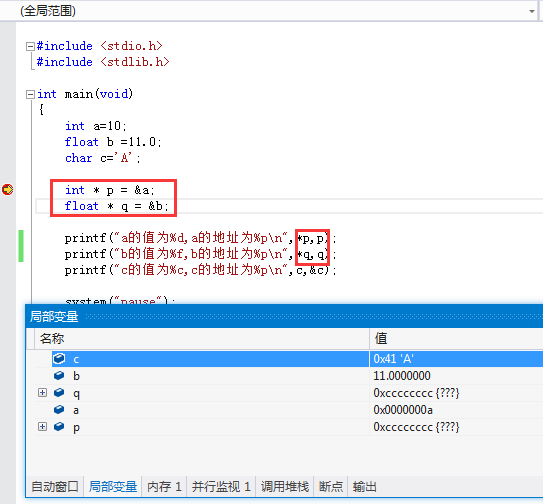




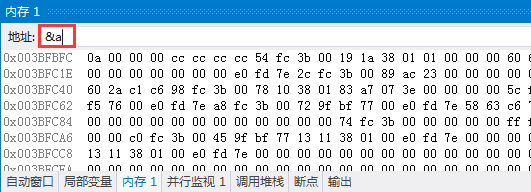
步骤2：内存查看变量的地址并输出，验证每个变量都有地址（多查看几次内存并对比执行结果）。



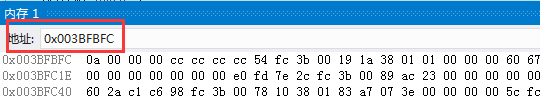
步骤3：指针变量 p，q的理解。指针p，q的数据类型为指针，理解p与&a，q与&b等价的原理，\*p，\*q的意义。



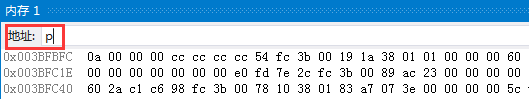
内存查看&a，p，&p的值，并对比分析其关系。

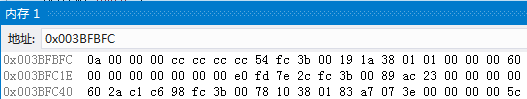


&a，即变量a的地址为0x003BFBFC。

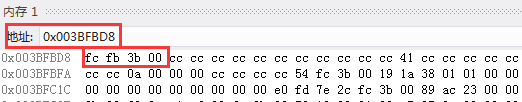


变量p的值为0x003BFBFC，即变量a的地址&a。





&p即p的地址0x003BFBD8，变量p占据4个字节。

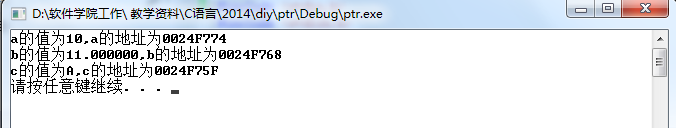


同理自行分析指针变量q，并设计指向变量c的指针。

**实验结果/结论**

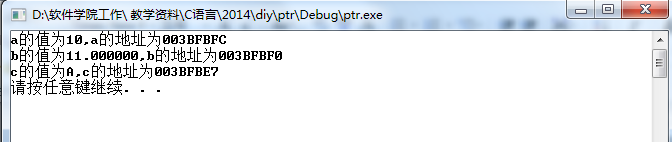
1. **实验结果**

步骤2的运行结果



注意：每次编译器为变量a，b，c分配的内存空间大小确定，受对应数据类型的约束，但是每个变量的地址未必同上次执行过程。

步骤3的运行结果。



注意：通过指针变量取值“\*p”，即通过地址取变量的值，\*运算优先级很高，右结合。地址捆绑空间的大小受所存对象的类型影响。如p存储的是整型变量a的地址，故而从这个地址起的4个字节空间存储a的值10。

2. **实验结论**

* C语言中所有的变量都是要占据内存的，并且其占据内存大小是由变量类型所决定的。
* 指针变量存储的是地址,指针变量也是要占内存的，所有类型的指针变量在32位机上都占4个字节。
* 指针类型为不完整类型，用\*表示其非普通类型，指针变量绑定空间大小受存放对象的类型影响。如int \* p表示，p为指针变量，这4个字节存放整型变量的地址。
* 指针变量有时候被简称为指针，注意“指针”具体是变量还是数据类型。

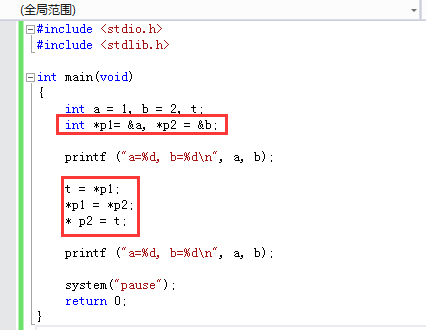
**二、 一重指针的使用以及程序分析**

**实验目的**

1. 掌握一重指针的使用规范
2. 对一重指针程序进行分析

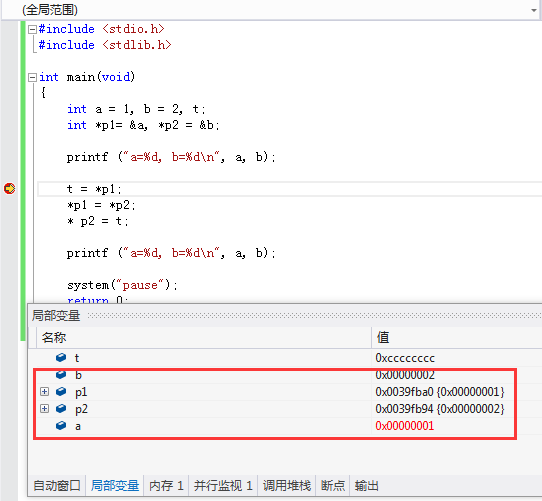
**实验步骤**

步骤1： 使用一重指针编写程序，源代码如下。指针定义的时候即可以赋初值，如p1和p2，赋值后即可参加运算。

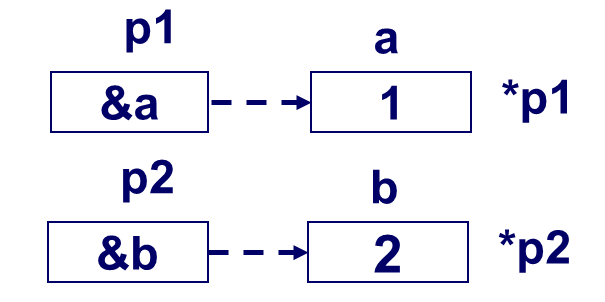


步骤2： 分析该程序

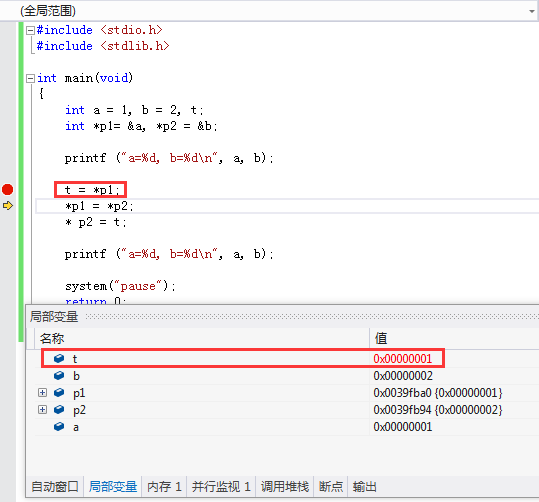
1.a，b，p1和p2变量遵循先定义后使用，先赋值后运算的原则。

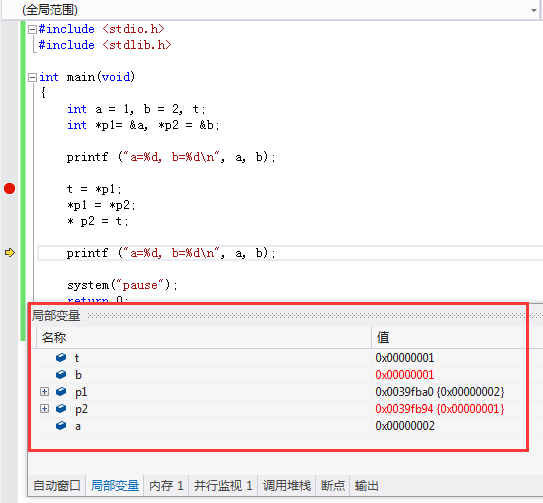


2.内存中的存储以及指向情况，可以用如下内存示意图来展示。

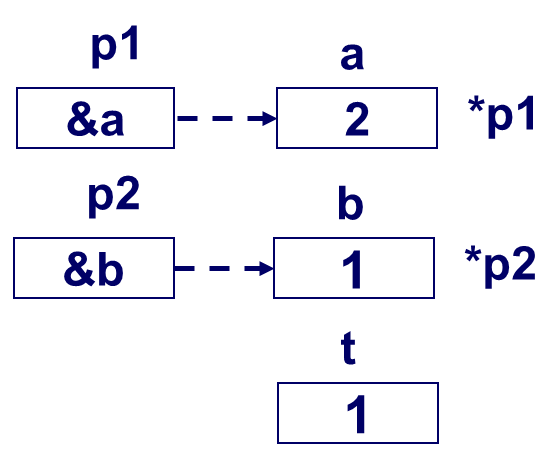


1. 执行3条赋值语句后的存储情况。



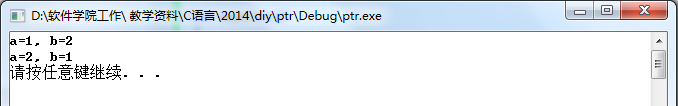


4.内存中的存储以及指向情况，可以用如下内存示意图来展示。



**实验结果/结论**

1. **实验结果**



2. **实验结论**

* 指针变量p的值为变量a的地址，称p指向a
* p是指针变量，\*p和p指向变量等价（\*p1即a，\*p2即b）。
* 涉及指针变量参与运算的程序，使用内存示意图的方式辅助分析程序是一个非常不错的手段。

**三、 多重指针的使用以及程序分析**

**实验目的**

1. 掌握多重指针的使用规范
2. 对多重指针程序进行分析

**实验步骤**

步骤1：

**实验结果/结论**

1. **实验结果**

* 略

2. **实验结论**

* 略

**四、 const指针**

**实验目的**

1. 掌握const指针的作用
2. 对const指针程序进行分析

**实验步骤**

步骤1：

**实验结果/结论**

1. **实验结果**

* 略

2. **实验结论**

* 略

## 6.2 理论题

### A类

1. **填空题**
2. 若d 是已定义的双精度变量，再定义一个指向d 的指针变量p 的代码是\_\_\_\_\_\_\_\_\_\_\_.
3. “\*” 称为\_\_\_\_\_\_\_\_\_\_\_ 运算符， “&” 称为\_\_\_\_\_\_\_\_\_运算符。
4. &后跟变量名，表示该变量的\_\_\_\_\_\_\_\_；\*后跟指针变量名，表示该指针变量\_\_\_\_\_\_\_. &后跟的是指针变量名，表示该指针变量的\_\_\_\_\_\_\_\_\_\_\_\_\_.

**二、选择题**

1. 设有定义“int a＝3， b，\*p＝&a；”，则下列语句中使 b 不为3 的语句是（ ）
2. b＝\* &a
3. b＝\* p
4. b＝a
5. b＝\*a
6. 设指针x 指向的整型变量值为25，则“printf（”%d\n“，++ \*x）；”的输出是（ ）
7. 23
8. 24
9. 25
10. 26
11. 若有说明：“ int i，j＝ 7，\*p＝&i；”，则与“ i＝j；”等价的语句是（ ）
12. i＝\*p；
13. \*p＝\* &j；
14. i=&j；
15. i＝\* \*p；
16. 若有如下定义和语句，则输出结果是（ ）

int \*\*pp， \*p，a＝10，b＝20；pp=&p； p＝&a； p＝&b；

printf（“%d%d\n”，\*p，\*\*pp）；

1. 10， 20
2. 10， 10
3. 20， 10
4. 20， 20
5. 有四组对指针变量进行操作的语句，以下判断正确的选项是（ ）。

(1)int \*p,\*q; q=p;

int a,\*p,\*q; p=q=&a;

(2)int a,\*p,\*q; q=&a; p=\*q;

int a=20,\*p;\*p=a;

(3)int a=b=0,\*p; p=&a; b=\*p;

int a=20,\*p,\*q=&a; \*p=\*q;

(4)int a=20,\*p,\*q=&a; p=q;

int p,\*q; q=&p;

1. 正确：(1) 不正确：(2),(3),(4)
2. 正确：(1),(4) 不正确：(2),(3)
3. 正确：(3) 不正确：(1),(2),(4)
4. 以上结论都不正确
5. 以下程序中调用scanf函数给变量a输入数值的方法是错误的，其错误的原因是 。

main()

{int \*p,\*q,a,b;

p=&a;

printf("input a:");

scanf("%d",\*p);

…

}

1. \*p表示的是指针变量p的地址
2. \*p表示的是变量a的值，而不是变量a的地址
3. \*p表示的是指针变量p的值
4. \*p只能用来说明p是一个指针变量

**三、综合题**

1. 读程序题,写出程序运行的结果。

#include <stdio.h>

void main()

{ int \*p1,\*p2,\*p;

int a=5,b=8;

p1=&a; p2=&b;

if(a<b) { p=p1; p1=p2; p2=p;}

printf("%d,%d\n",\*p1,\*p2);

printf("%d,%d\n",a,b);

}

2.写程序：输入3 个整数，按从大到小顺序输出。要求不改变存储这3个整数的变量的值。

### B类*（指针结合数组、函数,不作为基础知识）*

1. **填空题**
2. 若两个指针变量指向同一个数组的不同元素，可以进行减法运算和\_\_\_\_\_\_\_\_\_\_\_运算。
3. 设有以下定义和语句，则\*（\*（p＋2）＋l）的值为\_\_\_\_\_\_\_\_\_\_.

int a[3][2]={10,20,30,40,50,60},（\*p）[2]；p= a；

**二、选择题**

1. 若有说明语句“int a[10]，\*p＝a；”，对数组元素的正确引用是（ ）
2. a[p]
3. p[a]
4. \*(p+2)
5. p＋2
6. 下面各语句中，能正确进行赋字符串操作的语句是（ ）
7. char s[5] ={“ABCDE”}；
8. char s[5] ={‘A’，‘B’，‘C’，‘D’，‘E’}；
9. char \*s；s＝“ABCDE”；
10. char \*s；scanf （“%s”, &s）；
11. 若有以下定义，则数值为4 的表达式是（ ）

static int w[3][4]＝{{0，1}，{2，4}，{5，8}};int (\* p)[4]＝w；

1. \*w[1]＋l
2. p++，\*（p+1）
3. w[2][2]
4. p[1][1]
5. **综合题**
6. 以下程序的运行结果是 。

void sub (int x,int y,int \*z)

{

\*z=y-x;

}

int main(void)

{

int a,b,c;

sub(10,5,&a);

sub(7,a,&b);

sub(a,b,&c);

printf("%4d,%4d,%4d",a,b,c);

}

## 本章答案

### A类

1. **填空题**
2. double \*p＝&d.
3. “\*”称为取内容运算符， “&”称为取地址运算符。
4. &后跟变量名，就表示该变量的地址。\*后跟指针变量名，表示该指针变量所指变量的内容。若&后跟的是指针变量名，就表示该指针变量的地址。

**二、选择题**

1. D
2. D
3. B
4. D
5. D
6. B
7. **综合题**
8. 8,5

5,8

1. 设用三个简单变量x，y，z 存储输入的三个整数，如通过比较交换指针变量，当比较后发现要交换时，就交换变量的指针，而不交换变量的值，则在比较结束后，变量的值没有改变，但从指针方向来看，它们的值是从大到小排列的。

#include <stdio.h>

#include <stdlib.h>

int main(void)

{

int x,y,z;

int \*big=&x,\*mid=&y,\*sma=&z; /\*置三个指变量分别指向x，y， z\*/

int \*temp;

printf("Enter x，y，z.\n");

scanf("%d,%d,%d", big,mid,sma);/\*顺序为变量x，y，z 输入值\*/

if(\*big< \*mid) {temp=big;big=mid; mid=temp; }/\*使\*big>=\*mid\*/

if(\*big<\*sma) { temp=big;big=sma;sma=temp; }/\*使\*big>=\*sma\*/

if(\*mid<\*sma ) {temp = mid ; mid=sma ;sma=temp; }/\*使\*mid>=\*sma\*/

printf("%d\t%d\t%d\n", x,y,z);/\*按输入顺序输出x，y，z\*/

printf("%d\t%d\t%d\n",\*big,\*mid,\*sma);/\*按从大到小的顺序输出\*/

system("pause");

return 0;

}

### B类

1. **填空题**
2. 若两个指针变量指向同一个数组的不同元素，可以进行减法运算求它们所指元素相差多少元素。进行关系运算，判定它们所指元素的前后，或是否指向同一个元素等
3. 60

**二、选择题**

1. C
2. C
3. D
4. **综合题**

1. -5, -12, -7