

第7章 指针

§ 7.1 指针与指针变量

§7.2 指针变量的定义、初始化与引用

§ 7.3 指针变量的使用

§7.4 指针数组与多级指针

§7.5 指针类型程序举例

§7.1 指针与指针变量

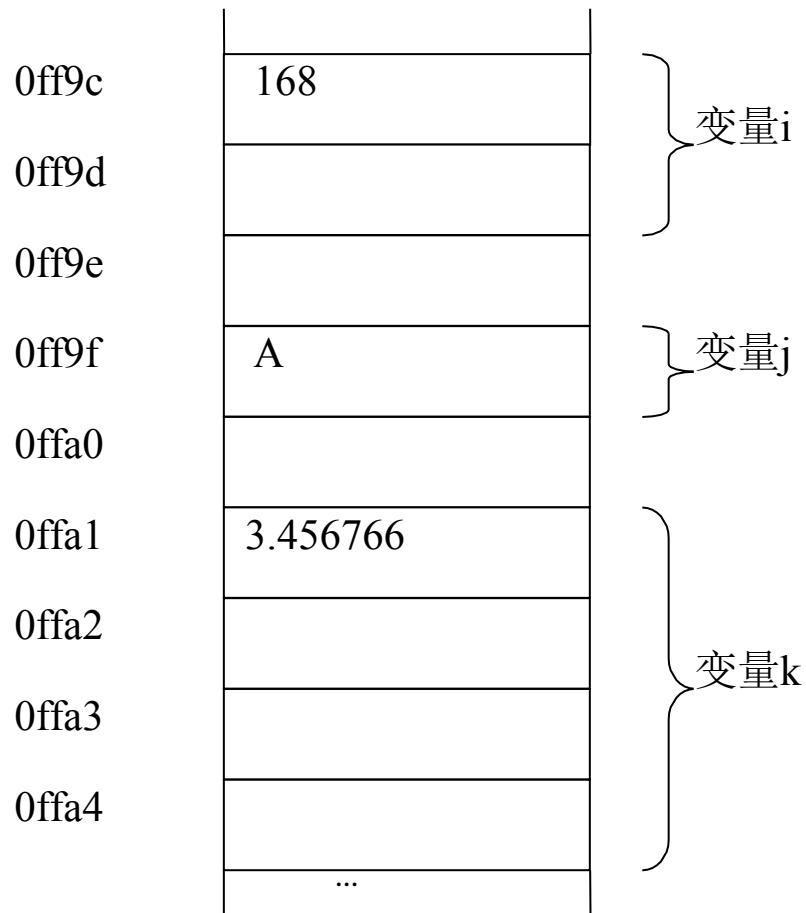
一、指针

在C语言所编写的代码中，基本元素除了各种类型的数据、数组之外，还有变量、语句、函数等元素。在代码被执行时，它们都需要被装入到内存储器中，而计算机的内存储器已被划分为一个个的内存储单元，存储单元并按照一定的规则编号，这个编号称为存储单元的地址。

§7.1 指针与指针变量

一、指针

例: `int i=168, char j=A, double k=3.456766;`



§7.1 指针与指针变量

一、地址

- **【例7-1】** 察看变量地址
- 分析：我们可以通过%x格式，把变量的地址输出出来进行观察。
- #include <stdio.h>
- main()
- { int a=32766, b=-32766, c[10];
- unsigned m=6553;
- float t=3.4e+37;
- printf("a=%d\n",a);
- printf("变量a的地址: %x\n",&a);
- printf("m=%u\n",m);
- printf("变量m的地址: %x\n",&m);
- printf("t=%e\n",t);
- printf("变量t的地址: %x\n",&t);
- printf("数组c[12]的地址: %x\n",c);
- }

§7.1 指针与指针变量

一、指针

我们可以看到，同样的程序在不同的编译环境下，系统给相同的变量开设的存储空间是不相同的，即使是在同一个编译环境下，在不同的时期，对同一个变量所赋予的存储空间也是不相同的。系统不仅对变量如此，对其它的基本元素也是一样的，在不同的编译环境下，在不同的运行时期，对同一个基本元素都会分配不同的存储空间，也就是会分配到不同的存储空间地址，为了方便开发程序，于是人们就用**指针**这一新的数据类型来表达不同的存储单元地址空间了。

§7.1 指针与指针变量

二、指针变量

指针变量是一种特殊的变量，它不同于普通的一般变量，对普通变量赋值时，我们可以赋予各种类型的数据，例如，`int a=10; float x=3.141592;`；而对于指针变量我们只能赋予指针这一特殊类型的数据，这也就是说指针变量所取的值只能为地址。

例如，在程序中声明了一个int型的变量a，其值为168，系统为变量a分配的首地址假设为6FDF4H，pa是一个存放变量a地址的指针变量，那么pa所取的值即为6FDF4H。

§7.2 指针变量的定义、初始化与引用

一、指针变量的定义、初始化

指针变量定义的形式为：

数据类型 *指针变量名1[=初始值], *指针变量名2[=初始值],

例如：int i;

float m;

char c;

int *pa_1=&i; /* pa_1是指向int型变量的指针变量）

float *pa_2=&m; /* pa_2是指向float型变量的指针

char *pa_3=&c; /*pa_3是指向char型变量的指针变量

§7.2 指针变量的定义、初始化与引用

二、指针变量的引用

指针变量经过初始化就会指向一个普通的变量，通过指针变量我们就可以对所指的变量进行各种操作运算。

§7.2 指针变量的定义、初始化与引用

【例7-2】 通过指针变量访问整型变量

分析：首先可以声明指针变量，然后通过初始化，将指针变量指向变量。

```
# include <stdio . h >
void main()
{ int a , b ;
  int * pointer _1 , * pointer _2 ;
  a=100;b=10;
  pointer _1=&a ;    /*把变量a的地址赋给 pointer_1 */
  pointer _2=&b ;    /*把变量b的地址赋给 pointer_2 */
  printf ( "%d , , %d\n" , a , b );
  printf ( "%d , , %d\n" , * pointer _1 , * pointer _2 );
}
```

§7.2 指针变量的定义、初始化与引用

【例7-3】 任意输入两个整数a和b，然后按先大后小的顺序输出a和b。

```
# include <stdio. h>
void main()
{ int * p1 , * p2 , * p , a , b ;
  scanf ("%d , %d", &a ,&b);
  p1=&a;
p2=&b;
  if(a<b )
  {
p1=&b; p2=&a; /* 交换指针变量的值，即p1、p2所指的变量交换
*/
}

  printf("a=%d, b=%d\n\n", a, b);
  printf("max=%d, min=%d\n", *p1, *p2);
}
```

§7.3 指针变量的使用

一、指向变量的指针变量的使用

1 给所指变量赋值

【例7-4】 通过指针变量给所指向的变量赋值

分析：通过指针变量给所指的变量赋值，其实就是直接在指针变量所取值处赋值，在scanf()语句中用指针变量代替&变量。

```
#include<stdio.h>
void main()
{
    int *p1, *p2 , a, b;
    p1=&a; p2=&b;
    scanf("%d,%d", p1,p2);
    printf("%d,%d", a, b);
}
```

§7.3 指针变量的使用

2 改变指针变量的指向来访问不同的变量

【例7-5】 使两个指针变量交换指向。

```
#include<stdio.h>
main()
{
    int *p1 , *p2 ,*p , a, b;
    scanf ("%d , %d" , &a , &b);
    p1=&a ; p2=&b ;                               /* p1指向a , p2指向b*/
    printf ("%d , %d\n" ,*p1 , *p2
    p=p1 ; p1=p2 ; p2=p ;                          /* p1和p2交换指向*/
    printf ("%d ,%d\n" , *p1, *p2);                /*输出指针所指向空间的内
    容*/
}
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/457026121130006144>