第8章 善于利用指针

8.1指针是什么

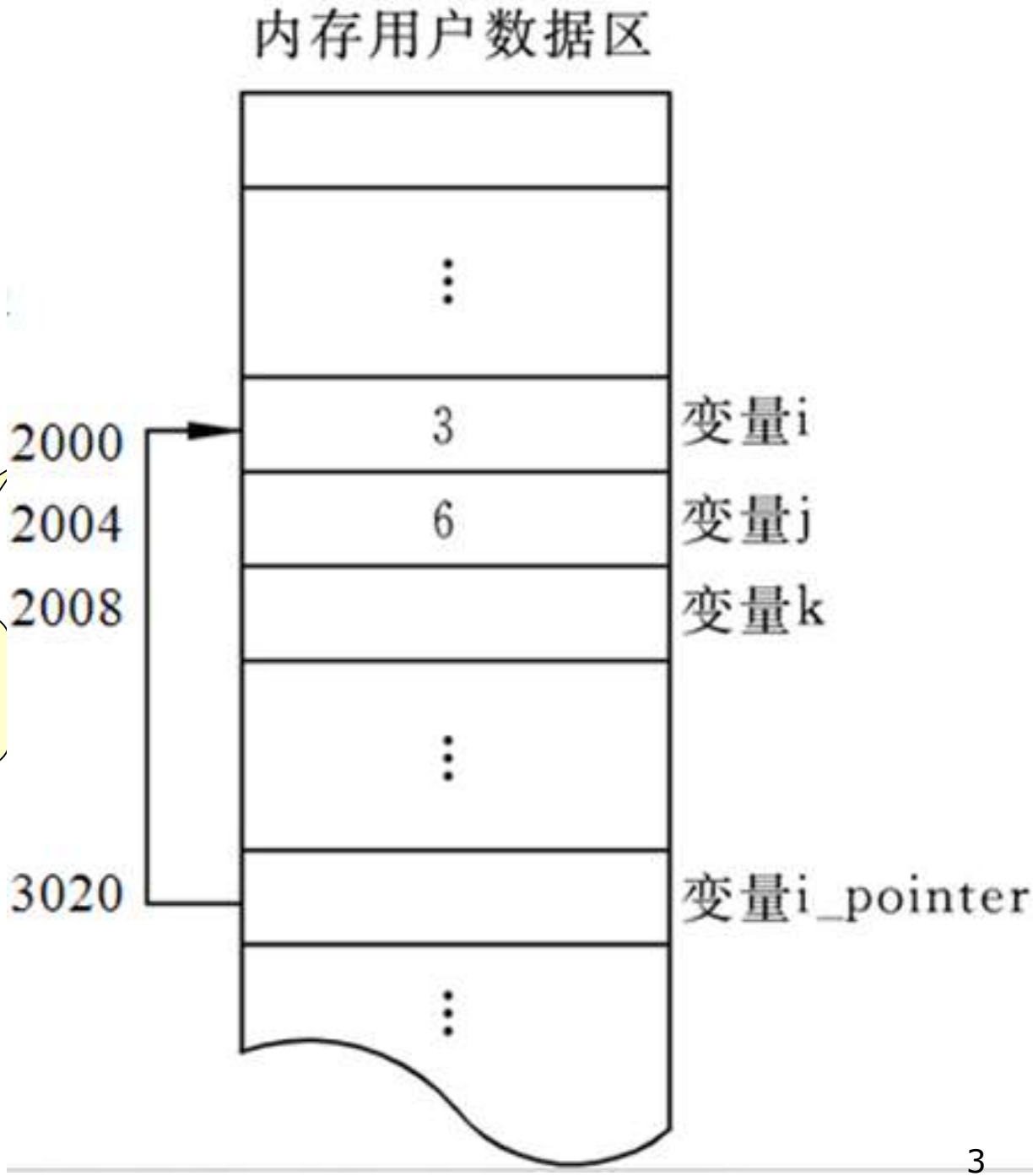
- 如果在程序中定义了一个变量，在程序运行时，系统就会给该变量分配内存单元
- 编译系统根据程序中定义的变量类型，分配一定长度的空间
- 例如，**GCC、VS2010/3**为整型**int**变量分配**4**个字节，对单精度浮点型**float**变量分配 4 个字节，对字符型变量分配 1 个字节
- 内存区的每一个字节有一个编号，这就是“地址”，它相当于我们教室中的座位号。
- 在地址所标识的内存单元中存放数据，这相当于教室座位上学习的学生一样。
- 由于通过地址能找到所需的变量单元，我们可以说，地址指向该变量单元。
- 将地址形象化地称为“指针”

int i=3,j=6,k;

printf(“%d”,i);

通过变量名i

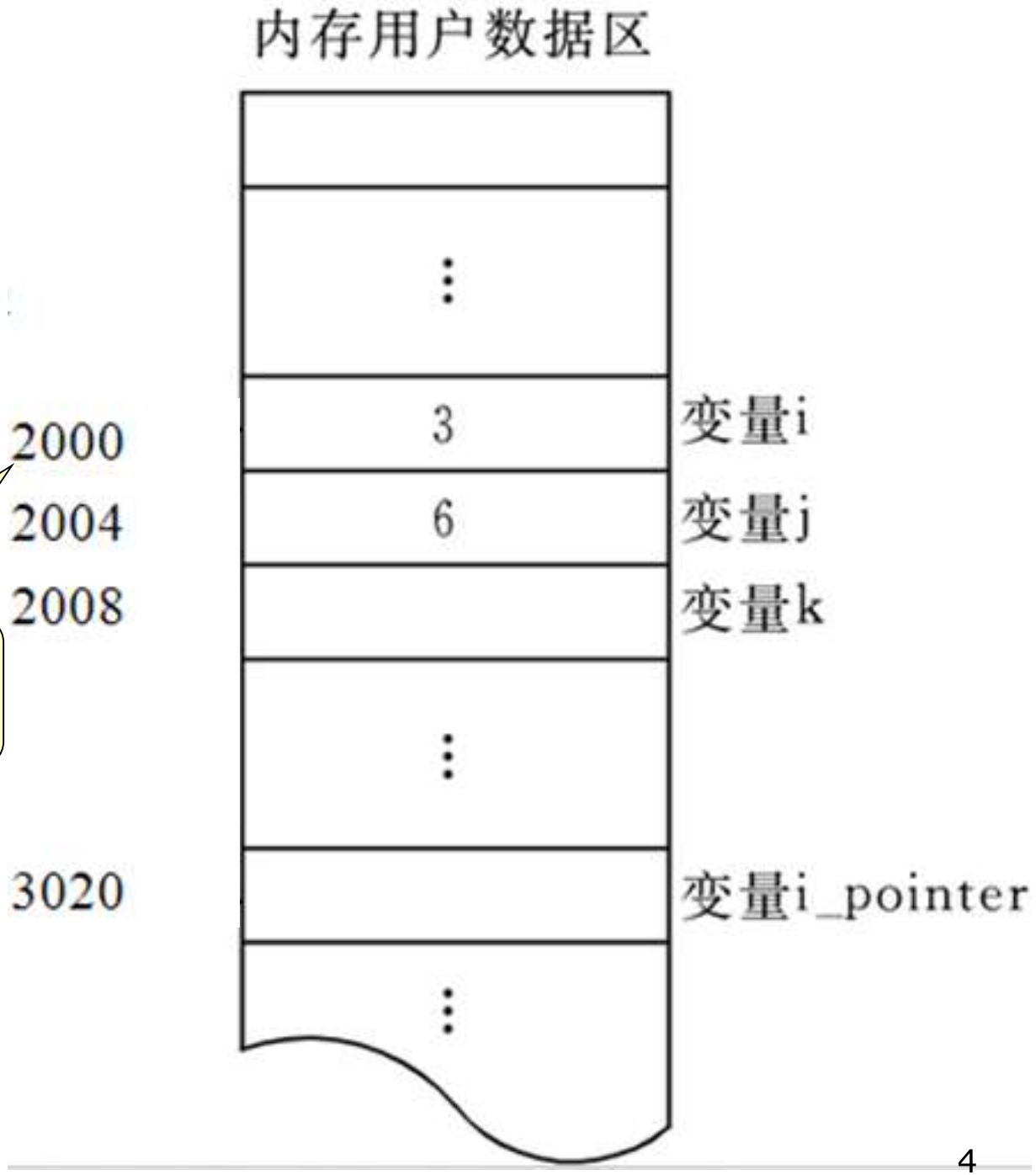
找到i的地址**2000**，从而从存储单元读取**3**



```
int i=3,j=6,k;

printf("%d",i);
```

通过变量名i

找到i的地址2000，从而从存储单元读取3



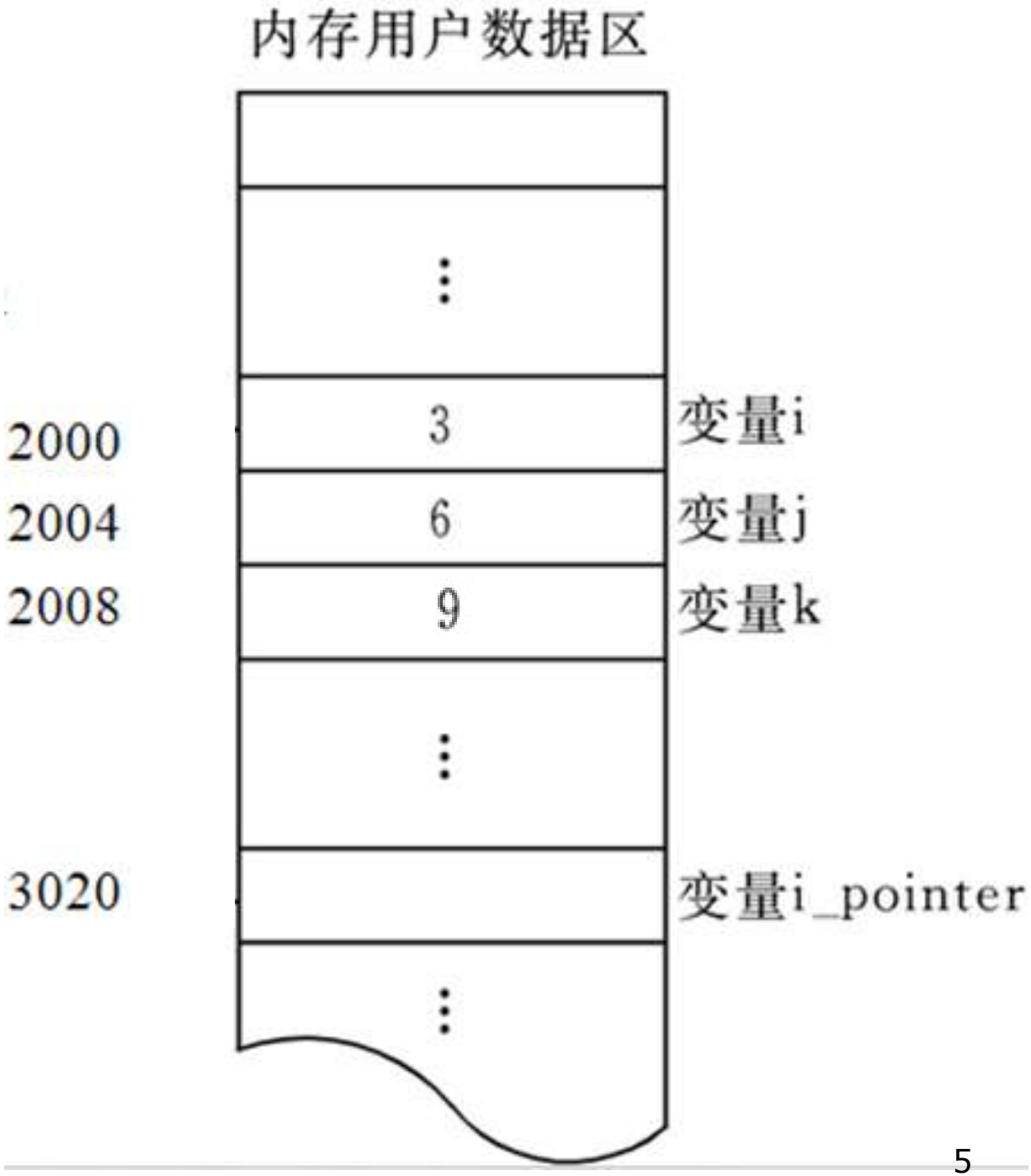
```
int i=3,j=6,k;

k=i+j;
```

从这里取3

从这里取6

将9送到这里

直接存取



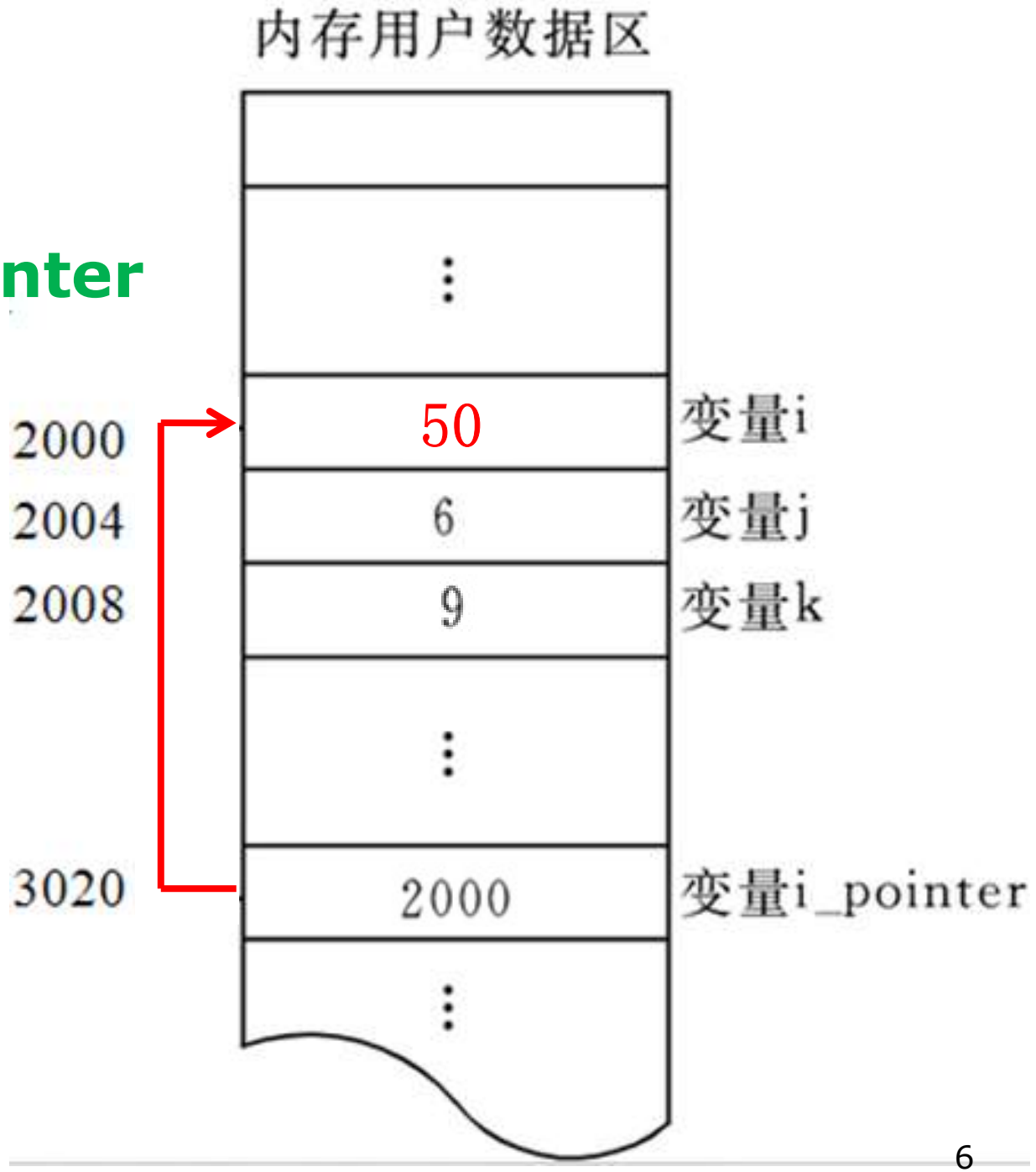
```
int i=3,j=6,k;
```

定义特殊变量i_pointer

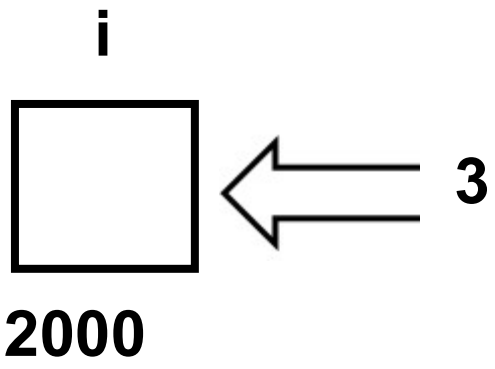
```
i_pointer=&i;
*i_pointer=50;
```

将i的地址存到这里

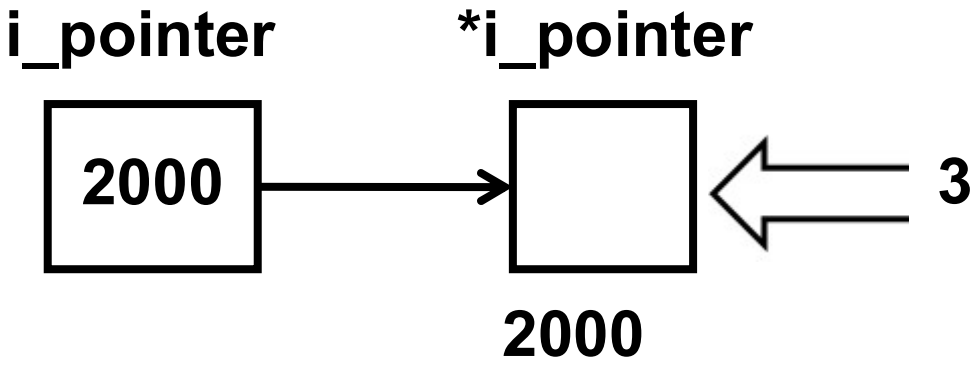
间接存取



直接存取



间接存取



- 指针是一种数据类型，是从函数类型或对象类型派生的，也称为引用类型。
- 指针类型用来描述对象，提供一种间接访问对象（通过内存地址访问对象）的方式。
- 指针与指针变量

8.2 指针变量

8.2.1怎样定义指针变量

8.2.2怎样引用指针变量

8.2.3使用指针变量的例子

8.2.4 指针变量作为函数参数

8.2.1 怎样定义指针变量

- 指针是一种数据类型，是从函数类型或对象类型派生的，也称为引用类型。
- 指针类型用来描述对象，提供一种间接访问对象（通过内存地址访问对象）的方式。
- 指针既然是一种数据类型，就可以定义该类型的变量，通过指针类型定义的变量，称为指针变量。定义指针变量的一般形式为：

  类型说明符 * 指针变量名; // T *variable;

  如：**int *p1, *p2;**

  ◆**int**是为指针变量所引用的类型，称为“基类型”

  ◆基类型指定指针变量可指向的变量类型

  ◆如**p1**可以指向整型变量，但不能指向浮点型变量

8.2.1 怎样定义指针变量

- 下面都是合法的定义和初始化：

```
float  *pointer_3;
char  *pointer_4;
int  a,b;
int  *pointer_1=&a,*pointer_2=&b;
```

```
pointer_3=3.1234;
pointer_4=&a;
```

错误

8.2.2 怎样引用指针变量

- 在引用指针变量时，可能有三种情况：
  - ◆给指针变量赋值。如：**p=&a;**
  - ◆引用指针变量指向的变量。如有
    p=&a; *p=1;
    则执行**printf("%d",*p);** 将输出**1**
  - ◆引用指针变量的值。如：**printf("%p",p);**
- 要熟练掌握两个有关的运算符：
  (1) & 取地址运算符。
  &a是变量**a**的地址
  (2) * 指针运算符（“间接访问”运算符）
  如果： **p**指向变量**a**，则***p**就代表**a**。
  k=*p; (把**a**的值赋给**k**)
  ***p=1;** (把**1**赋给**a**)

使p指向a

*p相当于a

以十六进制输出a的地址

8.2.3使用指针变量的例子

例8.1 通过指针访问整型变量。

```
#include <stdio.h>
int main()
{ int a=100,b=10;
  int *p1,*p2;
  p1=&a;
  p2=&b;
  printf("a=%d,b=%d\n",a,b);
  printf("*p1=%d,*p2=%d\n",*p1,*p2);
  return 0;
}
```

此处的*出现在数据类型名的后边，用来定义两个指针变量

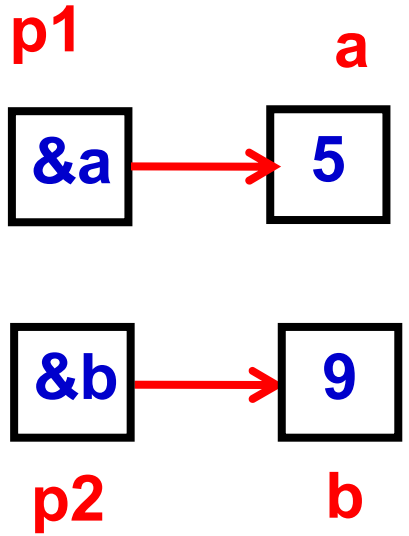
使p1指向a

使p2指向b

直接输出变量a和b的值

间接输出变量a和b的值此处的*用在指针变量名的前面，用来表示指针变量所引用的对象

例8.2 输入a和b两个整数，按先大后小的顺序输出a和b(a中值大，b中值小)，要求用指针处理。



分析：

(1)定义两个指针变量p1和p2，分别指向a和b

(2)若a<b，则通过指针交换两个变量的值

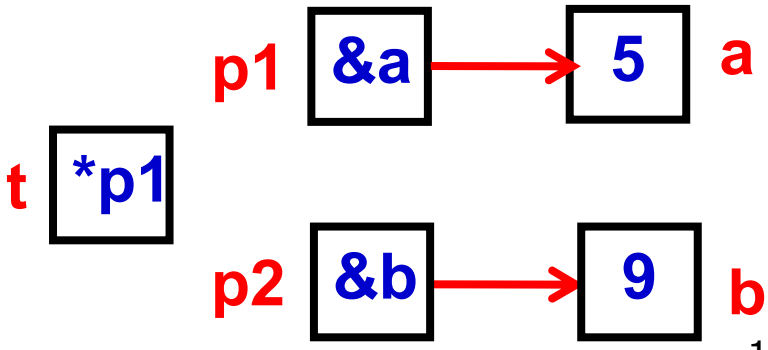
直接交换a,b的值int t=a;a=b;b=t;

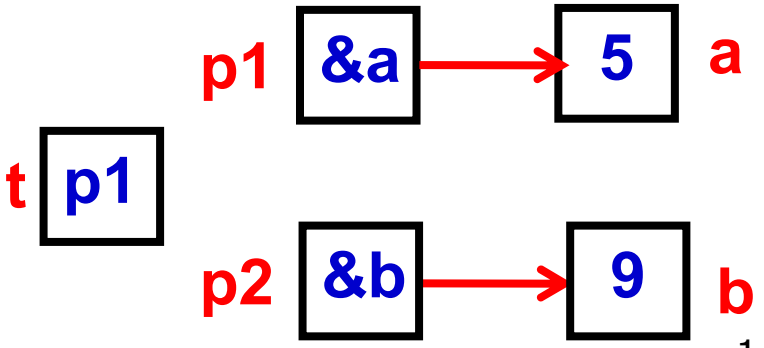
a和b的值用指针来表示，修改上述语句即可

int t=*p1; *p1 = *p2; *p2 = t;

(3)输出结果

若将上式改为：int *t = p1; p1 = p2; p2 = t;结果？





14

8.2.4 指针变量作为函数参数

例**8.3** 题目要求同例**8.2**，即输入**a**和**b**两个整数，按先大后小的顺序输出**a**和**b(a**中值大，**b**中值小**)** 。要求用函数处理。

- 分析：定义一个函数**swap**如下：

```
void swap(int x, int y){
    int t = x; x = y; y = t;
}
```

- 结果原因分析：
  - ◆局部变量作用域及生存期
  - ◆函数调用参数传递方式：值传值
- 解决办法：
  - ◆函数调用参数传递方式采用：地址传递
  - ◆修改**swap**函数，形参由值类型改为指针类型

```
void swap(int *p1,int *p2)
{
    int t=*p1;
    *p1=*p2;
    *p2=t;
}
```

- 如果想通过函数调用得到 n 个要改变的值：
  ① 在主调函数中设 n 个变量，用 n 个指针变量指向它们
  ② 设计一个函数，有**n**个指针类型的形参。在这个函数中改变这 n 个形参的值
  ③ 在主调函数中调用这个函数，在调用时将这**n**个指针变量作实参，将它们的地址传给该函数的形参
  ④ 在执行该函数的过程中，通过形参指针变量，改变它们所指向的 n 个变量的值
  ⑤主调函数中就可以使用这些改变了值的变量

- 课后作业：
  结合例**8.2**分析课本例题**8.4**、**8.5**