

第十三节：赋值语句的覆盖性。

【13.1 什么是赋值语句的覆盖性？】

```
a=b;
```

上述代码，执行完这条赋值语句后，会把右边变量 b 的数值复制一份给左边变量 a，a 获得了跟 b 一样的数值，但是 a 原来自己的数值却丢失了，为什么会丢失？就是因为被 b 复制过来的新数据给覆盖了，这就是赋值语句的覆盖性。

【13.2 例程的分析和练习。】

既然赋值语句有覆盖性的特点，那么如何让两个变量相互交换数值？假设 a 原来的数据是 1，b 原来的数据是 5，交换数据后，a 的数据应该变为 5，b 的数据应该变为 1，怎么做？很多初学者刚看到这么简单的题目，会马上根据日常生活的思路，你把你的东西给我，我把我的东西给你，就两个步骤而已，看似很简单，现在按这个思路编写一段程序看看会出什么问题，代码如下：

```
/*---C 语言学习区域的开始。-----*/

void main() //主函数
{
    unsigned char a=1;    //定义的变量 a 被分配了 1 个字节的 RAM 空间，保存的数据被初始化成 1。
    unsigned char b=5;    //定义的变量 b 被分配了 1 个字节的 RAM 空间，保存的数据被初始化成 5。

    b=a; //第一步：为了交换，先把 a 的数赋值给 b。
    a=b; //第二步：为了交换，再把 b 的数赋值给 a。

    View(a);    //把第 1 个数 a 发送到电脑端的串口助手软件上观察。
    View(b);    //把第 2 个数 b 发送到电脑端的串口助手软件上观察。

    while(1)
    {
    }
}

/*---C 语言学习区域的结束。-----*/
```

在电脑串口助手软件上观察到的程序执行现象如下：

开始...

第 1 个数
十进制:1

十六进制:1

二进制:1

第 2 个数

十进制:1

十六进制:1

二进制:1

分析:

第 1 个数 a 和第 2 个数 b 居然都是 1! 这不是我们想要的结果。我们要的交换结果是: 交换后, a 变为 5, b 变为 1。在哪个环节出了问题? 把镜头切换到上述代码的“第一步”和“第二步”, 由于 b 的数据在执行完“第一步”后, b 自己原来的数据 5 被覆盖丢失了变成新的数据 1, 接着执行“第二步”后, 此时相当于把 b 的新数据 1 赋值给 a, 并没有 5! 所以 a 和 b 的数据都是 1, 不能达到交换后“a 为 5, b 为 1”的目的。其实就是赋值语句的覆盖性在作祟。

上述交换数据的程序宣告失败! 怎么办? 既然赋值语句具有覆盖性, 那么两变量想交换数据, 就必须借助第三方变量来寄存, 此时只需要多定义一个第三方变量 t。正确的代码如下:

```
/*---C 语言学习区域的开始。-----*/
```

```
void main() //主函数
```

```
{
```

```
    unsigned char a=1;    //定义的变量 a 被分配了 1 个字节的 RAM 空间, 保存的数据被初始化成 1。
```

```
    unsigned char b=5;    //定义的变量 b 被分配了 1 个字节的 RAM 空间, 保存的数据被初始化成 5。
```

```
    unsigned char t;      //定义一个第三方变量 t, 用来临时寄存数值。
```

```
    t=b; //第一步: 为了避免 b 的数据在赋值后被覆盖丢失, 先寄存一份在第三方变量 t 那里。
```

```
    b=a; //第二步: 把 a 的数赋值给 b, b 原来的数据虽然丢失, 但是 b 在 t 变量那里有备份。
```

```
    a=t; //第三步: 再把 b 在 t 变量里的备份赋值给 a。注意, 这里不能用 b, 因 b 原数据已被覆盖。
```

```
    View(a);    //把第 1 个数 a 发送到电脑端的串口助手软件上观察。
```

```
    View(b);    //把第 2 个数 b 发送到电脑端的串口助手软件上观察。
```

```
    while(1)
```

```
    {
```

```
    }
```

```
}
```

```
/*---C 语言学习区域的结束。-----*/
```

在电脑串口助手软件上观察到的程序执行现象如下：

开始...

第 1 个数

十进制:5

十六进制:5

二进制:101

第 2 个数

十进制:1

十六进制:1

二进制:1

分析：

实验结果显示，两变量的数值交换成功。

【13.3 如何在单片机上练习本章节 C 语言程序？】

直接复制前面章节中第十一节的模板程序，练习代码时只需要更改“C 语言学习区域”的代码就可以了，其它部分的代码不要动。编译后，把程序下载进带串口的 51 学习板，通过电脑端的串口助手软件就可以观察到不同的变量数值，详细方法请看第十一节内容。