

第3章

最简单的 C 程序设计——顺序程序设计

3.1 怎样区分表达式和表达式语句？C 语言为什么要设表达式语句？什么时候用表达式，什么时候用表达式语句？

解：略。

3.2 C 语言为什么要把输入输出的功能作为函数，而不作为语言的基本部分？

解：略。

3.3 用下面的 scanf 函数输入数据，使 $a=3, b=7, x=8.5, y=71.82, c1='A', c2='a'$ 。问在键盘上如何输入？

```
#include <stdio.h>
void main()
{int a,b;
 float x,y;
 char c1,c2;
 scanf("a=%d b=%d",&a,&b);
 scanf("%f %e",&x,&y);
 scanf("%c %c",&c1,&c2);
 printf("a=%d,b=%d,x=%f,y=%f,c1=%c,c2=%c\n",a,b,x,y,c1,c2);
}
```

解：可按如下方式在键盘上输入：

a=3 b=7 ✓
8.5 71.82A a ✓

输出为：

$a=3, b=7, x=8.500000, y=71.820000, c1=A, c2=a$

请注意：在输入完 8.5 和 71.82 两个实数给 x 和 y 后紧接着输入字符 A，中间不要有空格，由于 A 是字母而不是数字，系统在遇到字母 A 时就确定输入给 y 的数值已结束。字符 A 就送到下一个 scanf 语句中的字符变量 c1。如果在输入 8.5 和 71.82 两个实数后输入空格符，会怎么样呢？

a=3 b=7 ✓
8.5 71.82 A a ✓

这时 71.82 后面的空格字符就被 c1 读入, c2 读入了字符 A。在输出 c1 时就输出空格。

输出为:

a=3,b=7,x=8.500000,y=71.820000,c1= ,c2=A

如果在输入 8.5 和 71.82 两个实数后输入回车符,会怎么样呢?

a=3 b=7 ✓
8.5 71.82 ✓
A a ✓

输出为:

a=3,b=7,x=8.500000,y=71.820000,c1=
,c2=A

这时“回车”被作为一个字符送到内存输入缓冲区,被 c1 读入(实际上 c1 读入的是回车符的 ASCII 码),字符 A 被 c2 读取,所以在执行 printf 函数输出 c1 时,就输出一个回车符,输出 c2 时就输出字符 A。

在用 scanf 函数输入数据时往往会出现一些想象不到的情况,例如在连续输入不同类型的数据(特别是数值型数据和字符数据连续输入)的情况。要注意回车符是可能被作为一个字符读入的。

读者在遇到类似情况时,上机多试验一下就可以找出规律来。

3.4 用下面的 scanf 函数输入数据,使 a=10,b=20,c1='A',c2='a',x=1.5,y=-3.75,z=67.8,请问在键盘上如何输入数据?

```
scanf ("%5d%5d%c%c%f%f* f,%f",&a,&b,&c1,&c2,&x,&y,&z);
```

解:

```
#include <stdio.h>
void main()
{int a,b;
 float x,y,z;
 char c1,c2;
 scanf ("%5d%5d%c%c%f%f* f,%f",&a,&b,&c1,&c2,&x,&y,&z);
 printf ("a=%d,b=%d,c1=%c,c2=%c,x=%6.2f,y=%6.2f,z=%6.2f\n",a,b,c1,c2,x,y,z);
}
```

运行情况如下:

UUU10UUU20Aa1.5U-3.75U2.5,67.8 ✓ (此行为输入的数据)
a=10,b=20,c1=A,c2=a,x=U1.50,y=U-3.75,z=U67.80 (此行为输出)

说明：按%5d格式的要求输入a与b时，先输入3个空格，然后再输入10与20。
%*f是用来禁止赋值的。在输入时，对应于%*f的地方，随意输入了一个实数2.5，该值不会赋给任何变量

3.5 设圆半径 $r=1.5$ ，圆柱高 $h=3$ ，求圆周长、圆面积、圆球表面积、圆球体积、圆柱体积。用scanf输入数据，输出计算结果，输出时要求有文字说明，取小数点后2位数字。请编程序。

解：可编程序如下：

```
#include <stdio.h>
void main ()
{float h,r,l,s,sq,vq,vz;
 float pi=3.141526;
 printf("请输入圆半径 r,圆柱高 h: ");
 scanf("%f,%f",&r,&h);           //要求输入圆半径 r 和圆柱高 h
 l=2*pi*r;                        //计算圆周长 l
 s=r*r*pi;                        //计算圆面积 s
 sq=4*pi*r*r;                    //计算圆球表面积 sq
 vq=3.0/4.0*pi*r*r*r;            //计算圆球体积 vq
 vz=pi*r*r*h;                    //计算圆柱体积 vz
 printf("圆周长为:           l=%6.2f\n",l);
 printf("圆面积为:           s=%6.2f\n",s);
 printf("圆球表面积为:       sq=%6.2f\n",sq);
 printf("圆球体积为:         v=%6.2f\n",vq);
 printf("圆柱体积为:         vz=%6.2f\n",vz);
}
```

运行情况如下：

请输入圆半径 r,圆柱高 h: 1.5,3↵

圆周长为: l=9.42

圆面积为: s=7.07

圆球表面积为: sq=28.27

圆球体积为: v=7.95

圆柱体积为: vz=21.21

说明：如果用 Visual C++ 6.0 中文版对程序进行编译，在程序中可以使用中文字符串。在输出时也能显示汉字。如果用 Turbo C 或 Turbo C++，则无法使用中文字符串，读者可以改用英文字符串。

3.6 输入一个华氏温度，要求输出摄氏温度。公式为：

$$c = \frac{5}{9}(F - 32)$$

输出要有文字说明，取两位小数。

解：相应的程序如下所示。

```
#include <stdio.h>
void main()
{float c,f;
 printf("请输入一个华氏温度：");
 scanf("%f",&f);
 c=(5.0/9.0)*(f-32);          /* 注意 5 和 9 要用实型表示,否则 5/9 值为 0 */
 printf("摄氏温度为：%5.2f\n",c);
};
```

运行情况如下：

请输入一个华氏温度：87

得到结果：摄氏温度为：30.56

3.7 编程序,用 `getchar` 函数读入两个字符给变量 `c1`、`c2`,然后分别用 `putchar` 函数和 `printf` 函数输出这两个字符,并思考以下问题：

(1) 变量 `c1`、`c2` 应定义为字符型或整型？还是二者皆可？

(2) 要求输出 `c1` 和 `c2` 值的 ASCII 码,应如何处理？用 `putchar` 函数还是 `printf` 函数？

(3) 整型变量与字符变量是否在任何情况下都可以互相代替？如：

① `char c1,c2;`

② `int c1,c2;`

是否无条件等价？

解：可编程序如下：

```
#include <stdio.h>
void main()
{
 char c1,c2;
 printf("请输入两个字符 c1,c2: ");
 c1=getchar();
 c2=getchar();
 printf("用 putchar 语句输出结果为:");
 putchar(c1);
 putchar(c2);
 printf("\n");
 printf("用 printf 语句输出结果为:");
 printf("%c %c\n",c1,c2);
}
```

运行结果：

请输入两个字符 `c1,c2`: ab

用 `putchar` 语句输出结果为：ab

用 printf 语句输出结果为: a b

请注意: 连续用两个 getchar 函数时是怎样输入字符的。如果用以下方法输入:

a↵
b↵

得到以下运行结果:

用 putchar 语句输出结果为: a

(空一行)

用 printf 语句输出结果为: a

(空一行)

因为第 1 行将 a 和回车符输入到内存的输入缓冲区, 因此 c1 得到 a, c2 得到一个回车符。在输出 c2 时就会产生一个回车换行, 而不会输出任何可显示的字符。在实际操作时, 只要输入了“a↵”, 系统就会认为用户已输入了两个字符。所以应当连续输入 ab 两个字符然后再按回车键, 这样就保证了 c1 和 c2 分别得到字符 a 和 b。

回答思考问题:

(1) c1 和 c2 可以定义为字符型或整型, 二者皆可。

(2) 可以用 printf 函数输出, 在 printf 函数中用 %d 格式符, 即:

```
printf("%d,%d\n", c1, c2);
```

(3) 字符变量在计算机内占 1 个字节, 而整型变量占 2 个或 4 个字节。因此整型变量在可输出字符的范围内 (ASCII 码为 0~255 之间的字符) 是可以与字符数据互相转换的。如果整数在此范围外, 不能代替。

请分析以下 3 个程序:

程序 1:

```
#include <stdio.h>
void main()
{
    int c1, c2;                /* 定义整型变量 */
    printf("请输入两个整数 c1, c2: ");
    scanf("%d, %d", &c1, &c2);
    printf("按字符输入结果: \n");
    printf("%c, %c\n", c1, c2);
    printf("按 ASCII 码输入输出结果为: \n");
    printf("%d, %d\n", c1, c2);
}
```

运行结果:

请输入两个整数 c1, c2: 97, 98↵

按字符输出结果:

a, b

按 ASCII 码输出结果：

97,98

程序 2：

```
#include <stdio.h>

void main()
{
    char c1,c2;                                /* 定义字符型变量 */
    int i1,i2;                                  /* 定义整型变量 */
    printf("请输入两个整数 c1,c2: ");
    scanf("%c,%c",&c1,&c2);
    i1=c1;                                      /* 赋值给整型变量 */
    i2=c2;
    printf("按字符输入结果: \n");
    printf("%c,%c\n",i1,i2);
    printf("按整数输入结果: \n");
    printf("%d,%d\n",c1,c2);
}
```

运行结果：

请输入两个字符 c1,c2: a,b ↵

按字符输出结果：

a,b

按整数输出结果：

97,98

程序 3：

```
#include <stdio.h>

void main()
{
    char c1,c2;                                /* 定义字符型变量 */
    int i1,i2;                                  /* 定义为整型变量 */
    printf("请输入两个整数 i1,i2: ");
    scanf("%d,%d",&i1,&i2);
    c1=i1;                                      /* 将整数赋值给字符变量 */
    c2=i2;
    printf("按字符输入结果: \n");
    printf("%c,%c\n",c1,c2);
    printf("按整数输入结果: \n");
    printf("%d,%d\n",c1,c2);
}
```

运行结果：

请输入两个整数 i1,i2: 298,330 ↵

按字符输出结果：*,J

按整数输出结果：42,74

请注意 `c1` 和 `c2` 是字符变量,只占 1 个字节,只能存放 0~255 范围内的整数,而现在输入给 `i1` 和 `i2` 的值已超过 0~255 的范围,所以只将整型变量 `i1` 和 `i2` 在内存存储单元中的最后一个字节(低 8 位)赋给 `c1` 和 `c2`。可以看到： $298-256=42$, $330-256=74$ 。读者可以写出 298 和 330 的二进制形式,取其低 8 位,即可一目了然。与 ASCII 码 42 和 74 相应的字符为“*”和“J”,因此。按字符形式输出结果时输出：*,J。请读者注意分析。