C 语言程序设计要点 2017-09-26

今天的主要内容是表达式。

1. 表达式（expression）是 C 语言最基本的计算单元。
2. 表达式由（1 至 2 个）操作数（operand）和一个操作符（operator）构成。例如：

   ```
   i + j
   ++i
   ```

   其中：
   - 如果运算符只连接一个操作数，那么该运算符称为“单目运算符”
   - 如果运算符连接两个操作数，那么该运算符称为“双目运算符”。这里，运算符左边的操作数称为 lhs，右边的称为 rhs
   - 推广：连接 n 个操作数的运算符称为“n 目运算符”
     - 预告：C 语言有一个三目运算符
   - 运算符有优先级和结合性的特点。教材上有这两项的描述。
3. 每一个表达式都会完成两项工作：

   a) 进行指定的运算，得到结果

   b) 表达式本身也有一个结果，称为“表达式的值”

   例如有：

   ```
   int i = 2, j = 3;
   ```

   那么表达式

   ```
   i + j
   ```

   首先要完成加法运算，得到运算结果 5；而这个结果也成为整个加法表达式的值
4. 任何一个合法表达式后加上分号就构成了一条语句。例如：

   ```
   i + j;
   ```

   以上语句会执行表达式的运算，但最终的结果（表达式的值）因为没有任何单元接收它就被忽略了。

常用表达式

1. 赋值表达式，形如：

   ```
   a = b
   ```

   它完成的操作是：

   1）将 b 的值赋予 a，即 a 单元的值被 b 单元的值取代。实际上，这是一种复制(copy)操作。

   2）整个赋值表达式的值是赋值号=右边表达式的值。
   - 赋值运算符是个双目运算符。它有一个副作用，就是要改变 lhs 的值。相比之下，很多的双目运算符不改变任何一个操作数。
   - 赋值运算可以级联(cascading)，例如：

     ```
     a = b = c
     ```

     但以上赋值的顺序是：

     ```
     a = (b = c)
     ```

     即：b 是被 c 赋值，而 a 是被表达式 b=c 赋值。但从最终效果来看，可以认为 a 也是被 c 赋值。

- 要完成赋值操作，a 必须是个左值(lvalue)，即它的值是可以改变的。相较之下，表达式 i + j 的结果是个右值（rvalue），则不能出现在赋值号的左边，即

  i + j = 2

  是错误的。

  注：所有的字面常量（例如：12，0.1，'c'，"abc"）都是右值。

2. 复合赋值表达式，形如：

   lhs @= rhs

   其中，@是一个双目运算符。

   复合赋值表达式可以认为是完成了以下工作：

   1）完成指定的运算 lhs @ rhs，得到运算结果 r；

   2）将 r 赋给 lhs；

   3）整个表达式的值等于 r

   - 只要有赋值操作，那么 lhs 就会被改变
   - 复合赋值也可以级联，例如：

   如果有 int i = 6;

   那么表达式

   i += i *= i /= 2

   的值是 18。

   以上表达式的执行顺序是：

   i /= 2 ➔ i 的值是 3，该表达式的值也是 3

   i *= 3 → i 的值是 9，该表达式的值也是 9

   i += 9 → i 的值是 18，该表达式的值也是 18

3. 算数表达式，形如：

   - 双目：lhs @ rhs

   其中，@是+、-、*、/、%之一

   注意：

   1) 在做除法时，如果 lhs 和 rhs 都是整数，那么除法就是整除（截断整除），结果是整数（除法的商）。这意味着：如果 lhs 小于 rhs，那么结果就是 0。

   2) %是取余数运算。一般地，lhs 和 rhs 都是正整数。

   3) 双目算数运算不改变任何一个操作数的值

   - 单目：@lhs 或者 lhs@，分别称为前缀和后缀。

   其中，@是++、--之一

   前缀和后缀运算是有区别的。但无论如何，都要改变操作数的值。

   以++i 为例：

   1）完成的运算是使 i 自增 1

   2）整个前缀表达式的值是自增后的 i 的值

   以 i++为例：

   1）完成 i 的自增 1

   2）整个后缀表达式的值是自增前 i 的值

   实例：

```
int i = 6, j;

j = ++i;  //此后 ij 的值都是 7
```

```
j = i++;  //此后 i=8, j=7
```

基于此，请在程序中尽量使用前缀自加！

示例程序：将一个三位数倒过来并输出。

```
#include <stdio.h>

int main()
{
    int abc, a, b, c;

    scanf("%d", &abc);

    a = abc / 100;
    b = abc % 100 / 10;
    c = abc % 10;

    printf("%d, %d\n", abc, c * 100 + b * 10 + a);

    return 0;
}
```