

C 语言局部变量和全局变量

在《[C 语言函数的参数和返回值](#)》中提到，形参变量只在被调用期间才分配内存单元，调用结束立即释放。这一点表明形参变量只有在函数内才是有效的，离开该函数就不能再使用了，称为变量的作用域。**所谓作用域 (Scope)，就是变量的有效范围。**

不仅对于形参变量，C 语言中所有的变量都有自己的作用域。变量定义的方式不同，其作用域也不同。

C 语言中的变量，按作用域范围可分为两种：局部变量和全局变量。

局部变量

定义在函数内部的变量称为局部变量(Local Variable)，它的作用域仅限于函数内，离开该函数后再使用就是错误的。例如：

```
1. int f1(int a) {  
2.     int b, c; //a, b, c 仅在函数 f1() 内有效  
3.     return a+b+c;  
4. }  
5. int main() {  
6.     int m, n; //m, n 仅在函数 main() 内有效  
7.     return 0;  
8. }
```

几点说明：

- 1) main 函数中定义的变量也是局部变量，只能在 main 函数中使用；同时，main 函数中也不能使用其它函数中定义的变量。main 函数也是一个函数，与其它函数地位平等。
- 2) 形参变量、在函数体内定义的变量都是局部变量。**实参给形参传值的过程也就是给局部变量赋值的过程。**
- 3) 可以在不同的函数中使用相同的变量名，它们代表不同的对象，分配不同的单元，互不干扰，也不会发生混淆。

4) 在语句块中也可定义变量，它的作用域只限于当前语句块。例如：

```
1. #include <stdio.h>
2.
3. int main() {
4.     int i=2, j=3, k;
5.     k=i+j;
6.     {
7.         int k=8;
8.         printf("%d\n", k);
9.     }
10.    printf("%d\n", k);
11.    return 0;
12. }
```

运行结果：

8

5

本程序在 main 中定义了 i、j、k 三个变量，其中 k 未赋初值。而在复合语句内又定义了一个变量 k，并赋初值为 8。应该注意这两个 k 不是同一个变量。在复合语句外由 main 定义的 k 起作用，而在复合语句内则由在复合语句内定义的 k 起作用。因此程序第 3 行的 k 为 main 所定义，其值应为 5。第 6 行输出 k 值，该行在复合语句内，由复合语句内定义的 k 起作用，其初值为 8，故输出值为 8，第 8 行输出 i，k 值。i 是在整个程序中有效的，第 6 行对 i 赋值为 3，故以输出也为 3。而第 8 行已在复合语句之外，输出的 k 应为 main 所定义的 k，此 k 值由第 3 行已获得为 5，故输出也为 5。

全局变量

在所有函数外部定义的变量称为全局变量 (Global Variable)，它的作用域是整个源程序。例如：

```
1. int a, b; //外部变量
```

```

2. void f1() {
3.     // Code
4. }
5. float x, y; //外部变量
6. int fz() {
7.     // Code
8. }
9. int main() {
10.    // Code
11.    return 0;
12.}

```

a、b、x、y 都是在函数外部定义的全局变量。x、y 定义在函数 f1 之后，在 f1 内无效。a、b 定义在源程序最前面，因此在 f1、f2 及 main 内都有效。

【示例】输入正方体的长宽高 l、w、h。求体积及三个面 x*y、x*z、y*z 的面积。

```

1. #include <stdio.h>
2.
3. int s1, s2, s3; // 面积
4. int vs( int a, int b, int c) {
5.     int v; // 体积
6.     v = a*b*c;
7.     s1 = a*b;
8.     s2 = b*c;
9.     s3 = a*c;
10.    return v;
11.}
12.
13. int main() {
14.     int v, l, w, h;
15.     printf("Input length,width and height: ");
16.     scanf("%d %d %d",&l,&w,&h);
17.     v = vs(l, w, h);
18.     printf("v=%d, s1=%d, s2=%d, s3=%d\n", v, s1, s2, s3);
19.     return 0;
20.}

```

运行结果：

Input length,width and height: 3 4 5

v=60, s1=12, s2=20, s3=15

【示例】全局变量和局部变量同名。

```
1. #include <stdio.h>
2.
3. int a=3, b=5; // a, b 为全局变量
4. int max(int a, int b) { //a, b 为局部变量
5.     int c;
6.     c = a>b ? a : b;
7.     return c;
8. }
9.
10. int main() {
11.     int a = 8;
12.     printf("%d\n", max(a, b));
13.     return 0;
14. }
```

运行结果：

8

在同一个源文件中，如果全局变量与局部变量同名，那么在局部变量的作用范围内，外部变量被“屏蔽”，也就是它不起作用。