

C语言强制类型转换和自动类型转换

2023年3月17日 16:49

上一节字符型数据参加的算术运算实际上包含了 C 程序中的数据类型转换功能。在 C 语言中变量的数据类型是可以转换的，转换方法有两种：自动转换和强制转换。

自动转换是不同数据类型的数据在进行混合运算时，由编译系统自动完成的。自动转换遵循以下规则：

- 1) 若参与运算的变量数据类型不同，则先转换为同一类型，然后再运算。
- 2) 转换按数据长度增加的方向进行，以保证精度不降低。如 int 型和 long 型混合运算时，先把 int 型数据转换成 long 型后再运算。
- 3) 所有浮点运算都是以双精度进行的，即使仅仅含有 float 型单精度数参加运算，也会先转换为 double 型再运算。char 型和 short 型参加运算时，会先转换为 int 型。
- 4) 在赋值语句中，两侧的数据类型不同时，右侧变量的类型先转换为左侧变量类型再赋值。

```
int S = 5 * 5 * 3.14159; //运行后 S 的值为 78
```

执行时，5 和 3.14159 都被转换为 double 型计算，结果也为 double 型 78.53975，但变量 S 为整型，故最终结果舍去了小数部分。

强制转换是通过类型转换运算来实现的，其一般形式为：

```
(类型说明符)(表达式) //把表达式的运算结果转换成类型说明符所示的类型
```

例如：

```
printf ("%d", (int)3.14159); //运行后屏幕显示为 3
```

执行时，先将 float 型 3.14159 强制转换为 int 型 3，然后再在屏幕显示。

强制转换的类型说明符和表达式都必须加括号（单个变量或数据时可以不加括号）。如果把 (int)(a+b) 写成 (int)(a)+b，则含义就变为先将变量 a 转换成 int 型后，再与变量 b 相加。

无论是强制转换还是自动转换，都只是为了满足本次运算的需求而对变量的数据类型进行的临时性转换，并不会改变变量声明时对该变量所定义的类型。

C语言中变量类型的自动转换和强制转换实例分别如代码清单 1 和代码清单 2 所示。

代码清单 1：C语言变量类型自动转换示例

```
1. #include<stdio.h>
2. #include<stdlib.h>
3. int main()
4. {
5.     float PI = 3.14159; //定义 PI 为 float 型并赋值为 3.14159
6.     int S, C, r = 5; //定义 S、C、r 为 int 型
7.     S = r * r * PI; //PI 以 double 型参与计算，结果也为 double 型，因 S 为 int 型，故结果
        被自动转换为 int 型赋值给 S
8.     C = 2 * r * PI; //PI 以 double 型参与计算，结果也为 double 型，因 C 为 int 型，故结果
        被自动转换为 int 型赋值给 C
9.     printf ("\nr = 5的圆面积为: %d\n", S);
```

```
10. printf("r = 5的圆周长为: %d\n",C);
11. system("pause");
12. return 0;
13. }
```

运行结果为:

```
r = 5的圆面积为: 78
```

```
r = 5的圆周长为: 31
```

代码清单 2: C语言变量类型强制转换示例

```
1. #include<stdio.h>
2. #include<stdlib.h>
3. int main()
4. {
5.     float PI = 3.14159; //定义 PI 为 float 型并赋值为 3.14159
6.     int S, C, r = 5; //定义 S、C、r 为 int 型
7.     S = r * r * (int)PI; //PI 被强制转换为 int 型参与面积计算
8.     C = 2 * r * (int)PI; //PI 被强制转换为 int 型参与周长计算
9.     printf("\nr = 5的圆面积为: %d\n", S);
10.    printf("r = 5的圆周长为: %d\n", C);
11.    system("pause");
12.    return 0;
13. }
```

运行结果为:

```
r = 5的圆面积为: 75
```

```
r = 5的圆周长为: 30
```

来自 <<https://www.54benniao.com/a/40.html>>