

1.16, enum 关键字

很多初学者对枚举(enum)感到迷惑, 或者认为没什么用, 其实枚举(enum)是个很有用的数据类型。

1.16.1, 枚举类型的使用方法

一般的定义方式如下:

```
enum enum_type_name
{
    ENUM_CONST_1,
    ENUM_CONST_2,
    ...
    ENUM_CONST_n
} enum_variable_name;
```

注意: enum_type_name 是自定义的一种数据类型名, 而 enum_variable_name 为 enum_type_name 类型的一个变量, 也就是我们平时常说的枚举变量。实际上 enum_type_name 类型是对一个变量取值范围的限定, 而花括号内是它的取值范围, 即 enum_type_name 类型的变量 enum_variable_name 只能取值为花括号内的任何一个值, 如果赋给该类型变量的值不在列表中, 则会报错或者警告。ENUM_CONST_1、ENUM_CONST_2、...、ENUM_CONST_n, 这些成员都是常量, 也就是我们平时所说的枚举常量(常量一般用大写)。enum 变量类型还可以给其中的常量符号赋值, 如果不赋值则会从被赋初值的那个常量开始依次加 1, 如果都没有赋值, 它们的值从 0 开始依次递增 1。如分别用一个常数表示不同颜色:

```
enum Color
{
    GREEN = 1,
    RED,
    BLUE,
    GREEN_RED = 10,
    GREEN_BLUE
}ColorVal;
```

其中各常量名代表的数值分别为:

```
GREEN = 1
RED = 2
BLUE = 3
GREEN_RED = 10
GREEN_BLUE = 11
```

1.16.2, 枚举与#define 宏的区别

下面再看看枚举与#define 宏的区别：

1), #define 宏常量是在预编译阶段进行简单替换。枚举常量则是在编译的时候确定其值。

2), 一般在编译器里, 可以调试枚举常量, 但是不能调试宏常量。

3), 枚举可以一次定义大量相关的常量, 而#define 宏一次只能定义一个。

留两个问题：

A), 枚举能做到事, #define 宏能不能都做到? 如果能, 那为什么还需要枚举?

B), sizeof (ColorVal) 的值为多少? 为什么?

枚举类型的目的是为了程序的提高可读性和可维护性。如果要处理颜色, 使用red和blue比使用0和1更直观。注意, 枚举类型只能在内部使用。如果要输入color中orange的值, 只能输入1, 而不是单词orange。或者, 让程序先读入字符串"orange", 再将其转换为orange代表的值。

因为枚举类型是整数类型, 所以可以在表达式中以使用整数变量的方式使用enum变量。它们用在case语句中很方便。