

# 嵌入式C语言之- 关系运算符

讲师：叶大鹏

助力你成为优秀的电子工程师！



# 整体介绍

- 关系运算符包括

$<$ 、 $<=$ 、 $>=$ 、 $>$ 、 $==$ 、 $!=$

- 关系运算实质上是比较运算，关系运算符组成的表达式，称为关系表达式：

$a < b$

- 关系表达式的结果是bool类型，要么是0 (false)，要么是1(true)，应用于条件判断语句和循环控制语句中；

# 整体介绍

```
uint8_t a = 10;  
uint8_t b = 9;  
printf("a > b = %d \n", a > b);  
printf("a >= b = %d \n", a >= b);  
printf("a < b = %d \n", a < b);  
printf("a <= b = %d \n", a <= b);  
printf("a == b = %d \n", a == b);  
printf("a != b = %d \n", a != b);
```

- 输出结果:

```
a > b = 1  
a >= b = 1  
a < b = 0  
a <= b = 0  
a == b = 0  
a != b = 1
```

# 运算符优先级

运算符（优先级从上往下）	运算符说明及应用场景	结合性
() [] -> .	括号（函数等），数组，结构体指针变量的成员访问，普通结构体变量的成员访问	由左向右
! ~ ++ -- + -	逻辑非，按位取反，自增1，自减1，正号，负号	由右向左
* & (类型) sizeof	间接，取地址，强制类型转换，求占用空间大小	
* / %	乘，除，取模	由左向右
+ -	加，减	由左向右
<< >>	左移，右移	由左向右

# 运算符优先级

运算符（优先级从上往下）	运算符说明及应用场景	结合性
< <= >= >	是否小于, 是否小于等于, 是否大于等于, 是否大于	由左向右
== !=	是否等于, 是否不等于	由左向右
&	按位与	由左向右
^	按位异或	由左向右
	按位或	由左向右
&&	逻辑与	由左向右
	逻辑或	由左向右
?:	条件	由右向左
= += -= *= /= %= &= ^=  = <<= >>=	各种赋值运算符	由右向左
,	逗号（顺序）	由左向右

赋值运算符优先级 < 关系运算符优先级 < 算数运算符优先级

# 应用案例

```
uint8_t a = 10;
uint8_t b = 10;
if (a == b)
{
    printf("a equal to b \n");
}
```

- 输出结果:

```
a equal to b
```

# 编程规范

1. 对于二元运算符，建议运算符前后加空格：

```
uint8_t a = 10;
```

```
if (a > b)
```

2. 如果要判断变量数值所在区间，下面写法是错误的，但是初学者经常容易和数学表达式混淆：

```
if (8 < a < 10)
```

正确书写为：

```
if (a > 8 && a < 10)
```

# 编程规范

## 3. 工程师经常误将 == 写成 =, 导致逻辑执行错误

```
uint8_t a = 10;
```

```
if (a = 9) //a = 9转换为bool类型结果是1, 所以会执行printf语句
```

```
{
```

```
    printf("a equal to b \n");
```

```
}
```

在工程实践中, 如果关系运算符右侧为常量数值, 可以将常量放在左侧, 变量放在右侧:

```
uint8_t a = 10;
```

```
if (9 == a) //当误写为9 = a时, 编译会出错
```

```
{
```

```
    printf("a equal to b \n");
```

```
}
```

```
main.c(51): error: #137: expression must be a modifiable lvalue  
if (9 = a) //当误写为9 = a时, 编译会出错
```



**THANK YOU!**