

# 操作/运算符的优先级与结合律

优先级	运算符	结合律
从 高 到 低 排 列	() [] -> .	从左至右
	! ~ ++ -- (类型) sizeof	从右至左
	<span style="border: 1px solid red; padding: 2px;">+ -</span> * &	
	* / %	从左至右
	<span style="border: 1px solid red; padding: 2px;">+ -</span>	从左至右
	<< >>	从左至右
	< <= > >=	从左至右
	== !=	从左至右
	&	从左至右
	^	从左至右
		从左至右
	&&	从左至右
		从 <span style="color: red;">左到右</span>
	?:	从右至左
	= += -= *= /= %= &= ^=  = <<= >>=	从左至右

## 一.关系运算:

### 一,是什么?

“ == ” “ > ” “ < ” “ >= ” “ <= ” “ != ”

### 二,有什么用?

对数据进行比较,看是否成立.

### 三,怎么用?

把运算符放在两个要比较的数据中间就可以.条件成立=真(1),条件不成立=假(0).

### 四,什么时候用?

当需要对两个数据进行比较.

### 五,有什么特点?

1.关系运算符运算结果是一个int值.0或1.

2.优先级:>,<,>=,<=,优先级相等, 但大于==和!=.同时==与!=优先级相等.

3.如果优先级相同的两个运算,先后顺序看结合律(结合律指要么从左往右运算,要么从右往左运算),关系运算符结合律是从左往右,即:先看优先级,再看结合律.如下:

int b=1!=2>3<4; 等效于:2>3为假=0. 0<4为真=1, 1!=1为假=0.所以最终结果=0;

```

1. //下面式子用到>,<,>=,<=,四种平级的运算方式,顺序位从左到右。
2. int a =!(1>2<3<=1>=0);
3. printf("%d\n",a);
4. //下面的式子用到==和!=这两种运算方式,
5. int b =1==2!=2;
6. printf("%d\n",b);
7. //六种方法都用到,系统按照优先级进行运算
8. int c =3>5==5!=2<3>=4<=2;
9. printf("%d\n",c);
10. //右边为真,所以d=1
11. int d =1>2||1;
12. printf("%d\n",d);
13. //左边为假,所以d1=1
14. int d1 =1>2&&1;
15. printf("%d\n",d1);
16. //1不大于4,所以为假,有!所以反转为真,d2=1
17. int d2 =!1>4;
18. printf("%d\n",d2);

```

## 二.逻辑运算:

### 一,是什么?

“ || ” 或 “ && ” 与(且) “ ! ” 非

### 二,有什么用?

对多个表达式进行并列真假判断.

### 三,怎么用?

||只要有一边的表达式为真,则结果为真1,两边都不成立,则结果为假0.而&&必须两边全部为真才为真,只要有一边的表达式为假,结果为假.如:1>2||2>1;结果为真(1),但如果1>2&&2>1.结果为假(0);!则是取当前结果的反面.真假调换.

### 四,什么时候用?

当需要对多个表达式进行并列真假判断.

### 五,有什么特点?

- 1.所有数据都有真假值.0为假,其它的数都为真(规定). 结合律:从左往右执行,
  2. 如果||表达式左边表达式已经为真,则右边不执行.而&&只要左边为假右边不执行.
  - 3.优先级顺序为: 小括号() > 负号(-) > ! > 算术运算符 > 关系运算符 > && > ||
- 如:1+2<3||!-3<4&&5>0 等效于(1+2)<3 || ( (!(-3) )<4 && (5>0) ) 结果是:1.

||(或)运算时,如果左边已经为真,则右边不再执行!!!

```

1. //由于e2左侧为真,右侧不进行运算,所以最终e=1,e2=1
2. int e =1;
3. int e2 =1||e++;
4. printf("%d\n%d\n",e,e2);
5. int q =1;
6. int q1 =2&&q++;
7. printf("%d\n%d\n",q,q1);
8. int w =1;
9. int w1 =1+2<1||w++;
10. int w2 =1+2<1||++w;
11. printf("%d\n%d\n%d\n",w,w1,w2);
12. int z =1;
13. int z1 =1-1<1&&z++;
14. int z2 =1-1<1&&++z;
15. printf("%d\n%d\n%d\n",z,z1,z2);

```

```

1. int a = 0;
2. int b = a=0||a&&a++;
3. printf("a=%d\n",a);
4. printf("b=%d\n",b);
5. return 0;

```

### 三.三目运算符:

#### 一,是什么?

<真假判断表达式1>?<表达式2>:<表达式3>;

#### 二,有什么用?

可以根据条件成立与否执行不同的表达式,返回不同的值,简化代码.

#### 三,怎么用?

先求表达式1的值,如果为真,则执行表达式2,并返回表达式2的结果;如果表达式1的值为假,则执行表达式3,并返回表达式3的结果.如:int a=1>2?3:4; 结果为a=4.

#### 四,什么时候用?

当需要根据条件执行不同的表达式时.

#### 五,有什么特点?

1.三个位置为表达式.不仅仅是数值.但必须有运算结果.不能定义变量.

如: 1+2>4?2+2:3+4; 这个运算过程为:

1+2=3, 3>4为假.则运行3+4这个表达式.所以最终结果为7.

```

#include <stdio.h>
int main(){
    int a=1>2;
    a=1>2?3:4;
    printf("%d\n",a);
    return 0;
}

```

```
int a=1>2?3+2:int b=3;  
printf("%d\n",a);  
return 0;
```

← 不能跟变量