

简单信息的表达和运算

# 特殊运算符

赵英良



# 内容

---

- ▶ 赋值运算符( = )
- ▶ 字节数运算符( sizeof() )
- ▶ 条件运算符 (问号, ? : )
- ▶ 逗号运算符( , )

# 1.赋值运算符

---

- ▶ 等号"="是赋值运算符

- ▶ 格式

    <变量> = <表达式>

- ▶ 作用

- (1)将右边表达式的值存入左边的变量中

    int a=3,b=5,c;

    c=a+b; //赋值语句

# 等号作用

---

(2)作为表达式，它的值就是右边<表达式>的值

```
int a=3,b=5,c;
```

```
cout<<(c=a+b)<<endl; //显示8
```

```
cout<<c<<endl;      //显示8
```

# 优先级和结合顺序

- ▶ 优先级较低，低于逻辑运算，和复合算术运算和复合位运算是同级的
- ▶ 结合顺序是从右向左。

```
int a=1,b=5,x,y,z;
```

```
    x=  y=  z = a+b    ;
```

```
//  x=( y= ( z = a+b ) );
```

```
z=x+y=a+b; ?
```



```
z=a+b;
```

```
y=z;
```

```
x=y;
```

[Error] lvalue required as left operand of assignment

# 左值和右值

## ▶ 左值lvalue

- 能够出现在等号左边的式子
- 能够被赋值的量
- 变量、数组元素、对象等

## ▶ 右值rvalue

- 有值的表达式

`z=x+y=a+b; // 不正确`

`z=x+(y=a+b); // 正确`

```
cout<<(c=a+b)<<endl; // 不推荐
x= y=z=a+b ;    // 不推荐
z=x+(y=a+b);    // 不推荐
```

## 2. 字节数运算符

---

### ▶ 字节数运算符sizeof()

### ▶ 格式

- sizeof(<常量、变量、数组、对象>) // 表达式的特例、
- sizeof(<类型>) // 用法象函数，叫运算符
- sizeof(<表达式>)

### ▶ 作用

- 求某类型的数据或表达式的值占的字节数

### ▶ 举例

- `cout<<sizeof(int)<<" "<<sizeof(x)<<endl;`

### 3. 条件运算符

---

- ▶ 条件运算符、问号运算符 ? :
- ▶ 三目运算
- ▶ 格式  
    <条件> ? <表达式1> : <表达式2>



# 条件运算符举例

---

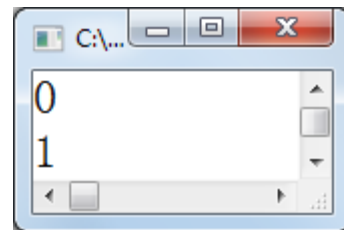
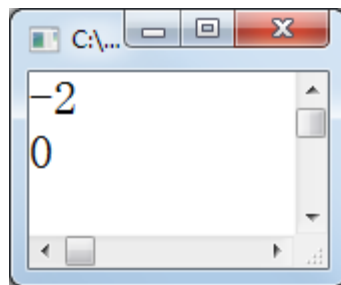
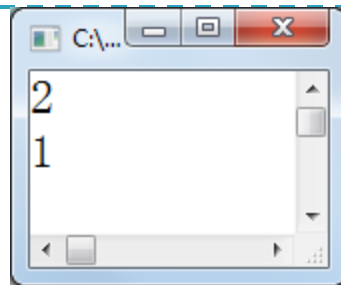
- ▶ 有这样的函数

$$y(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

`y = ( t >= 0 ? 1 : 0 );`

# 求分段函数值的完整程序

```
#include<iostream>
using namespace std;
int main()
{
    double t,y;
    cin>>t;
    y=( t>=0 ? 1:0 );
    cout<<y<<endl;
    return 0;
}
```



# 使用条件运算符注意

---

- ▶ <表达式1> 和 <表达式2> 的值应具有相同的类型，或能转换为相同的类型

`y=( x>1 ? 1 : "error" ); //错误`

## 4.逗号运算符

---

- ▶ 逗号","是一个运算符
- ▶ 多个表达式用逗号隔开，构成逗号表达式
- ▶ 格式  
    <表达式1> , <表达式2> , ... , <表达式n>

### 作用

从左向右依次计算n个表达式的值  
整个表达式的值为<表达式n>的值

# 逗号运算符举例

```
int a=1,b=2,c=3;
```

```
int y;
```

```
y=(a=a+1,b=b+2,c=c+3);
```

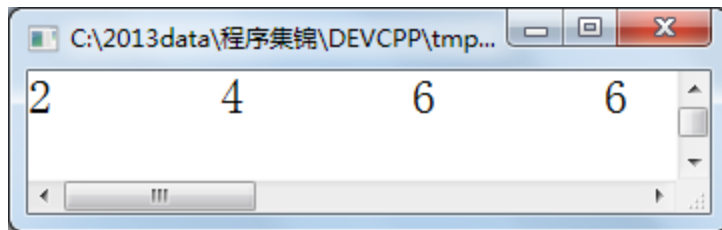
相当于

```
a=a+1;
```

```
b=b+2;
```

```
c=c+3;
```

```
y=c;
```



# 5. 优先级和结合顺序

高



低

**sizeof(), 按位反 逻辑非! , 正、负号**

算术 乘、除、求余 \* / %

算术加、减 + -

按位左移<<、右移>>

关系: 大于小于 >, >=, <, <=

关系: 等于不等于 ==, !=

按位与

按位异或

按位或

逻辑 与 &&

逻辑或 ||

**条件 ? :**

**赋值=、复合运算**

**逗号 ,**

从右向左1

从右向左2

从右向左3

# 总结

- ▶ 赋值运算 =
  - 起赋值作用
  - 等于右边表达式的值
  - 左边必须是变量
- ▶ 字节数运算符：sizeof(<表达式、类型>)
- ▶ 逗号：<表达式1>,<表达式2>, ..., <表达式n>  
y=( <表达式1>,<表达式2>, ..., <表达式n> )
- ▶ 条件运算符
  - y=(<条件> ? <表达式1> : <表达式2>)

所有运算符都使用英文符号!!!