

C++语言基础

迂者 - 贺利坚

<http://blog.csdn.net/sxhelijian/>

<http://edu.csdn.net>





本节主题：

重载运算符的规则

C++的运算符

- (1)算术运算符： +(加)-(减) *(乘)/(除) %(整除求余)++(自加)--(自减)
- (2)关系运算符： > (大于) < (小于) == (等于) >= (大于或等于) <= (小于或等于) != (不等于)
- (3)逻辑运算符： &&(逻辑与) ||(逻辑或) !(逻辑非)
- (4)位运算符： <<(按位左移) >>(按位右移) &(按位与) |(按位或) ^(按位异或) ~(按位取反)
- (5)赋值运算符 (= 及其扩展赋值运算符)
- (6)条件运算符 (?:)
- (7)逗号运算符 (,)
- (8)指针运算符 (*)
- (9) 引用运算符和地址运算符 (&)
- (10)求字节数运算符 (sizeof)
- (11)强制类型转换运算符((类型) 或类型())
- (12) 成员运算符 (.)
- (13) 指向成员的运算符 (->)
- (14) 下标运算符 ([])
- (15) 其他 (如函数调用运算符 ())

优
先
级

结
合
性

重载运算符的规则

- ❏ 不允许创造新的运算符，只能对已有的C++运算符进行重载
- ❏ C++允许重载的运算符：不能重载的运算符只有5个
 - ❏ 成员运算符 (.)、成员指针访问运算符 (. *)、域运算符 (::)、sizeof、条件运算符 (? :)
- ❏ 重载不能改变运算符运算对象(即操作数)的个数
- ❏ 重载不能改变运算符的优先级别
- ❏ 重载不能改变运算符的结合性
- ❏ 重载运算符的函数不能有默认的参数
- ❏ 重载的运算符必须和用户定义的自定义类型的对象一起使用，参数至少有一个是类对象或其引用

```
Complex Complex::operator + (Complex &c2)
```

```
{return Complex(real+c2.real, imag+c2.imag);}
```

```
Complex operator + (int &a, Complex &c)
```

```
{return Complex(a+c.real, c.imag);} //用友元函数
```

重载运算符的功能应类似该运算符作用于标准类型数据时所实现的功能

- 以 += 运算为例

- int i=3;

- 表达式 i+=2的功能

 - i 的值变为5

 - i+=2是一个表达式，其值为改变后的i，即5

- 例：cout<<(5+(i+=2))<<endl; 的输出

 - 10

- 问题：

 - operator+=(Complex &c2)如何实现？

 - Complex c1(3, 4), c2(5, -10);

 - c1+=c2;

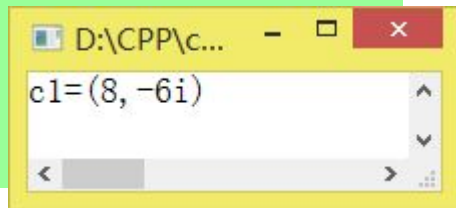
实现operator+=(Complex &c2)?

```
class Complex{
public:
    Complex( ) {
        real=0;
        imag=0;
    }
    Complex(double r,double i) {
        real=r;
        imag=i;
    }
    Complex & operator += (const Complex & c);
    void display( );
private:
    double real;
    double imag;
};
```

```
Complex & Complex::operator += (const Complex & c){
    real += c.real; //this->real += c.real;
    imag += c.imag; //this->imag += c.imag;
    return *this;
}

void Complex::display( ){
    cout<<"("<<real<<","<<imag<<"i)"<<endl;
}

int main( ){
    Complex c1(3,4),c2(5,-10);
    c1+=c2; //相当于c1.operator+=(c2);
    cout<<"c1=";
    c1.display();
    return 0;
}
```



THANKS

本课程由 迂者-贺利坚 提供

CSDN网站：www.csdn.net
企业服务：<http://ems.csdn.net/>
人才服务：<http://job.csdn.net/>
CTO俱乐部：<http://cto.csdn.net/>
高校俱乐部：<http://student.csdn.net/>
程序员杂志：<http://programmer.csdn.net/>

CODE平台：<https://code.csdn.net/>
项目外包：<http://www.csto.com/>
CSDN博客：<http://blog.csdn.net/>
CSDN论坛：<http://bbs.csdn.net/>
CSDN下载：<http://download.csdn.net/>

