

# 运算符重载规则小结

- 1) 重载不能改变运算符操作数的个数、优先级别、结合性；
- 2) 重载运算符的函数不能有默认的参数；
- 3) 重载运算符必须和用户定义的自定义类型的对象一起使用，其参数至少应有一个是类对象(或类对象的引用)；
- 4) 用于类对象的运算符必须重载，但有两个例外，运算符“=”和“&”不必用户重载；
- 5) 重载运算符的功能类似于该运算符作用于标准类型数据时所实现的功能；
- 6) 运算符重载函数可以是类的成员函数或类的友元函数；
- 7) 友元的关系是单向的而不是双向的；友元的关系不能传递。

## 第4次作业（必做题）

### 1. 完成类中友元函数（申明形式如红线）的定义形式

```

1. class Date {
2.     int d, m, y;
3. public:
4.     Date(int dd=0, int mm=0, int yy=0);
5.     void addDay();
6.     friend Date operator ++(Date& D);           //声明前置++
7.     friend Date operator ++(Date& D, int);      //声明后置++
8.     void Print() { cout << d << "," << m << ","
9.                     << y << endl; }
10.    ...
11. };
    
```

2. C++在C基础上增加一个新的数据类型—字符串类型（string），其实本质上它是C++在标准库中申明了一个字符类。请完成string类申明，并采用类的成员函数重载==，>，<，+，= 5种运算符。然后，在5子棋盘前3次需求基础上，使用上述某些运算符，通过比较玩家名称（英文表示）来决出谁先手（名字大者先下棋）。

■ 也可自己设计需求，将上述5种运算符都应用在五子棋程序中。

## 第4次作业（选做题）

第1道题：阅读附录资料一友元，思考如下问题，分析为什么？

- 1) 友元函数是否可以申明在类中的private?;
- 2) 类中申明的友元函数是否有this指针? ;
- 3) 如何使得友元函数获得操作对象? ;
- 4) 为什么说，使用友元函数破坏了类的封装性，而使用友元类增强了类的封装性？运用类封装性和友元类的原理来分析？