

实验七 多态：运算符的重载

1 实验目的

学习面向对象程序设计中多态机制的实现方法之一多态性，包括以下内容：

(1) 学习如何定义运算符的重载；

(2) 学习如何使用重载的运算符。

2 实验内容

2.1 改进《实验三 面向对象初步》中的日期类

在《实验三 面向对象初步》中，设计了日期类 `Date`。本次实验将 `daysTo` 函数重载为减号运算符，使得 `t1 - t2` 的值为从日期 `t2` 到日期 `t1` 的天数。如果 `t1` 在 `t2` 之前，则为负数，即计算 `t1` 比 `t2` 晚多少天。

重载大于运算符，如果 `t1` 在 `t2` 之后，则表达式 `t1 > t2` 为真；

重载小于运算符，如果 `t1` 在 `t2` 之前，则表达式 `t1 < t2` 为真。

2.2 使用改进后的日期类，修改《实验四 对象作为数据成员》中的雇员类 (`Employee`)

(1) 修改 `getDaysWorked`，使其调用日期类中重载的减号运算符。

(2) 添加一个静态成员函数，调用日期类中重载的“>”运算符，通过比较雇佣日期，在雇员对象数组中，找出工作年限最长的雇员。该函数的说明如下：

```
/*  
*参数employees[]是雇员对象的数组；n是雇员对象数组的元素个数  
*返回值：工作年限最长的雇员对象的引用  
*/  
static const Employee& getMostFaith(const Employee employees[], int n);
```

2.3 测试 `getMostFaith` 静态成员函数

编写一个主函数，生成含有 5 个雇员对象的数组，然后调用 `Employee` 类的 `getMostFaith` 静态成员函数，找出工作年限最长的雇员，打印其信息以及工作了多少天。