实验1：C++面向对象程序设计

--Mydate类的功能实现

1. 实验目的
2. 熟悉并掌握类的声明和实现分离的方法；
3. 熟悉C++类内数据与函数的使用方法和思考逻辑；
4. 深入理解和掌握类的数据域初始化的重要性和方法；
5. 熟悉运算符重载的基本方法；
6. 掌握C++程序调试排错的基本技能。
7. 实验环境
8. 支持C++11标准的编译执行环境
9. 实验结果测试平台Ubuntu 18.04 LTS、G++
10. 实验内容

设计并实现一个日期类 MyDate

1. 实验步骤

1、设计一个日期类 MyDate (1)

(1) 提 供 转 换 构 造 函 数 , 把 整 数 转 成 日 期 , 规 则 为 整 数 代 表 自 1970-01-01 00:00:00 UTC 起经过的秒数。

(2) 提供构造函数,接受三个整数分别代表年、月、日三个字段。

(3) 提供成员函数 bool equals(const MyDate&) const当二个对象年月日均相等 时 返 回 true , 否 则 返 回 false 。

提 供 成 员 函 数 bool lessThan(const MyDate&) const,当此对象的日期早于形参日期时返回 true,否则返回 false。

类似的,提供以下的成员函数:

bool lessThanOrEquals(const MyDate&) const

bool greaterThan(const MyDate&) const

bool greaterThanOrEquals(const MyDate&) const

(4) 提供成员函数 MyDate add(int) const,意味着返回此对象增加相应天数的日期。类似的,提供以下成员函数 MyDate subtract(int) const。

(5) 提供成员函数 int subtract(MyDate) const,返回值为此对象与形参之间的距离,当此对象比形参晚时返回正整数。

(6) 提供成员函数 string toString() const,输出符合 ISO8601 标准。

2、完善日期类 MyDate

(1) 要求支持公元前9999年1月1日至公元9999年12月31日，公元前年份的表 达方式遵守 ISO8601 标准:公元 1 年为 1，公元前 1 年为 0，公元前 2 年为-1， 以此类推。

(2) 在公元1752年9月2日(周三)之前使用儒略历，其后一天起使用格里历，调 整为 1752 年 9 月 14 日。星期则连续，2 日为周三，14 日为周四。

(3) 以下运算符重载为成员函数:<, <=, >, >=, ==。功能与 lessThan, lessThanOrEquals, greaterThan, greaterThanOrEquals，equals 相同。可以直接调用上述函数。

(4) 以下运算符重载为成员函数:重载二元+，-。功能:可以对一个日期加/减一个 整数，整数的含义为天数。功能与 add(int days), subtract(int days)相同。可以直 接调用上述函数。只需要考虑形如 myDate+5 这种情况，不考虑 5+myDate。

(5) 以下运算符重载为成员函数:重载二元-。功能:二个日期类之间可以做减法， 结果为相差的天数，功能与 subtract(const MyDate&)相同。

(6) 重载<<以便使用cout输出。功能是将toString()的返回值直接输出即可。输出格 式为 ISO8601 标准，即形式为:2017-01-01;公元前 1 年记为 0，形式为:0000-01-01; 公元前 2 年记为-1，形式为:-0001-01-01。

(7) 增加函数 int getWeekday()，获得当前对象的星期数。返回值的含义:0 代表周 日，1 代表周一......6 代表周六。已知 1970 年 1 月 1 日是星期四。

(8) 构造函数要进行输入合法性判断，如果输入不合法，则以 1970-01-01 存储之。

1. 实验方式

每位同学独立上机编程实验，实验指导教师现场指导。

1. 参考内容

1）《C++程序设计（原书第3版）》，[美] Y.Daniel Liang 著；刘晓光，李忠伟，任明明 等 译，机械工业出版社，2015.01

2）《C++ Primer Plus（第6版 中文版）》，[美] Stephen Prata 著；张海龙，袁国忠 译，人民邮电出版社，2012.06

1. 相关软件下载

无

1. 实验报告要求

项目代码