

OOP上机实验(5)

(Version: 0.75 Date: 2024/4/24 Author: wangxp@fudan.edu.cn)

1 实验目标

通过本实验，拟达到如下目标

- 编写完整的类，熟悉类的构造函数、析构函数、拷贝构造函数；熟悉操作符重载（含输入输出操作符）；
- 熟悉成员函数与非成员函数的区别；熟悉友员函数；熟悉类的静态属性；
- 理解完整的类代码包含成员属性、成员函数、静态属性、静态成员函数；以及非成员函数，特别是输入输出重载操作符，不能作为成员的两元或多元操作数和关系操作符；
- 理解基于测试的程序开发过程。

2 实验内容

2.1 编写完整的类

ISO是一个标准化组织,致力于多个方面的标准开发。本次实验的内容主要是练习ISO制定的标准week date[1]，下面的代码已经处理并进行了规范。

从标准中，我们抽取了year这个类，其接口定义在year.h文件中。本次实验提供的文件包括：

- year.h;
- year.cpp
- year_test.cpp
- year_main.cpp
- CMakeLists.txt

其中year.h和test_year.cpp无需修改，year.cpp需要补充，最后仅需提交year.cpp，且需要补充的内容在//TODO 标记下。

关于year.h有如下说明：

- year类是一个纯数据类，不包含资源，因此无需定义拷贝构造函数、拷贝赋值操作符、移动构造函数、移动赋值操作符、析构函数；相反，只定义了默认的构造函数，以及接受一个 int 值的构造函数；
- year类定义了两个静态函数 min/max，限制所表示的年分必须在min和max的范围之间，按照 C++的惯例表示的范围为[min, max)，即左闭右开区间；
- year类定义了一元操作符+/-；二元操作符+/-/=；自增(前缀、后缀)操作符；自减(前缀、后缀)；--这些操作符统统定义为成员函数；
- 6个关系操作符，两元+/-操作符(重载了 year/years 类型)，输出操作符<<，以及字符常量(literals)的重载操作符[2];--这些操作符定义为非成员函数。
- 所有需要补写的内容在标记"//TODO{1,2,3,4}"后面有说明。

关于year_test.cpp有如下说明：

- 使用googletest框架;
- 在test_year.cpp中，针对year类的构建、比较操作、算术运算等操作符进行了调用，并确认这些正确。
- 如果补充的代码功能正确，那么将year.cpp和year_test.cpp编译出来的可执行文件，在运行时将会输出"PASSED"; 否则看到错误信息。

关于CMakeLists.txt有如下说明：

- 编译lab5需要使用了googletest框架;

- 在CMakeLists.txt中通过FetchContent_*自动下载相关的库(第12~17行);
- 在第20行添加了gtest库: GTest::gtest GTest::gtest_main GTest::gmock GTest::gmock_main

2.2 配置googletest

实验中给出了测试代码year_test.cpp, 其中使用了goolge的单元测试框架googletest[3]。

按照上面的设置, lab5自带的CMakeLists.txt可以正常工作; 在运行之后, 如果补充的代码正确, 应该输出PASSED。

2.3 提交要求

提交内容: 只要提交year.cpp

3 参考资料

1. ISO week date. 参考 https://en.wikipedia.org/wiki/ISO_week_date
2. 自定义的字符常量. 参考 <https://blog.csdn.net/K346K346/article/details/85322227>
3. googletest测试框架. 参考 <https://github.com/google/googletest>