**Lab 7**
**C++ Designing and Implementing Classes**

**Purpose**: Understand the concept of classes as types and objects as instances of a class. Implement class methods including constructors, accessors, and mutators.

| **Grading:** | | |
|---|---|---|
| | Documentation & Style (indentation, spacing, etc) | 5 points |
| | Makefile | 5 points |
| | Test Program (thoroughly tests all Date class member functions) | 10 points |
| | Date Class | 20 points |
| | Total possible | 40 points |

---

**(Date Class)** This is essentially problem 16.15 on page 621 in the textbook.

Warning! Do not copy the code from chapter 17.

**Date.h (see GradeBook.h in the notes or on page 608 for an example)**

- Design and implement a Date class that includes data for month (int), day (int) and year (int).
- For each member function, the prototype should be placed in the header file.
- Provide a constructor with three parameters and use the parameter values to initialize the data members (in the order listed above).
- Use default parameters in the constructor prototype. Default month = 1. Default day = 1. Default year = 2016.
- Do not assign variables directly in the constructor. Call the mutator functions from within the constructor.
- Follow naming conventions. Use camel case for names of data members and member functions.
- Provide mutator “set” functions, for each data member. Provide accessor “get” functions, for each data member.
- Data members are private. Member functions are public.

**In Date.cpp (see GradeBook.cpp or Time.cpp on pg 638 for an example)**

- Write the member function definitions – the complete function, not just the prototype. Remember to use the binary scope resolution operator with the class name `Date::` in front of each member function name.

- Validate data as follows:
  month – must be between 1 and 12
  day – must be between 1 and the number of days in the month (you may ignore leap year)
  year – must be positive

  Throw an invalid_argument exception if the value passed to the mutator is not valid data.

  In your test program, use a try __ catch block to test each mutator.

**In DateTest.cpp**

Write a main function that will declare several Date objects. Write sufficient test calls to insure that all of the Date member functions work correctly.

**Makefile**

Create a makefile that will compile your programs, and link them together to produce an executable named DateTest.