COP3330 Programming Assignment 2

Date and day of week calculator for 2019

Objectives

* Implement a simple class with public and private members and multiple constructors.
* Gain a better understanding of the building and using of classes and objects.
* Practice problem solving using OOP.

Overview

You will implement a date and day of week calculator for the 2019 calendar year. The calculator repeatedly reads in three numbers (in a line) from the standard input that are interpreted as *month dayofmonth daysafter***,** calculates the dates in the year and days of week for the dates, and outputs the information. For example, input “1 1 31” is interpreted as the following: the first 1 means the 1st month in the year, January; the second 1 means the 1st day in January; and the 31 means 31 days after the date January 1, 2019 (we assume the year is 2019 to simplify the program), which is February 1, 2019. The program also calculates the days of week for each of the dates. More specifically, for input “1 1 31”, the calculator should produce the following output:

“31 days after Tuesday, January 1, 2019 is Friday, February 1, 2019. “

The first input number must be from 1 to 12 representing the 12 months in 2019, the second input number must be a day in the month (e.g. for 1-31 for January, 1-28 for February for leap year, and so forth). The third number is larger than or equal to 0. The program should report an error (and exit) if the input is incorrect. If a day is not in 2019, the program should output that. Following are a sample input file (redirect to be the standard input) and the corresponding output.

Input file:

1 1 20

1 1 31

2 1 0

1 1 32

4 5 0

2 1 28

1 1 59

6 10 100

7 20 300

12 20 2

Output:

20 days after Tuesday, January 1, 2019 is Monday, January 21, 2019.

31 days after Tuesday, January 1, 2019 is Friday, February 1, 2019.

0 days after Friday, February 1, 2019 is Friday, February 1, 2019.

32 days after Tuesday, January 1, 2019 is Saturday, February 2, 2019.

0 days after Friday, April 5, 2019 is Friday, April 5, 2019.

28 days after Friday, February 1, 2019 is Friday, March 1, 2019.

59 days after Tuesday, January 1, 2019 is Friday, March 1, 2019.

100 days after Monday, June 10, 2019 is Wednesday, September 18, 2019.

300 days after Saturday, July 20, 2019 is a date not in 2019.

2 days after Friday, December 20, 2019 is Sunday, December 22, 2019.

Details

1. This project has two components. The first component is to implement a *date2019* class specified in the following. The second component is a driver program that uses the date2019 class to realize the calculator.
2. The date2019 class should have two private data members of the *int* type, **d** and **m** with **m** encoding the month (1 – January, 2-February, …) and **d** being the day in the month. For example to represent April 5, **d = 5**, **m = 4**. To represent a date not in 2019, m=-1, d=-1.
3. The date2019 class should have the following public functions:

date2019();

date2019(int dd);

date2019(int dd, int mm);

void setdate(int dd, int mm);

void print();

void plusday(int days);

1. The default constructor date2019() should set the default date of the object to be January 1, 2019.
2. The parameter for the constructor with 1 parameter date2019(int dd) ranges from 1 to 365 (the code should report error and exit if the parameter is not in the range) and is day of year for the date in 2019. The constructor converts day of year to month and day of month. For example the 70th day of the year is March 11: the constructor should make m =3 and d = 11 when date2019(70) is invoked.
3. For the two parameter constructor date2019(int dd, int mm), you just need to make sure that month (mm) and day (dd) are legitimate and set d = dd, and m = dd. If either the day or the month is not legitimate, the code should report error and exit.
4. Setdate is similar to the two parameter constructor.
5. The print() function calculates the day of week for the date2019 object and outputs to the standard output the date in the form of ‘dayofweek, month day, 2019’. For example, when the object has d=1, m=1, the print should output “Tuesday, January 1, 2019”. For a date not in 2019 (m=-1 and d =-1), this routine should output “a date not in 2019”.
6. The *plusday(int days)* function modifies the m and d for the new date, which is *days* after the current date.
7. You should declare the class in a file called proj2.h; and the implement of the class (class functions) in a file called proj2.cpp. You should have a statement ‘#include “proj2.h”’ in proj2.cpp.
8. Write a driver program called proj2\_driver.cpp that reads in lines of three numbers as specified earlier and generates the corresponding output as specified using the date2019 class. The driver is very small (about 20 lines of code) performing the basic IO.

Submission

The due time for this assignment is Feb 11 (Monday), 2019. 9.00 AM.

Your program files are proj2.h, proj2.cpp, proj2\_driver.cpp.