<u>**Lab 1:**</u> Basic Math   <u>**Due:**</u> 9/4/2020

In this lab you will practice basic math operations in C++. First you will review a program that demonstrates the use of basic math, then you will write your own C++ code that uses basic math.

Example Program

Our example program is called the Dateable Equation. There is a common rule of thumb that you should not date a person who is younger than your age divided by two, plus seven. The Dateable Equation is:

DateableAge = (YourAge / 2) + 7

The **DateableEquation.cpp** program calculates the lowest dateable age for a given person's age. Download this program and create a project for it in Visual C++. Read and understand the code and run the program. Try changing the given age to see the dateable age for different ages.

Notice that the Dateable Age is always an integer. If your age is an odd number, dividing it by two should result in a decimal number, so why is the Dateable Age always an integer? The reason is that in C++ when you divide two integers the result is an integer (the decimal part is truncated). Even if the dateable variable was of type double, the result of the division would not have a decimal part. In order to have a decimal result from division, at least one of the numbers involved in the division must be a floating point number (such as type double) and if we are saving the result into a variable, the variable must be a floating point type. In this case an integer is the correct result because we always deal with ages as whole numbers. But in other cases integer division will give an incorrect result. In another lab we will look at an example where integer division results in an incorrect result.

Your Program

In this program you will calculate your magic number. Your magic number is calculated by multiplying the month of your birth by the year of your birth, and taking the modulus of that result and the day of your birth. The equation is:
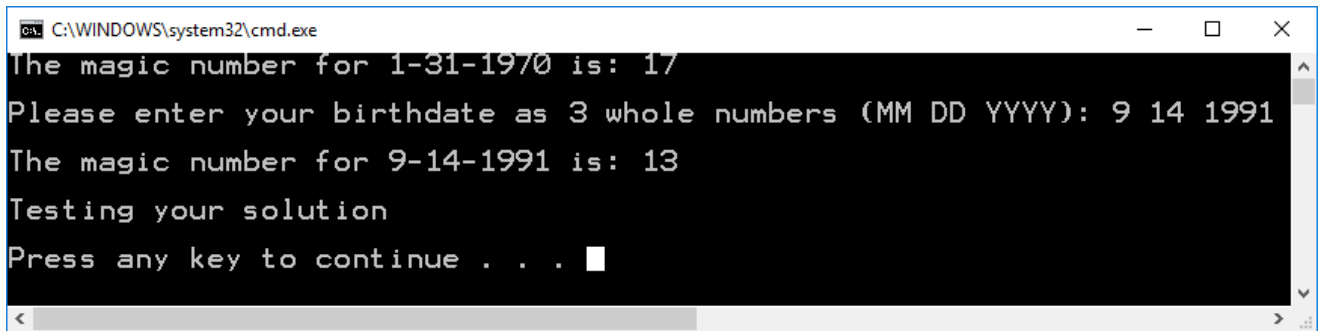
*MagicNumber = (month * year) % day*

Remove **DateableEquation.cpp** from your project and **add the existing item** named **lab01_FML.cpp** (replace F, M. and L with the initials of your first, middle (if any), and last names) in your project.

Implement the algorithm provided in the source code as comments.

Note:
- Do NOT remove or modify the statements that I use to test certain things in your program.
- Notice how we use descriptive variable names so it is easy to tell the meaning of the variable.
- You must use the variables above for the calculation and printing. Do NOT use any literal numbers in your calculation
- Run my sample solution to know how your program must behave. Pay attention to the input and the output formats. Your solution must behave exactly like mine.
- Carefully analyze the following figure and use it as a reference to ensure you do the right things.

```
C:\WINDOWS\system32\cmd.exe                              —    □    ×
The magic number for 1-31-1970 is: 17

Please enter your birthdate as 3 whole numbers (MM DD YYYY): 9 14 1991

The magic number for 9-14-1991 is: 13

Testing your solution

Press any key to continue . . . ■
```

- Test and compare your solution with mine for different birthdates to ensure they always produce the same outputs.

Include at the top of the program the comments shown below with your information (name, class and section number, etc.)

```
/////////////////////////////////////////////////////////////////
//
// Name: <Put your name here>
// Date: <Today's date>
// Class: <Your class number and section number, like: CSCI 1370.02>
// Semester: <This semester, like: Spring 2012>
// CSCI/CMPE 1370 Instructor: <Your lecture instructor's name>
//
// Program Name: Magic Number
// Program Description: Calculate a person's magic number based on their birthday.
//
/////////////////////////////////////////////////////////////////
```

       When done, submit your solution through Blackboard using the "Assignments" tool *(see thread "How to submit an assignment" in Discussions please).* ***Do Not email it.***

The following is the basic criteria to be used to grade your submission:

You start with 100 points and then lose points as you don't do something that is required.

-5: Missing comments at the top of the program
-50: Program does not compile;
-10: Wrong variable names;
-10: Wrong variable types
-5: No comments or too few comments in source code;
-10: Wrong input format;
-10: Wrong output format;
-20: Program does not implement the algorithm provided;
-5: Incorrect file name (lab1_FML.cpp);
-10: Late;

Important: more points may be lost for other reasons not specified here.