**CIS 246 – Spring 2020**

|  |  |
| --- | --- |
| **Program:** | **1** |
| **Points:** | **20** |
| **Chapter(s):** | **1, 2** |
| **File(s) to Submit:** | **Program1.cpp** |
| **Due:** | **February 5, 2020 11:59pm** |

**Description**

Write a C++ program that prompts the user for three numbers representing a month, day, and year, and then displays the date to the screen in the format mm/dd/yy.

**Requirements for the Program1 File**

Within a Dev-Cpp project, create a .cpp file called Program1.

Inside the main function, perform the following tasks:

1. Create three integer variables for the day, month, and year.
2. Prompt the user to type in the day.
3. Read the value typed and store it in the appropriate variable from step 1.
4. Prompt the user to type in the month.
5. Read the value typed and store it in the appropriate variable from step 1.
6. Prompt the user to type in the year.
7. Read the value typed and store it in the appropriate variable from step 1.
8. Display the output using the following format. You DO NOT have to use any special technique for formatting. Just send the appropriate values to stdout using one C++ statement.

The date is mm/dd/yy.

See the sample runs at the end of this document for the proper display.

You **DO NOT** have to check for valid months, days, or years. Just print what the user types.

**General Requirements**

For complete credit, you must:

1. **MEET ALL REQUIREMENTS ACCORDING TO THE INSTRUCTIONS** – Follow the instructions as written for completing this project, even if you [think you] know a “better” way to do something. I will deduct points for not following instructions as written.
2. **INCLUDE COMMENTS** – Include comments in your code. There must be a comment at the top of each source code file that includes your name, the program number, and a description of the class. There must be comments at each important step in your algorithm that describes that step.
3. **FOLLOW BEST PRACTICES** – Follow best practices in C++ programming, including, but not limited to, appropriate use of private/public, appropriate use of classes and/or header files, sets & gets, white space, alignment, meaningful variable names, naming conventions, using directives, etc. Points will be deducted for sloppy code that is hard to read, even if it works, so pay attention to these details.
4. **SUBMIT ALL FILES BEFORE THE DUE DATE** – Submit a .zip of ONLY source code files to the dropbox for this assignment on Canvas before the due date. Do not submit anything except .cpp and/or .h, within a zip. Do not submit .exe files. Do not submit project files from an IDE.

**Sample Runs**





