cisc1110, fall 2010, assignment for unit II, part 2

This is the second part of the assignment for unit II, simple data types. You are expected to complete the assignment in the C++ language and submit your ".cpp" files. You must complete and submit the assignment on or before the due date of **September 23**. This assignment is worth **2 points**.

Submission instructions:

- Submit your assignment to me via email: `sklar@sci.brooklyn.cuny.edu`
- Your email subject line should be: **CISC 1110 Lab II.2 submission**
- Attach your C++ (**magic.cpp**) file to your email.
- Make sure your name is in the body of the email message.
- Make sure your name is also included in the header comments at the top of your C++ file.

For this assignment, you will create a program called **magic.cpp**. There are **5 steps**, described below, that you should follow in order to develop your program. Make sure that for each step, you compile, run and test your program to make sure it works as described in the instructions for that step. You should make copies of the program along the way, after each step, so that you save the progress you make in your work—in case of disaster or you need to backtrack to a previous step that worked.

step 1 (0.4 points)

- Write a program that declares two **character** variables, named `first` and `last`.
- Store the value of your first initial in the variable `first`. For example, mine is `'E'`;
- Store the value of your last initial in the variable `last`. For example, mine is `'S'`;
- Output the values of `first` and `last`.
- Compile, run and test your program. Go back and fix it if it doesn't work properly.

step 2 (0.4 points)

- Modify your program as follows.
- Create an **integer** variable called `mon_num` that stores the *ones* digit of the number of your birth month. For example, if you were born in January, your birth month number is 1, so you would store 1 in `mon_num`. If you were born in July, you would store 7 in `mon_num`. If you were born in December, your birth month number is 12, so you would store 2 in `mon_num`.
- Output the value of your birth month number and label it as follows:

  ```
  cout << "Here is my birth month number: " << mon_num << endl;
  ```
- Compile, run and test your program. Go back and fix it if it doesn't work properly.

step 3 (0.4 points)

- Modify your program as follows.
- Create a **character** variable called `mon_char` that stores the *ones* digit of the number of your birth month *as a character*. For example, if you were born in January, then you would store `'1'` in `mon_char`; if you were born in December, then you would store `'2'` in `mon_char`;
- Output the value of your character birth month number and label it as follows:

  ```
  cout << "Here is the character version of my birth month number: " << mon_char << endl;
  ```

- Compile, run and test your program. Go back and fix it if it doesn't work properly.

step 4 (0.4 points)

- Modify your program as follows.
- Create 3 more **integer** variables and call them `first_code`, `last_code` and `mon_char_code`.
- Store in each of these the numeric ASCII character code of the corresponding character values, `first`, `last` and `mon_char`, respectively. Use the `(int)` conversion operator to tell the computer to use the numeric versions of these values. For example:

  ```
  first_code = (int)first;
  ```

- Output the 3 character code values in a user-friendly way! For example:

  ```
  cout << "The character code for my first initial is: " << first_code << endl;
  ```

- Compile, run and test your program. Go back and fix it if it doesn't work properly.

step 5 (0.4 points)

- Modify your program as follows.
- Create one more **integer** variable.
- Compute your "magic" number and store it in that variable, using the following formula:

$$magic\_number = (first\_code + last\_code + mon\_char\_code) * mon\_num$$

- Output the value of your magic number in a user-friendly way.
- Compile, run and test your program. Go back and fix it if it doesn't work properly.