CS 2024 – ASSIGNMENT #3

Assigned: **9/12/2019**
Due: **9/18/2019**

PROBLEM:

This week in lecture we learned some basic syntax for functions and control statements. We're going to use what we've learned (along with our experience with classes from Assignment 2) to build a simple text-based menu system just like we had when I was a kid!

I'll start by showing a sample run of what this should look like:

```
---- MENU ----
(1) Print My Name 3 Times
(2) Square A Number
(3) Quit
Enter Choice> 1
Ron
Ron
Ron
 ---- MENU ----
(1) Print My Name 3 Times
(2) Square A Number
(3) Quit
Enter Choice> 2
Enter a number to be squared>6
6 squared is 36
 ---- MENU ----
(1) Print My Name 3 Times
(2) Square A Number
(3) Quit
Enter Choice> 3
Quitting!
```

How do we do this?

First, create a `Menu` class that can be used to display the menu. The menu will use an enumeration to define the following "constants" :

- cUnknownItem
- cPrintMyNameItem
- cSquareANumberItem
- cQuitItem

You can put these in an old style C/C++ enumeration or a new C++11 scoped enumeration.

The class definition would look something like this (this is the last time I will likely give you a class definition for a homework assignment):

```
class Menu
{
 public:
  void showMenu();
  MenuItemCode promptUser();
};
```

The class methods do the following (note that there is NO constructor and NO private member variables):

`showMenu`	Simply prints out the menu to the user. It determines the "number" to put in parenthesis before each menu choice, but as an "advanced hint" you should make this number match up with whatever the enumerated constant is for the corresponding menu choice (for example, if `cQuitItem` is going to have an integer value of 3, then the "Quit" option should be shown as " (3) Quit").
`promptUser`	This will prompt the user for a choice, accept an integer as input (don't worry if the user doesn't type in an integer, just let everything fail) and then will return the corresponding *enumerated constant* for the value entered. So, for example, if `cQuitItem` ends up being "3" and the user enters a "3", this function should return `cQuitItem`. Note that the return type is `MenuItemCode`, so you must use this as the name of your enumeration and you must return the enumerated constant (such as `cQuitItem`) instead of the actual integer value (such as 3). If the user enters a valid integer that doesn't correspond to any of the menu choices, you should return `cUnknownItem`.

For this assignment your class definition should be in a file name `MenuH.h`, and the class member function definitions should be in a file named `Menu.cpp` (NOTE: We use "MenuH.h" instead of just "Menu.h" to get around a limitation in the CMS system. CMS does not allow two different files to be submitted with the same base name and different extensions).

In your `main.cpp` file, implement two simple functions (along with your `main`() function) that do the following:

1. One function will use a loop to print your name 3 times. Call it `printMyName`
2. The other function will prompt the user for an integer and then print out the square of that number. Call it `squareANumber`.

Your `main`() function will need to do the following:

1. Declare an instance of `Menu`
2. Enter into a loop that does the following:
    a. Shows the menu by calling the `showMenu` method in the `Menu` class
    b. Prompts the user for a menu choice by calling the `promptUser` method in the `Menu` class
    c. Enters into a switch statement that switches on the value returned by `promptUser` and has the following cases:
        i. If the user entered the choice for "Print My Name 3 Times", call `printMyName`() and break.
        ii. If the user entered the choice for "Square a Number", call `squareANumber`() and break.
        iii. If the user entered the choice for "Quit", print "Quitting" and break.
        iv. If the user entered something else, print out "Unknown selection" and break.
    d. The loop terminates when the user enters Quit

HINTS/DISCUSSION:

Pay attention to the sample printout above. It will help remind you what you should be doing!

When implementing `Menu::promptUser()`, you will likely be reading into an `int` variable. If you decide you'd like to return that value (and that member function is defined to return a `MenuItemCode`) you may get a syntax error. Such an error could be circumvented by *casting* the integer variable to a `MenuItemCode` value. This can be done with either of the following (assuming your integer variable's name is `input`):

- `return (MenuItemCode) input;`
- `return MenuItemCode(input);`

Your MenuItemH.h header file should define the enumerated type `MenuItemCode`. It will appear in the header file but outside of the class definition.

Both your `Menu.cpp` and `main.cpp` file will need to include the `Menu.h` file. Do not try to include a .cpp file. While it is legal, it is not usually done.

FILES TO SUBMIT:

1. main.cpp
2. MenuH.h
3. Menu.cpp
4. Writeup (.doc, .docx, .pdf, .txt accepted)