CSCI 1300 Introduction to Programming [Fall 2014]
Instructor: Boese
Assignment #6
Due: Fri **Oct 17, by 10am**.

Calculations in C++

Objectives

- Read in command-line arguments
- Read in data from the user
- Practice C++

Problem Statement

For each of the problems, generate a separate .cpp file with a name that identifies the problem number.

- Problem 1 should be called **Hw6Problem1.cpp**
- Problem 2 should be called **Hw6Problem2.cpp**
- Problem 3 should be called **Hw6Problem3.cpp**

Zip the files for each answer together and submit them to the Moodle as ***Firstname_Lastname_HW6.zip***.

Also include comments in your code to describe what your code is doing. Comments should also include your name, date, the assignment number, and problem number.

Note: Your files need to be named exactly as described here, and zipped together to generate a .zip file, or points will be deducted from your grade.

The print statements requested in each question should also be formatted exactly as described.

Problems:

1. **Passer Rating**

   In football, there is a statistic for quarterbacks called the *passer rating*. There are five input parameters to the calculation: pass completions, pass attempts, total passing yards, touchdowns, and interceptions.

   Write a program that prompts the user for 5 values, one at a time, in this order:

   i. Pass attempts
   ii. Pass completions
   iii. Total passing yards
   iv. Touchdowns
   v. Interceptions

   Your five values read in from the user should then be used in the passer rating calculation as follows:

   C = (completions per attempt – 0.30) * 5
   Y = (yards per attempt – 3) * 0.25
   T = touchdowns per attempt * 20
   I = 2.375 – (Intercepts per attempt * 25)
   PasserRating = (C+Y+T+I)/6 * 100

   Note: You need to calculate "completions per attempt" and "yards per attempt", etc. based on the data you read in from the user.

   Your program should then output the passer rating in the following print statement: "*The passer rating is <X>*", where *X* is the number calculated for PasserRating, rounded to exactly one place after the decimal. (*Note: you will get weird numbers if you input fake data. However, if you use data from nfl.com or espn.com, you should get the same result for passer rating as these websites.*)

   An example run for D. McNabb's data (see table below). Note that the numbers in blue are user-inputs during the execution of the program.

```
Enter # of pass attempts
316
Enter # of pass completions
180
Enter total passing yards
2647
Enter # of touchdowns
18
Enter # of interceptions
6
The passer rating is 95.5
```

To test your program, look up actual data on www.nfl.com or use the following information from 2007:

Quarterback	Completions	Attempts	Yards	Touchdowns	Interceptions
D. McNabb	180	316	2647	18	6
T. Brady	319	516	3529	24	12
P. Manning	362	557	4397	31	9

Use this calculator to test your program:
http://www.csgnetwork.com/quarterbackratecalc.html

2. **Time**
A day has 86,400 seconds (24*60*60). Given an integer number in the range of 86,400, output the current time as hours, minutes, and seconds for a 24-hour clock. For example, 70,000 seconds is 19 hours, 26 minutes, and 40 seconds. Your program should have one command-line argument that is the number of seconds to convert, and then use that number in your calculations. Your output should be displayed as "*The time is <X> hours, <Y> minutes, and <Z> seconds*", where each X, Y, and Z is an integer representing that increment of the time.

For a command-line argument of 70000, your program should output

```
The time is 19 hours, 26 minutes, and 40 seconds
```

3. **Census Data**
The U.S. Census provides information about the current U.S. population as well as approximate rates of change. Three rates of change are provided:

a. There is a birth every 7 seconds
b. There is a death every 13 seconds
c. There is a new immigrant every 35 seconds

Using those three rates of change, and a current U.S. population of 307,357,870, write a program to calculate the U.S. population in exactly one year. Your program should output the result in a nicely formatted print statement: "The population will be X". Note: You cannot have a part of a person!

Hint: You will need to calculate the number of seconds in one year

Example output:

```
The population will be 310338195 people
```

Requirements

- The name of the files must be
    - Problem 1 should be named **Hw6Problem1.cpp**
    - Problem 2 should be named **Hw6Problem2.cpp**
    - Problem 3 should be named **Hw6Problem3.cpp**

- Comments at the top of each of your programs
    - Your name
    - Date
    - Homework #6 Problem #[put either 1, 2, or 3]
    - Brief description of the problem (one or two lines max)
- The output must match exactly to the examples provided (given appropriate inputs).
- If the program requires command-line arguments and you do not use command-line arguments, that program will receive a zero.
- If the program requires prompting the user for the values, and instead you hard-code those values, that program will receive a zero.

Submit your zip file to Moodle.