CSCI 1300 - Intro to Computer Programming
Instructor: Knox
Assignment 2

Due Friday, Feb 3rd, by 12:30 pm

For this assignment, write all of your code in one file, called **main.cpp**, which is the default source file name in a CodeBlocks project. You can find the template of this file uploaded in moodle itself. You just need to insert your code in this file at appropriate places. The template file has all the function definitions including main, you need not modify that part. Once you have your code running on your virtual machine (VM), you must submit it to the autograder. You can modify your code and resubmit as many times as you need to, up until the assignment due date. You **must also** submit your code to Moodle to get full credit for the assignment.

Submitting Your Code to the Autograder

The computer science autograder, known as COG, can be found here:
https://web-cog-csci1300.cs.colorado.edu

Login to COG using your identikey and password.

- Select **Assignment 2** from the assignment drop-down box.
- Upload your **.zip** file and click Submit.

Your main.cpp file must be compressed into a zip folder. You may name this zip folder of your choosing, but the .cpp *must* be named main.cpp.

COG will run its tests and display the results in the window below the Submit button. If your code doesn't run correctly on COG, read the error messages carefully, correct the mistakes in your code, and upload a new file. You can modify your code and resubmit *as many times* as you need to, up until the assignment due date.

Before you submit your code to COG, make sure it runs on your computer. If it doesn't run on the VM, it won't run on COG. **If your code does not compile, you will receive a 0.**

What to do if you have questions
There are several ways to get help on assignments in 1300, and depending on your question, some sources are better than others. There is a Peer Discussion Forum on our Moodle page that is a good place to post technical questions, such as how to get user input, or treat that input as an integer. When you answer other students' questions on the forum, please do not post entire assignment solutions. The CAs are also a good source of technical information. If, after reading the assignment write-up, you need clarification on what you're being asked to do in the assignment, the TAs and the course instructors are better sources of information than the discussion forum or the CAs.

For this assignment, all functions must be of the form **void <function_name>(void) {}**. In other words, the functions *do not* take any arguments as inputs, nor do they make any returns to main. You must name the functions as indicated in each section below. **Importantly**, the cout formats provided for each problem are not suggestions – they **must** be followed precisely, word-for-word and including all punctuation marks, otherwise the autograder will not recognize your results and you will not receive credit.

Problem Set

Write a **function** for each of the following problems within your main.cpp file and make the appropriate calls to each function from within your **main function**.

1. 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 2 seconds
    b. There is a death every 7 seconds
    c. There is a new immigrant every 24 seconds

    Using those three rates of change, and a current U.S. population of 318,933,342, write a function to calculate the U.S. population in exactly one year (365 days).

    Your function *must* be named **calculatePopulation**. It should calculate and **print** the result in the following format, where *X* is your result:

    "The population is X."

2. A day has 86,400 seconds (24*60*60). Given a number of seconds in the range of 0 to 86,400, output the 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 function *must* be named **secondsToTime**. It should obtain the user input within the function's scope (i.e., inside of the function), perform the necessary calculations, and then **print** the result in the following format:

    "The time is X hours, Y minutes, and Z seconds."

3. In science, temperature is always described in Celsius, but in the U.S. we tend to use Fahrenheit temperatures. Write an algorithm to convert a Celsius temperature into Fahrenheit.

    Your function *must* be named **celsiusToFahrenheit**. It should obtain the user input within the function's scope, perform the conversion, and then **print** the result in the following format:

    " X degrees Celsius is Y degrees Fahrenheit."