**Program 1 - Getting Familiar with Your Environment**

CS 580U - Spring 2019

*All programs will be tested on the assigned virtual machines. If your code does not run on the assigned virtual machines, it is considered non-functioning EVEN IF IT RUNS ON YOUR PERSONAL COMPUTER. You can write your code anywhere, but always check that your code runs on the assigned virtual machines before submitting.*

### Driver Code and Test Input Files

### Driver Code

* + [program1.c](https://drive.google.com/file/d/0B5NpY9dM1zfBR2lGdTFtandQZVk/view?usp=sharing)
    - *All of the function interfaces are provided for you. You just need to implement them.* ***You must use the driver code as I have given you and only make alterations where stated. DO NOT change it to take user input, etc.***

### Grading Rubric

***TOTAL: 10 points***

## Part 1 (1 point):

* + Compiles and outputs hello world when run: 1 point

## Part 2 (8 points):

* + Part A: 2 points
  + Part B: 3 points
  + Part C: 3 points

## Style Guidelines (1 point)

* + Follows requested program structure and submission format
  + Follows [formatting guidelines](https://drive.google.com/open?id=1PyRZpBay-PZ8CzQRDAaz2Oke0K0T1rIBHMKNzZWjgbI)

### Guidelines

This is an individual assignment. You must do the vast majority of the work on your own. It is permissible to consult with classmates to ask general questions about the assignment, to help discover and fix specific bugs, and to talk about high level approaches in general terms. It is not permissible to give or receive answers or solution details from fellow students.

You may research online for additional resources; however, you may not use code that was written specifically *to* solve the problem you have been given, and you may not have anyone else help you write the code or solve the problem. You may use code snippets found online, providing that they are appropriately and clearly cited, within your submitted code.

*By submitting this assignment, you agree that you have followed the above guidelines regarding collaboration and research.*

# Part 2 - Using Variables and Arithmetic

For the second part of the program we will be using variables and math. Helpful link: [printf](http://www.cplusplus.com/reference/cstdio/printf/)

# Part A

* + Uncomment the remainder of the code, and add the following expressions shown here in the main:
    - 3x3 - 5x2 + 6 for x = 2.5.
    - The result of (4 × 108 + 2 × 10-7) / (7 × 10-6 + 2 × 108)
      * read the comments in the code to know where to write your code
  + To round off an integer i to the next largest even multiple of another integer j, the following formula can be used:
    - int next\_multiple = i + j - i % j
      * For example, to round off 256 days to the next largest number of days evenly divisible by a week, values of i = 256 and j = 7 can be substituted into the preceding formula as follows:  
        *int next\_multiple = 256 + 7 - 256 % 7*
    - Write a function called findNextMultiple(int number1, int number2) to find the next largest even multiple for the following values of i and j:
      * **i j**  
        365 7  
        12258 28  
        996 4
  + Write a function, float convertFtoC(float fahrenheit), that converts 40° from degrees Fahrenheit (F) to degrees Celsius (C) using the following formula and returns the result:
    - C = (F - 32) / 1.8

# Part B

* + In the next part of the program we are going to see how choosing the wrong data types and careless casting can result in data loss. You should see inaccurate results.
    - Write functions to typecast a long integer to the following datatypes
      * int
      * double
      * char

# Part C

* + In the Fibonacci sequence, the first two Fibonacci numbers, called f0 and f1, are defined to be 0 and 1, respectively. Thereafter, each successive Fibonacci number f*i* is defined to be the sum of the two preceding Fibonacci numbers f*i*2 and f*i*1. So f*i*2 is calculated by adding together the values of f*i*0 and f*i*1.
    - Write a function that generates the first 20 Fibonacci numbers (including 0 and 1) using a loop.
    - You should return the final resulting value

Part 3 - Submission

* Required code organization:
  + **<user\_id>\_program1.c**
* While inside your program1 folder, create a zip archive with the following command
  + zip -r **<user\_id>\_program1** **<user\_id>\_program1.c**
    - This creates an archive of all file and folders in the current directory called **<user\_id>\_program1.zip**
    - **Do not zip the folder itself, only the files required for the program**
* Upload the archive as specified for the course