Final Project Assignment
CS590-01 Summer 2016

This project is your final exam (the project grade will be your final exam grade).

Due: August 9, 2016
The assignment will be submitted through Canvas.
As before, you need include a Makefile, then tar and gzip your files and upload the gz file.

Goal: In this assignment, you will practice using the `pthread` library to write a C program that creates a number of producer and consumer threads. The producer threads will generate random numbers to be stored in a circular queue. The consumer threads will remove elements from the queue.

Assignment Details

1. The program must ask the user for the number of producer and number of consumer threads to create.
    a. The number of producers and consumers do not have to be balanced. In other words, there may be 50 producers and 1 consumer, or 50 consumers and 10 producers, and so on. So, be sure to test in a variety of ways.
2. Producers and consumers will operate on a circular queue.
    a. The queue must have a capacity of 100 integers.
    b. You can search for information on circular queues if you are not familiar with that data structure. Note that a circular queue can be implemented with an array.
3. Each producer thread must generate a random integer from 1 to 1000, store the integer in the queue, sleep for a random amount of time from 1 and 3 seconds, and repeat.
    a. If the queue is full, the producer must print a message that the queue is full and block until more space is available.
    b. When the producer places an item on the queue, it should print a message with its thread id, the value of the item placed on the queue, and the slot into which it was placed (e.g., Producer 52833823 added 999 to slot 50).
4. Each consumer thread must consume a single integer from the queue, sleep for a random amount of time from 1 and 3 seconds, and repeat.
    a. If the queue is empty, the consumer must print a message that the queue is empty and block until more items are available (an element has been added).
    b. The consumer must print the item removed, the slot number from which it was removed, along with its thread id (e.g. Consumer 54326483 removed 346 from slot 10).

5. The program should run forever.
6. Be sure to use a thread safe function for generating random numbers.
7. You must use a mutex and condition variable to protect the queue.

Name your main program file `queue.c`.

Your program must be able to be compiled using `gcc` under Linux. Remember that you need to use the `–pthread` option when using the `pthread` library.

There are several example programs in the book chapter on threads that demonstrate the techniques you will need to use for this program.