Spring, 2015 ECEN 4003, CSCI 4830-012

Concurrent Programming

Homework 1

Assigned: 01/16/2015; Due: 01/23/2015, before class.

This is warm-up homework assignment. The goal is to make sure that you can compile and run concurrent Java code, access the class Moodle, and access `ecen5033.colorado.edu`. Please submit your homework through Moodle.

Problem 1

(a) Write a Java program to create 128 threads. Each of these threads should print it's thread ID. Use the `Thread.join` function to block the main thread until all of the 128 threads finish execution.

(b) Modify the code written in problem (a) such that each thread increments a single global counter variable common to all the threads. Initial value of the counter variable should be set to 0. Print the value of this counter variable after all of the 128 threads finish execution.

Submit a well-commented `.java` file. You can use the server
`ecen5033.colorado.edu` to test your solution.

Problem 2

Modify the program written in problem 1 such that the logic for incrementing the counter variable is protected by a lock or a synchronized block or method. Explain briefly (comment in your `.java` file is sufficient) how you ensure mutual exclusion for accesses to the shared variable.

Submit a well-commented `.java` file. You can use the server
`ecen5033.colorado.edu` to test your solution.