Programming Assignment #1: Multithreading
CSC 4103: Operating Systems, Spring 2020

Due Date: February 27, Thursday (by 11:59 PM)
Total Points: 10

Instructions: Compile and test-run your code on the classes server. Submit your work as instructed and verify your submission. The verify command will display your submission date/time. Include your name, email, and classes login ID in all source code files. Late submissions will be penalized at the rate of 10% per day late and no more than 3 calendar days late.

Objective
To learn the use of POSIX Pthreads or Java Threads

Background
Modern operating systems provide features for a process to use multiple threads to speed up accesses. In the class, we have learned the concepts associated with multithreaded computer systems, such as multithreading models. There are various thread libraries. POSIX Pthreads and Java Threads are widely used for creating and managing threads. The textbook (Threads, Chapter 4) gives examples of multithreaded programs using these libraries.

Programming Language
C/C++ or Java

Programming Task
You will implement the **Project – Sudoku Solution Validator** given in the textbook (see the Project 1, Page 197 in the 9th edition). Your program will need to validate whether a given Sudoku solution is correct or not. Please read the description in the textbook, and use either **Pthreads** or **Java threads** to implement this program.

Input: Your program should accept an input file that describes a Sudoku solution. An example input file containing a valid Sudoku solution is available on the course website for testing your code.

Output: Your program should generate the validation result of each row, column, and subgrid. Each line of the output should contain the following information:

(1) The thread ID of the thread that performed the validation.
(2) The row, column, or subgrid that was examined.
(3) The validation result of the row, column, or subgrid (valid or not).

Below is an example output that shows three lines of the validation results for Row #2, Column #5, and the top-left Subgrid (Row 1-3 and Column 1-3).

```
Thread 1, Row 2, Valid
Thread 2, Column 5, Invalid
Thread 3, Subgrid R123xC123, Valid
...
```

NOTE:

(1) Compile and test-run your code on the classes.csc.lsu.edu server. Your code should be compilable and runnable in the Linux environment of the classes server. Windows code will NOT be accepted.
(2) Include an README file to clearly explain how to compile and run your code, such as the command, arguments, and expected input and output, etc. The README file should also include your full name, LSU ID, and email address.
(3) Submit your work by the deadline. Verify your submission, which will display your submission date/time. Note that if you make multiple submissions, the prior submission will be overwritten. The final submission date/time will be the latest one.
(4) Late submissions will be penalized by 10% per day late and no more than 3 days late.

Submitting Your Work

All files you submit must have a **header** with the following (enclosed in comment lines):

Name:	Your Name (Last, First)
Email:	Your LSU email
Project:	PA-1 (Multithreading)
Instructor:	Feng Chen
Class:	cs4103-sp20
Login ID:	cs4103xx

You need to use the server "**classes.csc.lsu.edu**" to work on the assignment. You can login to your account in the server using SSH. Create a directory **prog1** (by typing **mkdir prog1)** in your home directory in which you create your program or source code.

Make sure that you are in the **prog1** directory while submitting your program. It is suggested to pack all the files in a tar tall (or a zip file) before submission. Submit your assignment to the grader by typing the following command:

~cs4103_chf/bin/p_copy 1

This command copies everything in your prog1 directory to the grader's account.

Verify that all the files have been submitted successfully:

~cs4103_chf/bin/verify 1