

CSCE 314 [Section 100] Programming Languages – Summer 2017

Anandi Dutta

Assignment 5

Assigned on Wednesday, June 29, 2017

Electronic submission to eCampus due at 23:59, Sunday, 07/02/2017

By electronically submitting this assignment to eCampus by logging in to your account, you are signing electronically on the following Aggie Honor Code: “On my honor, as an Aggie, I have neither given nor received any unauthorized aid on any portion of the academic work included in this assignment.”

Note 1: This homework set is individual homework, not a team-based effort. Discussion of the concept is encouraged, but actual write-up of the solutions must be done individually.

Note 2: Turn in one yourLastName-yourFirstName-hw5.tar or yourLastName-yourFirstName-hw5.zip file on eCampus, nothing else. What to submit is detailed below.

Note 3: All Java code that you submit must compile without errors.

Note 4: Remember to put the head comment in your files, including your name, your UIN, and acknowledgements of any help received in doing this assignment. Again, remember the honor code.

Problem [1] and [2] contains 30 points each. If you write a detailed design document [submit a pdf file for problem [2] and a .java file in that case] and detailed code for problem [2] describing your implementation, you will get 30 points. You will get 30 points if it's not compiled correctly. If you solve the Problem [2] and it compiled correctly [submit your .java file only, no need to submit a pdf file], you will get 5 points as extra credit. These extra credit will be added to your final. **Please add a comment on top of problem [2] if you are able to solve that perfectly. The comment should mention: “This program should run perfectly”.**

[1] Suppose you have multiple processors, therefore you can speed up the matrix multiplication. Implement the following method in parallel. Moreover, write a sequential matrix multiplication method as well.

```
public static double[][] parallelMultiplyMatrix( double[][] a, double[][] b)
```

Write a test program that measures the execution time for multiplying two 2,000 * 2,000 matrix using the parallel method and sequential method, respectively. You can write your test inside main method.

Your output should produce this message on the console:

```
Parallel time for matrix multiplication is      milliseconds
```

```
Sequential time for matrix multiplication is    milliseconds
```

Please see the example codes (ParallelMergeSort.java and MergeSort.java) provided to help you out on measuring time.

[2] You are working as a programmer for developing a game show. The host of the show asked you to develop a Java program that will help the game show running smoothly. The rules for the show is that it will ask the player a couple of questions. If the player can answer the question correctly in 10 seconds, he/she will score one/1 point. If the player's answer is wrong or if he/she does not answer the question within 10 seconds, the correct answer is displayed and the player does not score any points. After getting an answer or running out of time, the next question is displayed.

From the rules above you can get the sense this program needs to run in parallel manner to handle different tasks at the same time. You can use locks/conditions to ensure thread synchronization. We suggest that you should use three threads to handle three tasks. Below is a skeleton of logic that you might follow:

```
printQuestion thread{
// start timer thread
// wait for answer or timeout
// save response from user
// print "correct", "wrong" or "timeout"
}
reader_thread{
// check for end of program
// read answer;
// move answer to global variable
```

```
// release printer thread
}
timer_thread(int num) {
// wait 10 seconds; you can use sleep() here
// still waiting for this question?
// put TIMEOUT value as response
// release printer thread
}
```

Again, it is not necessary that you have to follow this skeleton code. If you have better/different idea/implementation, you are welcome to follow that. Moreover, add a test in main to interact with the user. You can use the Scanner class for that.