

Homework 3 (HW3)

EE 363 (Fall 2018)

Department of Electrical & Computer Engineering
Clarkson University

Due: October 28, 2018 at 11 pm.

Instructions

Please read the instructions carefully before submitting your work.

Note: Solve all problems and upload your answers to Moodle. Whenever you write solutions on paper, you need to scan it and upload the file(s).

Note: *user* stands for your login ID on Polaris (*polaris.clarkson.edu*). This should be the exact same as your CU ID.

Note: Make sure any code you write works on Polaris before uploading your files. You will lose many points if your code doesn't compile.

Note: Please do not upload any executable or intermediate files as answers to problems, unless specifically asked to do so.

Total points: **50**

Problem 1 (50 points)

In this problem, you will write a multithreaded Java program. We have discussed two ways of doing this: by subclassing *java.lang.Thread* and by implementing the *java.lang.Runnable* interface. Some multithreaded programs that were discussed in class are in Section 6 ("Multithreaded programming") on Moodle.

Design a class *RThread* that allows the creations of multiple threads that can be used to concurrently compute the averages of multiple lists of numbers:

- every list is specified by a left (L) and right endpoint (R) where $L < R$
- whenever a thread is created, it is given two positive integers L, R that specify a numeric range; the thread is supposed to calculate the average of the square roots of all numbers in this range
- e.g., if the endpoints for a thread are $L=4$ and $R=12$, it will average the numbers $\text{sqrt}(4)$, $\text{sqrt}(5)$, ... , $\text{sqrt}(11)$, $\text{sqrt}(12)$
- each thread prints the average it computes as shown in `parsrtavg10.txt`
- make sure that every thread can be given a (String) name
- make sure that $0 \leq L \leq 50$ and $100 \leq R \leq 800$

Test your *RThread* implementation by modifying the client *ParSqrtAvgCalc* to use your implementation for computing the averages of the square roots of 50 lists of numbers. **Make sure the modified version your *ParSqrtAvgCalc* meets all the requirements given above.**

Store the output of this execution in `user_parsqrtavg50.txt`.

Note: the output will (in general) vary every time you run it; that is because the endpoints of the lists (and hence the lists themselves) are randomly generated.

Note that answers approximate up to two decimal places will be accepted as correct so you do not have to worry about precision beyond that.

Deliverable: Upload `RThread.java` and `user_parsqrtavg50.txt` to Moodle.

End