Due Date: April 13th, 2010

Program 4: Threads

UMD's Nuclear Reactor is overheating!! The power it produces is just too much for one processor! The only way to save it from the meltdown is to write a multithreaded prime-finding program! You are given access to a 10-processor nuclear reactor machine, for which you are to implement a program using multithreading concepts, SwingWorker and JProgressBar classes. To help you get started, check out the really good examples in sections 29.18 and 29.19. The code from these sections has been posted on VLT.

Your program is to do a distributed processing task. You can choose the task from the following two choices (choose one only):

1) **Compute the first N prime numbers**
   If N is 5, the first 5 prime numbers are {2, 3, 5, 7, 11}
2) **Compute Prime numbers up to N**
   If N is 5, the prime numbers up to and including 5 are {2, 3, 5}

For either choice, your program is to:

- Display a GUI where user can enter N and get notified when the task is done.
- Start 10 threads in the background to do the task (use SwingWorker to keep GUI responsive)
- Display the total progress of all the threads using JProgressBar
- Write the prime numbers in order to a file upon program completion.

Policies

- You may work with another person on this assignment
- Due Date: April 13th, 2010. Late penalty applies after due date (10% off the score)

Deliverables:

- The GUI program Java source file(s) submitted through VLT
- A single-page paper with a brief write-up of your project. See Paper section below.
- If working with another person, both names are to be specified on the assignment

Bonus1 (2%)

Ability to start a different number of threads, where number of threads is specified through GUI.

Bonus2 (3%)

Allow your program to do both tasks 1 or 2, based on user choice

Paper

A single-page paper telling me the option you chose and any specific things I am to know about your project. If you do any computations for your project, include them in your paper. Imagine you are writing a quick note to another programmer who will be later maintaining and extending your project.

Due Date: April 13th, 2010

Grading

- 90% - program
- 10% - paper
- 2% - bonus1
- 3% - bonus2

Things I will be looking for when grading

- Good GUI design
- GUI responsiveness [use of SwingWorker]
- 10 threads being started
- The total work is split evenly among all the threads
  [all threads should finish processing at about the same time]
- JProgressBar displaying true progress
  [50% should mean that 50% of work has been done time-wise]
- Primes written to a file in order smallest to largest.