CMSI 387/587

OPERATING SYSTEMS

Spring 2010

Assignment 0413

This assignment is yet another "classic" operating systems programming task dealing with process synchronization, supported by a smattering of exercises.

Not for Submission

- Read Chapter 6 of SGG and the synchronization papers given out in class.
- Sometime over spring break, you will get feedback from me regarding your project ideas. Once you get this, it is recommended that you start some initial reading and/or literature search. Start committing any relevant files (LaTeX, code, etc.) to */projects/cmsi387* or */projects/cmsi587*, as appropriate.

For Submission

The Dining Philosophers Problem

Implement a solution to the dining philosophers problem using POSIX threads and semaphores. The bounded buffer code given out in class may be used as a basis for your solution. In addition, Sections 6.6.3 and 6.7.2 provide outlines for solving the problem.

Make sure to include well-placed output statements to report what's happening in your program and the state of things at any given time — that's how we'll know whether your solution is working. Use the *assert* function where possible. Commit your code to */homework/cmsi387/dp* or */homework/cmsi587/dp*, as appropriate.

Exercises

Name two things (for a total of four observations) that may happen in *incorrect* critical section solutions to the dining philosophers and sleeping barber (SGG Exercise 6.39) problems.

Submit your answers on hardcopy or in LaTeX. For LaTeX, commit your source file(s) to */homework/cmsi387/dp/doc* or */homework/cmsi587/dp/doc*, as appropriate.

Extra Credit

The Sleeping-Barber Problem

You will get extra credit if you also implement a solution to the sleeping barber problem (SGG Exercise 6.39). Commit this to */homework/cmsi387/sb*.