CSc 422/522 — Homework 3

due Tuesday, April 4

The first four problems are worth 10 points each. The last problem is worth 20 points. Graduate students are to solve all problems (60 points). Undergraduates are to solve any combination that adds to 40 points.

Be sure to read each question carefully and to do all that is asked. As usual, the work you turn in must be your own. Please explain your answers clearly.

1. MPD book, exercise 4.17 — alternate way to solve the Dining Philosophers problem.

2. MPD book, exercise 4.31 — the Search/Insert/Delete problem.

3. MPD book, exercise 5.8 — the Savings Account problem. Note that a `deposit` might satisfy the needs of more than one pending `withdraw`.

4. MPD book, exercise 5.9 — the Water Molecule problem.

5. Write a program to simulate the One-Lane Bridge problem described in exercise 5.7 of the MPD book. Each car should be a process that crosses the bridge `trips` times, first in one direction, then in the other direction, and so on. Use `northCars` instances of cars that first cross in a northbound direction, and use `southCars` instances of cars that first cross in a southbound direction.

Write a monitor to control entry to the bridge. It should have operations that a car calls to ask permission to cross the bridge and to indicate that it has finished crossing the bridge. You will want to have two or four operations depending on how you program the cars processes. (After you think about the choices, you will understand what the last sentence means!) Your solution does not have to ensure fairness, although that would be a nice touch.

You may write your program in SR, Java, or C plus Pthreads. If you use SR, you will have to simulate monitor entry, exit, wait, and signal using semaphores as shown in Figure 6.7. If you use Java, see Section 5.4 for how to program threads and synchronized methods. If you use Pthreads, see Section 5.5 for how to program monitors.

Section 1.5 of the SR book contains an example of how to do a simulation that is somewhat similar to the one you are to write. (However, do not use a process and the `in` statement to implement the monitor; use procedures and semaphores.) Source files for this SR program as well as for the Java and Pthreads programs in Sections 5.4 and 5.5 are stored in the directory `/home/cs522/SamplePrograms`.

Your program should have three command-line arguments, for the values of `trips`, `northCars`, and `southCars`. Have the cars sleep for a random amount of time between crossings of the bridge and have them sleep for a smaller random amount of time to simulate the time it takes to cross the bridge. Stop the simulation after each car has crossed the bridge `trips` times.

The output from your program should be a trace of the significant events that occur, such as a car wanting to cross the bridge, starting to cross, or leaving the bridge. Each line of output should contain a time stamp, the identity of the car associated with the event, and a short descriptive message. Write the trace to standard out.

Give us a commented listing of your program when you turn in the rest of your homework. Also use `turnin` to submit your program electronically. The assignment name is `hw3.problem5`. *Turn in both your program and a makefile that we can use to compile your program.* We should be able to execute `make` to produce an executable file, and then we should be able to execute `a.out ...` to execute your program with command-line arguments as specified above.