Homework/Lab 3

due Wednesday, February 26, 2003, 11:59 PM

There are several files to turn in for this assignment. They should all be included in a file named `hw03.tar` that you submit using the `turnin` utility. *Please use the filenames specified* and be sure to include your name in each file.

(10 points) Write a C or C++ program using pthreads that implements a domain decomposition approach to the matrix-matrix multiplication example we have been studying in class. However, instead of dividing up the work by rows, your program should divide up the work by assigning square submatrices to each thread. To simplify things, you may assume that the number of threads is a perfect square, and that the square root of the number of threads evenly divides the number of rows and columns in the matrices. You may also define the number of threads as a constant in your program rather than on the command line. You may use one of the class demo programs as your starting point.

Once your program works, run for 750 × 750 matrices, using 4 threads on the four-processor nodes of bullpen (*wetteland* or *rivera*; `ppn=4` in PBS). Submit these runs through PBS. Compare the timings for the compute phase of your submatrix decomposition implementation with those for each of the multithreaded matrix-matrix multiplication examples from the class demos. Explain the differences in running times. Write your answer to this part in a plain text file `hw03.txt`.

Your submitted tar file should include your `Makefile`, your C source code (including the timer code from class, if you choose to use it), your PBS script(s), a brief `README` file expaining how to run your program, and the `hw03.txt` file. Please do *not* include object files or your executable.

Honor code guidelines: While the program is to be done individually, along the lines of a **laboratory program**, I want to encourage you to ask questions and discuss the program with me, our TA, and with each other, as you develop it. However, no sharing of code is permitted. If you have any doubts, please check first and avoid honor code problems later.

Grading guidelines: Your grade for the program will be determined by correctness, design, documentation, and style, as well as the presentation of your timing results.