

DUE DATE: Friday March 5th 2010 @ 11:59 pm

Homework 4

The objective of this assignment is to compute the product of two matrices A and B. Each element of your product matrix corresponds to computing the sum of the product of elements of a row from matrix A and a column from matrix B. Thus, element $[i,j]$ of the product matrix is produced using row i from matrix A and column j from matrix B. The input matrices will be read from a file, and the product matrix will be written to a file.

A. Storing and Reading matrices using files (3 points)

You are required to provide code for storing a matrix to file and also for reading this matrix from the file. Individual matrices will be stored in a file: elements of this file will be stored as text separate by spaces. Matrices will be stored in row major format. The first two elements of the stored file will indicate the dimensions (number of rows and columns) for the matrix, and the remainder of the file will contain elements from the matrix.

B. Using thread pools for computing the product (7 points)

As part of this assignment you will design and implement your own thread pool. You can use either PThreads or Java threads for this assignment. You cannot however use preexisting thread pools in these libraries. If you are using Java, you are not allowed to use the `java.util.concurrent.*` package.

Your thread pool will have two sets of workers: the work generators and the calculators. The generators and calculators are responsible for coordinating the process of computing the product of the two matrices.

The work generators are responsible for generating the rows and columns to be used in the matrix multiplication. The unit of work produced by the generator is a row from matrix A and a column from matrix B. This unit of work will be placed in a work queue (bounded buffer can be implemented using a data structure of your choice). A generator produces one unit of work and places it in the work queue during one round of its execution.

The calculator workers will be responsible for computing the value for a specific entry in the product matrix. It does this by accessing the work queue, removing a work unit from the queue, performing the calculation, and reporting the result to a reporter. The reporter is responsible for ensuring the storage of the computed matrix to a file. You must wait for an entire row of the product matrix to be available before writing it to the file system.

Update

A calculator waits for a unit of work to be available before it proceeds with its processing. If a calculator has no work to do, it will block till such time that there is work available. When a unit of work is available the calculator (1) performs the computation (2) sends the result to the reporter thread and then (3) checks if there is more work to be performed.

The worker threads – calculators and generators – are created exactly once. When the thread pool manager determines that the ENTIRE work has been performed, it terminates the workers and exits.

C. Managing the configuration file (2 points)

Your program needs to take a configuration file as input. This configuration file will have the several elements. A sample configuration file is listed below:

```
Number_of_Generators=3
Number_of_Calculators=3
MatrixA=fileA.txt
MatrixB=fileB.txt
MatrixC=productFile.txt
Buffer_Size=30
```

D. Error Checking (3 points)

1. You are required to check if there is a problem in the matrix that you are reading
 - a. If the stored dimensions do not match the number of elements that you are actually reading from the file.
2. If the dimensions of the specified matrices preclude the possibility of computing a product matrix.

Grading

This assignment will contribute a maximum of 7.5 points towards your final grade. The grading for this assignment will be done on a 15-point scale, and the breakdown of points for each of these problems has been listed.

You are required to work alone on this assignment.

Late Policy

See <http://www.cs.colostate.edu/~cs451/assignments.html>