Lab Assignment 2: Threads

Notes

You need to work on this project individually.
This project must be implemented in C.

Objectives

- To get familiar with thread programming using the Pthread library.
- To better understand processes and threads.

Overview

You are required to implement a multi-threaded matrix multiplication program. The input to the program is two matrixes A(x*y) and B(y*z) that are read from corresponding files. The output is a matrix C(x*z) that is written to an output file. A parallelized version of matrix multiplication can be done using one of these two methods: (1) a thread computes each row in the output C matrix, or (2) a thread computes each element in the output C matrix.

Requirements

- Implement the multi-threaded matrix multiplication using both methods described above.
- Compare the two implementations according to the following: (1) the number of thread created and (2) the execution time taken.
- Your program need to handle any errors and terminate gracefully.

Your programs should do the following:

- Your program is executed as: `./matMultp Mat1 Mat2 MatOut`, where Mat1 and Mat2 are the names of the files to read the first and second matrixes, respectively. MatOut is the name of the file to write the output matrix. If the user does not enter this information, the default is a.txt and b.txt, for input matrixes A and B, respectively, and c.out for the output matrix.
- Read the number of rows and columns of the input matrixes. They are written in the first line of the file as ”row=x col=y”.
- Read the input matrixes from their corresponding files. Each row is on a separate line, columns are separated by tabs.
- Use threads to calculate the matrix that results from multiplying the input two matrixes.
- Output the resulting matrix in a file.
- Output the number of threads created and the time taken on the standout.

Deliverables

- Complete source code in **C**, commented thoroughly and clearly. You also need to submit a make file that we can use to compile/build your code. Note that you need to call the executable **matmult.out**.
- A report that describes the following: (1) how your code is organized, (2) its main functions, (3) how to compile and run your code, (4) sample runs, and (5) a comparison between the two methods of matrix multiplication.
- All deliverables are to be put in one directory named lab2_XX, where XX is your ID and then zipped.
- You need to send your code to cs333f17@gmail.com on October 21 before 8 AM. The subject line should be: "Lab Assignment 2 - SID:xx"
- The assignment will also be discussed in the lab.

Hints

To measure the execution time, you will need to use code that is similar to this:

```
#include <sys/time.h>

main()
{
struct timeval stop, start;

gettimeofday(&start, NULL); //start checking time

//your code goes here

gettimeofday(&stop, NULL); //end checking time
    printf("Seconds taken %lu\n", stop.tv_sec - start.tv_sec);
    printf("Microseconds taken: %lu\n", stop.tv_usec - start.tv_usec);
}
```