CSE331 OPERATING SYSTEMS DESIGN FALL 2018

ASSIGNMENT I

Last Submission Date:
6 December 2018, 23:59

UNIX PROCESSES AND POSIX THREADS

In this assignment, you will compare three programming concepts: multiprocess, multithreaded and single process programming. So you will write a multi process, multi threaded and a single process program. For each programs, you will write a code which do gamma encoding to images. The images will be given as a raw image matrices in a dat input file and with different sizes to you.

IMPLEMENTATION

a) Multi process program (30 pts):

Before gamma coding, the parent process will do the following:

1) The parent process will read first line of the file and get the size N of the image.

2) Parent process will create a two dimension NxN array.

3) Parent process will read the remaining lines of the file and fill the matrix created by itself. The value of the element is between 0 and 255.

4) Parent process will update elements of the matrix with dividing each elements to 255.

For gamma coding, parent process will create *N* child processes which run concurrently and do the following operations below. Parent process will start to wait for completion of the operation of all child processes.

i) A child process will do gamma coding to a row associated with it. The first created child does gamma coding to the first row, the second child do gamma coding to second row, and so on.

ii) The gamma coding is done using the following formula. where γ is the coding exponent. In this assignment use γ as 1/2.

$$X_{new} = X_{old} {}^{\wedge} \gamma$$

iii) Any one of the child process calculates new value for an element in the matrix using this formula.

iii) When a child process completes the gamma coding to its row, it will write the contents of the row to a file associated with its number and terminate.

After termination of all child processes, the parent process will read the content of all files created by its child processes and create an output file with the content of the files created by the child processes.

b) Multi Threaded Program (20 pts):

Before gamma coding, the main thread will do the following:

1) Main thread will read first line of the file and get the size *N* of the image.

2) Main thread will create a two dimension *NxN* array.

3) Main thread will read the remaining lines of the file and fill the matrix created by itself. The value of the element is between 0 and 255.

4) Main thread will update elements of the matrix with dividing each elements to 255.

For gamma coding, main thread will create **N** threads which run concurrently and do the following operations below. Main thread will start to wait for completion of the operation of all threads.

i) A thread will do gamma coding to a row associated with it. The first created thread does gamma coding to the first row, the second thread do gamma coding to second row, and so on.

ii) The gamma coding is done using the following formula. where γ is the coding exponent. In this assignment use γ as 1/2.

$$X_{new} = X_{old} \char`\^ \gamma$$

iii) Any one of the thread calculates new value for an element in the matrix using this formula.

iii) When a thread completes the gamma coding to its row, it will write the contents of the row to a file associated with its number and terminate.

After termination of all threads, the main thread will read the content of all files created by other threads and create an output file with the content of the files created by these threads.

c) Single Process Program:

1) This program will read first line of the file and get the size **N** of the image.

2) This program will create a two dimension **NxN** array.

3) This program will read the remaining lines of the file and fill the matrix created by itself. The value of the element is between 0 and 255.

4) This program will update elements of the matrix with dividing each elements to 255.

5) The gamma coding is done using the following formula. where γ is the coding exponent. In this assignment use γ as 1/2.

$$X_{new} = X_{old} \char`\^ \gamma$$

6) This program calculates new value for an element in the matrix using this formula.

7) When it completes gamma coding, it will write the contents of the matrix to an output file.

REPORT (40 points)

You are wanted to write a report that expresses the design, implementation and execution flow of your programs including your understandings from processes and threads. By obtaining execution times from each program you have written, make comparisons between multi-process and multi-threaded programming concepts. Express the total execution time variations for each program with different input image files. **In addition, compare those programming concepts with another choice of your own metric which you consider as crucial for the expression of the difference.** Explain your results clearly and support them with graphics in your report. (40 pts)

,

SUBMISSION RULES

- Writing clean, readable code and using comments are recommended.

- Including the student name and ID in the top of the code, as comments is required.

- **Filenames of the C files must be hw1a.c, hw1b.c and hw1c.c. For report files, only PDF format is accepted. The filename of the report file must be Report1.pdf. Zip your files and name the zip file as <YOURNAME>_<STUDENT_ID>.zip**

- **This homework will be sent to COADSYS. Do not send assignment via e-mail. Assignments sent via e-mail will not be evaluated.**

- **Your C files will be checked for plagiarism using MOSS system.**

- **Any copied work will be evaluated as 0 point and the assignment will not be accepted as submitted !!!**

- You are welcomed to ask questions, come up with new ideas about the homework, but reading the instructions explained here carefully, also studying from the course book are highly recommended to have a general understanding before asking questions. Playing with the program implementation is encouraged. More creative ideas can get higher points.

You should do this assignment individually. Works done with collaboration is treated as copied work.

Due date for this assignment is strict 6 December 23:59 and submit your own work to stay away from punishment. Good luck...