CSE 344 MIDTERM PROJECT REPORT

EMRE SEZER
1901042640

CLIENT:

Client communicates between server_Y and server_Z.
Sends its request to server_Y. It can get its response from server_Y's or server_Z's workers.

Struct request includes client's process id (int), matrix's content (int array) and n of the n x n matrix (int). Struct response includes end time of the total process, invertible data of the matrix (1 if invertible, 0 if not invertible).

When client reads the matrix from the input file, Requested prints is done. When response is gathered from the one of the server's workers, requested prints is done on the command line.

SERVER_Y:

It gets the request from the Client. It creates the Server_Z as a child process and called execve() function. Later, t child processes are created as worker_Y's.

When request is gathered from the client, it lets one of the worker_Ys work. It starts checking from 1to t. When i'th worker_Y is available it lets that worker_Y work. If there is no worker_Ys is available, it forwards job to the server_Z through a pipe. It writes matrix data and client's ID to the server_Z.

Server_Y communicates with its worker_Y's through a pipe. It writes matrix data and client's ID to the desired worker_Y. Server_Y waits until a request comes from the client. It works one time when it comes, then again waits for a new request.

When SIGINT happens it kills all of the worker_Ys and server_Z.

WORKER_Y:

It communicates between server_Y through a pipe. It increases the invertible matrix counter by 1 whenever it gets an invertible matrix. When SIGINT happens, it writes to the pipe the total number of invertible matrices. server_Y reads all of its worker_Y's umber of invertible matrices and sums them. Worker_y calculates all of the matrix related calculations. It sends the client's response through a FIFO.

SERVER_Z:

It gets the matrix data and client's ID through a pipe from the server_Y. It uses shared memory in order to communicate with it's worker_Zs. It keeps all of the calculated matrix number. Worker_Z synchronization is provided using semaphores. It waits for new string from server_Y. Until new string comes it simply waits.

When SIGINT happens it kills all of the worker_Zs.

WORKER_Z:

It communicates between server_Z through the shared memory. It increases the invertible matrix counter by 1 whenever it gets an invertible matrix. When SIGINT happens, it writes to the pipe the total number of invertible matrices. Worker_Z calculates all of the matrix related calculations. It sends the client's response through a FIFO. It waits for the semaphore value to become 1 for working. This situation occurs when server_Y sends a new string through the pipe to the server_Z.

NOTES:

Needed header and c files from the book is added under the "lib" folder. My homework codes are in the main folder.

You need to execute server_Y in order to test my work.

I added a sample log file to the main folder. (example_log.txt)
I added a sample input file to the main folder (input.csv