AZMİ UTKU SEZGİN

131044048

CSE344

System Programming

Final Project

Problem Definition

Providing a service for cheater kids to calculate their cosine providing different quality options with different prices and allowing kids to choose an option between fast result, high quality and low prices. Utilizing multithreading to utilize the main CPU as efficient as possible.

Problem Solution

The main server gets a port number to open a socket, data file to read provider informations from and a log file to save the logs, then creates new threads for each provider and giving function pointer of function called **providerThread**, opens up the socket with local address, and binds it to the given port number and accepts clients. After clients start to connect, server creates a new thread for each of them, creates thread using **clientThread** function pointer. Server accepts clients all the time but providers leave after their specified duration timed up and even if the server doesn't have any provider left, it will still accept new clients but will inform them that there are no available providers, it will only stop once the user gives terminate signal to server.

clientThread, finds the best possible and available provider, depending on client's priority, for the client and puts it to provider's queue and informs the **providerThread**. Note that every provider can have maximum number of 2 clients at the same time hence why the thread may not always be the best provider but only the best available provider. And then thread waits untill provider calculates the cosine value and informs the **clientThread** that the task is done. After the task is completed **clientThread** gives the output to the client and closes their socket.

providerThread, after getting the information from the **clientThread** providerThread calculates the cosine value of given degree using Taylor Series, it informs the **clientThread** that the task is done, if it's queue is empty thread waits for new task and if it's not empty, it moves to the next task.

Missing Pieces, Failures

When the server is being shutdown or there are no more active providers, the info server passes to client has some extra garbage characters after the intended message, i could not figure it out why. Sometimes the threads would never join, probably a deadlock of some sort that i didn't undersand why so the server doesn't wait for it's threads to finish their job and exit but there are no leaks The way my algorithm work makes provider queues almost never full, i might have used a mutex wrong and the way it works is, you can't add new elements while provider is completing a task, it has to wait and by the time provider finishes a task, there will always be a new opening in the queue hence why queue will "always" have a spot.And last but not least, sending signal sometimes won't stop the program.

These are all the known errors and missing features, i didn't have much time to fix them as i had to re-write the whole thing in less than 12 hours, fixes or advices are much appreciated!