Allon Finezilber

Michael Ferrera

CSC214 Fall 2014

# Producer Consumer Assignment 5 Multithreads

For the client database, we parsed the data from the given database text file. The database is an array that acts like a hashtable which contains client information. Each client has a unique ID which was used as the index in our array to identify each client and their personal information.

The producer and the consumer both share a single queue, by doing this we have control over the order of which the producer produces and the consumer consumes their bookorders. The array we implemented is a circular linked list for efficiency. We used a Mutex to protect the data in the array from the producer and consumer.

We created all our threads in the main(). We first made the producer and let it start its job. We then for each category prompted through the command line we created a consumer thread for that specific category. After creating the threads the main then waits for all the threads to finish their job and eventually join().

If the producer is done executing early the producer thread will continue to execute until there is nothing left in the queue. After that happens the main will gain control and print out all the final reports and free any leftover memory useage.

The database also needed its own mutex such that threads that are processing order can access the database in a thread safe method. Whenever a consumer takes control on the lock of the queue mutex it wont unlock it until its done meddling with the data. Because of this only one producer runs at any given time. By doing this, the database is safe from being executed and in a controlled specific order and isn't generated randomly and will always output the same successful, failed and final orders.

This program is running in the O(n). Searching and accessing a particular client because of its unique index is O(1), Accessing every client is in O(1) and parsing the customer receipt is also in O(1). Finally, the program runs at O(n) because traversing through n orders is set at the O(N)

Execution: run the Makefile. -> ./orders <database file> <orders file> <categories>