Harris Seth
Von Quilon

PA5 Read Me

Welcome to the read me for the fifth assignment: Multithreaded Book Order System. In this assignment we were to implement a book ordering system that worked through the use of threads.

The program starts by taking in database.txt, orders.txt, and categories.txt files. The database.txt file is used to create a database of customers with information such as their name, address, and credit limit. This is simply an array of customer nodes where each node contains the information. The categories.txt file is used to create an array of the available categories. The orders.txt file is used to create a queue. There is one queue which is accessed by all of the threads.

After having built database and categories the program creates a producer thread. The producer thread parses the order.txt file and uses that to build a queue. The enqueue operation for the queue is mutex locked so that when the producer is inserting none of the consumer threads are taking anything out of the queue. Once the producer has inserted into the queue the thread is unlocked.

There are multiple consumer threads, one for each category. The consumer threads check to see if the queue is empty. If the the queue is empty and the flag has been set for the orders.txt having been completely parsed the consumer threads exit. Otherwise if the flag has not been set but the queue is empty the threads wait around for the queue to be filled up by unlocking. If the queue is not empty but the information in the head of the queue is useable by the current thread it will release the queue by unlocking it's mutex and will let another thread see if it can use the queue. If the queue is useable by a thread the thread will process the thread by printing out whether or not the order succeeds, and changing around database to reflect a new balance if there is one, and updating the success fail lists.

Once all the consumer threads have been exited the program prints out the resulting database table and the program ends.