

README

Data Structures:

The consumer information is all stored in a hash table. Each node will hold a consumer's information as well as a linked list of their book orders by the end of execution. Key, value pairs are held as <id, consumerNode>

The queues which hold and organize the book orders before they get processed for a consumer are held in another hash table with key, value pairs as <category, queue>.

—

Initially, the consumer database file is loaded into the hash table in main. The producer thread then takes over and starts to parse the orders file. As each entry is parsed, it gets put into a node and thrown into its corresponding queue in the hashtable according to its category. Whenever the producer moves a node to a queue, it checks to see if the consumer thread for that queue is running. If it is, it continues execution as normal. If not, then it joins with the consumer thread and creates it again. This way, the consumer can run at the appropriate times without pulling a busy-wait.

When the consumer thread processes an order, it dequeues it from the queue and attaches it to the appropriate user according to the id number as either a successful or unsuccessful order.