

Helen See
Hyun Mo Yang

Structures:

There are three main structures. The categories structure, a linked list, will be built beforehand. It contains the category name and a category ID which is equivalent to some thread ID. The database structure, composed of linked lists, will have the database.txt file set before the threads get access to it. The order queue will first be created in the producer thread.

Shared Memory:

There are three shared data structures. Two out of three will each have a mutex. One is a shared queue and the other is a shared database. The producer thread will have access to only the shared queue data. The category threads will have access to shared queue and shared data. We allowed the category threads have access to two shared data structures by using a third shared data structure which contained both shared data without a mutex.

Memory:

Most of the memory is allocated to the overall database structure. In the end, everything is freed. The shared structures are also freed.

Thread Implementation:

The producer thread and all category threads interact together with the shared queue. Producer will have one node created. After that, it will try to lock the mutex for the shared queue. If it does not manage to lock the shared queue, that means one of the category threads has locked it. Since the shared queue will be empty, the category thread will send a signal to the producer thread to let it know that there is space in the shared queue. The category thread will temporarily lose ownership of the mutex until the producer thread puts one node into the shared queue. Once the producer thread does that, it will send a signal to the category thread that is waiting for a signal from the producer. That category will have access to the first node.

The producer thread and the category threads will continue to alternate, at random, control of the shared queue. As that is happening, the category threads also have the ability to lock the shared database but can only lock either shared database or shared queue so not at the same time. This will allow one thread to collect data from the shared queue while another thread will be able to store any data it is holding. After the producer thread has ended, the category threads will continue to run until the shared queue is empty. Once those category threads are done, main will finish freeing anything that was not freed. Main will also print the final report into the terminal as well as print into a finalreport.txt. If this code is run continuously, the finalreport.txt will just be overwritten.