Machine Problem 2 : Synchronization routines for SThread

Team name : Manner Maketh Man

Member : kim han sung(20112944), kim seong won(20112618)

0) your working environment and assignment details to each team member.

working environment : ubuntu 14.04 LTS

assignment

- Kim, han sung

    implement sync.c and sync.h

- Kim, seong won

    implement test code for mutex and write report

1)implementation details about each requirement

We implemented mutex. At first, 'sthread_mutex_init' set count(=key) to 0, current state to NULL. 'sthread_mutex_lock' increase count to expresses lock state. When any other thread try accessing to locked mutex, suspend it and put in to queue for sequential operation. 'sthread_mutex_trylock' checks whether lock or not. If not, thread that try accessing operates work. 'sthread_mutex_unlock' decrease count to express unlock state and wake suspended thread at front of queue to work. 'sthread_mutex_destroy' notify end, so just return 0.

2) screen shot

3) summary of "What is the synchronization in the threads ?" based on your implementation.

Synchronization in the threads is revising two more thread can access shared data with regular term. When synchronizing by using mutex, let one thread access data and locking mutex to prevent another access. Other threads are suspended. After the thread finished, destroy that thread and unlock for another thread accessing. And loop again and again. In our implement using queue for saving suspended thread. It cannot check ranking but very safe way so we used queue.