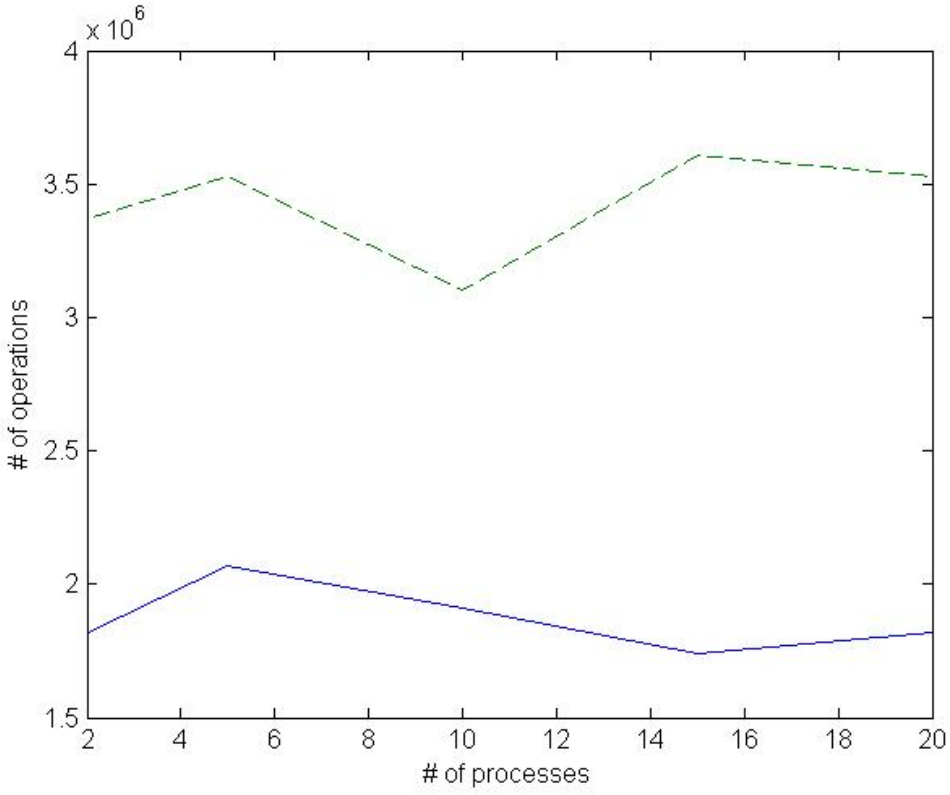
CPSC 416
Assignment 2 Report
Wei You
77610095



In the above graph, the solid line represents the 30 second benchmark while the dotted line represents the 60 second benchmark.

In order to protect the shared queue from concurrent access by the different threads, I use a mutex lock to lock the queue whenever a operation is in action. This is necessary because the program needs to know the true head and tail of the queue for dequeue and enqueue operation. The trade off of this approach is while one thread is performing an action (enqueue or dequeue) on the queue, all other threads must wait until it finishes to ensure the correctness of the queue.

There is no specific relationship between the number of operations and the number of threads used to perform the operations, because there can only be at most one thread performing an operation at a time while all other threads are busy waiting. The program does not benefits from multi-threading. The graph fluctuates because of the number of enqueue/dequeue operations performed is random. Because of the above reason, I cannot double the number of operations performed by doubling number of threads.