Assignment 3
Solving Producer Consumer Problem using Semaphores and Locks

Name : Shivashish Suman
Roll No.: CS17BTECH11037

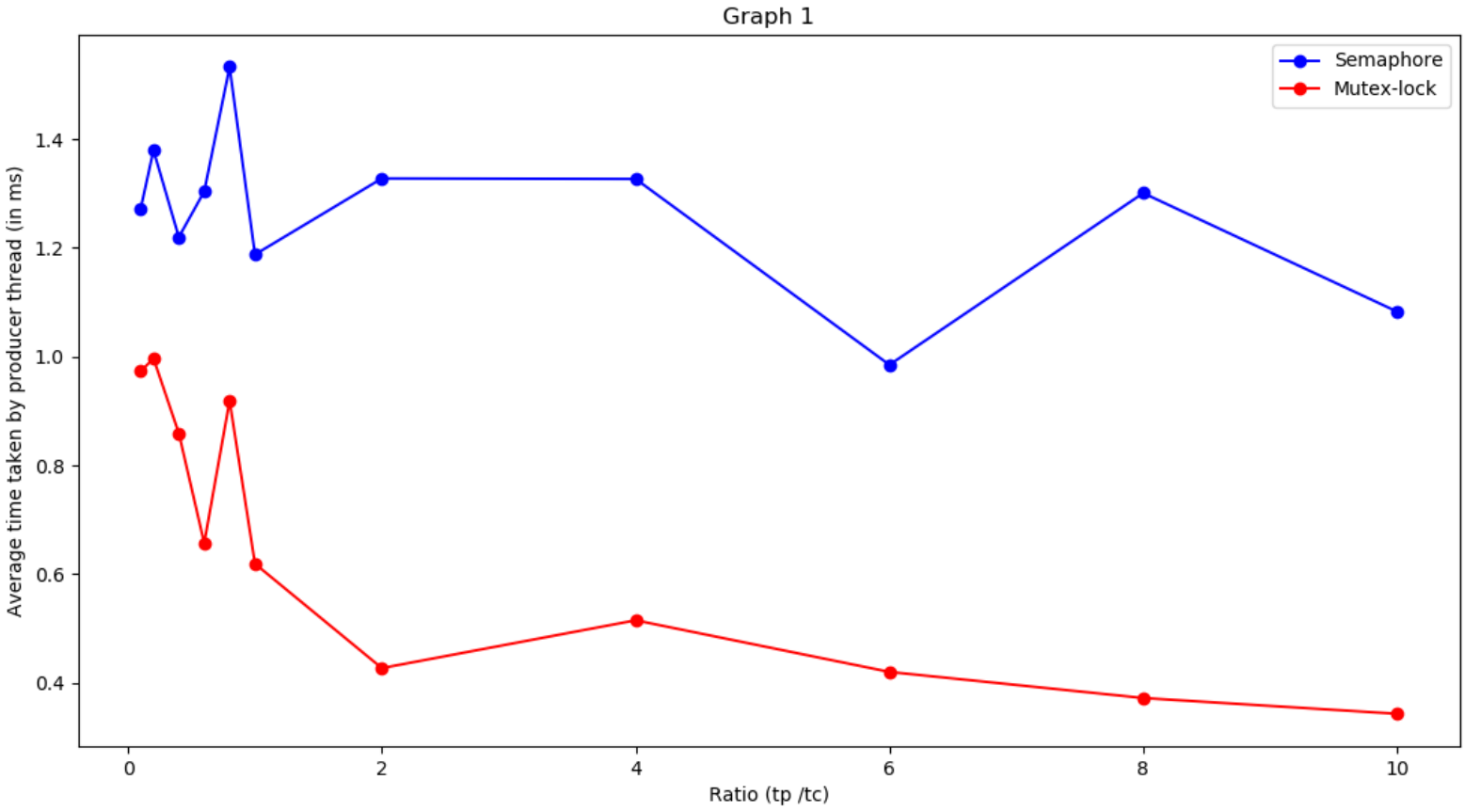
In this assignment, we have given solution of producer-consumer problem using mutex lock and semaphore.

In the program of semaphore, we have used POSIX semaphores which is initialised using init function and parameters contain the value of that counting semaphore. sem_wait() is used for waiting if semaphore is not available and when the task is done we use sem_post() to signal the semaphore.

In the program of mutex-lock, we have used pthread_mutex which is locked just before accessing the shared variable ‘count’ which maintains the count of items in the buffer and we release the lock after we read the value.

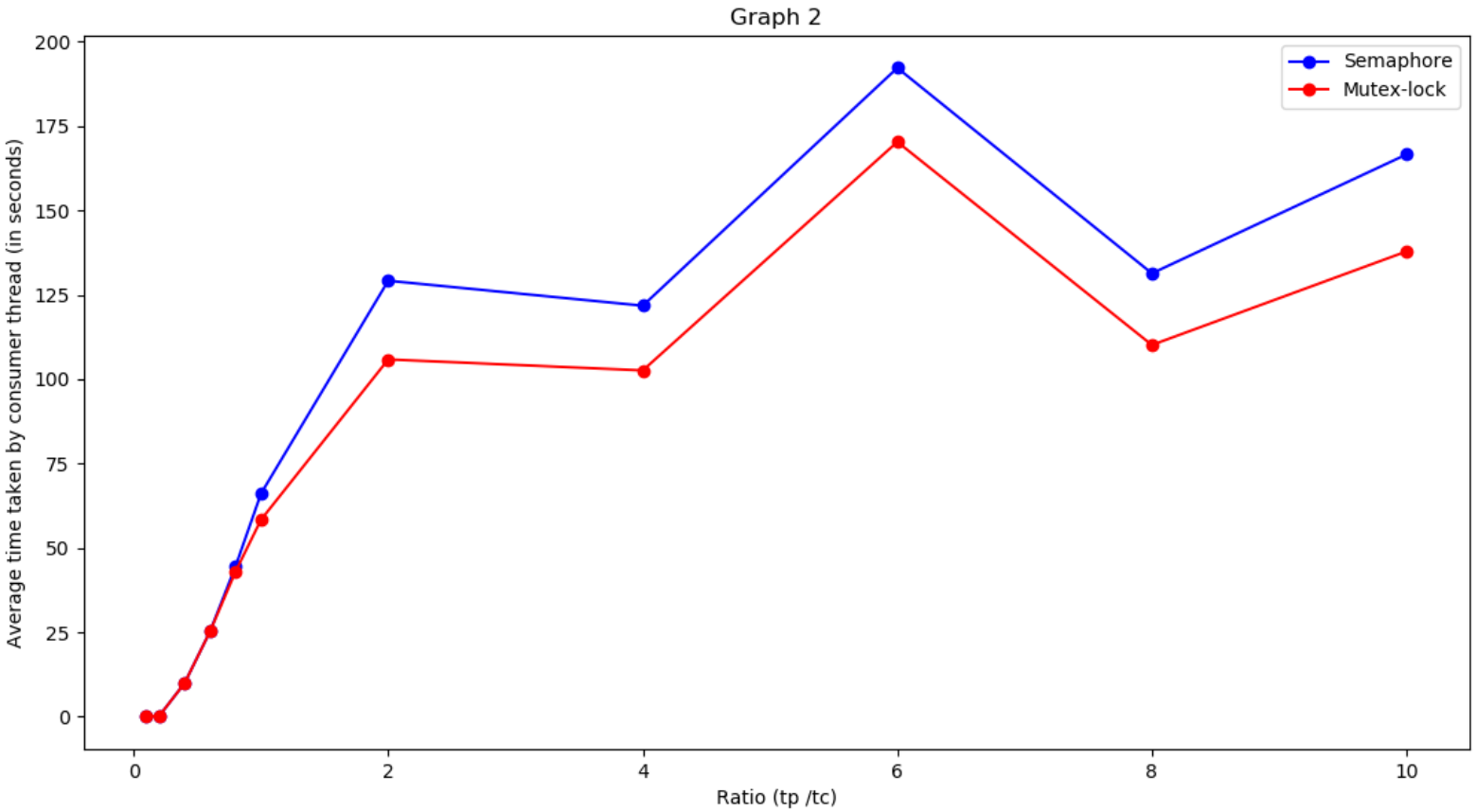
We have drawn two graphs because the average time taken by producer threads was in order of milli-seconds whereas the average time taken by consumer threads was in order of seconds.

Graph 1:
Average time taken by producer threads (in ms)



In this graph, we observe that the average time taken by producer threads in case of mutex lock is much smaller (1.5 times smaller) as compared to that of semaphores.

Graph 2:
Average time taken by consumer threads (in seconds)



In this graph, we observe that for ratio of average delay less than 0.8(i.e. for ratio of 0.1,0.2,0.4,0.6) the average waiting time for consumer threads is almost same in both the programs whereas for ratios greater than 0.8 we can observe large gap between the two programs.

The mutex-lock program performs better than semaphore one as waiting time for consumers is much less in case of mutex as compared to semaphore.