OS Assignment-3

Jakkala Naga Rohith Kumar
CS23BTECH11021

March 2025

1 Introduction

This is to compare the performance of producer and consumer threads under μ_c/μ_p and cnt_c/cnt_p.
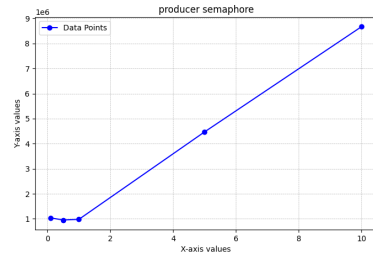
2 files
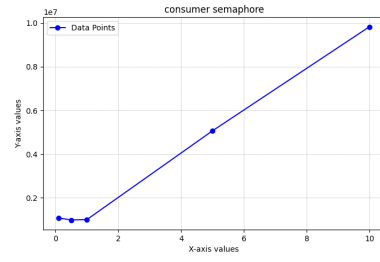
this assignment contains these files:

1. prod_cons-sems-cs23btech11021.cpp - which has implementation of semaphores

2. prod_cons-locks-cs23btech11021.cpp -which is implementation of locks

3. makefile to compile code

4. readme.txt - instructions for running program
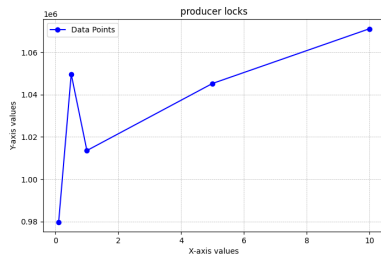
5. graph.py -to create graph

3 graphs

3.1 Under μ_c/μ_p

(a) Producer using semaphores

(b) Consumer using semaphores

(c) Producer using locks

(d) Consumer using locks

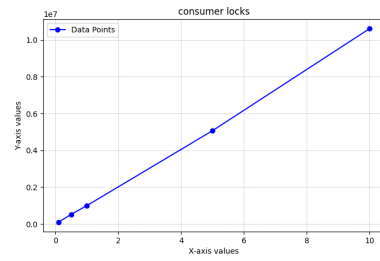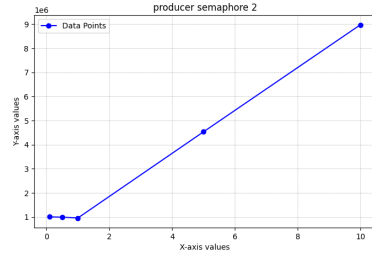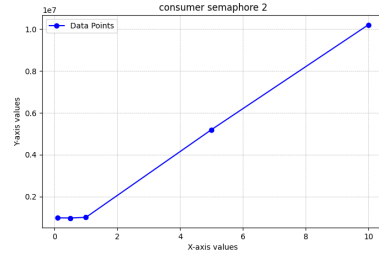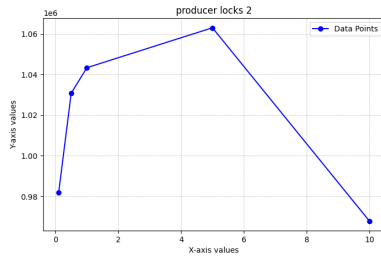Figure 1: Comparison under μ_c/μ_p
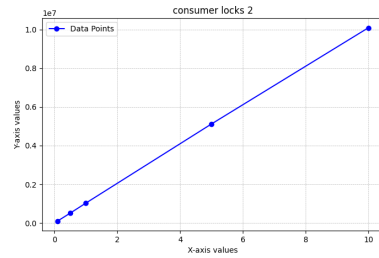
3.2 Under cnt_c/cnt_p

(a) Producer using semaphores

(b) Consumer using semaphores

(c) Producer using locks

(d) Consumer using locks

Figure 2: Comparison under cnt_c/cnt_p

4 analysis

4.1 based on μ_c/μ_p

1. producer using semaphores, consumer using semaphores first decrease and then increase with increase in cnt_c/cnt_p

2. consumer using locks are pretty much increase with increase in cnt_c/cnt_p

3. but the producer using locks is different from the other graphs

4. initially it took less time as no of tasks is less but after that it increased as its waits for the consumer to finish

4.2 based on cnt_c/cnt_p

5. producer using semaphores, consumer using semaphores first decrease and then increase with increase in cnt_c/cnt_p

6. consumer using locks are pretty much increase with increase in cnt_c/cnt_p

3

7. but the producer using locks is different from the other graphs

8. initially it took less time as no of tasks is less but after that it increased as its waits for the consumer to finish and after that it decreased

4.3 general

9. semaphores show almost equal times with locks but the semaphores are slightly betters

10. but in some cases it is almost twice as fast