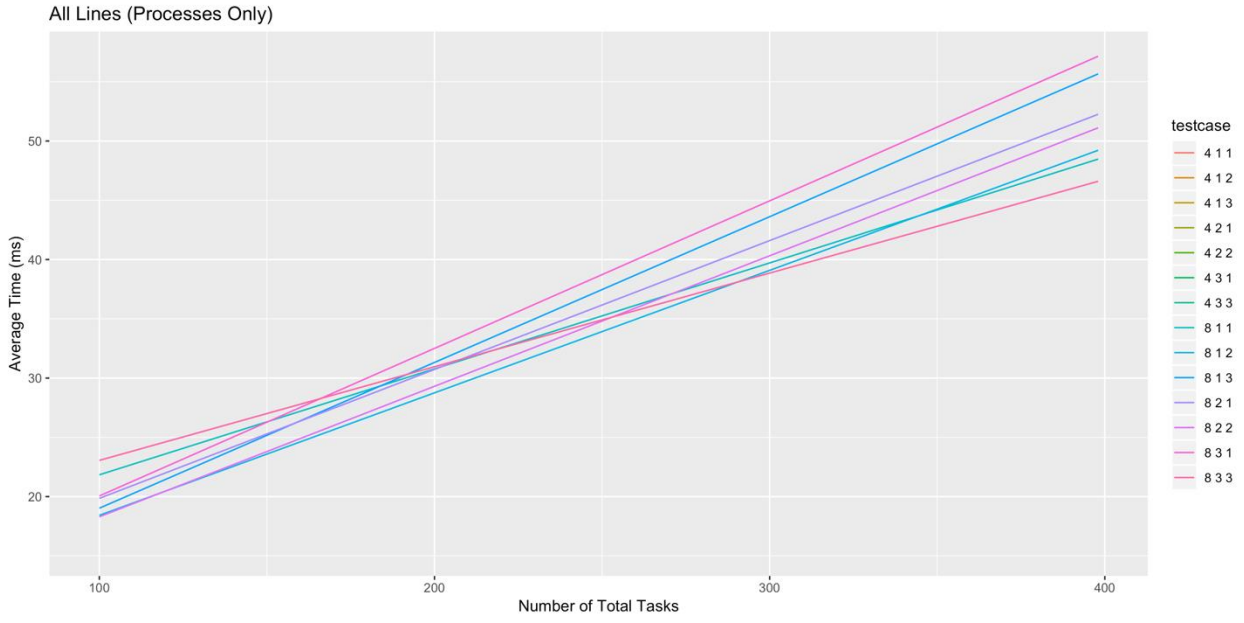
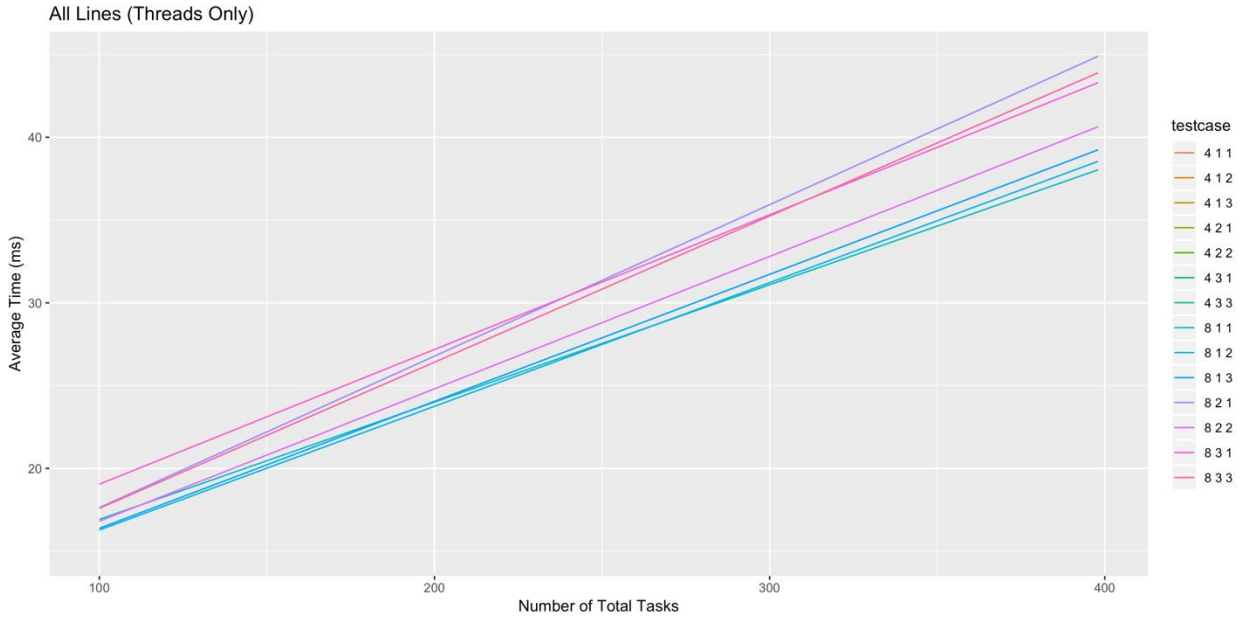
# Lab 3 Report:

Table of average timing and standard deviation measurement:

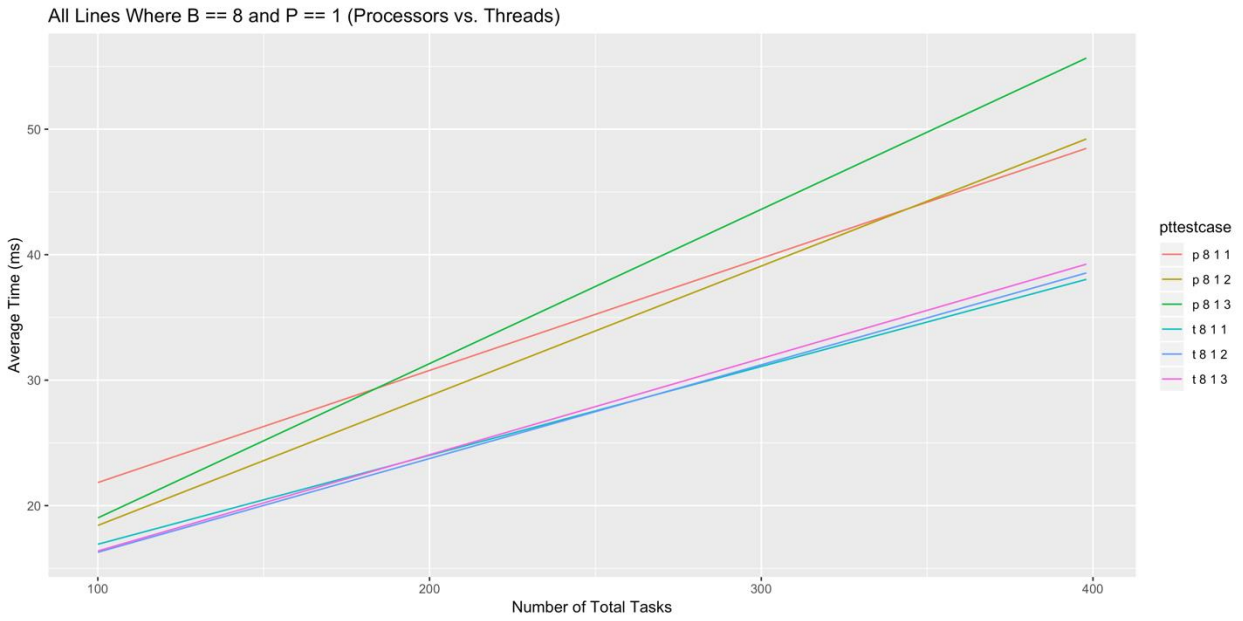
| N | B | P | C | AVG Time Processes | AVG Time Threads | STD Time Processes | STD Time Threads |
|---|---|---|---|---|---|---|---|
| 100 | 4 | 1 | 1 | 15.394694 | 18.271276 | 3.16425013 | 2.78711728 |
| 100 | 4 | 1 | 2 | 15.491816 | 16.40133 | 4.01192172 | 4.08294235 |
| 100 | 4 | 1 | 3 | 19.819686 | 15.003216 | 4.29685812 | 2.72242077 |
| 100 | 4 | 2 | 1 | 19.961146 | 17.951596 | 4.89742728 | 2.82672831 |
| 100 | 4 | 3 | 1 | 19.579208 | 18.848402 | 5.50861545 | 2.32825814 |
| 100 | 4 | 2 | 2 | 20.037968 | 17.926262 | 3.98607911 | 3.13730815 |
| 100 | 4 | 3 | 3 | 23.990856 | 17.47346 | 3.21538919 | 2.04846435 |
| 100 | 8 | 1 | 1 | 21.838822 | 16.921692 | 4.33415459 | 2.83961686 |
| 100 | 8 | 1 | 2 | 18.420256 | 16.280728 | 4.20879143 | 2.91078368 |
| 100 | 8 | 1 | 3 | 19.02039 | 16.383124 | 4.26845067 | 3.25995789 |
| 100 | 8 | 2 | 1 | 19.85311 | 17.634148 | 5.63183324 | 3.49488044 |
| 100 | 8 | 3 | 1 | 20.05484 | 19.050412 | 5.38729711 | 2.33154648 |
| 100 | 8 | 2 | 2 | 18.29362 | 16.81041 | 3.9830807 | 2.89034548 |
| 100 | 8 | 3 | 3 | 23.056076 | 17.588156 | 3.67544434 | 2.32839468 |
| 398 | 8 | 1 | 1 | 48.47887 | 38.035058 | 13.7589338 | 10.4998442 |
| 398 | 8 | 1 | 2 | 49.222884 | 38.544808 | 11.5672543 | 9.82696234 |
| 398 | 8 | 1 | 3 | 55.66906 | 39.248046 | 13.3072927 | 8.48848417 |
| 398 | 8 | 2 | 1 | 52.25754 | 44.897082 | 17.0586843 | 11.5030619 |
| 398 | 8 | 3 | 1 | 57.157522 | 43.300432 | 18.4468181 | 11.3451655 |
| 398 | 8 | 2 | 2 | 51.119446 | 40.635418 | 13.9866045 | 9.56382552 |
| 398 | 8 | 3 | 3 | 46.604294 | 43.899142 | 10.791907 | 8.64969359 |

- Compare the timing results of multi-thread with shared memory and multi-process with message queue. Discuss the advantages and disadvantages of these two approaches to solve the same problem.

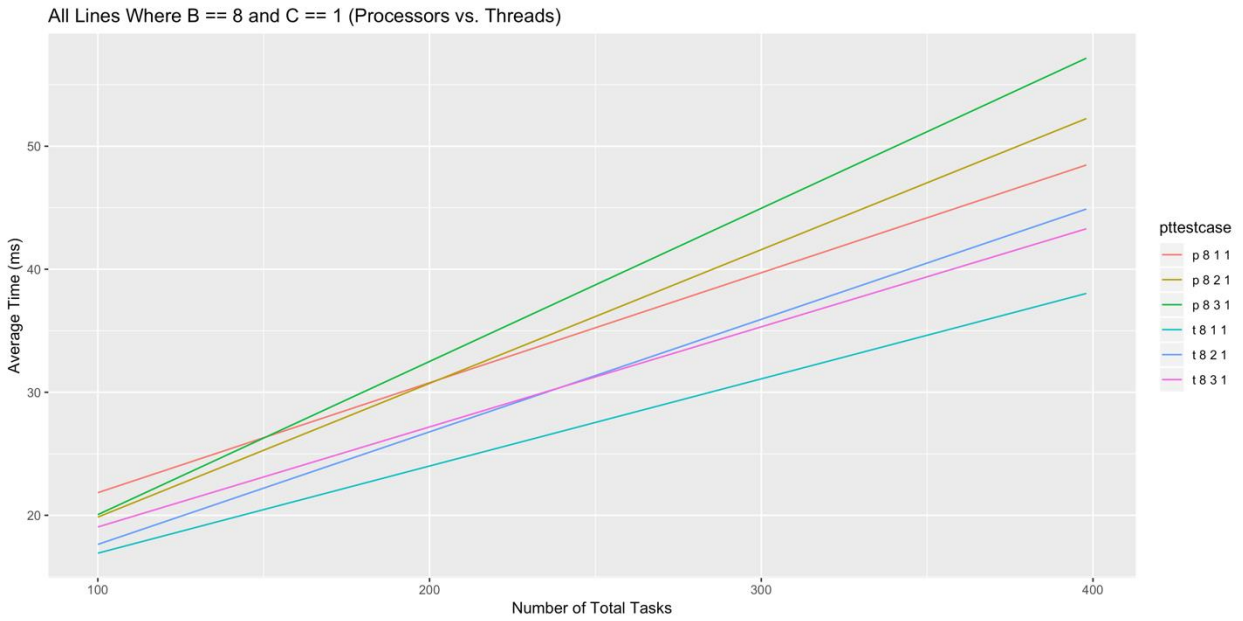


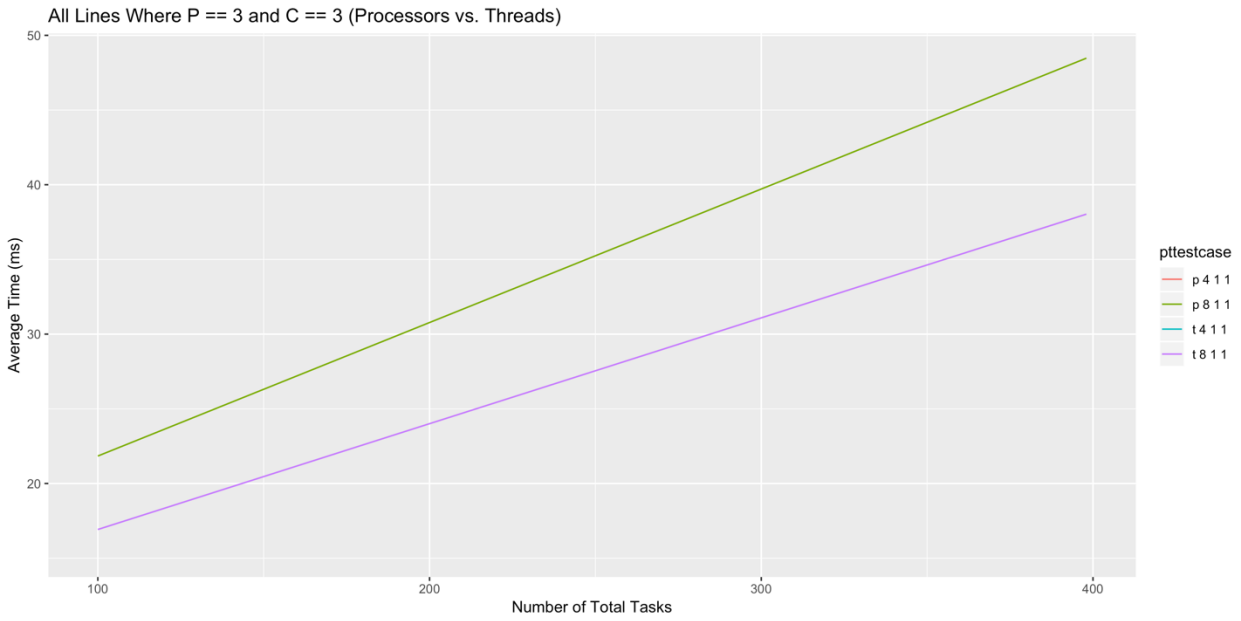

The buffer size wouldn't affect the average time for both multi-thread and multi-process.




When Buffer size is 8 and producer is 1, the average time of the threads with the shared memory is much faster than the average time of the processes with the message queue. More consumers would slow the average time.




When Buffer size is 8 and consumer is 1, the average time of the threads with the shared memory is much faster than the average time of the processes with the message queue. More producers would slow the average time.




When there are three producers and three consumers, the average time of multi-thread with shared memory is much faster than the average time of multi-process with message queue.

# Advantages and disadvantages:

### Threads with shared memory:

Shared memory is faster than message queue for read or write operation. Each access is treated as one normal memory access and therefore there is no extra CPU time being used except for switching threads. However, the data is not safe in shared memory region, which can be modified by any thread that accesses the shared memory and it's not the data owner.

### Processes with message queue:

Message queues do not utilize shared memory, therefor there is no fear of changing variables that will affect other processes. The message queue is much easier to implement compared to shared memory as we do not need to explicitly synchronize variable access. However, processes are slower when compared to multi-thread as it requires the program to essentially duplicate itself, and then in addition, the overhead of switching processes is much larger then multi-thread as there is no shared memory.

- Compare how each variable (N/B/P/C) affects the timing of the applications

  The larger number of tasks (N) would increase the average time. It would take larger time to finish all tasks, as there are more numbers to process.

The buffer size (B) wouldn’t affect the average time for both multi-thread and multi-process. This is likely due to the fact that producers produce at a rate faster then consumers can consume.

The larger number of producers (P) would increase the average time with the same number of consumers. Producer should wait for free space in buffer and get blocked. This can slow the program to finish all tasks.

A larger number of consumers (C) would increase the average time with the same number of producers. More consumers mean that the rate of consumption is higher. If the rate of consumption becomes higher then production rate, then the consumer would be blocked. The waiting time would increase the program time, as you would have the overhead of switching between all these consumer processes/threads.

The Average time can provide more accuracy for the time results. The standard deviation time represents the extent of deviation for a group of time results across 500 times. It is important to average the time over a large number of samples as at a given time the operating system may have a unpredictable amount of processes running, which can vastly cause varied time results. By averaging the time, we get a more accurate representation of the time it takes for the execution to finish.