OS ASSIGNMENT 2

GROUP 44

Name	ID No.
Harsh Vardhan Gupta	2019B3A70630H
Aryan Ramchandra Kapadia	2019B3A70412H
Dhruv Gupta	2019B3A70487H
Anand	2019B3A70613H
Anirudh Shankar	2019B5A71494H
Harshit Thakkar	2020A7PS2084H

NOVEMBER 29, 2022

CS F372: OPERATING SYSTEMS

BITS PILANI HYDERABAD CAMPUS

Table of Contents

P1: Time taken vs Number of threads

The process P1 is reading the input from the text file and putting it in the shared memory.

Figure 1 depicts the time taken in seconds versus the number of threads required for matrix multiplication of matrices with dimensions 20x40 and 40x20. The resultant matrix will be of 20x20 and the workload will be 20x40x20.

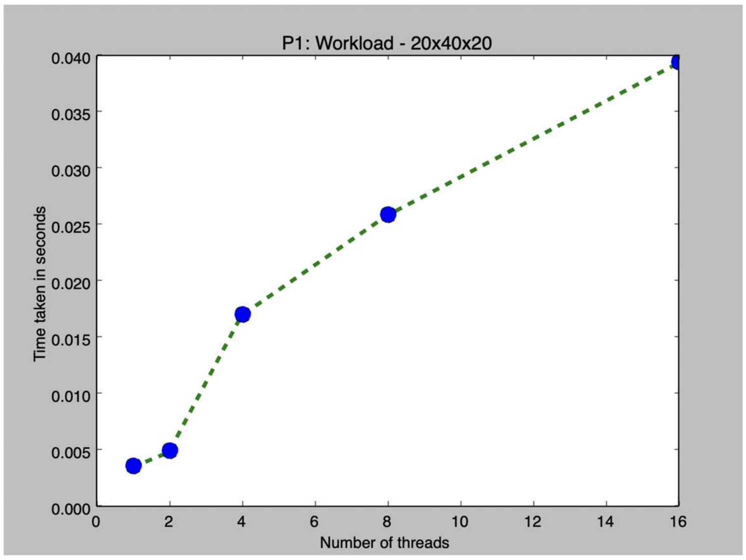

Figure 1: Time taken vs no. of threads for workload 20x40x20

Figure 2 depicts the time taken in seconds versus the number of threads required for matrix multiplication of matrices with dimensions 130x50 and 50x130. The resultant matrix will be of 130x130 and the workload will be 130x50x130.

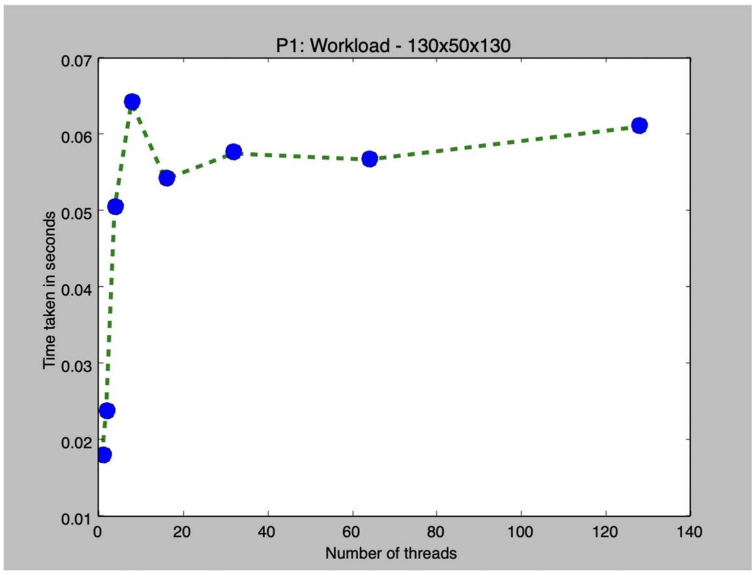

Figure 2: Time taken vs no. of threads for workload 130x50x130

Figure 3 depicts the time taken in seconds versus the number of threads required for matrix multiplication of matrices with dimensions 500x250 and 250x500. The resultant matrix will be of 500x500 and the workload will be 500x250x500.

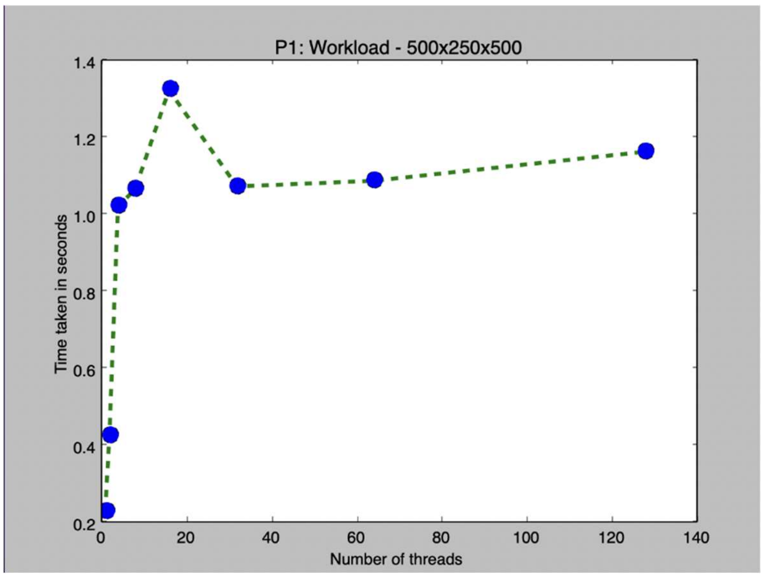

Figure 3: Time taken vs no. of threads for workload 500x250x500

Analysis of Plots

For every workload, the time taken to read a file increases as the number of threads increases. The reason is that a slowdown occurs when reading in parallel because the **magnetic hard disk head needs to seek the next read position** (taking about 5ms) for each thread. Thus, reading with multiple threads **effectively bounces the disk between seeks**, which wastes time and slows the process down. Thus, from our findings, the effective way to read a file is sequentially using one thread only.

P2: Time taken vs Number of threads

The process P2 is reading the data from shared memory and multiplying the matrices.

Figure 4 depicts the time taken in seconds versus the number of threads required for matrix multiplication of matrices with dimensions 20x40 and 40x20. The resultant matrix will be of 20x20 and the workload will be 20x40x20.

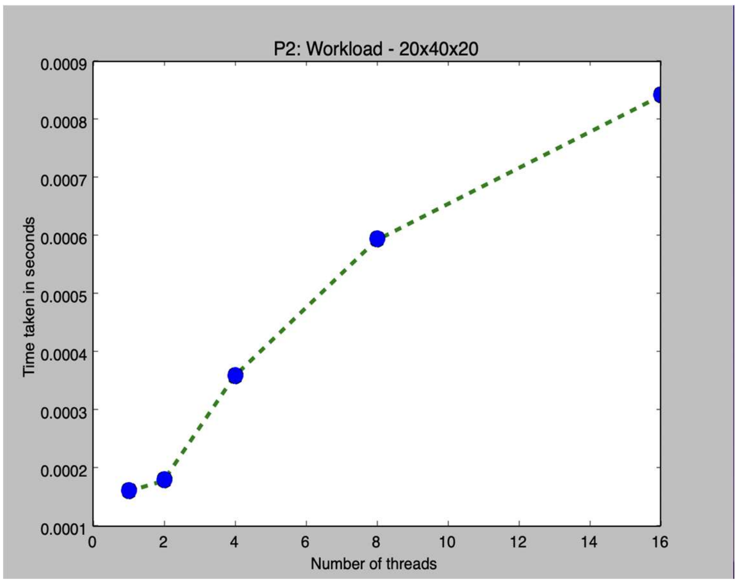

Figure 4: Time taken vs no. of threads for workload 20x40x20

Figure 5 depicts the time taken in seconds versus the number of threads required for matrix multiplication of matrices with dimensions 130x50 and 50x130. The resultant matrix will be of 130x130 and the workload will be 130x50x130.

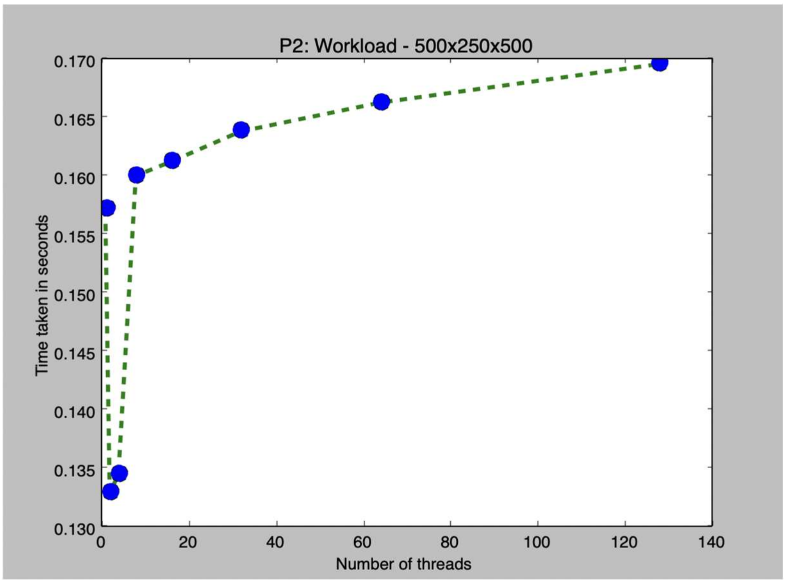

Figure 5: Time taken vs no. of threads for workload 130x50x130

Figure 6 depicts the time taken in seconds versus the number of threads required for matrix multiplication of matrices with dimensions 500x250 and 250x500. The resultant matrix will be of 500x500 and the workload will be 500x250x500.

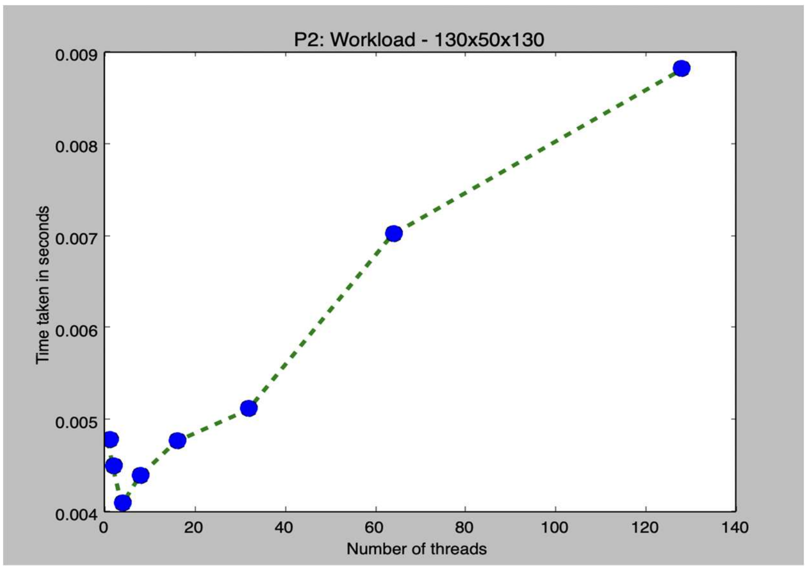

Figure 6: Time taken vs no. of threads for workload 500x250x500

Analysis of Plots

We can see for small workloads the time taken to multiply matrices increases as the number of threads increases. It is because the switching overheads negate any benefit of multithreading for such a small input.

Further, we notice that as we increase the workload there is a **substantial reduction** in time taken to finish the process when the number of threads used are **2 or 4**. But for more than 4 threads (on our system) the time taken again increases. It is because for more than 4 threads the switching overheads increase substantially and thus increase the overall time taken.

P3 : Scheduler scheduling P1 and P2 with round robin

The process P3 is scheduling P1 and P2 with round robin having time quantum 1ms and 2ms.

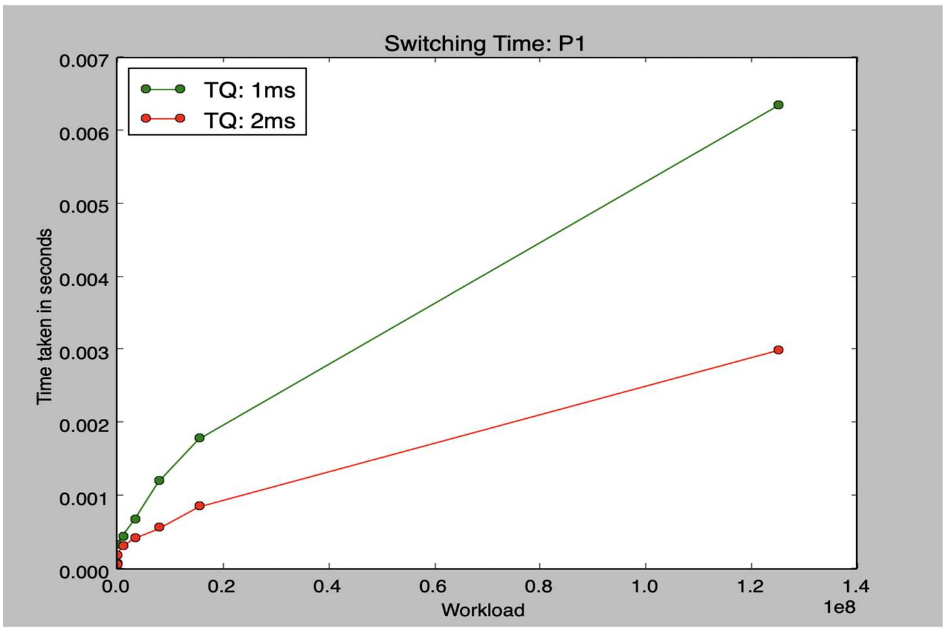

Figure 7: Switching time vs workload for P1

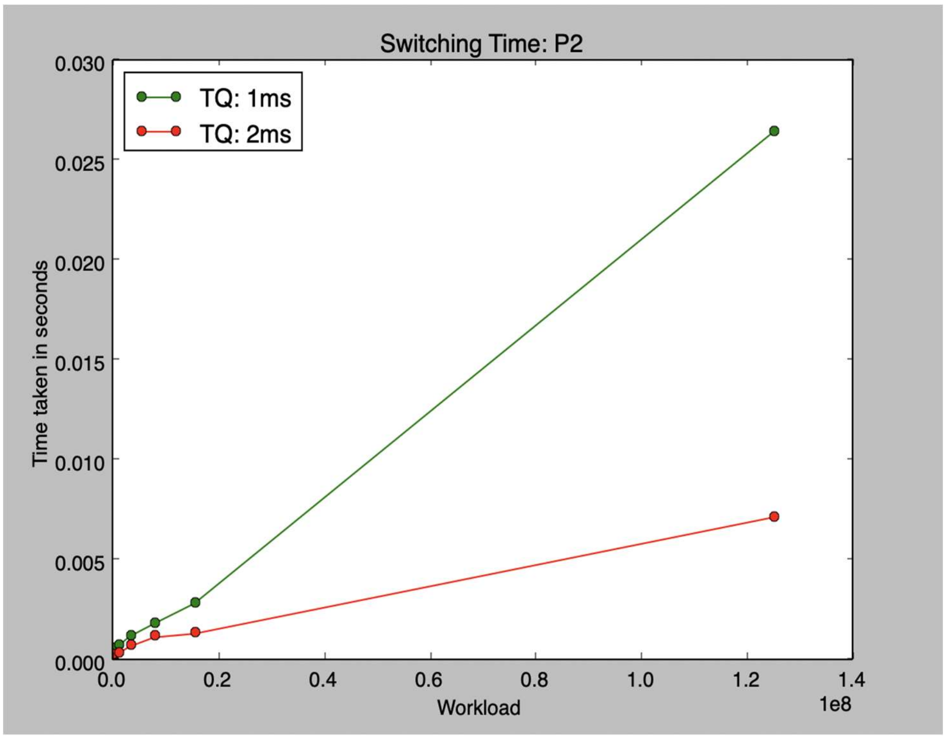

Figure 8: Switching time vs workload for P2

Analysis of Switching Time Plots

This plot shows that the switching overhead was more when the time quantum is 1ms. This makes sense as more switches are required when time quantum is 1 ms than when it is 2ms. We can also see that as the workload increases the difference increases which is as expected, because the number of switches increases for heavier workloads.

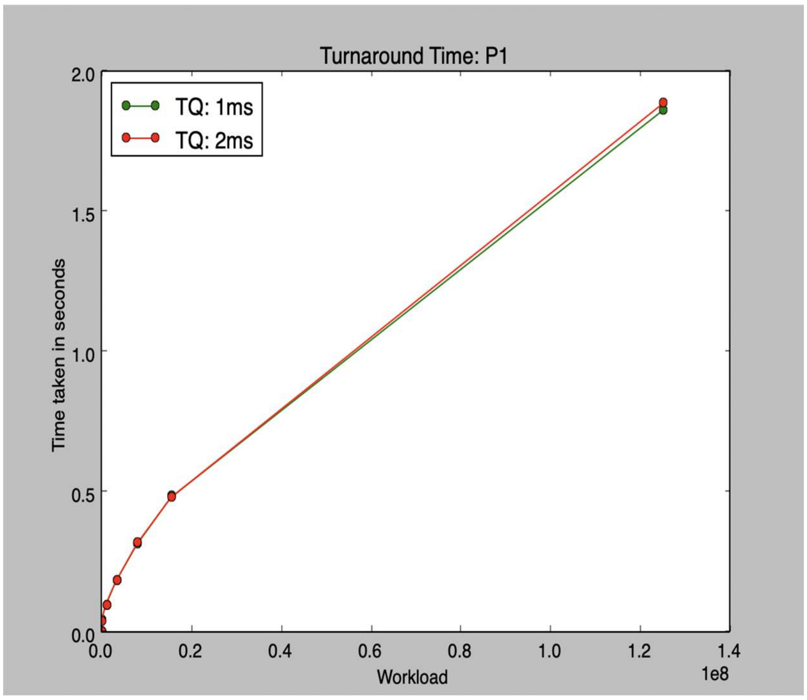

Figure 9: Turnaround time vs workload for P1

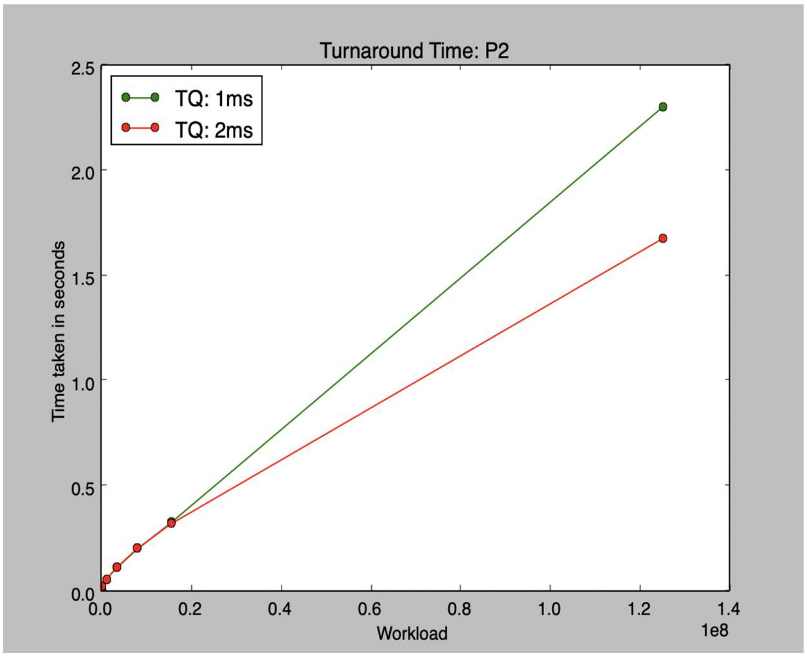

Figure 10: Turnaround time vs workload for P2

Analysis of Turnaround Time Plots

The turnaround time is almost similar in both the cases. We see an improvement in the turnaround time of process P2 for larger workloads when the time quantum is 2ms. This is because a smaller quantum size leads to CPU wastage in excessive switching between the two processes.

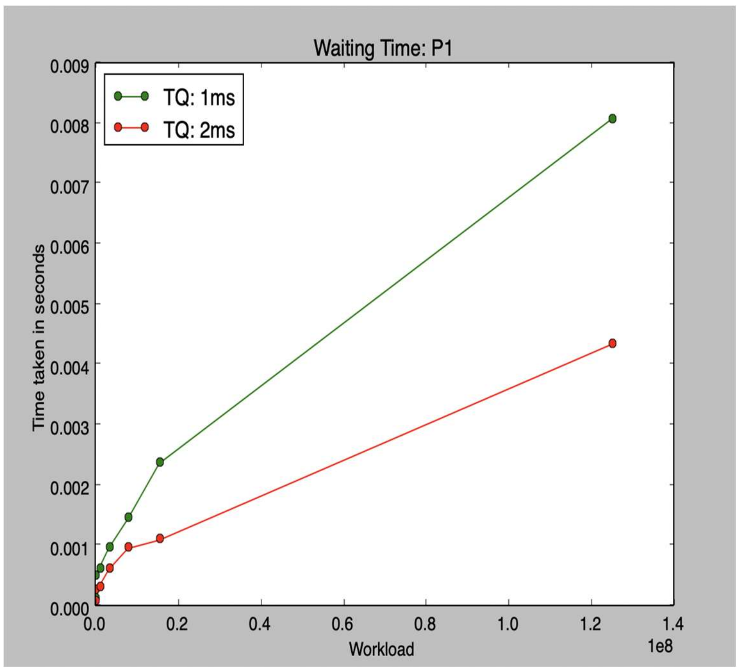

Figure 11: Waiting time vs workload for P1

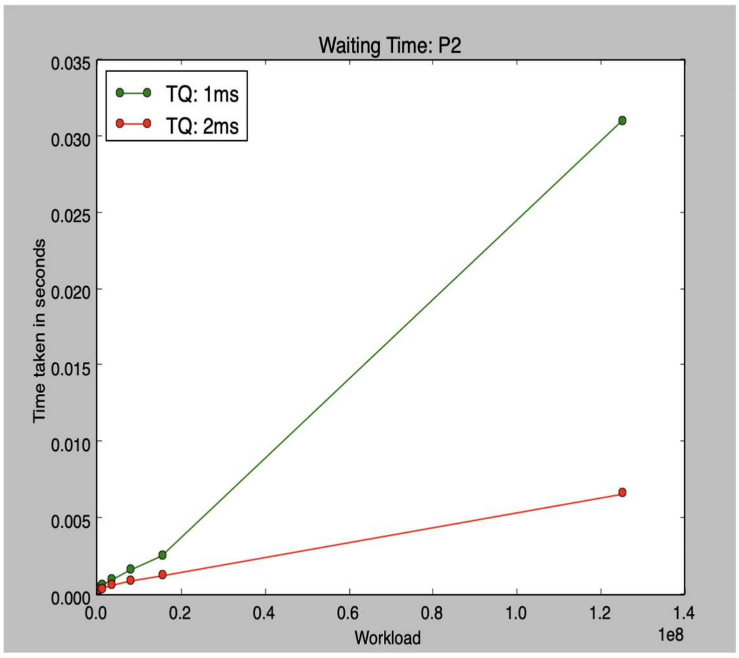

Figure 12: Waiting time vs workload for P2

Analysis of the Waiting Time Plots

The waiting time for both process P1 and P2, is more when the time quantum is 1ms. This is because a smaller time slice leads to more frequent switching and thus more overhead. This makes both P1 and P2 wait longer.

Conclusion

This report summarizes the various times taken for matrix multiplication using three processes (P1, P2, P3). P1 reads the input from text file and puts it into the shared memory. P2 reads the data from shared memory and performs matrix multiplication. P3 schedules P1 and P2 using Round Robin with time quanta 1ms and 2ms.