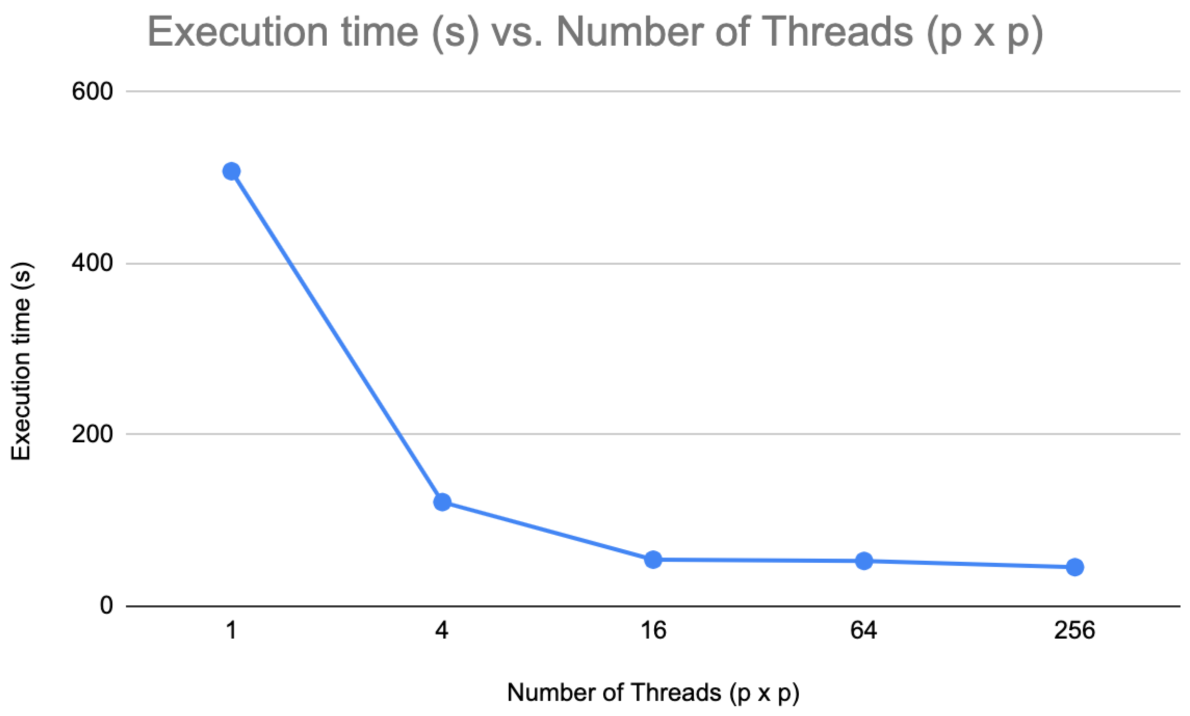
Tingqi (Ting) Wang

Problem 1: Parallel Matrix Multiplication

1.1 Local Output Variables

The following is the results of running a naive matrix multiplication of two 4K * 4K size matrixes that were parallelized using Pthreads with the number of threads being 1, 4, 16, 64, and 256.

The figures below (graph and table) illustrate the corresponding execution times in seconds for each number of threads. Evident in the graph, the execution times of the program decreases exponentially with the increase of the number of threads.



Number of Threads (p x p)	Execution time (s)
1	507.785293
4	121.588432
16	54.375915
64	52.88667
256	45.608205

Tingqi (Ting) Wang

In my program I partitioned the computation of the elements of the output matrix among the threads by dividing the total number of elements (n = 4096) by the number of threads (p x p) to ensure that each thread will take care of the same proportion of computations.

Terminal logs:

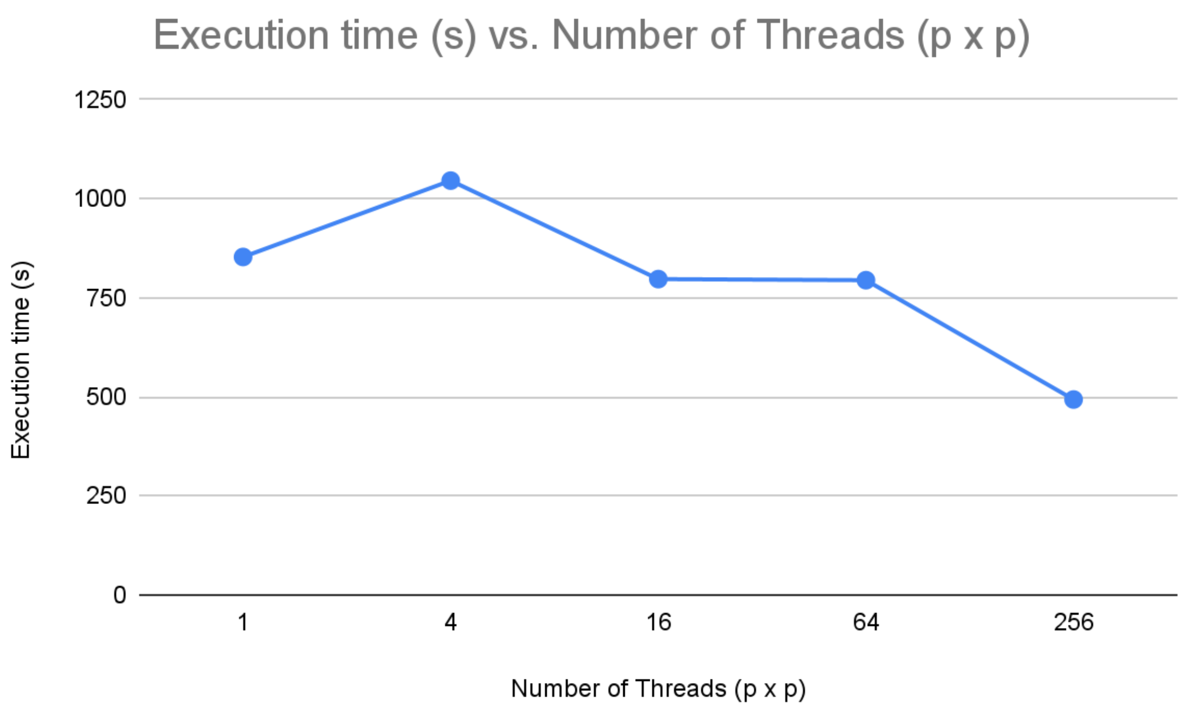
```
C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1a.exe 1
Number of FLOPs = 137438953472, Execution time = 507.785293 sec,
270.663517 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1a.exe 2
Number of FLOPs = 137438953472, Execution time = 121.588432 sec,
1130.362088 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1a.exe 4
Number of FLOPs = 137438953472, Execution time = 54.375915 sec,
2527.570400 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1a.exe 8
Number of FLOPs = 137438953472, Execution time = 52.886670 sec,
2598.744712 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1a.exe 16
Number of FLOPs = 137438953472, Execution time = 45.608205 sec,
3013.469885 MFLOPs per sec
C[100][100]=879616000.000000
```

1.2 Shared Output Variable

The following is the results of running a matrix multiplication of two 4K * 4K size matrixes that were parallelized using Pthreads with the number of threads being 1, 4, 16, 64, and 256. But this program has shared output variables and uses mutexes to prevent race conditions from happening.

Tingqi (Ting) Wang

The figures below (graph and table) illustrate the corresponding execution times in seconds for each number of threads. Looking at the graph, it can be seen that the runtime of the program first increases with the increase of threads from 1 to 4. After the increase however, the runtime decreases linearly with respect to the increase of the number of threads.



Number of Threads (p x p)	Execution time (s)
1	852.866209
4	1045.51414
16	797.013548
64	794.292303
256	493.166887

Similarly with the previous program, I partitioned the computation of the elements of the output matrix among the threads by dividing the total number of elements (n = 4096) by the number of threads (p x p) to ensure that each thread will take care of the same proportion of computations.

Terminal logs:

Tingqi (Ting) Wang

```
C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1b.exe 1
Number of FLOPs = 137438953472, Execution time = 852.866209 sec,
161.149489 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1b.exe 2
Number of FLOPs = 137438953472, Execution time = 1045.514140 sec,
131.455853 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1b.exe 4
Number of FLOPs = 137438953472, Execution time = 797.013548 sec,
172.442431 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1b.exe 8
Number of FLOPs = 137438953472, Execution time = 794.292303 sec,
173.033218 MFLOPs per sec
C[100][100]=879616000.000000

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p1b.exe 16
Number of FLOPs = 137438953472, Execution time = 493.166887 sec,
278.686499 MFLOPs per sec
C[100][100]=879616000.000000
```

Comparison:

The execution time for problem 1.1 was less than problem 1.2 for all numbers of threads. In addition, there was a continuous exponential decrease for problem 1.1 as the number of threads increases, but for problem 1.2 there was first an increase in execution time followed by an unconstant decrease in execution time. The reason for the overall higher execution time in problem 1.2 is due to the overhead times of acquiring and releasing the mutexes to avoid race conditions. The initial increase in execution time for problem 1.2 was also a result of significant increases in overhead from 1 thread to 4 threads.

Problem 2: Parallel K-Means

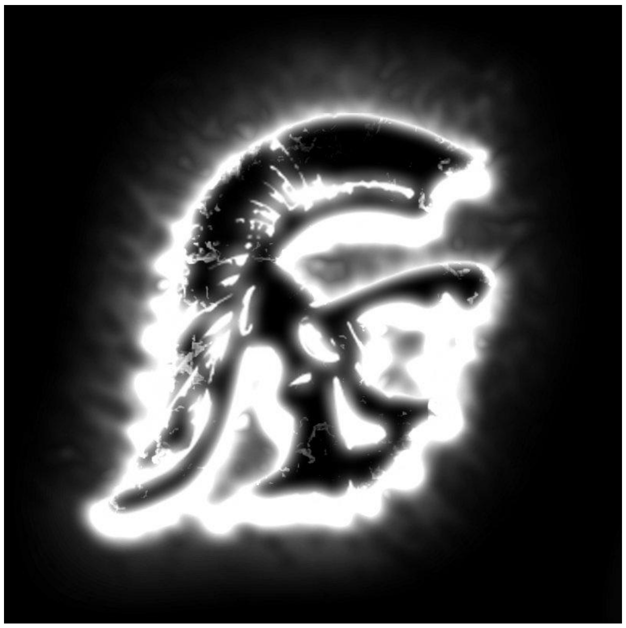
Tingqi (Ting) Wang

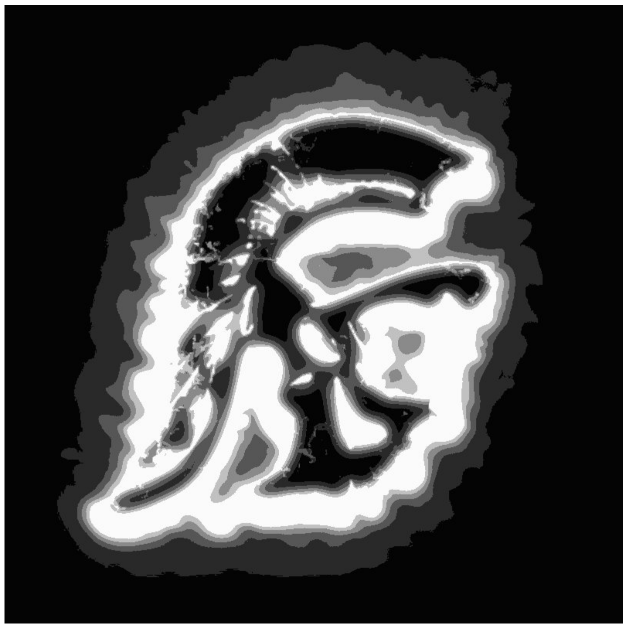
The following is the result of using Pthreads to parallelize the K-Means algorithm from PHW1 whilst also increasing the number of clusters from 4 to 6 and increasing the number of iterations from 30 to 50.

The runtime of the original serial program was 0.286369 sec
The runtime of the parallelized program with 4 threads was 0.237486 sec
The runtime of the parallelized program with 8 threads was 0.165518 sec

As the number of threads increased, the runtime of the program decreased linearly.

The images below demonstrate the images generated using imageJ from input.raw (left) and output.raw (right) files.





Terminal logs:

```
C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p2.exe 1
Time taken to run: 0.286369 sec

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p2.exe 4
Time taken to run: 0.237486 sec

C:\Users\tingt\OneDrive\Documents\USC\EE
451\EE451_HW\EE_451_F_2023_PHW_2\EE_451_S_2023_PHW_2>p2.exe 8
```

Tingqi (Ting) Wang

Time taken to run: 0.165518 sec