Jessica Salguero, UIN: 937005006
CSCE 735, Summer 2025

Question 1:

1. (75 points) Parallelize the code by inserting OpenMP directives to obtain a parallel implementation of Strassen's recursive algorithm.

Parallelized functions:

- strassens_product(Matrix A, Matrix B)
- standard_product(Matrix& A, Matrix& B)
- addition(Matrix& A, Matrix& B)
- subtraction(Matrix& A, Matrix& B)

Jessica Salguero, UIN: 937005006
CSCE 735, Summer 2025

Question 2

2. (25 points) Determine the speedup obtained by your code on a single node of Grace using all available cores for matrixes of size 2 k for k = 10,…,14. Speedup should be computed as the speed improvement over Strassen's algorithm using only a single thread. Experiment with the size of the leaf matrix 2 q to determine which size(s) give you the maximum speedup. Summarize your findings in a document that includes speedup and efficiency graphs as well as your insights into the results you have obtained. Lastly, include a brief description of how to compile and execute the code on Grace.

From one of the professor's responses on Piazza, he stated that "You do not need to check correctness for larger matrices if there is no error for smaller sizes". Because of this, I tested parallelization on k = 10, q = 3, 4, 5, 6. Since Error = 0, I was able to comment out standard_product() (in main line 284) to reduce runtime for the remaining combinations of k and q values. I uncommented this line for my final submission.

```
k = 10
Matrix size = 1024, Leaf matrix size = 8, Strassen's (s) =   3.2127 s, Standard =   0.2821 s, Error = 0
Matrix size = 1024, Leaf matrix size = 16, Strassen's (s) =   1.9435 s, Standard =   1.0397 s, Error = 0
Matrix size = 1024, Leaf matrix size = 32, Strassen's (s) =   0.7310 s, Standard =   0.8005 s, Error = 0
Matrix size = 1024, Leaf matrix size = 64, Strassen's (s) =   1.1887 s, Standard =   0.2699 s, Error = 0
```

I reran all combinations of k = 10, 11, 12, 13 and q = 3, 4, 5, 6 with this standard_product() line commented out. I was able to successfully run a majority of these combinations when k = 10, 11, 12. When k = 13, I was unsuccessful with q = 3.
The sbatch runs would consistently error out for k = 13 and 14 with these q values. I used OMP_NUM_THREADS = 48 for all the available cores on Grace, and used OMP_NUM_THREADS = 1 to represent serial implementation of this code.

```
OMP_NUM_THREADS=48
k = 10, OMP_NUM_THREADS=48
Matrix size = 1024, Leaf matrix size = 8, Strassen's (s) =   3.3804 s, Standard =   0.4508 s, Error = 0
Matrix size = 1024, Leaf matrix size = 16, Strassen's (s) =   2.1654 s, Standard =   0.2389 s, Error = 0
Matrix size = 1024, Leaf matrix size = 32, Strassen's (s) =   1.0093 s, Standard =   0.1019 s, Error = 0
Matrix size = 1024, Leaf matrix size = 64, Strassen's (s) =   0.4570 s, Standard =   0.5272 s, Error = 0

k=11, OMP_NUM_THREADS=48
Matrix size = 2048, Leaf matrix size = 8, Strassen's (s) =  28.7563 s, Standard =   2.4228 s, Error = 0
Matrix size = 2048, Leaf matrix size = 16, Strassen's (s) =   8.7628 s, Standard =   1.4637 s, Error = 0
Matrix size = 2048, Leaf matrix size = 32, Strassen's (s) =   3.9111 s, Standard =   1.6309 s, Error = 0
Matrix size = 2048, Leaf matrix size = 64, Strassen's (s) =   1.8288 s, Standard =   1.5555 s, Error = 0

k=12, OMP_NUM_THREADS=48
/var/spool/slurmd/job15630781/slurm_script: line 44: 447400 Killed                  ./strassen_omp.exe 12
3
Matrix size = 4096, Leaf matrix size = 16, Strassen's (s) =  89.4855 s, Standard =  34.8302 s, Error = 0
Matrix size = 4096, Leaf matrix size = 32, Strassen's (s) =  68.4575 s, Standard =  34.8581 s, Error = 0
Matrix size = 4096, Leaf matrix size = 64, Strassen's (s) =  18.5299 s, Standard =  34.0378 s, Error = 0

k=13, OMP_NUM_THREADS=48
/var/spool/slurmd/job15630781/slurm_script: line 51: 447988 Killed                  ./strassen_omp.exe 13
3
/var/spool/slurmd/job15630781/slurm_script: line 52: 448223 Killed                  ./strassen_omp.exe 13
4
/var/spool/slurmd/job15630781/slurm_script: line 53: 448375 Killed                  ./strassen_omp.exe 13
5
/var/spool/slurmd/job15630781/slurm_script: line 54: 448579 Killed                  ./strassen_omp.exe 13
6
slurmstepd: error: Detected 5 oom_kill events in StepId=15630781.batch. Some of the step tasks have been
OOM Killed.
```

```
OMP_NUM_THREADS=1
k = 10, OMP_NUM_THREADS=1
Matrix size = 1024, Leaf matrix size = 8, Strassen's (s) =   6.0927 s, Standard =   3.4161 s, Error = 0
Matrix size = 1024, Leaf matrix size = 16, Strassen's (s) =   3.2023 s, Standard =   3.5692 s, Error = 0
Matrix size = 1024, Leaf matrix size = 32, Strassen's (s) =   2.3159 s, Standard =   3.6759 s, Error = 0
Matrix size = 1024, Leaf matrix size = 64, Strassen's (s) =   2.0483 s, Standard =   3.3775 s, Error = 0

k=11, OMP_NUM_THREADS=1
Matrix size = 2048, Leaf matrix size = 8, Strassen's (s) =  37.1615 s, Standard =  32.3633 s, Error = 0
Matrix size = 2048, Leaf matrix size = 16, Strassen's (s) =  18.5672 s, Standard =  30.8797 s, Error = 0
Matrix size = 2048, Leaf matrix size = 32, Strassen's (s) =  11.9272 s, Standard =  37.5663 s, Error = 0
Matrix size = 2048, Leaf matrix size = 64, Strassen's (s) =   9.1074 s, Standard =  37.3189 s, Error = 0

k=12, OMP_NUM_THREADS=1
/var/spool/slurmd/job15630785/slurm_script: line 44: 3409554 Killed                  ./strassen_omp.exe
12 3
Matrix size = 4096, Leaf matrix size = 16, Strassen's (s) = 124.1060 s, Standard = 642.8457 s, Error = 0
Matrix size = 4096, Leaf matrix size = 32, Strassen's (s) =  78.4156 s, Standard = 639.0218 s, Error = 0
Matrix size = 4096, Leaf matrix size = 64, Strassen's (s) =  64.1862 s, Standard = 641.5489 s, Error = 0

k=13, OMP_NUM_THREADS=1
/var/spool/slurmd/job15630785/slurm_script: line 51: 3411159 Killed                  ./strassen_omp.exe
13 3
/var/spool/slurmd/job15630785/slurm_script: line 52: 3411322 Killed                  ./strassen_omp.exe
13 4
/var/spool/slurmd/job15630785/slurm_script: line 53: 3411448 Killed                  ./strassen_omp.exe
13 5
/var/spool/slurmd/job15630785/slurm_script: line 54: 3411634 Killed                  ./strassen_omp.exe
13 6
slurmstepd: error: Detected 5 oom_kill events in StepId=15630785.batch. Some of the step tasks have been
OOM Killed.
```

After I saw that these initial q values did not work with k = 13 and 14, I decided to try a larger q value. When using q = 7, successful runs occurred with k = 10, 11, 12, 13.

```
OMP_NUM_THREADS=48
k = 10, OMP_NUM_THREADS=48
Matrix size = 1024, Leaf matrix size = 128, Strassen's (s) =   1.2919 s, Standard =   0.3647 s, Error = 0

k=11, OMP_NUM_THREADS=48
Matrix size = 2048, Leaf matrix size = 128, Strassen's (s) =   1.9629 s, Standard =   2.7176 s, Error = 0

k=12, OMP_NUM_THREADS=48
Matrix size = 4096, Leaf matrix size = 128, Strassen's (s) =  12.8655 s, Standard =  30.9583 s, Error = 0

k=13, OMP_NUM_THREADS=48
Matrix size = 8192, Leaf matrix size = 128, Strassen's (s) = 243.6189 s, Standard = 310.9115 s, Error = 0
```

```
OMP_NUM_THREADS=1
k = 10, OMP_NUM_THREADS=1
Matrix size = 1024, Leaf matrix size = 128, Strassen's (s) =   1.5798 s, Standard =   3.8135 s, Error = 0

k=11, OMP_NUM_THREADS=1
Matrix size = 2048, Leaf matrix size = 128, Strassen's (s) =  10.4280 s, Standard =  43.5584 s, Error = 0

k=12, OMP_NUM_THREADS=1
Matrix size = 4096, Leaf matrix size = 128, Strassen's (s) =  65.4939 s, Standard = 635.8068 s, Error = 0

k=13, OMP_NUM_THREADS=1
Matrix size = 8192, Leaf matrix size = 128, Strassen's (s) = 431.5043 s, Standard = 6531.0746 s, Error =
0
```

I was able to see better results when adjusting combinations of k and q values. However, I was unable to determine successful combinations when k = 14. I started with the same initial values where q = 3, 4, 5, 6 but these consistently errored out.

```
OMP_NUM_THREADS=48
k=14, OMP_NUM_THREADS=48
/var/spool/slurmd/job15630789/slurm_script: line 30: 1336137 Killed                  ./strassen_omp.exe
14 3
/var/spool/slurmd/job15630789/slurm_script: line 31: 1336350 Killed                  ./strassen_omp.exe
14 4
/var/spool/slurmd/job15630789/slurm_script: line 32: 1336558 Killed                  ./strassen_omp.exe
14 5
/var/spool/slurmd/job15630789/slurm_script: line 33: 1336687 Killed                  ./strassen_omp.exe
14 6
slurmstepd: error: Detected 4 oom_kill events in StepId=15630789.batch. Some of the step tasks have been
OOM Killed.
```

```
OMP_NUM_THREADS=1
k=14, OMP_NUM_THREADS=1
/var/spool/slurmd/job15630791/slurm_script: line 30: 2399994 Killed                  ./strassen_omp.exe
14 3
/var/spool/slurmd/job15630791/slurm_script: line 31: 2400152 Killed                  ./strassen_omp.exe
14 4
/var/spool/slurmd/job15630791/slurm_script: line 32: 2400282 Killed                  ./strassen_omp.exe
14 5
/var/spool/slurmd/job15630791/slurm_script: line 33: 2400413 Killed                  ./strassen_omp.exe
14 6
slurmstepd: error: Detected 4 oom_kill events in StepId=15630791.batch. Some of the step tasks have been
OOM Killed.
```

I attempted to see if higher values of q could potentially improve results. However, when using k = 14, and q = 6,7 I was again met with unsuccessful runs that were killed due to insufficient memory.

```
OMP_NUM_THREADS=48
k=14, OMP_NUM_THREADS=48
/var/spool/slurmd/job15633009/slurm_script: line 30: 1862980 Killed                  ./strassen_omp.exe
14 6
/var/spool/slurmd/job15633009/slurm_script: line 31: 1863149 Killed                  ./strassen_omp.exe
14 7
slurmstepd: error: Detected 2 oom_kill events in StepId=15633009.batch. Some of the step tasks have been
OOM Killed.
```

```
OMP_NUM_THREADS=1
k=14, OMP_NUM_THREADS=1
/var/spool/slurmd/job15633011/slurm_script: line 30: 4044621 Killed                  ./strassen_omp.exe
14 6
/var/spool/slurmd/job15633011/slurm_script: line 31: 4044806 Killed                  ./strassen_omp.exe
14 7
slurmstepd: error: Detected 2 oom_kill events in StepId=15633011.batch. Some of the step tasks have been
OOM Killed.
```

Jessica Salguero, UIN: 937005006
CSCE 735, Summer 2025

Question 2 Table

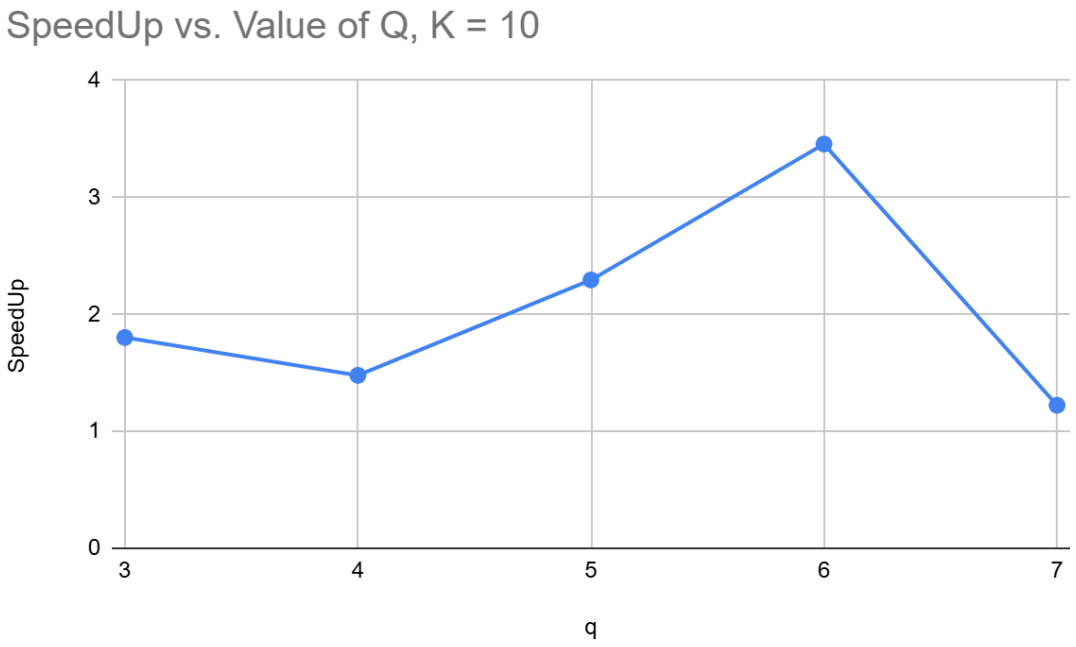
T1 Execution Time	P (# of Threads)	k	q	Tp Execution Time	SpeedUp	Efficiency
6.0927	48	10	3	3.3804	1.802360667	0.03754918057
3.2023	48	10	4	2.1654	1.478849173	0.03080935778
2.3159	48	10	5	1.0093	2.294560587	0.04780334555
2.0483	48	10	6	0.457	3.456892779	0.07201859956
1.5798	48	10	7	1.2919	1.222850066	0.02547604304
37.1615	48	11	3	28.7563	1.292290733	0.0269227236
18.5672	48	11	4	8.7628	2.118866116	0.04414304408
11.9272	48	11	5	3.9111	3.049576845	0.06353285095
9.1074	48	11	6	1.8288	4.979986877	0.1037497266
10.428	48	11	7	1.9629	5.312547761	0.1106780784
124.106	48	12	4	89.4855	1.386883909	0.02889341476
78.4156	48	12	5	68.4575	1.145463974	0.02386383279
64.1862	48	12	6	18.5299	3.463925871	0.07216512232
65.4939	48	12	7	12.8655	5.09066107	0.106055439
431.5043	48	13	7	243.6189	1.771226699	0.03690055622

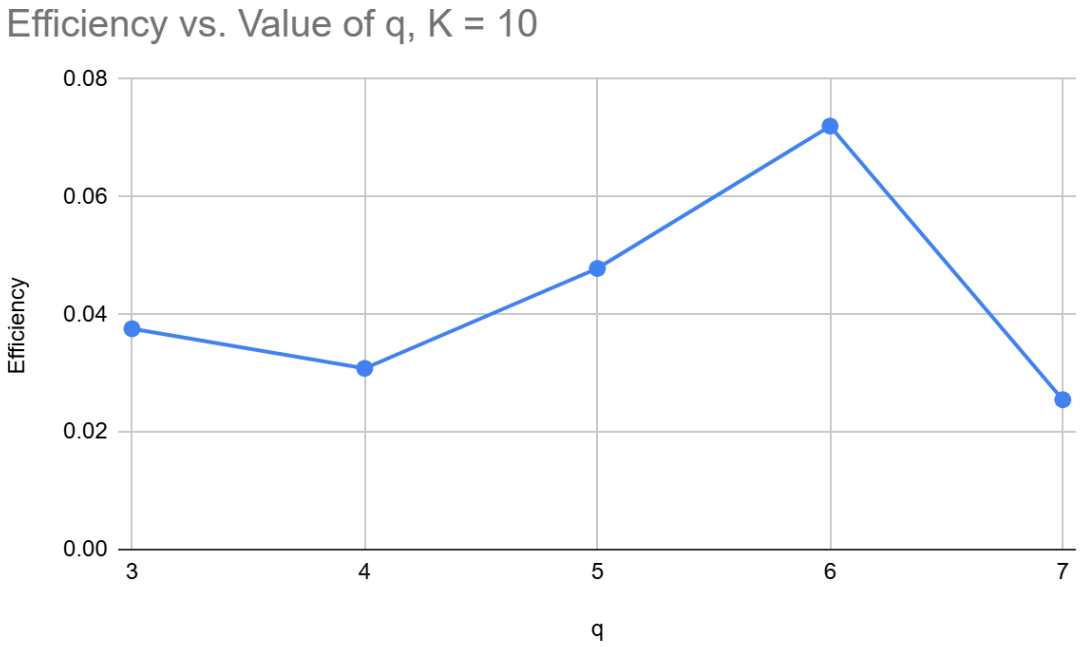
Jessica Salguero, UIN: 937005006
CSCE 735, Summer 2025

Question 2 Graphs

K = 10

When K = 10, we have a relatively small matrix size. SpeedUp gradually increases, with the best results being seen when q = 6. This is likely because the recursion is shallow enough that overhead and parallelism can strike a good balance. Then when q = 7, performance is hindered again and speedup drops drastically. This is likely because parallelism becomes less effective. Efficiency follows a very similar trend as speedup.

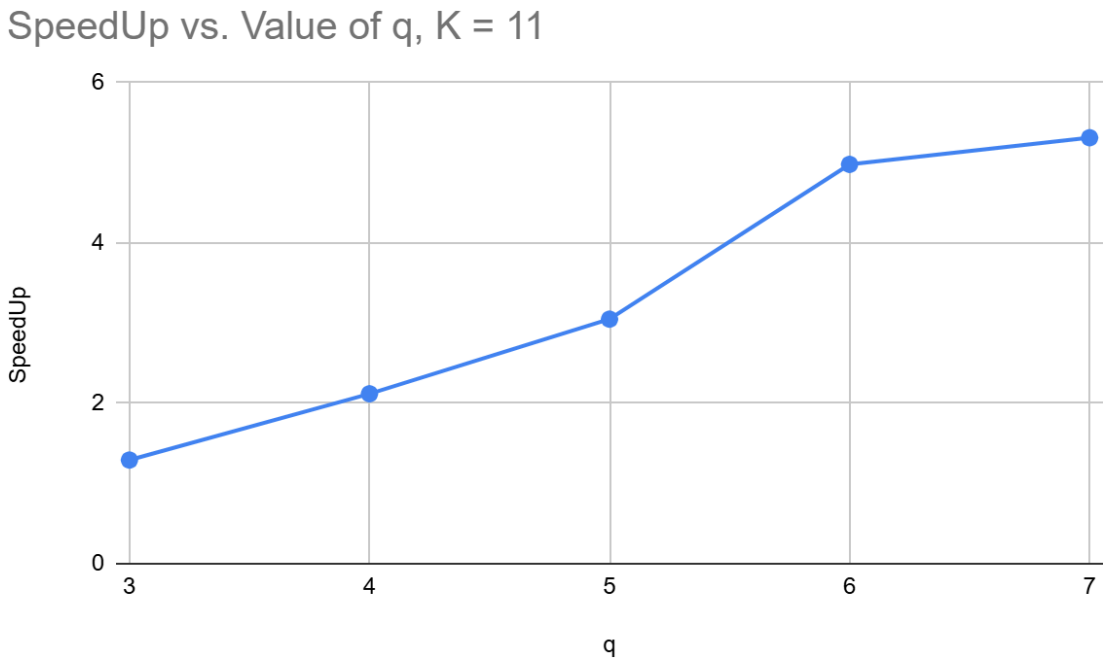


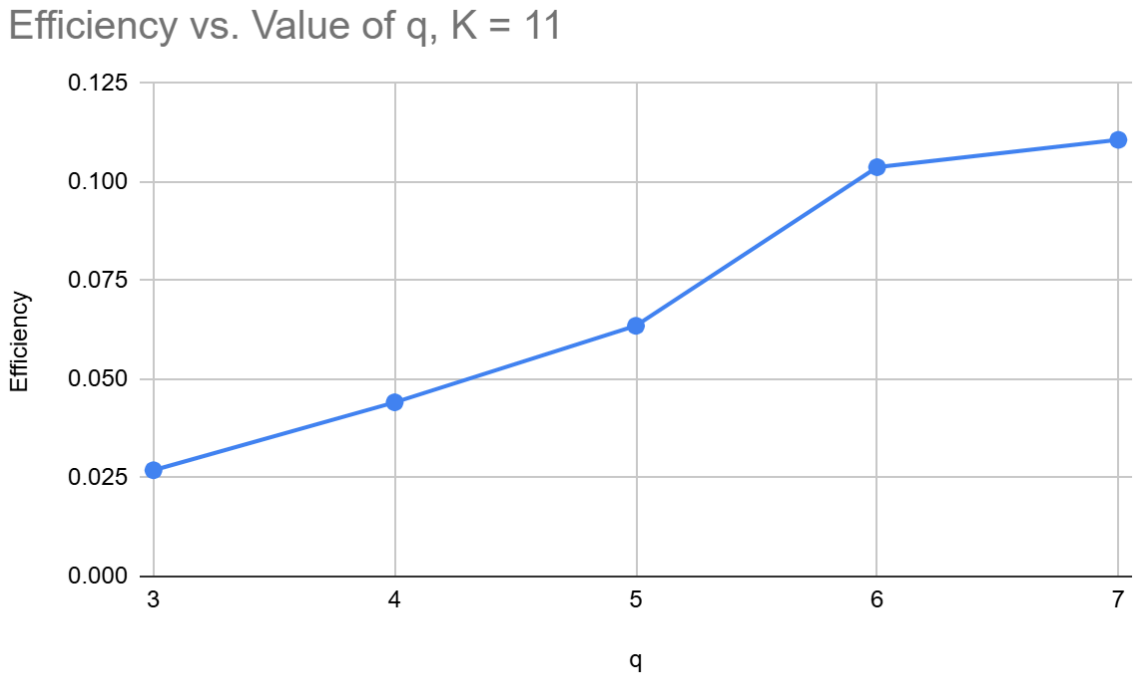


Jessica Salguero, UIN: 937005006
CSCE 735, Summer 2025

K = 11

When K = 11, SpeedUp gradually increases with the best performance occurring when q = 7. Efficiency follows a similar trend as speedup, suggesting that q = 7 is the ideal value of q and lower q values may involve recursion that requires more overhead for memory allocation, task management, etc.

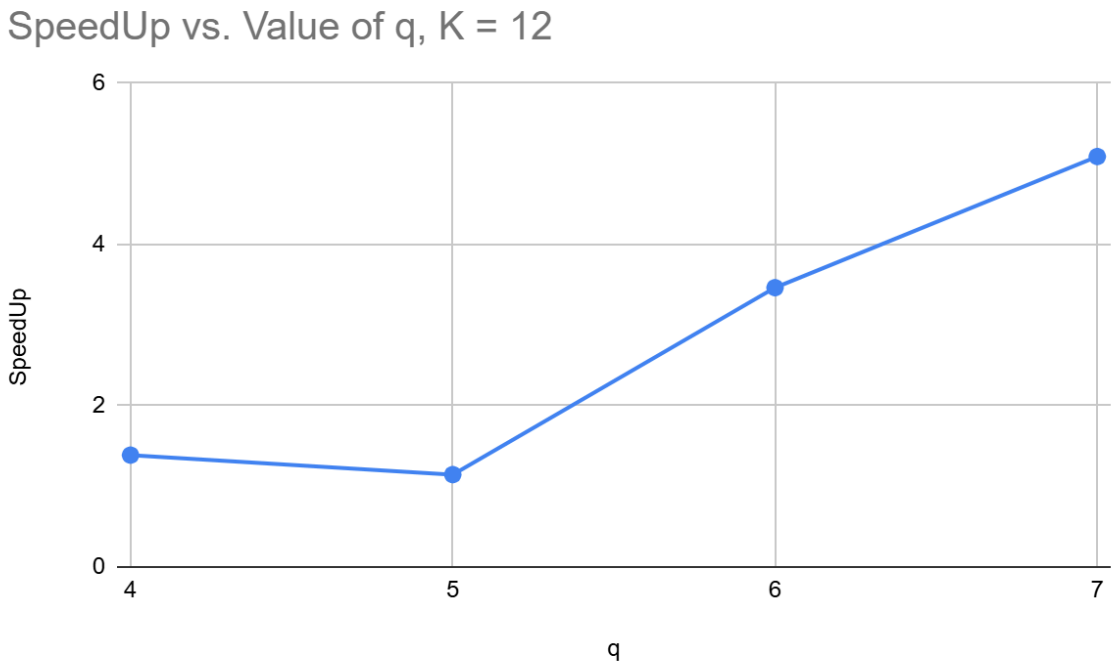


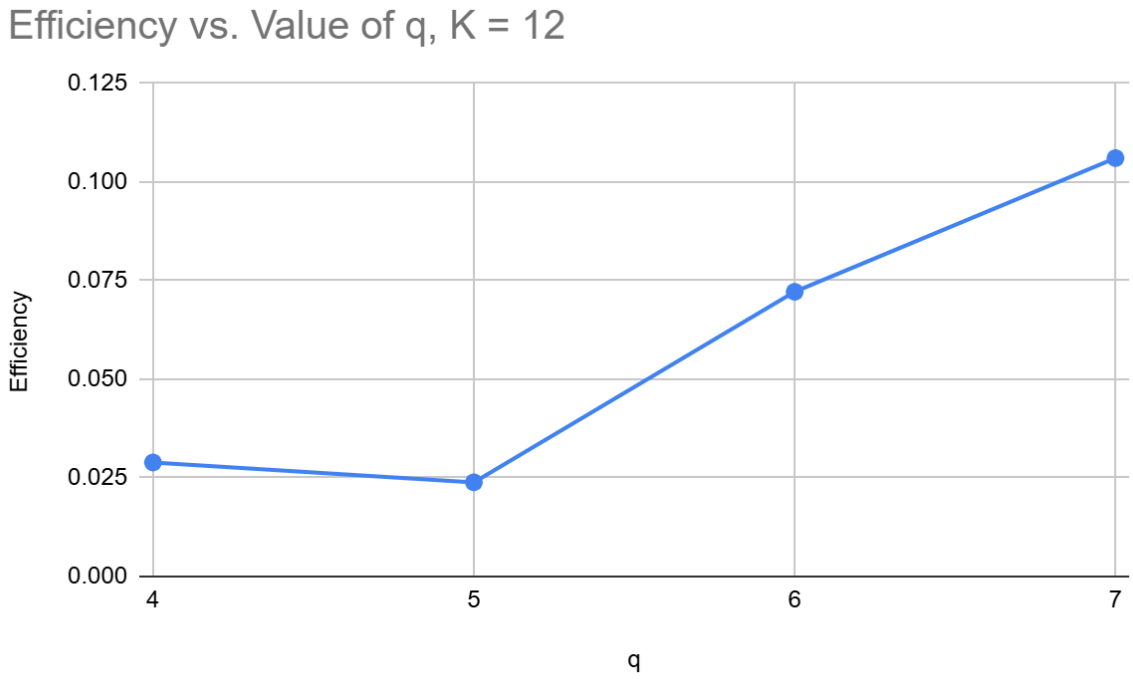


Jessica Salguero, UIN: 937005006
CSCE 735, Summer 2025

K = 12

When K = 12, the matrix becomes pretty large and begins to benefit significantly from the parallel implementation. The best speedup was seen at q = 7, with efficiency following a similar trend. The code was unable to successfully run when q = 3, likely due to the matrix's large size and excessive memory requirement that Grace was unable to provide.





K = 13

When K = 13, we begin to get to the extremely memory intensive matrices. Lower values of q = 3, 4, 5, 6 were unable to successfully run due to insufficient memory available on Grace. The only successful run occurred when q = 7, but efficiency was relatively low and speedup was decent but not extremely impressive. Parallel computing can only improve performance so much when large amounts of memory are required.

K = 14

When K = 14, the matrix size was too large for a single node to handle. All runs failed for q = 3, 4, 5, 6, 7 due to insufficient available memory. I expected higher values of q, such as q = 7, to be able to fix the memory issues, but I was unable to successfully run k = 14 on a single node on Grace.

Compile & Run on Grace

To compile and execute the code using grace, you would have to prepare an sbatch script containing these lines of code. Additionally, sbatch scripts were run using 48 cores, 192GB of memory, and 1 node.

```
module load intel
icpx -o strassen_omp.exe -qopenmp strassen_omp.cpp

#For my experiments I used 48 threads for parallel execution and 1 for serial execution
export OMP_NUM_THREADS = <number_threads>
./strassen_omp.exe <k> <q>  // k = matrix size, q = leaf matrix
```