Name: Shreya Palit
Email: palits@oregonstate.edu
Project Name: OpenCL Matrix Multiplication

CS 575 - Project #6

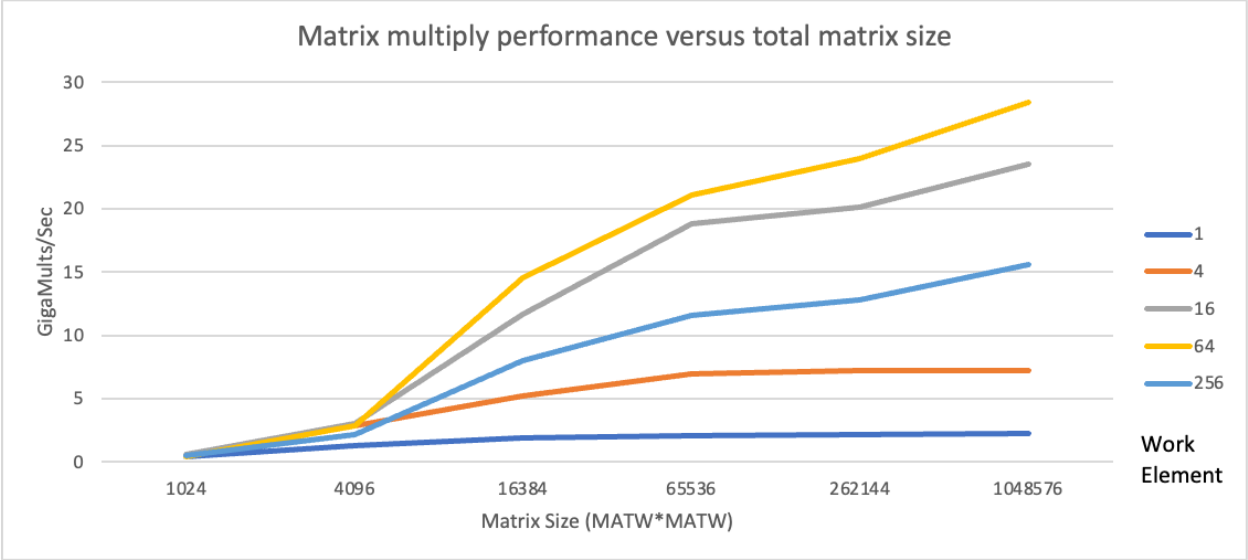
1. What machine you ran this on

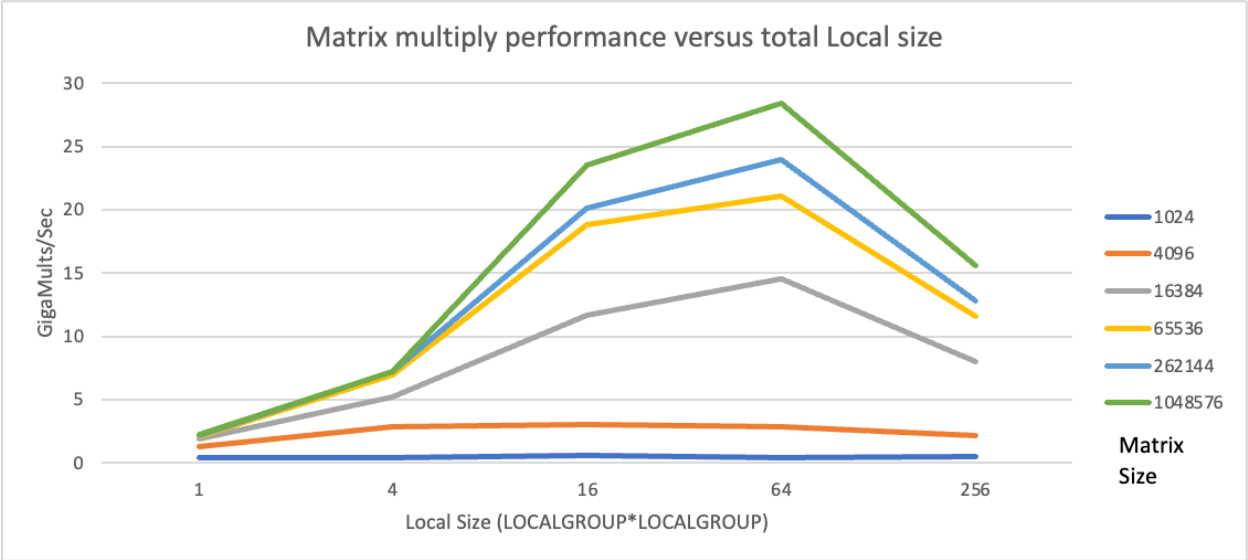
I ran this on the rabbit server which selected the vendor as NVIDIA for its GPU.

2. Show the table and graphs

Matrix Size	Work Element	GigaMults/Sec
1024	1	0.39
1024	4	0.41
1024	16	0.57
1024	64	0.38
1024	256	0.51
4096	1	1.27
4096	4	2.84
4096	16	3.06
4096	64	2.83
4096	256	2.13
16384	1	1.92
16384	4	5.19
16384	16	11.66
16384	64	14.5
16384	256	8.03
65536	1	2.09
65536	4	6.99
65536	16	18.83
65536	64	21.07
65536	256	11.57
262144	1	2.13
262144	4	7.17
262144	16	20.12

262144	64	23.99
262144	256	12.76
1048576	1	2.2
1048576	4	7.19
1048576	16	23.55
1048576	64	28.44
1048576	256	15.58





3. What patterns are you seeing in the performance curves? What difference does the size of the matrices make? What difference does the size of each work-group make?

   From the first graph it can be seen that with an increase in matrix size, the performance increases too. However, with an increase in local size, the performance doesn't increase after a certain point. For local group 8*8, highest performance is achieved after which performance starts decreasing. The maximum performance is for matrix size 1024*1024 with local group 8*8, which is 28.44 GigaMults/Sec. From the second graph it can be seen that for matrix size 32*32 and 64*64, the performance remains the same irrespective of the local size.

   With an increase in matrix size the performance increases too, except for matrix size 32*32 and 64*64 where the performance remains the same irrespective of the local size.

   The highest performance is achieved when the local work group is 8*8. After which, the performance starts dipping as can be seen in the second graph.

4. Why do you think the patterns look this way?

   Larger matrices demand more processing power and memory bandwidth, potentially creating bottlenecks and lowering performance. Another factor that may contribute to increasing contention and poor performance is the device's restricted resources or the workgroups' excessive memory utilization. It is also possible due to the GPU's design, utilizing a local work size around 8*8 is the most appropriate.