Student Name: - ATHARVA DESHPANDE

Student Email: - deshpana@oregonstate.edu

Project No: - Project#4

**Project Name**: - Vectorized Array Multiplication and Multiplication/Reduction using SSE

1. Tell what machine you ran this on.
   → I ran the program on a Windows 11 machine using the OSU Flip3 server. I wrote the code using vim and executed it on the server to get the results.

2. Show the 2 tables of performances for each array size and the corresponding speedups.
   → For Mul Speedup:

ArraySize	NonSimdMul	SimdMul	Speedup
1024	120.9	987.66	8.17
2048	121.32	1021.97	8.42
4096	120.92	957.87	7.92
8192	120.85	968.44	8.01
16384	120.99	951.39	7.86
32768	120.65	820.6	6.8
65536	120.46	779.54	6.47
131072	120.46	782.2	6.49
262144	119.82	768.55	6.41
524288	119.48	765.93	6.41
1048576	118.62	682.84	5.76
2097152	219.83	747.88	3.4
4194304	221.79	664.33	3

For MulSum Speedup:

ArraySize	NonSimdMulSum	SimdMulSum	Speedup
1024	121.82	978.21	8.03
2048	122.41	1017.95	8.32
4096	122.66	1027.34	8.38
8192	123.19	1005.76	8.16
16384	123.24	1006.33	8.17
32768	123.23	1002.23	8.13
65536	123.2	994.18	8.07
131072	122.59	1004.17	8.19
262144	122.53	1002.69	8.18
524288	122.27	987.2	8.07
1048576	230.41	1818.19	7.89
2097152	225.62	1178.08	5.22
4194304	226.37	1121.26	4.95

3. Show the graphs (or graph) of SIMD/non-SIMD speedup versus array size (either one graph with two curves, or two graphs each with one curve).
   ➔
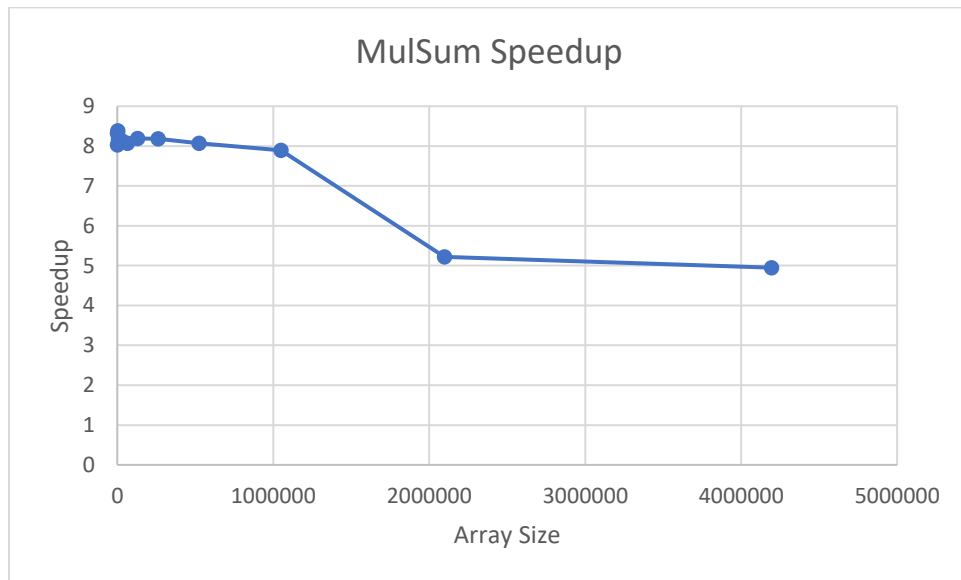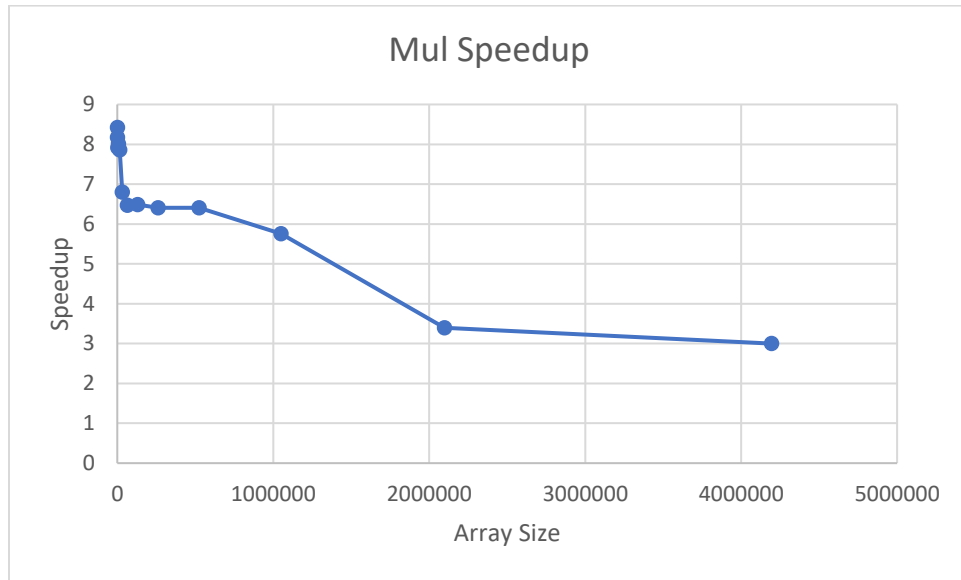
4. What patterns are you seeing in the speedups?
   - ➔ The speedups for both the graphs are concentrated around 8 with the maximum speedup for Mul being 8.42 and that for MulSum being 8.38. Further, the average speedup for Mul is 6.55 and that for MulSum is 7.67. We observe for both the graphs that with an increase in the array size, the speedup starts to decrease. For the MulSum Speedup graph, the speedup is consistent for range of array sizes from 1024 to 524288. After this, it starts decreasing. For Mul Speedup graph, the speedup falls right after an array size of 2048.

5. Are they consistent across a variety of array sizes?
   - ➔ No, the speedups are not consistent across a variety of array sizes. The reason behind this is that when the array size increases, the speedup decreases for both Mul as well as MulSum.

6. Why or why not, do you think?
   - ➔ The reason that the speedups decrease when the array size increases, and the speedups are not consistent is due to involvement of some overhead. There is some overhead involved in setting up and managing the SIMD operations, such as loading data and storing data into SIMD registers. Moreover, when the array size increases it takes more time to fetch the data from the memory and hence it can cause a decrease in speedup.