

## Project 04

1. What machine you ran this on.

I ran the code on the COE flip machine.

```
● flip4 ~/cs575/projects/project3 1007$ cat /etc/os-release
NAME="Rocky Linux"
VERSION="9.3 (Blue Onyx)"
ID="rocky"
ID_LIKE="rhel centos fedora"
VERSION_ID="9.3"
PLATFORM_ID="platform:el9"
PRETTY_NAME="Rocky Linux 9.3 (Blue Onyx)"
ANSI_COLOR="0;32"
LOGO="fedora-logo-icon"
CPE_NAME="cpe:/o:rocky:rocky:9::baseos"
HOME_URL="https://rockylinux.org/"
BUG_REPORT_URL="https://bugs.rockylinux.org/"
SUPPORT_END="2032-05-31"
ROCKY_SUPPORT_PRODUCT="Rocky-Linux-9"
ROCKY_SUPPORT_PRODUCT_VERSION="9.3"
REDHAT_SUPPORT_PRODUCT="Rocky Linux"
REDHAT_SUPPORT_PRODUCT_VERSION="9.3"
```

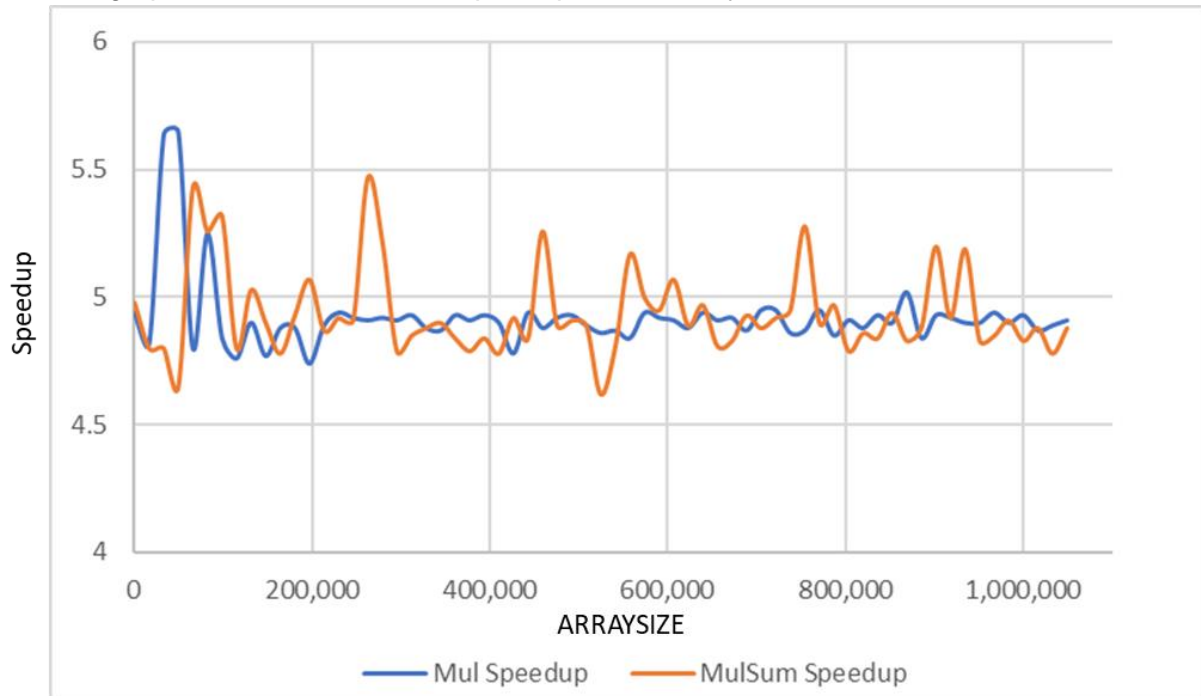
2. Show the tables of performances for each array size and the corresponding speedups.

ARRAYSIZE	NonSimdMul	SimdMul	Mul Speedup	NonSimdMulSum	SimdMulSum	MulSum Speedup
1024	351.62	1738.36	4.94	351.62	1749.42	4.98
17408	269.91	1300.65	4.82	398.25	1912.68	4.8
33792	379.35	2139.32	5.64	398.37	1911.99	4.8
50176	390.38	2205.64	5.65	409.39	1903.55	4.65
66560	389.07	1867.82	4.8	398.64	2164.82	5.43
82944	388.66	2041.2	5.25	398.62	2096.35	5.26
99328	388.54	1879.49	4.84	398.41	2118.1	5.32
115712	398.89	1898.59	4.76	407.84	1956.18	4.8
132096	388.84	1904.53	4.9	398.04	2001.57	5.03
148480	399.13	1902.22	4.77	398.11	1951.64	4.9
164864	388.46	1896.28	4.88	397.92	1901.9	4.78
181248	388.7	1897.65	4.88	403.24	1988.76	4.93
197632	398.99	1890.13	4.74	398.1	2019.31	5.07
214016	389.12	1904.7	4.89	397.99	1937.94	4.87
230400	387.77	1915.08	4.94	398.21	1957.18	4.92
246784	388.26	1911.03	4.92	398.15	1955.09	4.91
263168	388.56	1906.66	4.91	398.27	2177.71	5.47
279552	387.65	1907.82	4.92	398.2	2078.8	5.22
295936	388.51	1907.15	4.91	398.3	1908.54	4.79
312320	388.05	1912.11	4.93	398.27	1933.16	4.85
328704	388.08	1894.62	4.88	398.27	1941.99	4.88
345088	387.9	1889.54	4.87	398.27	1949.59	4.9

Yen-Chun Chen  
chenyenc@oregonstate.edu

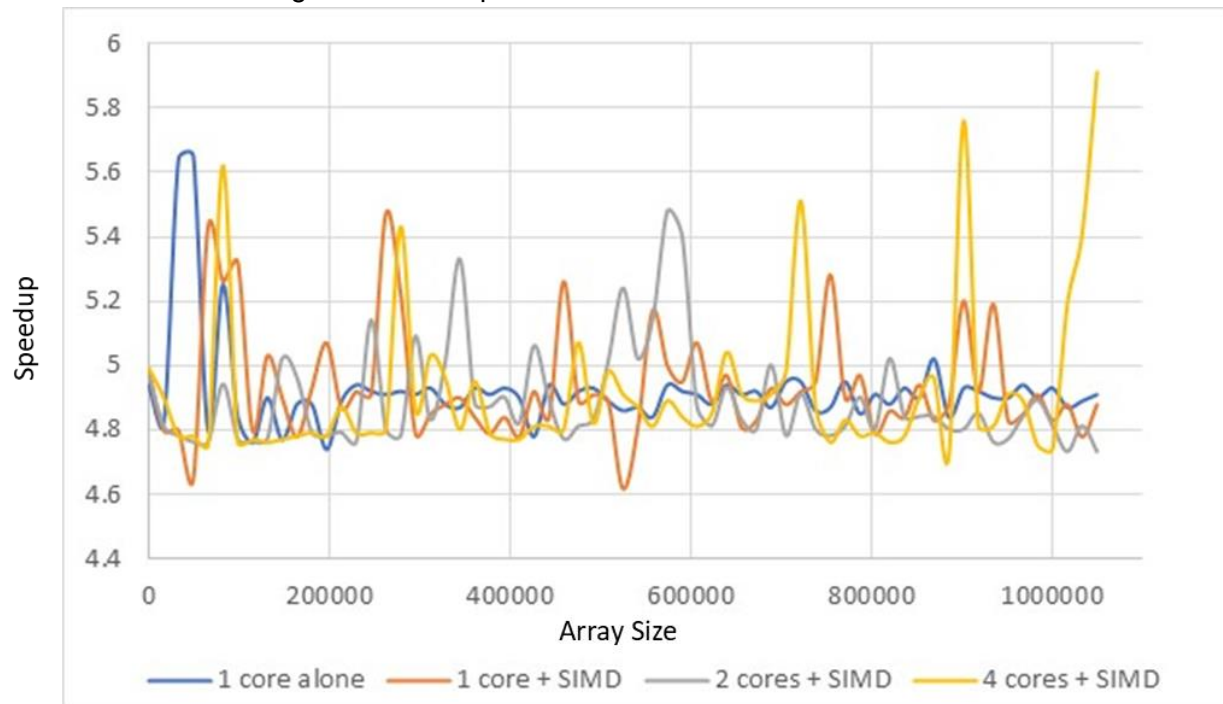
361472	388.11	1915.19	4.93	398.25	1928.03	4.84
377856	387.98	1903.44	4.91	397.97	1904.34	4.79
394240	386.2	1902.44	4.93	395.44	1915.56	4.84
410624	386.5	1892.03	4.9	404.03	1932	4.78
427008	400.55	1915.68	4.78	398.09	1959.8	4.92
443392	386.46	1908.96	4.94	396.02	1917.2	4.84
459776	386.87	1886.63	4.88	395.94	2081.11	5.26
476160	387.55	1907.53	4.92	396.27	1939.01	4.89
492544	387.28	1910.39	4.93	396.27	1945.93	4.91
508928	386.66	1889.64	4.89	396.32	1939.87	4.89
525312	386.35	1877.81	4.86	414.48	1915.84	4.62
541696	389.34	1896.12	4.87	396.46	1906.99	4.81
558080	386.71	1872.07	4.84	396.15	2046.98	5.17
574464	386.65	1909.79	4.94	396.51	1982.71	5
590848	387.26	1906.96	4.92	396.72	1962.31	4.95
607232	387.19	1902.75	4.91	396.37	2008.95	5.07
623616	387.1	1889.64	4.88	395.33	1932.49	4.89
640000	387.4	1913.47	4.94	396.39	1969.88	4.97
656384	387.08	1899.27	4.91	396.53	1908.57	4.81
672768	387.3	1906.81	4.92	396.38	1914.97	4.83
689152	387.23	1884.57	4.87	396.69	1954.03	4.93
705536	387.4	1916.32	4.95	396.65	1936.74	4.88
721920	386.31	1910.7	4.95	395.97	1949.51	4.92
738304	387.62	1885.67	4.86	395.47	1957.83	4.95
754688	390.73	1902.79	4.87	397.06	2098.29	5.28
771072	386.3	1911.68	4.95	395.59	1937.78	4.9
787456	390.68	1896.45	4.85	400.65	1989.36	4.97
803840	385.95	1896.05	4.91	400.61	1920.51	4.79
820224	386.32	1885.05	4.88	395.8	1922.61	4.86
836608	386.5	1904.59	4.93	395.48	1914.4	4.84
852992	385.98	1891.98	4.9	395.56	1952.74	4.94
869376	386.26	1938.47	5.02	395.08	1909.78	4.83
885760	391.48	1894.3	4.84	395.18	1929.53	4.88
902144	386.32	1903.33	4.93	395.67	2056.26	5.2
918528	386.42	1902.18	4.92	396.08	1946.92	4.92
934912	386.25	1894.19	4.9	395.97	2053.6	5.19
951296	385.63	1890.82	4.9	395.96	1913.98	4.83
967680	384.68	1900.72	4.94	395.71	1921.11	4.85
984064	382.93	1877.19	4.9	394.76	1937.66	4.91
1000448	384.73	1896.67	4.93	395.81	1911.81	4.83
1016832	385.65	1879.36	4.87	394.93	1925.49	4.88
1033216	391.31	1914.09	4.89	399.62	1911.9	4.78
1049600	385.8	1895.51	4.91	394.98	1926.84	4.88

3. Show the graphs of SIMD/non-SIMD speedup versus array size.



4. What patterns are you seeing in the speedups?  
The speedups for SIMD vs non-SIMD are generally around 4.95 regardless Mul or MulSum. When array size are 1024 and 50176 show slightly higher speedups around 5.6. And the performance decrease at larger array sizes, and it might suggest that overhead while too many people were using flip server at the same time.
5. Are they consistent across a variety of array sizes?  
Yes, the speedups are relatively consistent across different array sizes, and mostly around 4.9 to 5.0.
6. Why or why not, do you think?  
Due to the SIMD's parallel processing, which performs well across different array sizes. Those little variations might be because of the memory access patterns or system overheads.

7. Extra Credit: Combining SIMD with OpenMP.



This graph shows that 4 cores + SIMD has better performance. However, 2 cores + SIMD seems to have no explicit difference with 1 core + SIMD.