CS575 Project 4

Chiu-Chun, Chen

Email: chenchiu@oregonstate.edu

May 11, 2022

1. **What machine you ran this on**

```
flip1 ~/cs575/project4 230$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                24
On-line CPU(s) list:   0-23
Thread(s) per core:    2
Core(s) per socket:    6
Socket(s):             2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 44
Model name:            Intel(R) Xeon(R) CPU           X5650  @ 2.67GHz
Stepping:              2
CPU MHz:               2659.791
BogoMIPS:              5319.58
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              12288K
NUMA node0 CPU(s):     0,2,4,6,8,10,12,14,16,18,20,22
NUMA node1 CPU(s):     1,3,5,7,9,11,13,15,17,19,21,23
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr
tscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonst
pr pdcm pcid dca sse4_1 sse4_2 popcnt aes lahf_lm ssbd ibrs ibpb stibp t
```

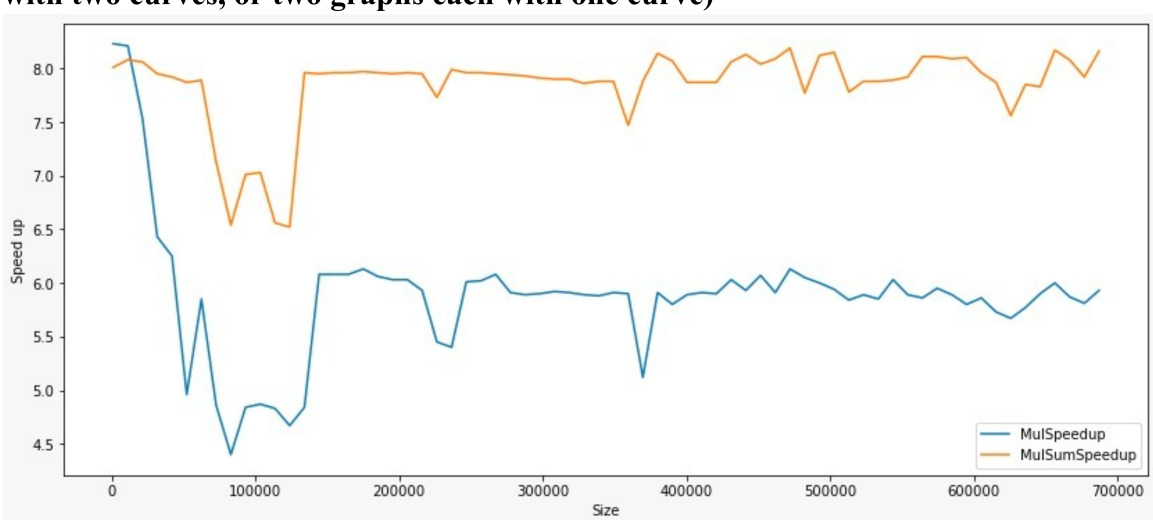
2. **Show the table of performances for each array size and the corresponding speedups**

ARRAYSIZE	NonSimdMul	SimdMul	Speedup	NonSimdMulSum	SimdMulSum	Speedup
1024	221.05	1818.88	8.23	223.52	1790.74	8.01
11264	222.34	1824.92	8.21	225.92	1825.6	8.08
21504	222.18	1674.08	7.53	225.99	1821.32	8.06
31744	221.66	1425.73	6.43	225.87	1796.59	7.95
41984	221.66	1385.8	6.25	225.83	1787.47	7.92
52224	221.48	1098.43	4.96	225.79	1777.31	7.87
62464	221.51	1295.63	5.85	225.78	1782.12	7.89

72704	221.44	1075.1	4.86	225.46	1608.57	7.13
82944	221.23	972.83	4.4	225.39	1473.14	6.54
93184	221.07	1071.06	4.84	225.5	1580.44	7.01
103424	220.99	1075.57	4.87	225.48	1585.92	7.03
113664	220.85	1067.18	4.83	225.36	1477.96	6.56
123904	220.81	1031.4	4.67	225.37	1470.27	6.52
134144	220.86	1069.11	4.84	225.87	1797.22	7.96
144384	221.38	1345.37	6.08	225.88	1796.81	7.95
154624	221.31	1346.48	6.08	225.87	1798.5	7.96
164864	221.3	1344.66	6.08	225.87	1798.31	7.96
175104	221.26	1357.14	6.13	225.87	1799.34	7.97
185344	221.21	1340.46	6.06	225.86	1797.87	7.96
195584	221.11	1332.78	6.03	225.79	1794.73	7.95
205824	220.98	1331.81	6.03	225.77	1796.33	7.96
216064	220.01	1303.67	5.93	225.79	1795.13	7.95
226304	219.9	1199.08	5.45	222.4	1720.1	7.73
236544	217.36	1173.44	5.4	224.71	1795.37	7.99
246784	220.16	1323.2	6.01	224.95	1791.23	7.96
257024	220.15	1325.99	6.02	224.97	1790.96	7.96
267264	220.17	1339	6.08	224.96	1787.97	7.95
277504	220.08	1301.74	5.91	224.93	1786.56	7.94
287744	220	1295.34	5.89	224.92	1782.65	7.93
297984	220.27	1299.74	5.9	225.07	1780.34	7.91
308224	220.46	1305.25	5.92	225.12	1777.83	7.9
318464	220.02	1300.3	5.91	224.86	1775.57	7.9
328704	220.05	1296.35	5.89	224.91	1767.77	7.86
338944	219.53	1291.06	5.88	224.91	1771.61	7.88
349184	219.86	1299.56	5.91	224.85	1772.07	7.88
359424	219.81	1296.55	5.9	224.81	1678.46	7.47
369664	213.97	1095.37	5.12	225.1	1773.04	7.88
379904	220.46	1302.11	5.91	227.26	1849.67	8.14
390144	224.07	1298.49	5.8	229.2	1850.4	8.07
400384	221.44	1304.41	5.89	225.07	1771.96	7.87
410624	220.2	1302.26	5.91	225.14	1772.03	7.87
420864	224.81	1327.27	5.9	225.02	1771.95	7.87
431104	223.26	1345.96	6.03	229.3	1847.5	8.06
441344	224.46	1331.46	5.93	227.42	1849.38	8.13
451584	222.02	1347.03	6.07	230.16	1849.55	8.04
461824	224.38	1325.11	5.91	228.98	1851.41	8.09
472064	221.81	1359.21	6.13	225.89	1849.61	8.19

482304	221.21	1339.34	6.05	228.12	1771.77	7.77
492544	221.95	1331.26	6	227.69	1847.89	8.12
502784	222.43	1320.24	5.94	226.69	1848.42	8.15
513024	221.4	1292.16	5.84	227.7	1770.36	7.78
523264	219.94	1295.18	5.89	224.82	1771.88	7.88
533504	222.32	1299.91	5.85	224.79	1771.33	7.88
543744	218.32	1315.67	6.03	224.53	1771.51	7.89
553984	219.37	1292.14	5.89	223.67	1771.59	7.92
564224	219.77	1288.6	5.86	227.91	1848.92	8.11
574464	222.38	1323.27	5.95	228.41	1852.84	8.11
584704	224.21	1319.69	5.89	228.34	1848.36	8.09
594944	221.87	1285.96	5.8	228.89	1853.49	8.1
605184	221.18	1295.06	5.86	226.71	1803.96	7.96
615424	218.95	1255.14	5.73	225.08	1771.23	7.87
625664	228.23	1293.63	5.67	235.19	1777.35	7.56
635904	229.39	1324.28	5.77	226.41	1776.49	7.85
646144	222.78	1315.24	5.9	226.87	1776.67	7.83
656384	221.47	1329.11	6	226.93	1853.92	8.17
666624	221.01	1297.46	5.87	226.31	1829.55	8.08
676864	220.63	1282.89	5.81	224.64	1779.4	7.92
687104	219.02	1298.47	5.93	227.36	1856.22	8.16

3. **Show the 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?**
At the beginning, both speed ups start from 8, but as the data set size (array size) increase till around 100000, performance decreases. Then they start to increase, note that the

increasing rates are quite similar. Afterwards, as the array size increase, both performances remain stable.

5. **Are they consistent across a variety of array sizes?**
   Yes, they are relatively high in consistency across a variety of array size.

6. **Why or why not, do you think?**
   Since SIMD represents "single instruction, multiple data", we can infer that with the same (single) instruction, which in this case is array*array, the result should be in relatively high consistency in terms of different array size.