CS 475/575 -- Spring Quarter 2022
Project #4
Vectorized Array Multiplication/Reduction using
SSE
60 Points
Due: May 11

Han-Hsing Pao
933651943
paoh@oregonstate.edu

What machine you ran this on?
I ran project4 on flip1 which on MobaXterm.

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

| ArraySize(KB)	NonSimdMul	SimdMul	NonSimdMulSum	SimdMulSum
1	222.47	561.36	238.09	653.5
2	223.73	541.21	239.44	636.7
4	225.08	674.54	245.67	655.02
8	220.62	655.01	246.29	631.39
16	216.4	653.21	246.45	629.59
32	218.78	622.98	239.08	649.64
64	222.79	572.79	247.32	632.2
128	217.81	611.01	247.68	631.74
256	219.98	608.03	247	655.61
512	221.08	623.58	246.73	570.3
1024	216.07	635.41	239.38	654.19
2048	223.92	660.29	248.98	660.32
4096	216.81	667.91	244.03	649.3
8192	215.05	546.73	245.05	577.5

Show the graph of SIMD/non-SIMD speedup versus array size (one curve only)
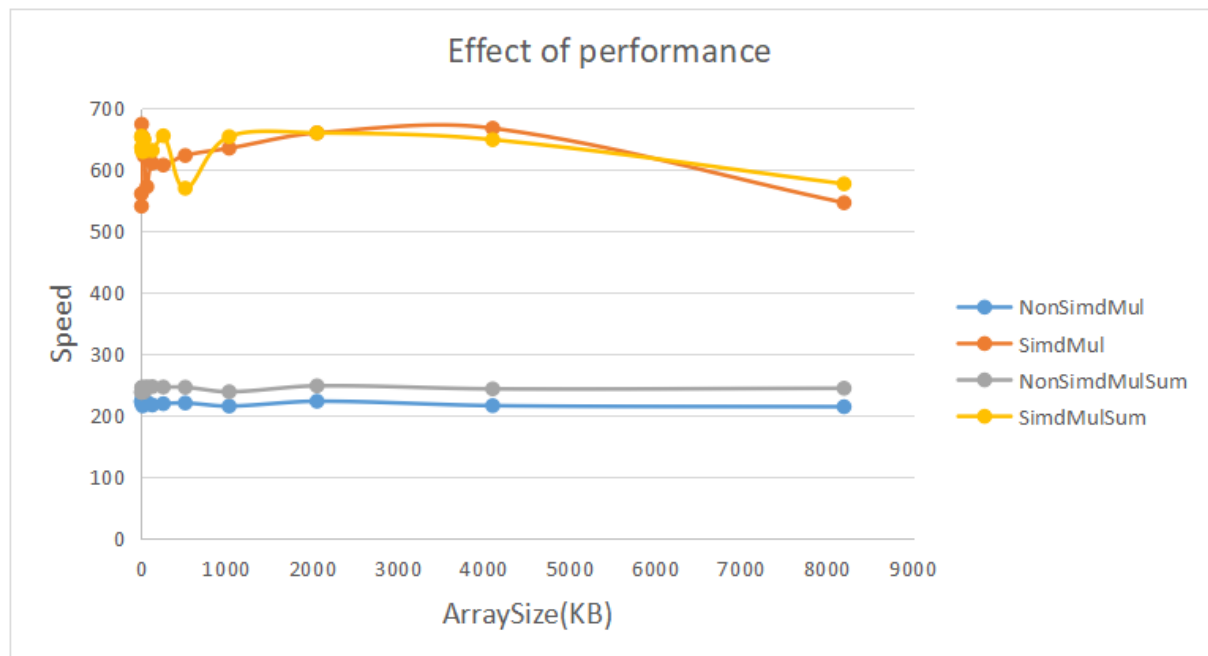
The Effects of Prefetching on SIMD
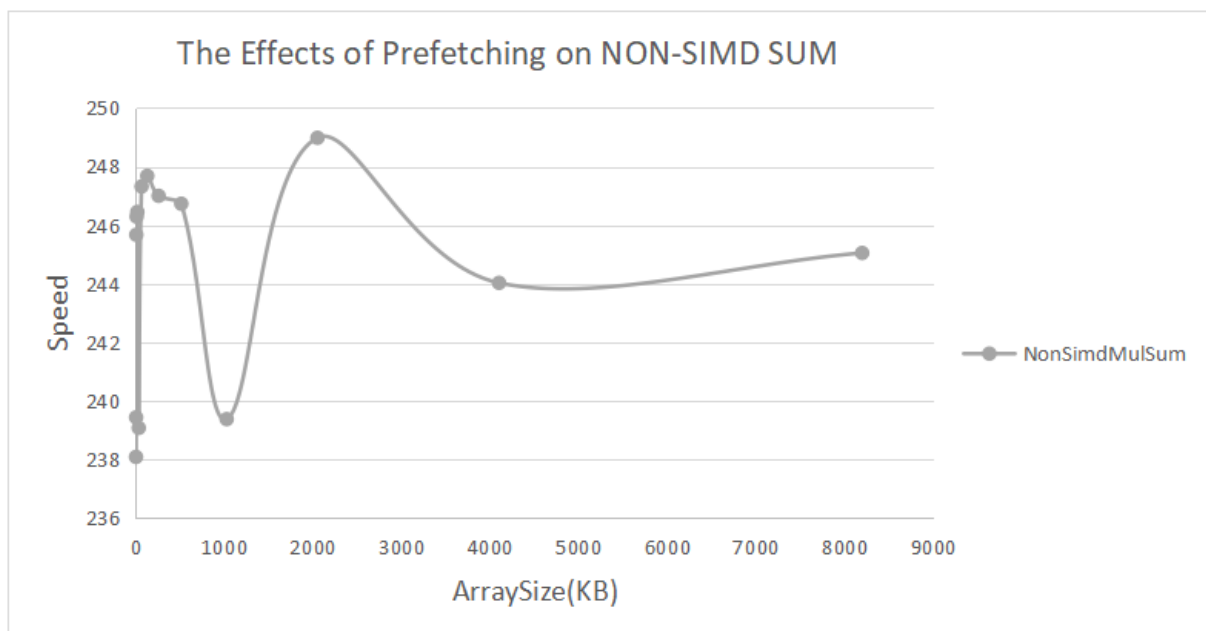

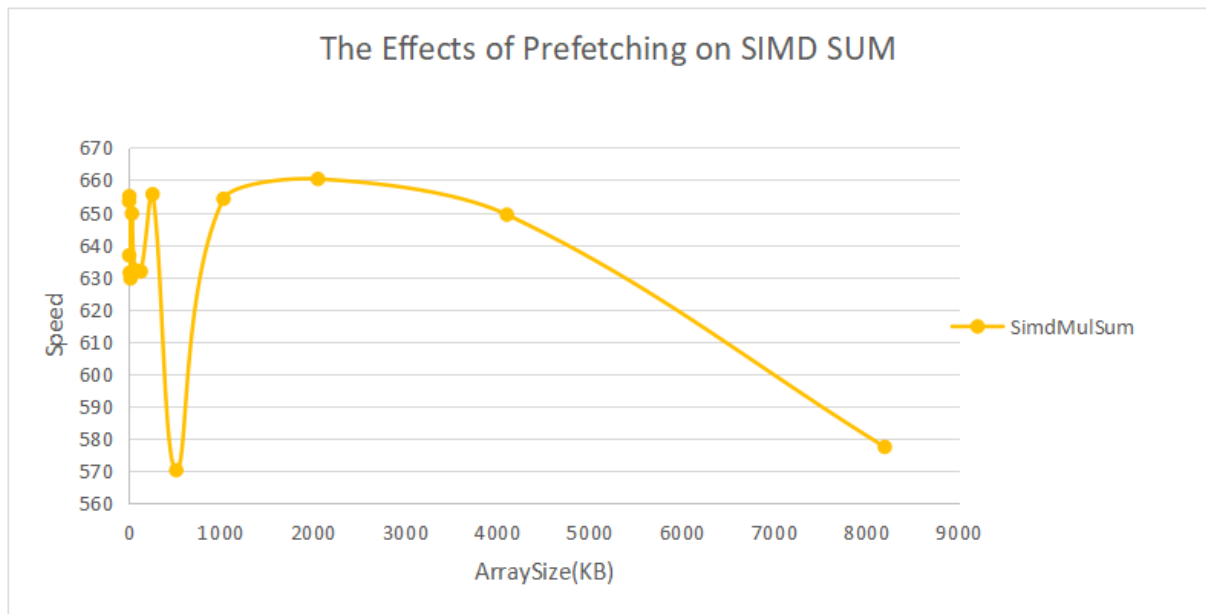# The Effects of Prefetching on NON-SIMD

The Effects of Prefetching on SIMD SUM
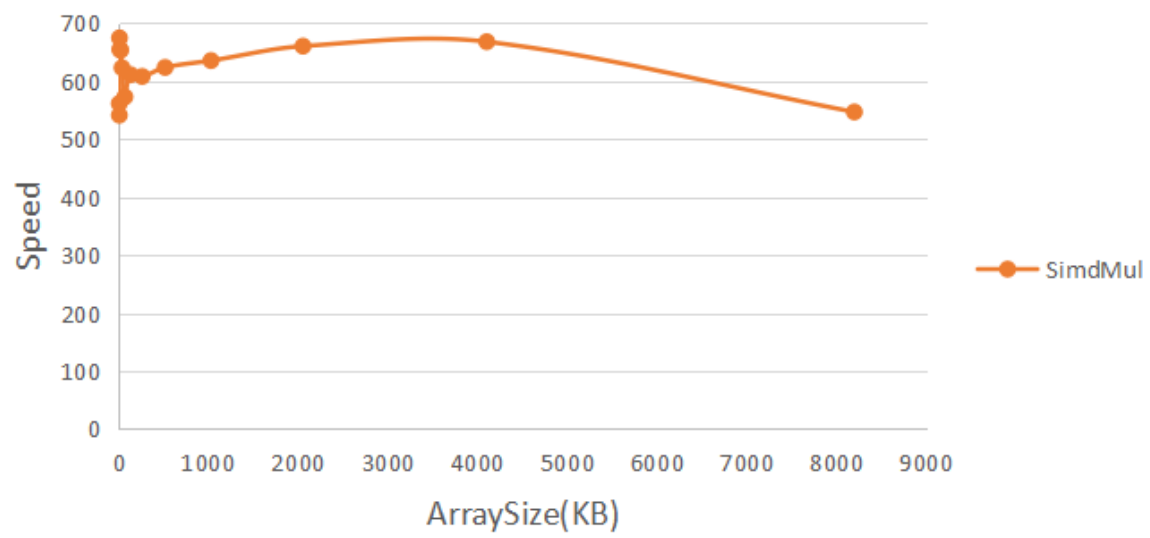
The Effects of Prefetching on NON-SIMD SUM
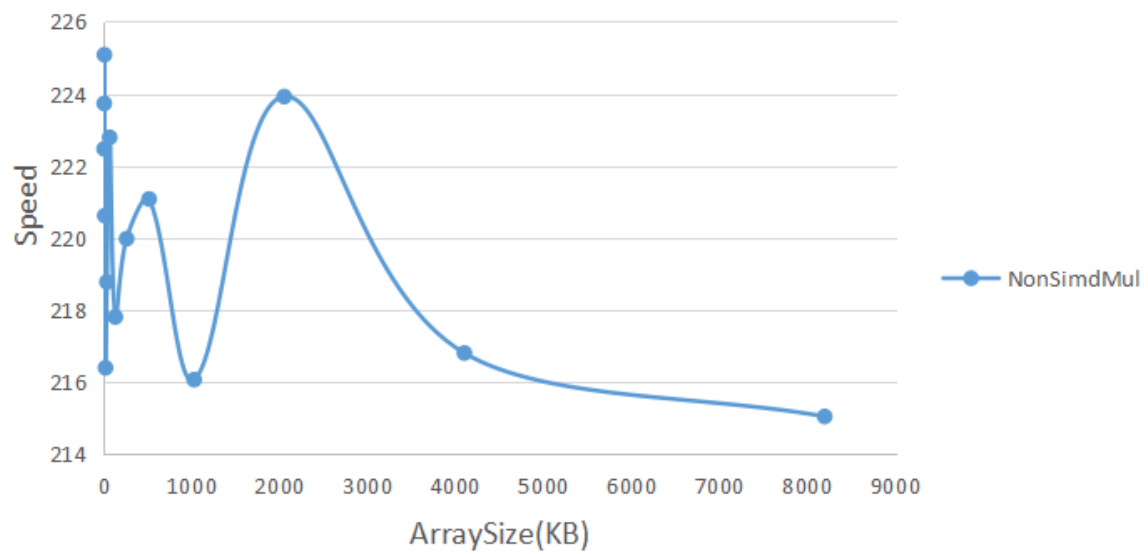
What patterns are you seeing in the speedups?
The pattern I see in the speedups is that SIMD has greater speedups than NON-SIMD. However, from 4000 to 5000kb, the acceleration of SIMD drops moderately. Compared with SIND, the speedup of NON-SIMD drops sharply from about 2MIB.

Are they consistent across a variety of array sizes? Why or why not, do you think?
The speedup is relatively steady for SIMD, but for NON-SIMD there is a high spike in speedup. I suppose it's likely that having numerous users simultaneously will result in a significant improvement in NON-SIMD speed. SIMD's speedup appears to increase with the size of the array since it consumes more memory as the array grows in size, resulting in an increase in speed.