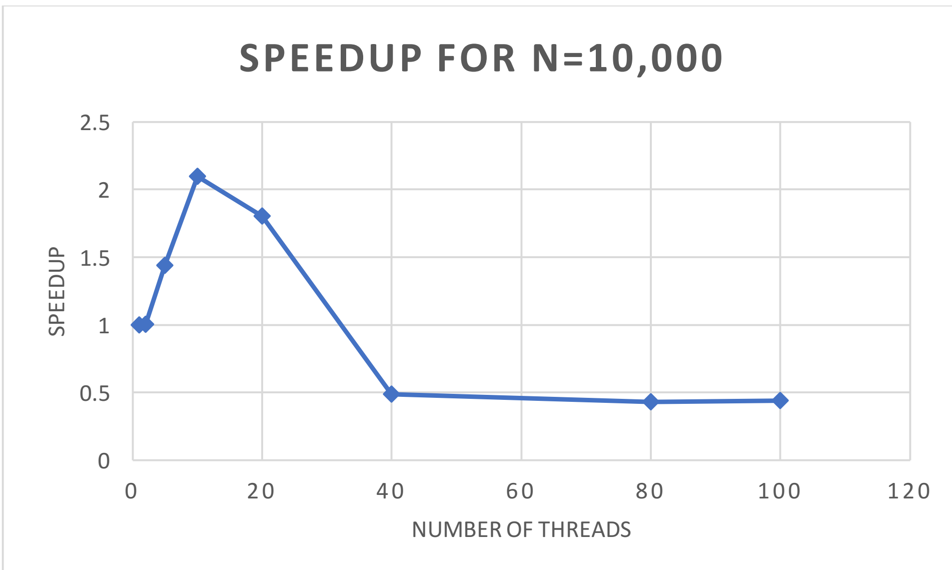
Alperen Karaoglu ak4410 N13012011

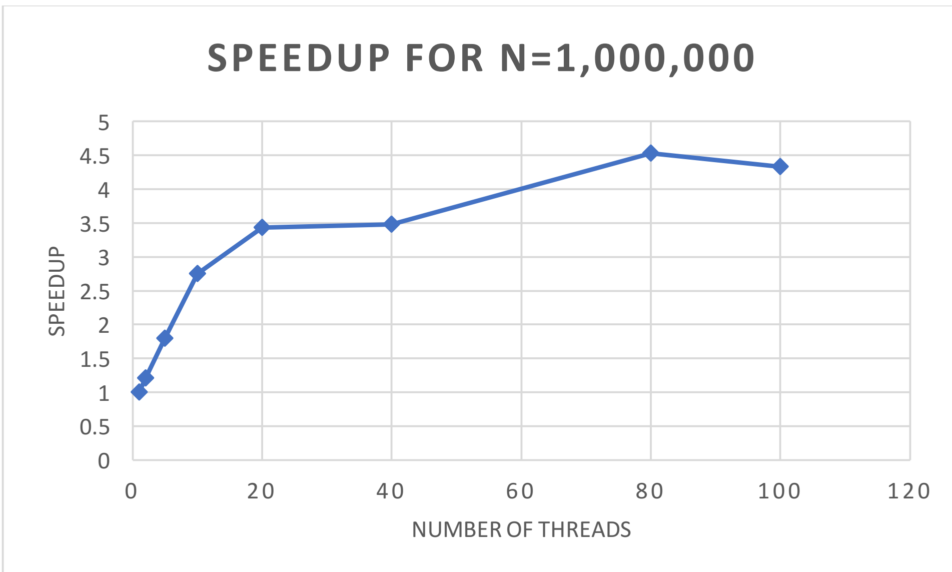
Parallel Computing lab-2 report

N=10,000		
T	time	Speedup
1	0.002997	1
2	0.002989	1.00267648
5	0.002085	1.437410072
10	0.001429	2.097270819
20	0.001661	1.804334738
40	0.006167	0.485973731
80	0.006995	0.428448892
100	0.00681	0.440088106



We can see that Speedup peaks at T = 10 and is positive until T = 20. This is due to the gain in computational time from running multiple threads. Up until 10 threads the gain from multithreading is bigger than the loss from thread communication for a data set of size 10 000 so we get a positive speedup, up to 2 times faster than running a single thread. When we reach 20 threads the gain we make in computational time is overpowered by the cost of thread communication on a data set of this size so speedup starts decreasing and even becomes negative.

N=1,000,000		
T	time	Speedup
1	0.479027	1
2	0.39625	1.208900946
5	0.266527	1.797292582
10	0.174353	2.747454876
20	0.139621	3.43090939
40	0.137726	3.478115969
80	0.105682	4.532720804
100	0.110457	4.336773586



We can see that Speedup peaks at T = 80 and is positive until T = 100. For a data set this large the gain from parallelizing the computations are very large. The load can be kept balanced and all threads can be busy doing computations. We only start getting speedup loss after 80 threads when the communication overhead from spanning too many threads causes a little loss in speedup. Because the data set is so large there is a lot more to be gained by parallelizing all the computations, so speedup is always positive and increasing until 80 threads. Speedup could be a little lower with 100 threads due to the CPU being busy with other tasks since it is still possible to get a speedup peak at 100 threads with a dataset this large when the CPU is not busy doing other computations.