Q2.)

Threads	Timings
1 Original	3.698s
1	3.574s
2	1.797s
3	1.210s
4	0.906s
6	0.631s
8	0.499s
12	0.344s
16	0.372s
24	0.402s
32	0.422s

B.) Yes, I do observe a N-times speedup in my results however as the number of threads gets larger this observation is less apparent as the speedup isnt as much as the threads being increased.

C.)As you add more threads, they start competing for resources like CPU time, memory access, and locks. This competition can slow things down because the threads have to coordinate and communicate more, which adds extra work and can cancel out the advantages of running things in parallel.

Q4.)

medium.txt

# threads	Observed Timing	Observed Speedup Compared to Original Timing	Expected Speedup
Original Program	22.4705s	1.0	1.0

1	23.6070s	0.9518575	1.0
2	13.1608s	1.70738101	2.0
3	8.5676s	2.62272982	3.0
4	6.5578s	3.42653024	4.0
8	3.2152s	6.98883429	8.0
16	2.3750s	9.46126316	16.0

hard.txt

# threads	Observed Timing	Observed Speedup Compared to Original Timing	Expected Speedup
Original Program	7.6477s	1.0	1.0
1	8.1061s	0.94344999	1.0
2	4.3351s	1.76413462	2.0
3	2.9222s	2.61710355	3.0
4	2.2382s	3.41689751	4.0
8	1.1066s	6.91098861	8.0
16	0.8630s	8.8617613	16.0

hard2.txt

# threads	Observed Timing	Observed Speedup Compared to Original Timing	Expected Speedup
Original Program	7.5572s	1.0	1.0
1	8.1446s	0.927887859	1.0
2	4.2170s	1.79207968	2.0
3	2.8320s	2.66850282	3.0
4	2.1397s	3.53189699	4.0

8	1.0769s	7.01755038	8.0
16	0.5731s	13.1865294	16.0

The results are all approximately what I expected them to be. For all the test files the observed speed up compared to the original was nearly the expected speedup if you round. The only places I was surprised was how in the high end especially with 16 threads, the execution for medium.txt and hard.txt was not ever more than 50% of what was expected.

As you add more threads, they start competing for resources like CPU time, memory access, and locks. This competition can slow things down because the threads have to coordinate and communicate more, which adds extra work and can cancel out the advantages of running things in parallel.