Lab 6 Report

Time in Seconds	Loop	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10	Avg	Std Dev
2 Threads	Less Than	30.152	29.96	30.208	30.094	30.168	30.179	30.011	30.083	30.228	30.206	30.1289	0.085057
	Modulus	30.173	29.932	30.099	30.087	30.108	30.289	29.978	30.011	30.068	30.126	30.0871	0.096098
4 Threads	Less Than	61.747	61.355	61.858	61.506	61.184	61.707	61.671	61.366	60.855	60.836	61.4085	0.34254
	Modulus	61.094	61.177	61.314	62.359	61.384	61.427	61.506	61.727	61.016	60.806	61.381	0.410643
6 Threads	Less Than	96.763	96.48	96.467	97.894	97.758	98.379	96.217	98.229	98.249	98.178	97.4614	0.826342
	Modulus	97.396	96.618	97.038	96.379	97.366	97.875	96.156	97.651	97.816	98.167	97.2462	0.643983
8 Threads	Less Than	130.987	132.919	130.687	131.052	130.719	129.225	130.601	132.62	130.794	131.14	131.0744	0.990115
	Modulus	132.9	132.107	131.565	131.614	131.176	134.525	131.643	130.878	130.858	131.139	131.8405	1.067318

At first, I thought that 30 seconds was a little long for my code to be running however I know the timing function I was using worked so I went along with it. When I got to 4 threads, I was amazed to see the time double. It didn't make sense and it felt like it was the same running time. When another 30 seconds was added for 6 threads, an idea crossed my mind. I reset it to 2 threads and timed the running on my phone. Lo and behold I got 30 seconds total run time. I did it again for 6 threads and also got 30. It was then that I realized that the time was cumulative for all threads. The output time is (loop run time) * (number of threads).