Kevin George
kmg328

Project 3

spell_t2_singleloop:			spell_t2_fastest:			spell_t4_singleloop			spell_t4_fastest		
Trial #	Time (ms)		Trial #	Time (ms)		Trial #	Time (ms)		Trial #	Time (ms)	
1	137		1	134		1	69		1	70	
2	135		2	137		2	68		2	70	
3	139		3	134		3	68		3	70	
4	133		4	134		4	68		4	68	
5	142		5	137		5	67		5	69	
6	144		6	132		6	67		6	68	
7	138		7	138		7	67		7	70	
8	146		8	136		8	67		8	68	
9	146		9	138		9	67		9	69	
10	125		10	133		10	67		10	68	
11	153		11	134		11	67		11	69	
12	142		12	135		12	68		12	69	
13	148		13	137		13	67		13	69	
14	140		14	139		14	68		14	69	
15	134		15	134		15	67		15	69	
	Average:	140.13		Average:	135.4		Average:	67.466		Average:	69
	Median:	140		Median:	135		Median:	67		Median:	69
	StdDev:	6.95764		StdDev:	2.099		StdDev:	0.6399		StdDev:	0.7559

I chose to parallelize the nested for-loops for the single-loop files because they require more time to run. Providing parallelism to those nested for loops would improve the run time more than the other loop with only one for loop. For all the files I chose to use the 'guided' scheduling strategy with a chunk size of 500. The number 500 is used simply because I was messing around testing a bunch of numbers and thought 500 worked well. As you can see from the numbers provided above, the number of threads used greatly improved the run time because more processes were able to be done simultaneously. But the amount of time between the files with the same amount of threads didn't change greatly because the loop that wasn't parallelized in the single-loop files don't have a significant impact on the run time of the program.