Mark Davis
CS 214 Lab11 CALVIN COLLEGE
May/02/2016

		SUM	**TIME**	**1**
1000numbers.txt	1000	523028	12340	1540557
10000numbers.txt	10000	5066337	127683	681657
100000numbers.txt	100000	50074388	1188179	1975183
1000000numbers.txt	1000000	500869305	3196199	4365188

Java.1 The time compared with Threaded_Array and 1 i
Java.2 The times using T > 1 are slower because the pr
Java.3 Under the circumstance that it's using more thar

1000numbers.txt	1000	523028	0.00014414	0.000159724
10000numbers.txt	10000	5066337	0.000845976	0.001363255
100000numbers.txt	100000	50074388	0.004103258	0.007460078
1000000numbers.txt	1000000	500869305	0.041013023	0.047614047

Ruby.1 *threaded_array_sum.rb* thread takes longer than
Ruby.2 The times T > 1 are progressively slower than th
Ruby.3 There are different scales but Ruby is slower.
Ruby.4 Ruby isn't great on process management. The in

1000numbers.txt	1000	523028	0.000014	0.00035
10000numbers.txt	10000	5066337	0.000083	0.000023

100000numbers.txt	100000	50074388	0.000745	0.000635
1000000numbers.txt	1000000	500869305	0.00233	0.0023556

Ada.1 *tasked_array_sum* is slower than array_sum.
Ada.2 times using T > 1 are slower than the single thre
Ada.3 Ada is slower than Java.

JAVA

4	8	16	24
1570745	841422	1271496	2723039
673582	912490	1207063	2901913
3100003	2709956	2865572	3228820
9386535	14347544	11152690	12588630

s slower than using Array_Sum
ogram is waiting for the addition to be done on each thread. The threads are waiting fo
8 threads because there are only 8 cores.

RUBY

0.000415587	0.002597163	0.00394565	0.005486713
0.000902685	0.00304811	0.004680004	0.002681752
0.007605637	0.009153804	0.009077849	0.013249894
0.070492923	0.068961572	0.078724567	0.082259611

array_sum.rb
e single thread. Each thread is waiting on the addition operation to finish so they are w

terpreter manges the context switching and all the 'threads' are running on a single co

ADA

0.000878	0.00734	0.002989	0.00398
0.000243	0.000778	0.001398	0.000932

0.0005747	0.000593	0.00192	0.000915
0.00116	0.001173	0.00183	0.01727

ead. Each thread is waiting upon the other thread for the addition operation to be done.

r each other.

aiting upon each other.

e.