



Estimate Secret Algorithms 2

Asher Kirshtein



About the Code

I have 4 methods so I can choose to run all of the algorithms at once or only choose certain ones to run. Each method does essentially the same thing and only really varies with what they print out and the size of the numbers I use. I use larger numbers in some algorithms with faster runtimes so I can get the best results. In each method I set n equal to some number. I set up the algorithm with that n . I then get the system `nanotime(starttime)` and then execute my algorithm. When the algorithm is finished I get the system `nanotime(endtime)`. I then subtract my finish from my start to get the time it took. I then print it out. Next I double n and repeat the process until I get to my desired value.

RunTime for Algorithm 1:

Secret Algorithm 1:

Time for 100 is: 0.0031344 Seconds.
Time for 200 is: 0.0031131 Seconds.
Time for 400 is: 0.0034726 Seconds.
Time for 800 is: 0.0329727 Seconds.
Time for 1600 is: 0.2150253 Seconds.
Time for 3200 is: 3.9132076 Seconds.
Time for 6400 is: 72.5982763 Seconds.
Time for 12800 is: 111.8566667 Seconds.
Time for 25600 is: 795.5538703 Seconds.

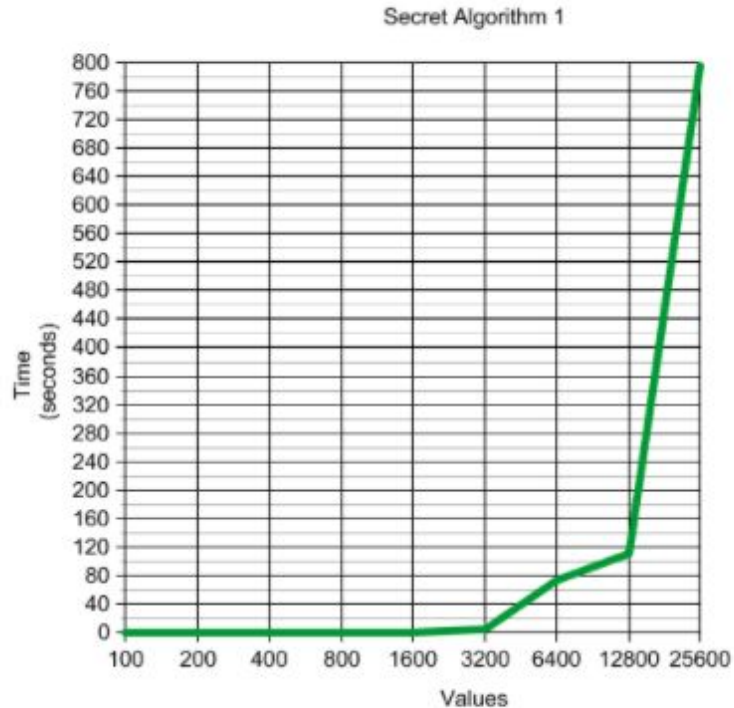
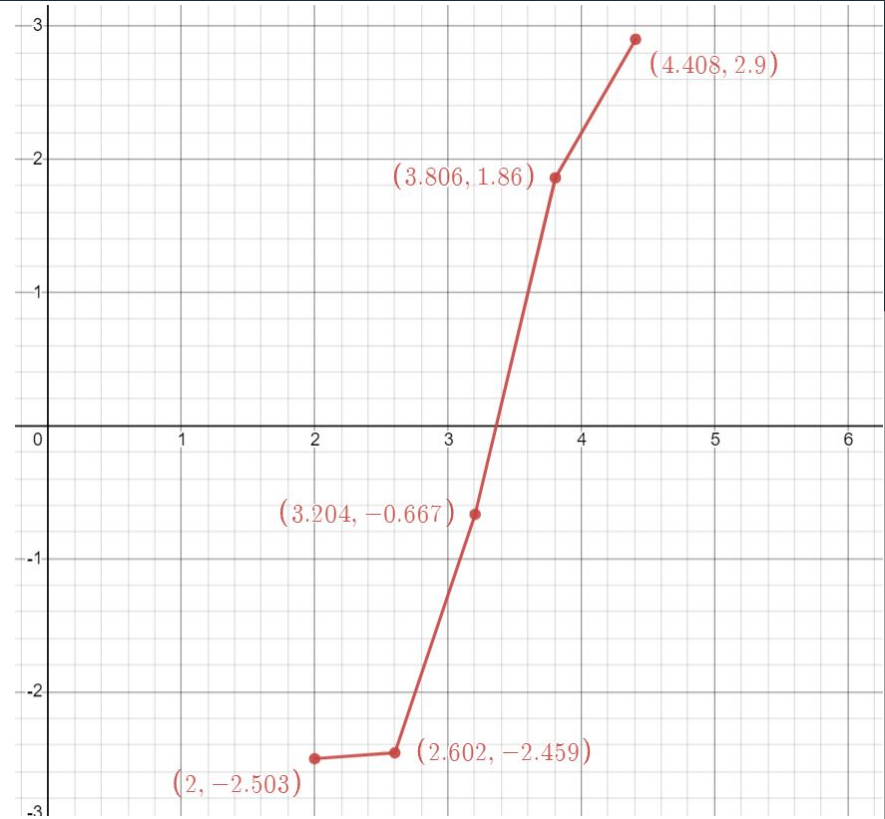


Table of the Log graph for Algorithm 1

	n	Log(n)	f(n)	Log(f(n))
	100	2	$3.1344 \cdot 10^3$	-2.503
	400	2.602	0.0034726	-2.459
	1600	3.204	0.2150253	-0.667
	6400	3.806	72.5982763	1.86
	25600	4.408	795.5538703	2.9

Log Graph

Our run time is $O(n^3)$ since the slope of the graph is approximately 3. At some points its less at some points it's more but 3 seems to be the average



Run Time for Algorithm 2:

Secret Algorithm 2:

Time for 100 is: 1.3E-5 Seconds.
Time for 200 is: 4.1E-6 Seconds.
Time for 400 is: 9.7E-6 Seconds.
Time for 800 is: 2.65E-5 Seconds.
Time for 1600 is: 5.74E-5 Seconds.
Time for 3200 is: 8.7E-5 Seconds.
Time for 6400 is: 1.521E-4 Seconds.
Time for 12800 is: 2.372E-4 Seconds.
Time for 25600 is: 5.219E-4 Seconds.
Time for 51200 is: 6.645E-4 Seconds.
Time for 102400 is: 7.328E-4 Seconds.
Time for 204800 is: 0.0013111 Seconds.
Time for 409600 is: 0.0010173 Seconds.
Time for 819200 is: 5.436E-4 Seconds.
Time for 1638400 is: 9.206E-4 Seconds.
Time for 3276800 is: 0.001473 Seconds.
Time for 6553600 is: 0.0024995 Seconds.
Time for 13107200 is: 0.004722 Seconds.
Time for 26214400 is: 0.0110661 Seconds.
Time for 52428800 is: 0.0180368 Seconds.

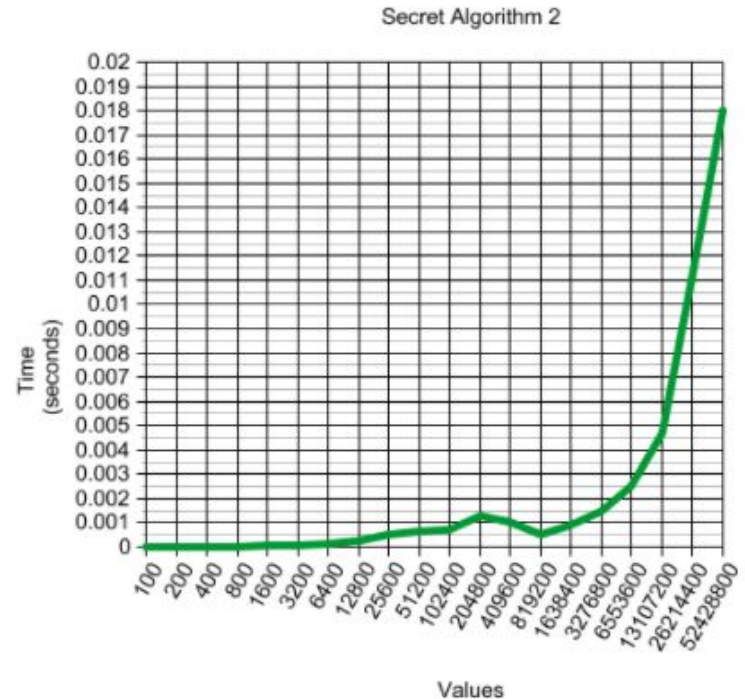
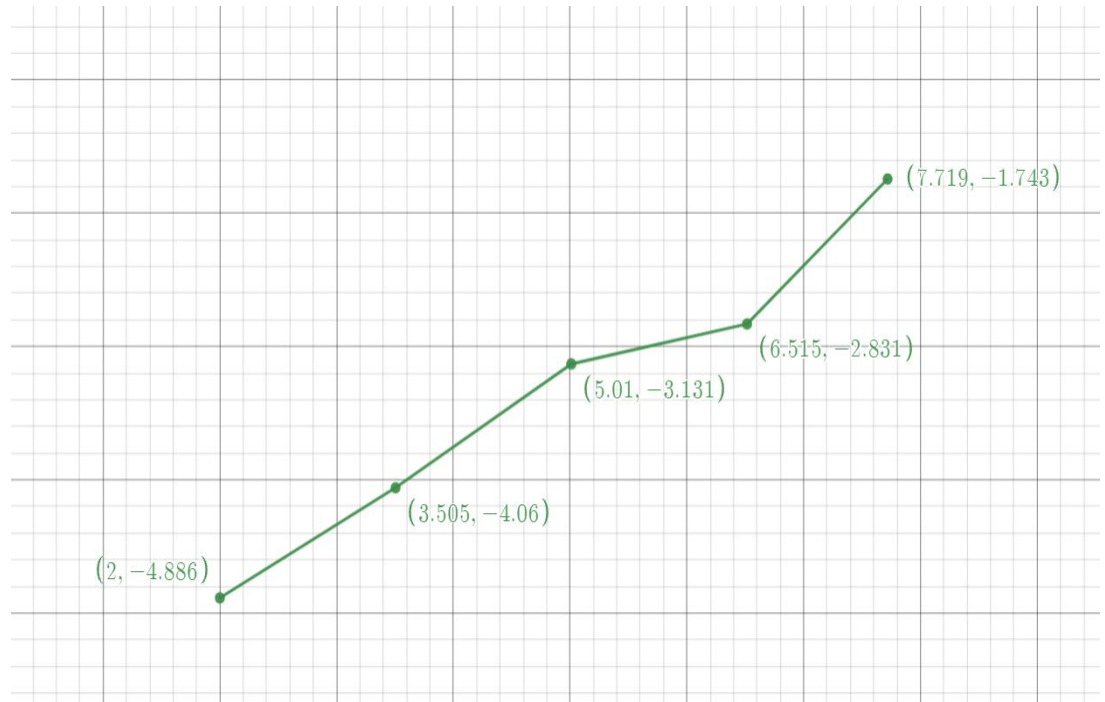


Table of the Log graph for Algorithm 2

	n	Log(n)	f(n)	Log(f(n))
	100	2	$3.1 * 10^5$	-4.9
	3200	3.505	$8.7 * 10^5$	-4.06
	102400	5.01	$7.328 * 10^4$	-3.131
	3276800	6.515	$1.473 * 10^3$	-2.831
	52428800	7.719	$1.80368 * 10^2$	-1.743

Log Graph

Since the slope of the graph is $\frac{1}{2}$
then it is logarithmic or $O(\log(n))$



Runtime for Algorithm 3:

Secret Algorithm 3:

Time for 100 is: 3.089E-4 Seconds.
Time for 200 is: 8.27E-4 Seconds.
Time for 400 is: 0.0031893 Seconds.
Time for 800 is: 0.0039825 Seconds.
Time for 1600 is: 6.989E-4 Seconds.
Time for 3200 is: 0.0074354 Seconds.
Time for 6400 is: 0.0058369 Seconds.
Time for 12800 is: 0.0283785 Seconds.
Time for 25600 is: 0.086056 Seconds.
Time for 51200 is: 0.3332976 Seconds.
Time for 102400 is: 2.8926081 Seconds.
Time for 204800 is: 5.6086535 Seconds.
Time for 409600 is: 22.511869 Seconds.
Time for 819200 is: 90.4146355 Seconds.
Time for 1638400 is: 381.0415919 Seconds.
Time for 3276800 is: 1810.4365406 Seconds.

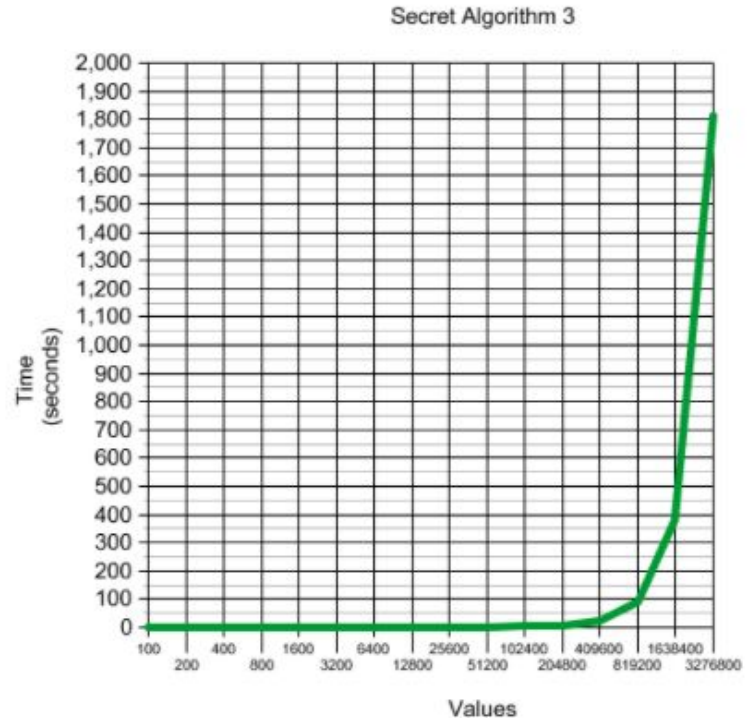
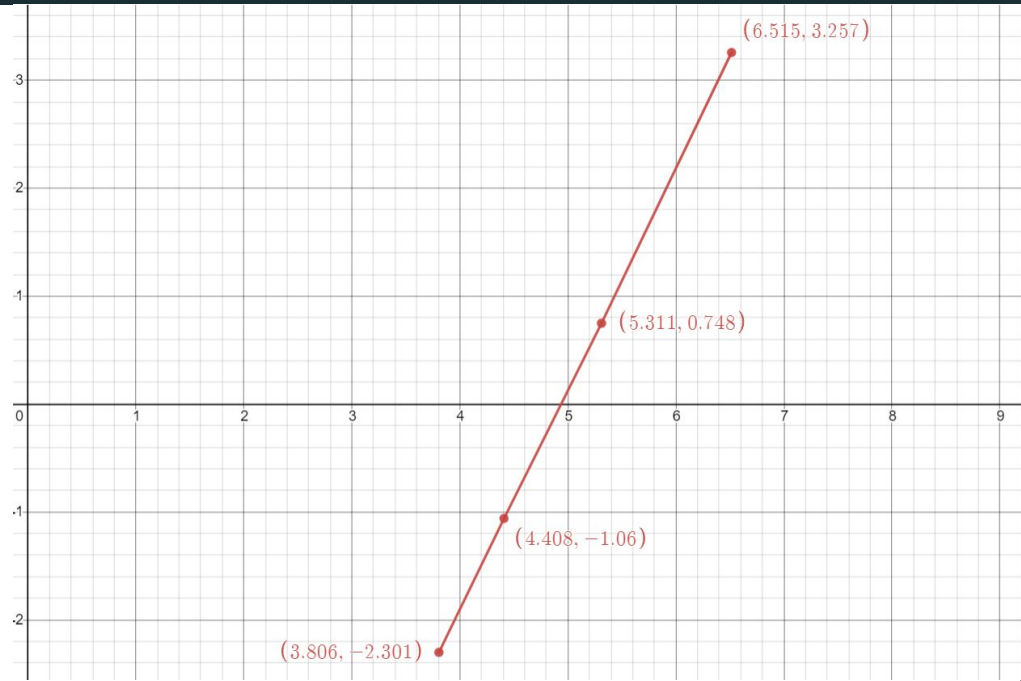


Table of the Log graph for Algorithm 3

	n	Log(n)	f(n)	Log(f(n))
	6400	3.806	0.005	-2.301
	25600	4.408	0.086	-1.06
	204800	5.311	5.608	0.748
	3276800	6.515	1810.436	3.257

Algorithm 3

The time for this algorithm is $O(n^2)$. As the slope of our $\text{Log}(n)$ graph is 2



Runtime for Algorithm 4:

Secret Algorithm 4:

Time for 1000000 is: 7.54E-5 Seconds.
Time for 2000000 is: 1.4E-6 Seconds.
Time for 4000000 is: 1.3E-6 Seconds.
Time for 8000000 is: 1.0E-6 Seconds.
Time for 16000000 is: 1.2E-6 Seconds.
Time for 32000000 is: 1.1E-6 Seconds.
Time for 64000000 is: 1.2E-6 Seconds.
Time for 128000000 is: 1.1E-6 Seconds.
Time for 256000000 is: 1.2E-6 Seconds.
Time for 512000000 is: 1.0E-6 Seconds.

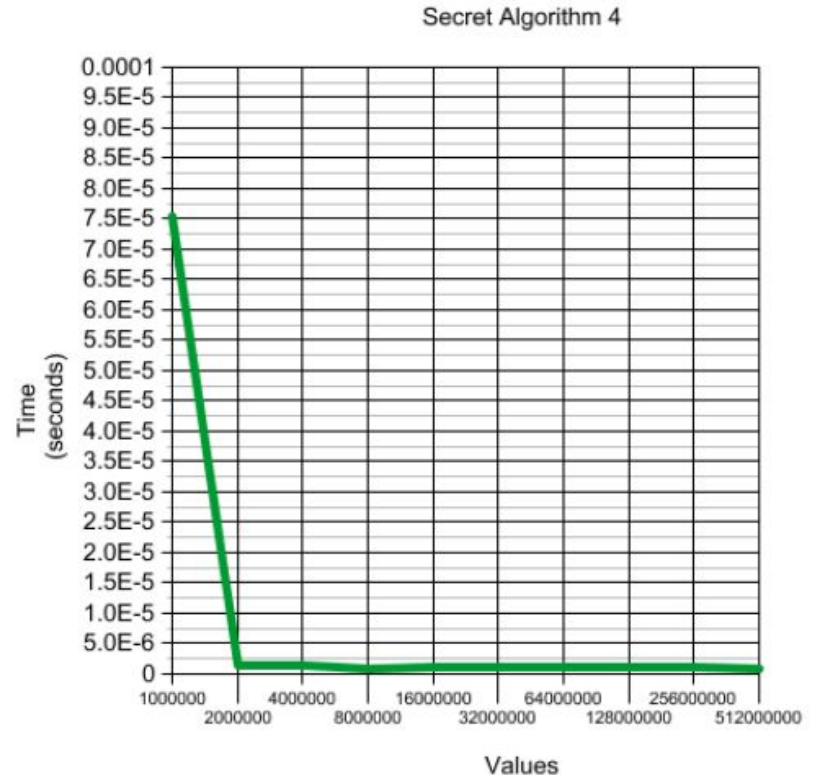


Table of the Log graph for Algorithm 4

	n	log(n)	f(n)	log(f(n))
	1000000	6	0.0000754	-4.122
	2000000	6.301	0.0000014	-5.853
	64000000	7.806	0.0000012	-5.920
	512000000	8.709	0.0000001	-6

Log Graph

This graph actually gives me a negative slope so since there is nothing faster than constant time it must be constant or $O(1)$.

