

Assignment 1

ESO 207 Report

Siddhartha Saxena

150719

For the question, I have taken the following values for a, b and c
A = 3, B = 4, C = 7, M = 5.

Time	-5	-4	-3	-2	-1	0	1
Max value of n for recursive algorithm	1	13	19	26	31	36	41
Max value of n for iterative algorithm	1	300	4400	115000	1700000	20000000	200000000
Max value of n for matrix method	1	3	>10 ¹⁹	>10 ¹⁹	>10 ¹⁹	>10 ¹⁹	>10 ¹⁹

For the recursive algorithm

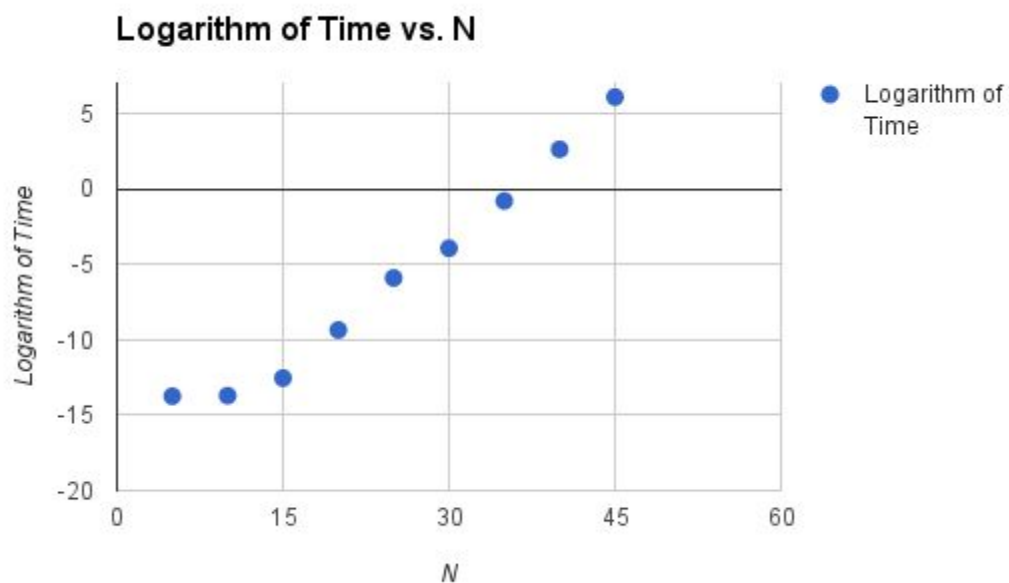


Table for the above plot

N	Logarithm of Time	N	Logarithm of Time
5	-13.74174401	30	-3.932890001
10	-13.70274988	35	-0.793104264
15	-12.53925115	40	2.627551977
20	-9.352252632	45	6.093015387
25	-5.900642847		

For the iterative algorithm

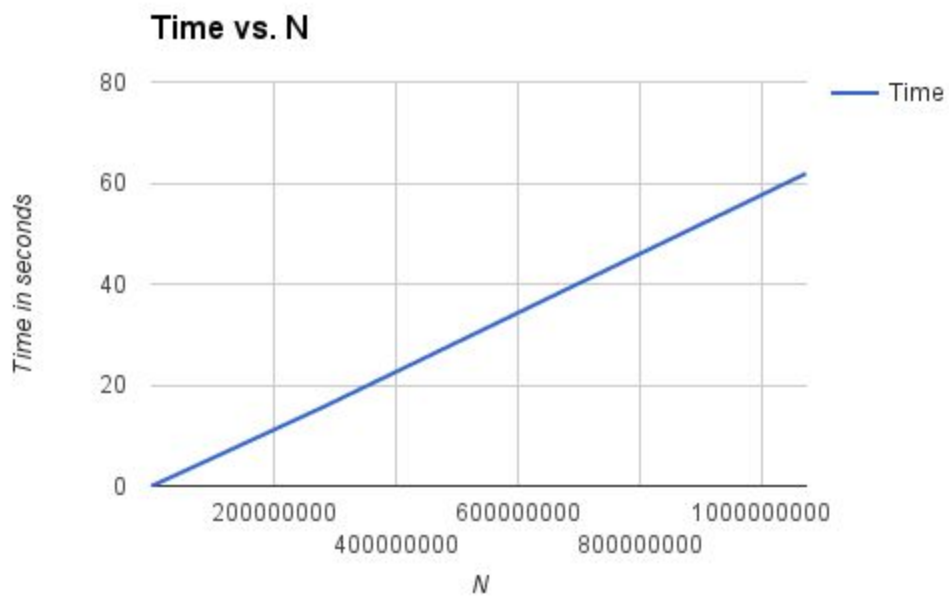


Table for the above plot

N	Time	N	Time
1	0.000095	9765625	0.546994
1024	0.000244	60466176	3.352676
59049	0.012895	282475249	15.709877
1048576	0.079398	1073741824	61.880483

For algorithm using a matrix

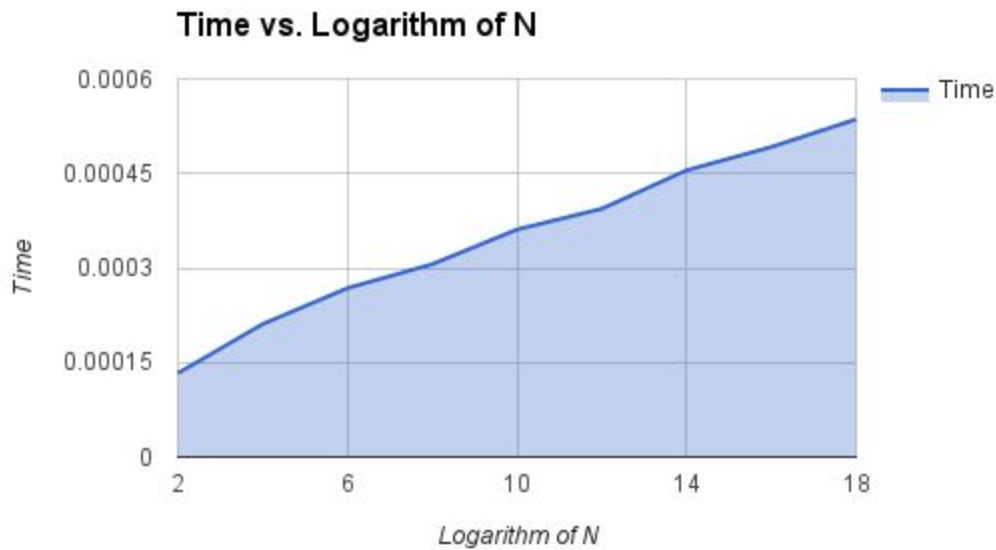


Table for above plot

Logarithm of N	Time	Logarithm of N	Time
2	0.000133	12	0.000394
4	0.000211	14	0.000455
6	0.000268	16	0.000492
8	0.000306	18	0.000536
10	0.000361		

All the above graphs are tending to be straight lines as it should be. This is because the first one has an exponential time complexity. So the logarithm of time vs N is linear, although it is not perfectly one because initially the number of computations are lower, so comparable to other operations done. In the iterative algorithm, the time complexity is linear leading to a perfect straight line in the time vs N curve. Finally in the matrix multiplication algorithm, the time complexity is logarithmic. Thus time vs log of N makes a linear curve too.

Another thing to note is that initially the matrix version was slowest, that is because in this we need to do several operations like matrix creation, multiplications etc which can be achieved in much fewer steps in the other 2 algorithms.