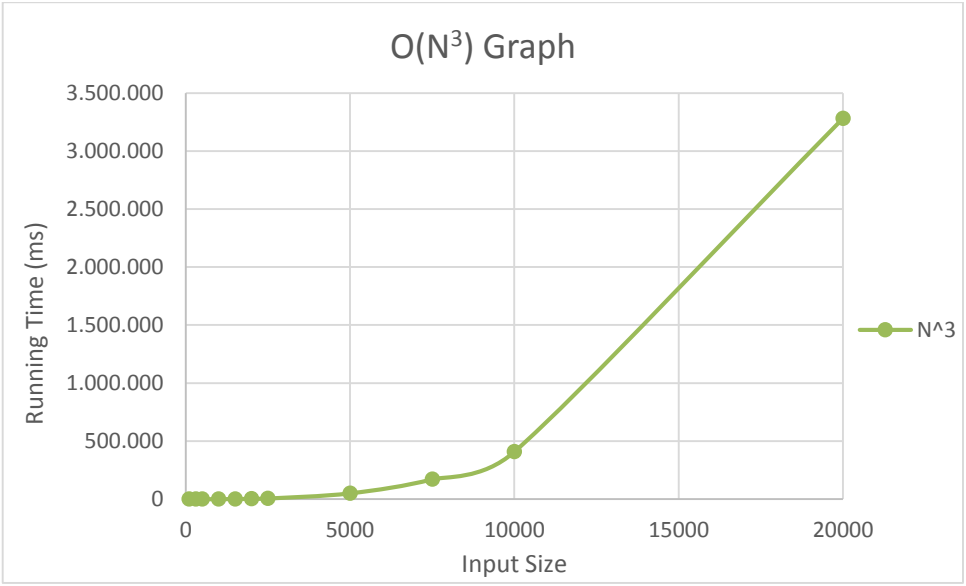
Ömer Mesud TOKER
Section 2 - 21302479

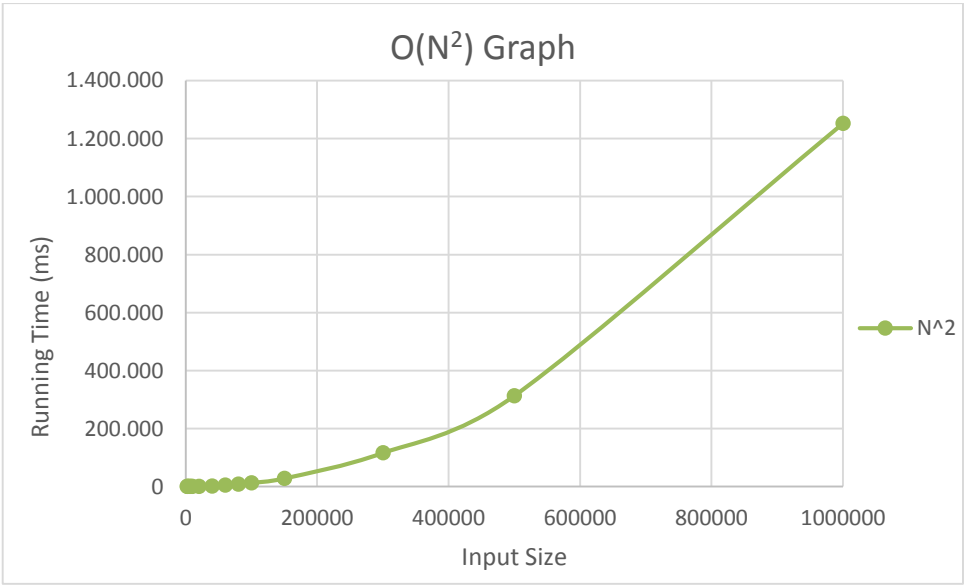
CS 201 HOMEWORK 2

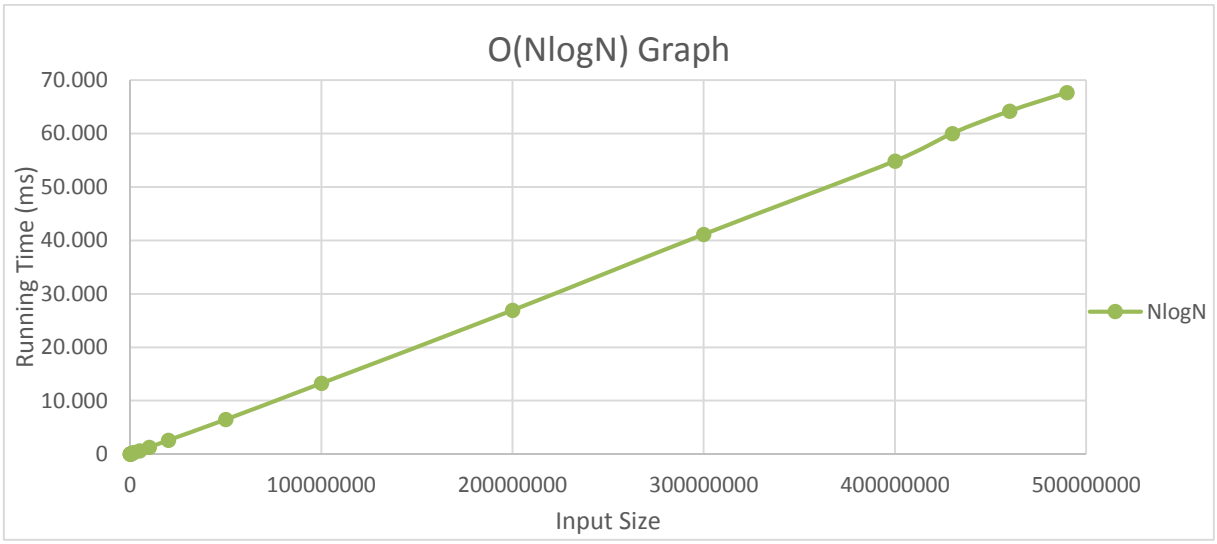
Table:

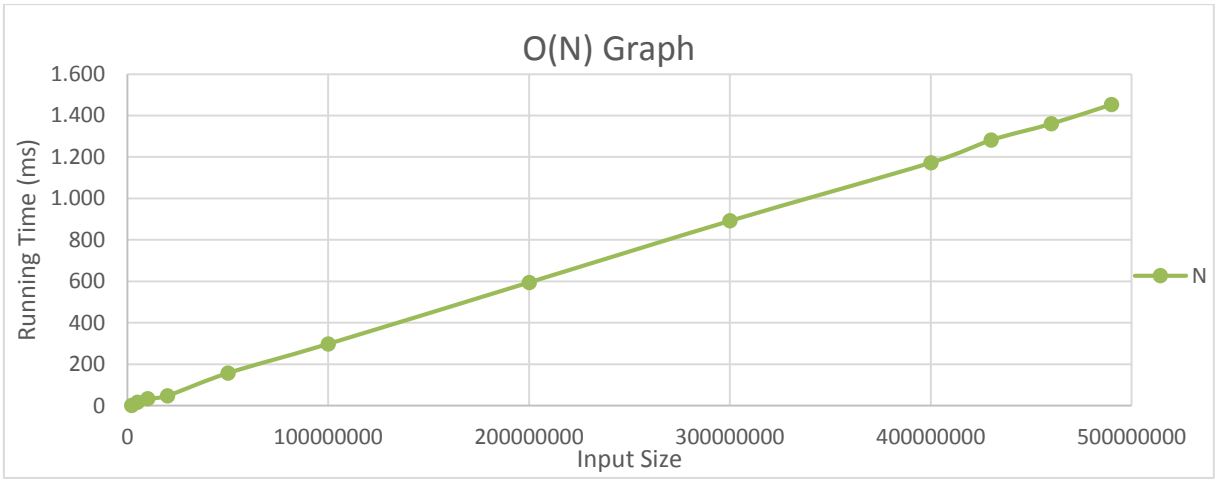
	Running Time (ms)			
Input Size	$O(N^3)$	$O(N^2)$	O(NlogN)	O(N)
N = 100	0	0	0	0
N = 300	15	0	0	0
N = 500	47	0	0	0
N = 1000	406	0	0	0
N = 1500	1375	0	0	0
N = 2000	3240	0	0	0
N = 2500	6281	15	0	0
N = 5000	50649	31	0	0
N = 7500	171604	63	0	0
N = 10000	408207	125	0	0
N = 20000	3.28016×10^6	515	0	0
N = 40000	NA	2015	0	0
N = 60000	NA	4515	16	0
N = 80000	NA	8000	15	0
N = 100000	NA	12844	16	0
N = 150000	NA	28172	15	0
N = 300000	NA	116103	47	0
N = 500000	NA	312895	62	0
N = 1000000	NA	1.25195×10^6	125	0
N = 2000000	NA	NA	328	0
N = 5000000	NA	NA	625	16
N = 10000000	NA	NA	1265	32
N = 20000000	NA	NA	2610	47
N = 50000000	NA	NA	6484	156
N = 100000000	NA	NA	13250	297
N = 200000000	NA	NA	26916	594
N = 300000000	NA	NA	41125	891
N = 400000000	NA	NA	54844	1172
N = 430000000	NA	NA	60025	1281
N = 460000000	NA	NA	64188	1360
N = 490000000	NA	NA	67667	1453

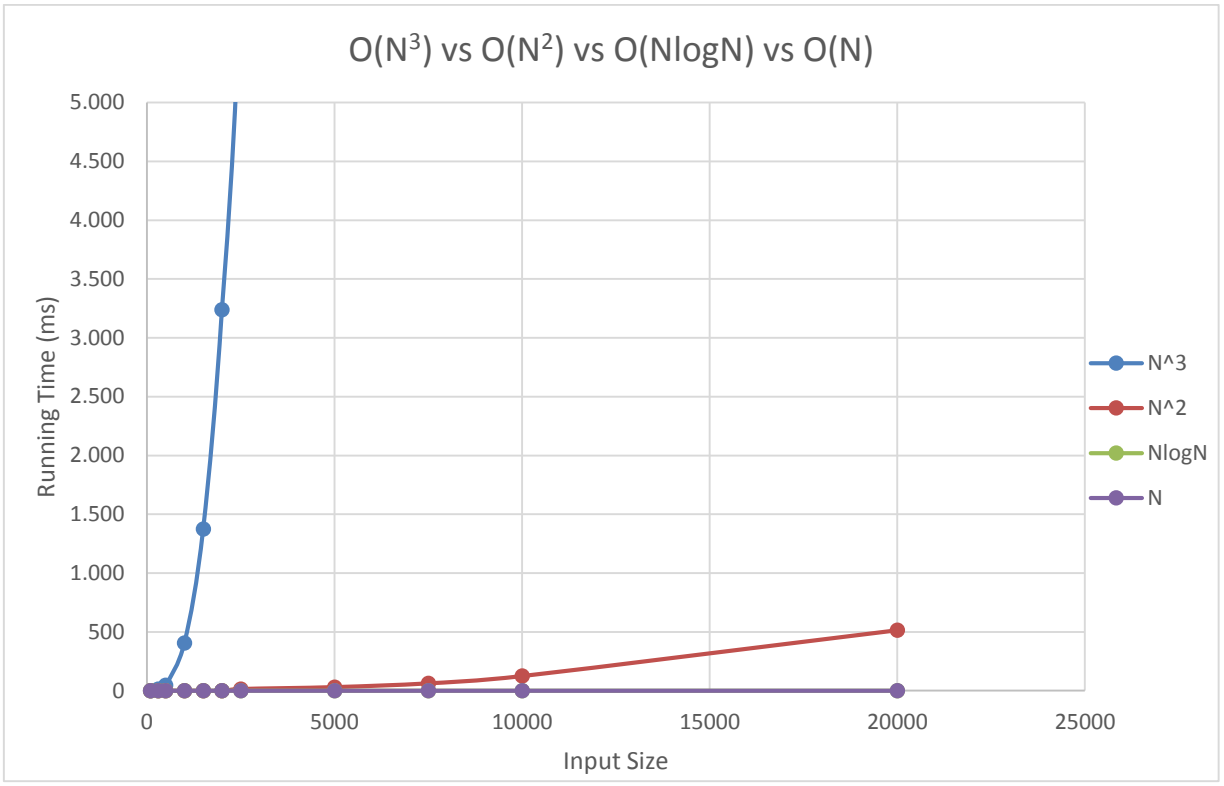
Graphs: First of all, I am going to show each algorithm's graph individually to illustrate their behavior properly. Then, I am going to show comparable ones together to understand the algorithms' running time behaviors appropriately.

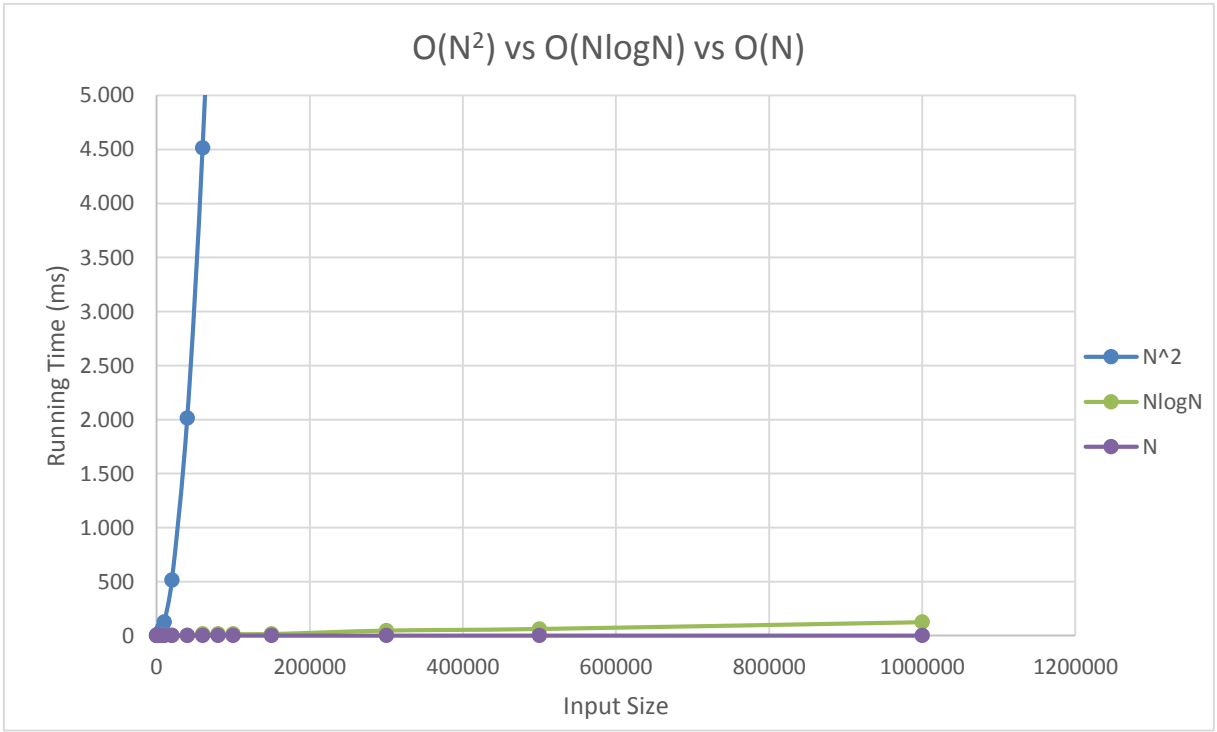


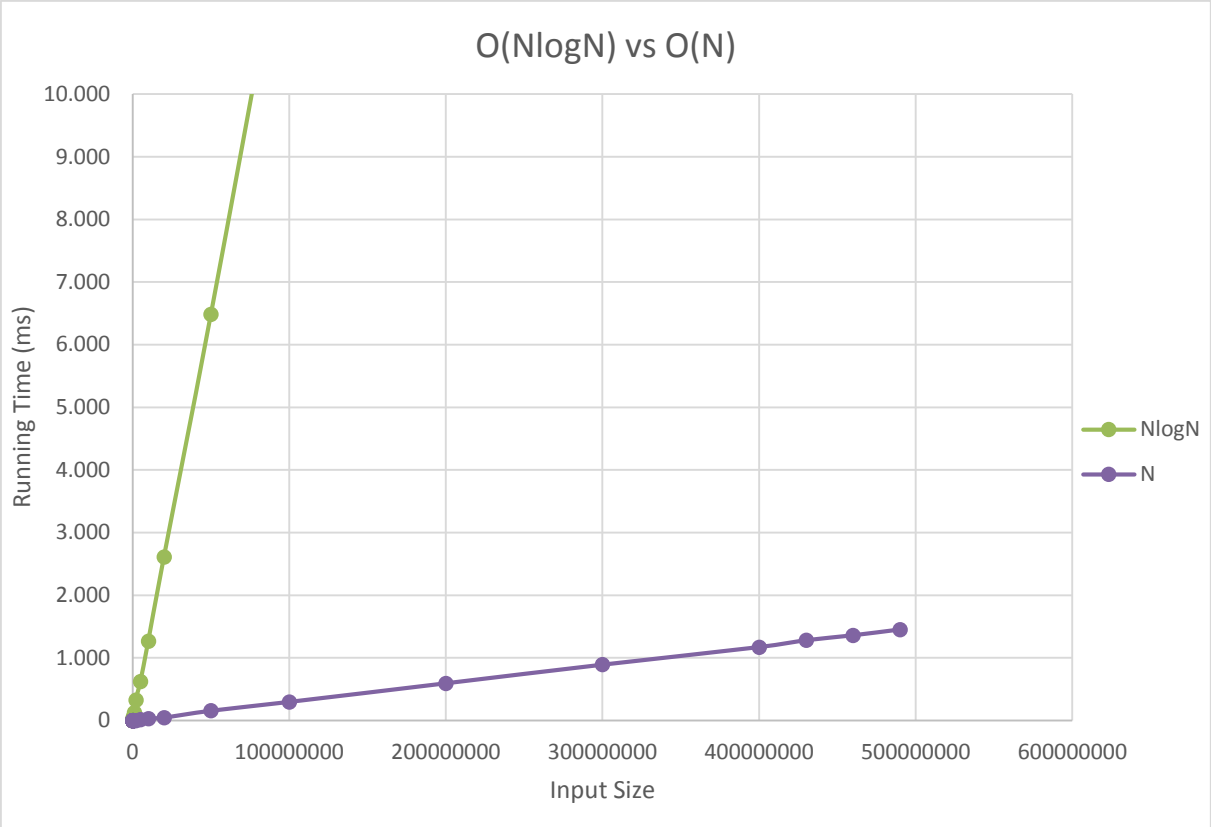



O(NlogN) Graph
70.000
60.000
50.000
40.000
30.000
20.000
10.000
0
Running Time (ms)
0
100000000
200000000
300000000
400000000
500000000
Input Size
NlogN


O(N) Graph
1.600
1.400
1.200
1.000
800
600
400
200
0
Running Time (ms)
0
100000000
200000000
300000000
400000000
500000000
Input Size
N


O(N3) vs O(N2) vs O(NlogN) vs O(N)
Running Time (ms)
5.000
4.500
4.000
3.500
3.000
2.500
2.000
1.500
1.000
500
0
0
5000
10000
15000
20000
25000
Input Size
N^3
N^2
NlogN
N


O(N2) vs O(NlogN) vs O(N)
Running Time (ms)
5.000
4.500
4.000
3.500
3.000
2.500
2.000
1.500
1.000
500
0
0
200000
400000
600000
800000
1000000
1200000
Input Size
N^2
NlogN
N



Discussion:

I cannot show all algorithms in a graphic whose x axis demonstrates all integers. In order to show the behavior of the algorithms, I needed to manipulate on the axis format of the graphs. For example, $O(N^3)$ graph's y axis is from 0 to 3.500.000 and input size is from 0 to 20.000. However, for O(N) graph, y axis is from 0 to 1600 and its input size is from 0 to 500.000.000. Moreover, I compare comparable ones with appropriate input sizes to comprehend the behaviors of all algorithms.

What I learnt from those graphics is that for large numbers, algorithms are so significant. When I need to wait almost 1 hour to calculate the sum of the maximum subsequence of the array with $O(N^3)$ algorithm, O(N) algorithm gives the result immediately. (input size is 20000)

Specification of the Computer Used:

- Windows 8.1 64-bit
- Intel Core i7 – 4510U CPU @ 2.00GHz 2.60GHz
- 8 GB RAM
- The used compiler: Code::Blocks 13.12