CS201 HOMEWORK 2

Onurcan Ataç

22002194

UPPER BOUNDS OF THE ALGORITHMS

Algorithm 1: Since algorithm 1 works over arr1 for each element of arr2 by nested for loop, the complexity of the algorithm is O(m*n).

Algorithm 2: Since in binary search, if an array is n elements, it reduces the area inspected to half every time and has to find 1 value at last. Hence, if k is number of processes; n/2^k = 1 has to be true at last. If k is derived, it would be k = logn and time complexity of binary search would be O(logn). Since binary search is performed for each element of arr2, compexity of the algorithm is 0(m*logn).

Algorithm 3: In this algorithm, first arr1 is passed over in order to create a frequency table (which is a new array)for it. After that, arr2 is passed over in order to decrease values on the frequency table. Hence, complexity of the algorithm is O(n+m).

PARAMETERS OF THE COMPUTER USED

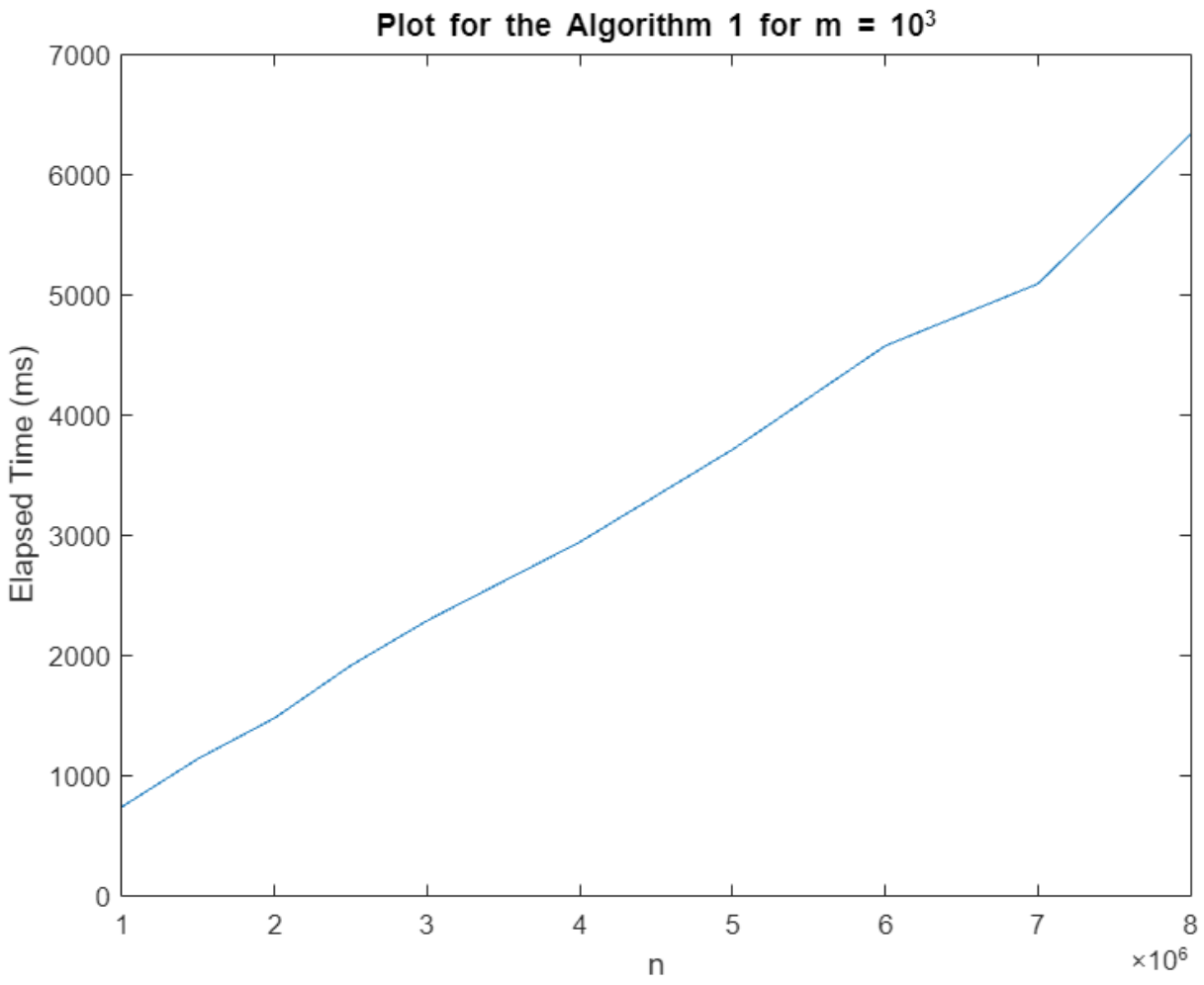
RAM: 16.0 GB

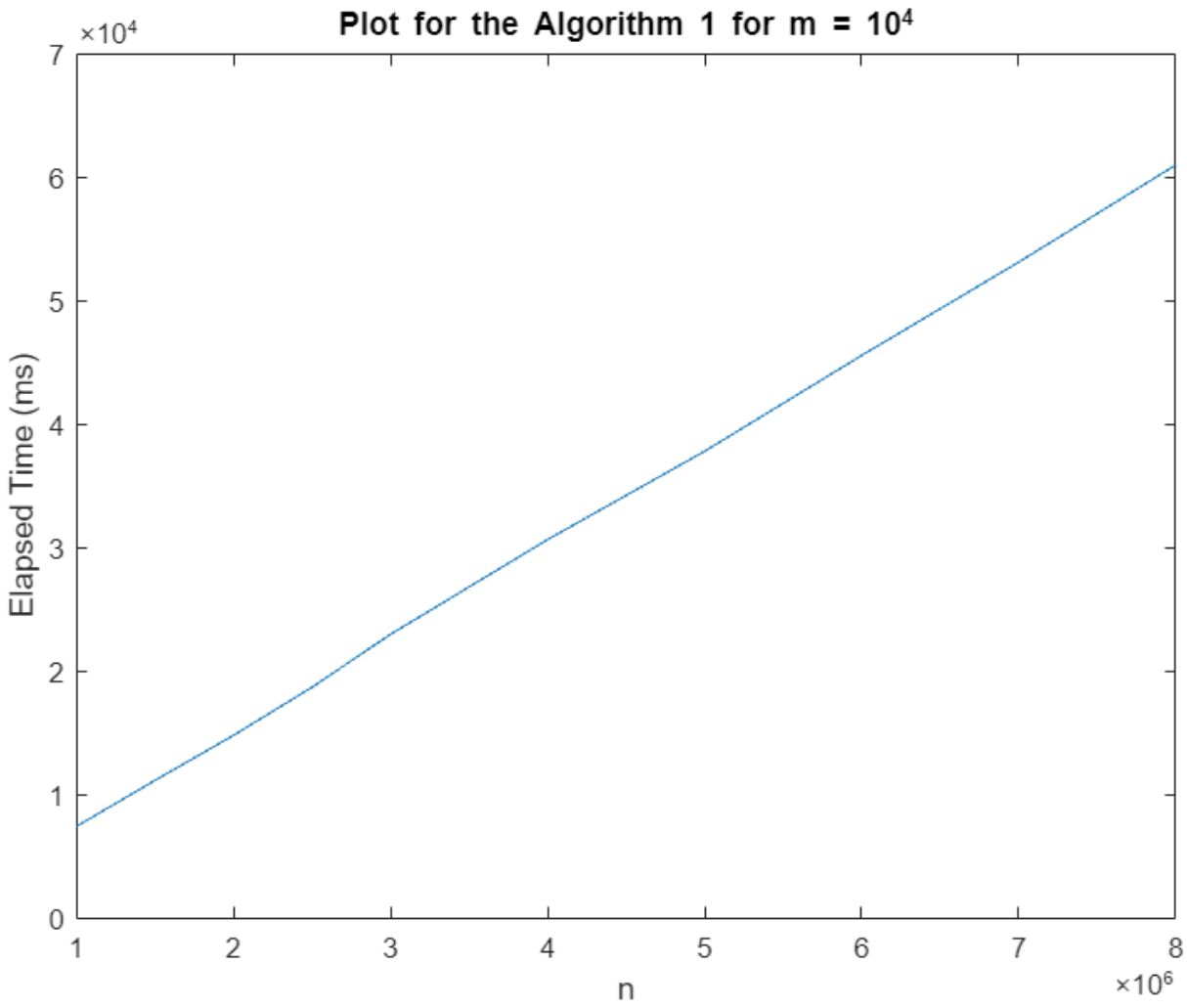
Processor: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz

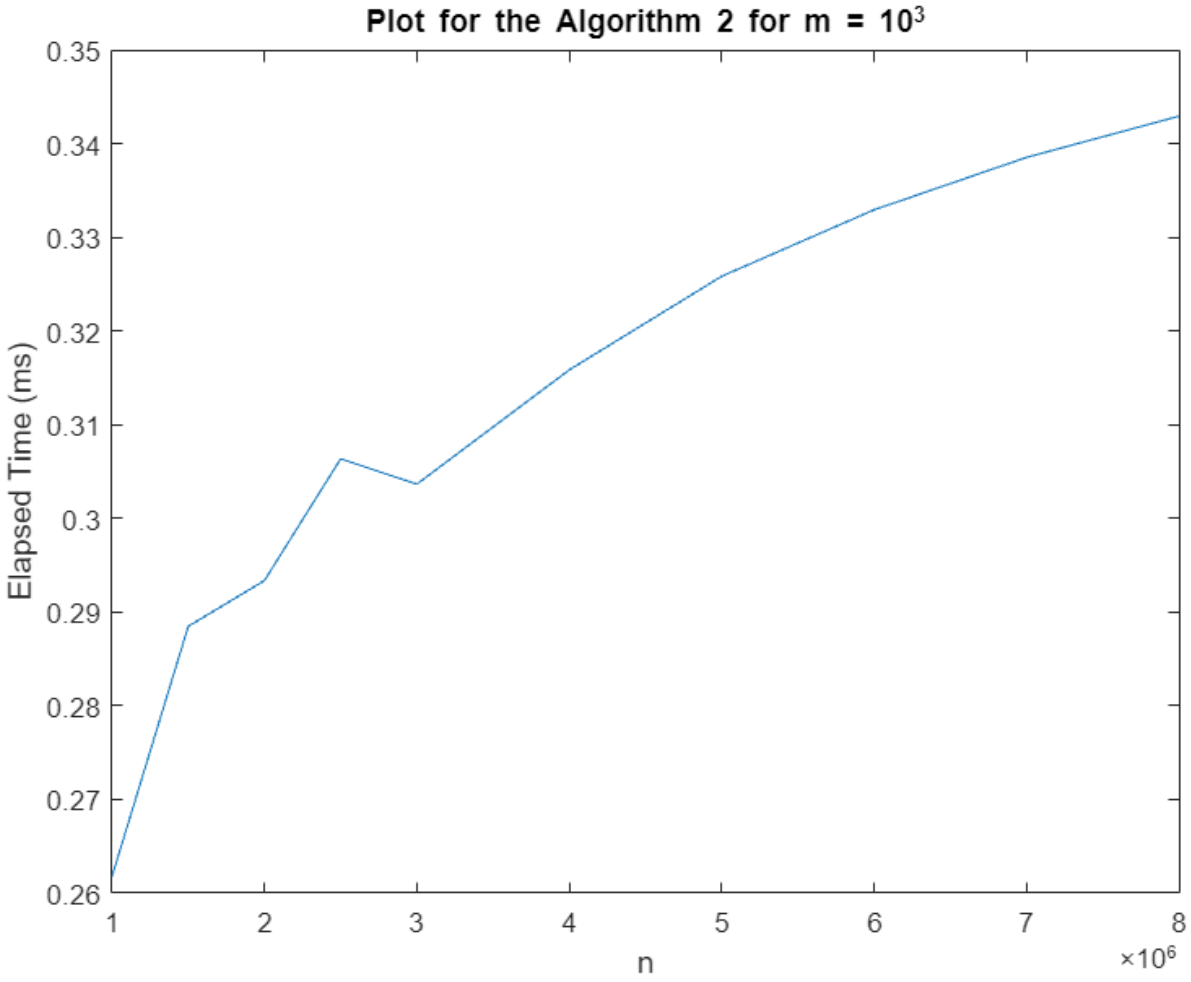
RESULT TABLE

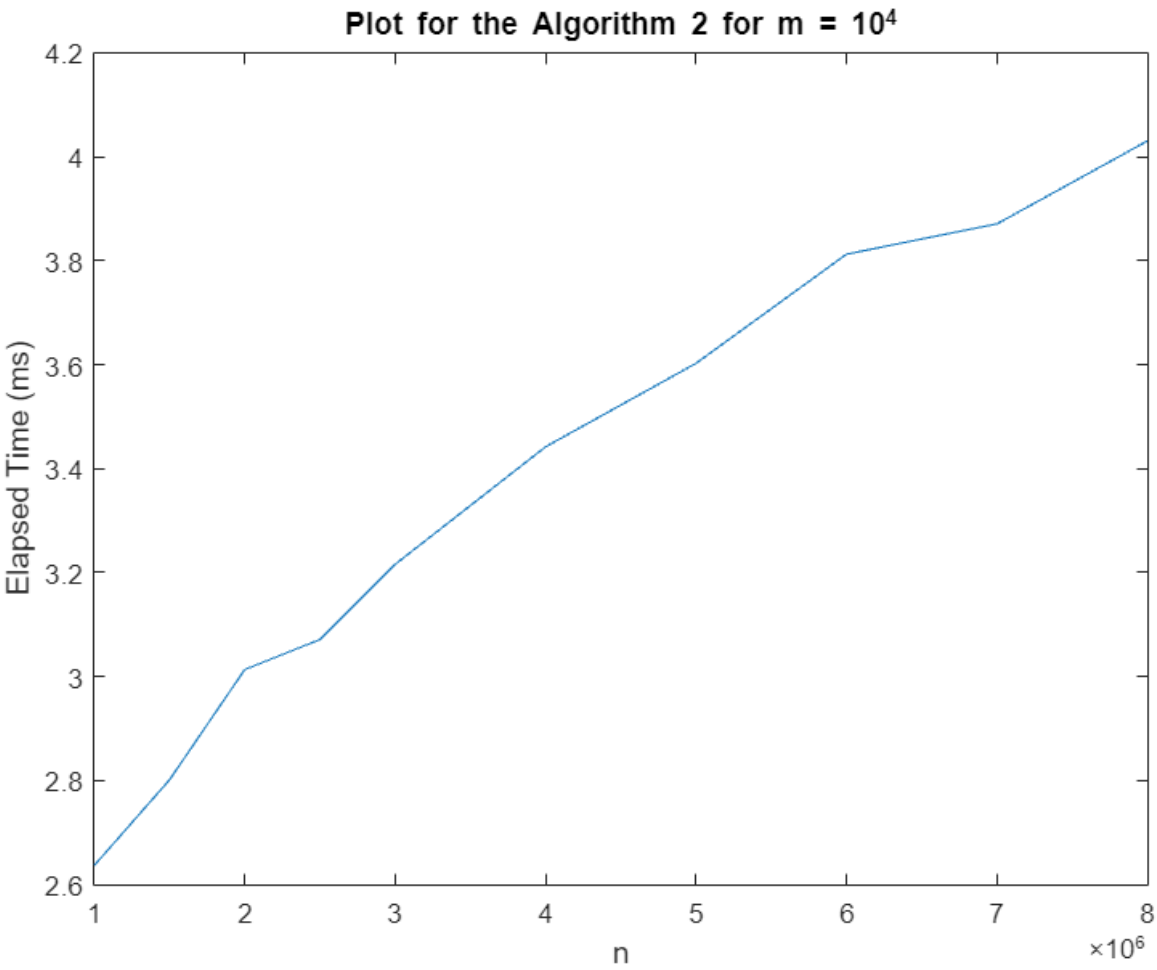
n	m = 10^3	m = 10^4	m = 10^3	m = 10^4	m = 10^3	m = 10^4
10^6	735.6 ms	7387.9 ms	0.2618 ms	2.6365 ms	5.563 ms	5.342 ms
1.5 * 10^6	1137.4 ms	11119.8 ms	0.2885 ms	2.8012 ms	8.425 ms	8.063 ms
2 * 10^6	1475 ms	14774.1 ms	0.2934 ms	3.0134 ms	11.501 ms	10.933 ms
2.5 * 10^6	1912.3 ms	18671.8 ms	0.3064 ms	3.0711 ms	14.198 ms	13.611 ms
3 * 10^6	2286.7 ms	22963.2 ms	0.3037 ms	3.2159 ms	17.046 ms	16.444 ms
4 * 10^6	2941.1 ms	30644.8 ms	0.3159 ms	3.442 ms	22.765 ms	22.994 ms
5 * 10^6	3710.8 ms	37804.1 ms	0.3259 ms	3.6027 ms	28.467 ms	30.995 ms
6 * 10^6	4574.3 ms	45577.6 ms	0.333 ms	3.8126 ms	34.019 ms	36.529 ms
7 * 10^6	5091 ms	53140.5 ms	0.3386 ms	3.8709 ms	39.727 ms	42.996 ms
8 * 10^6	6333.6 ms	61019 ms	0.343 ms	4.0305 ms	50.173 ms	55.532 ms

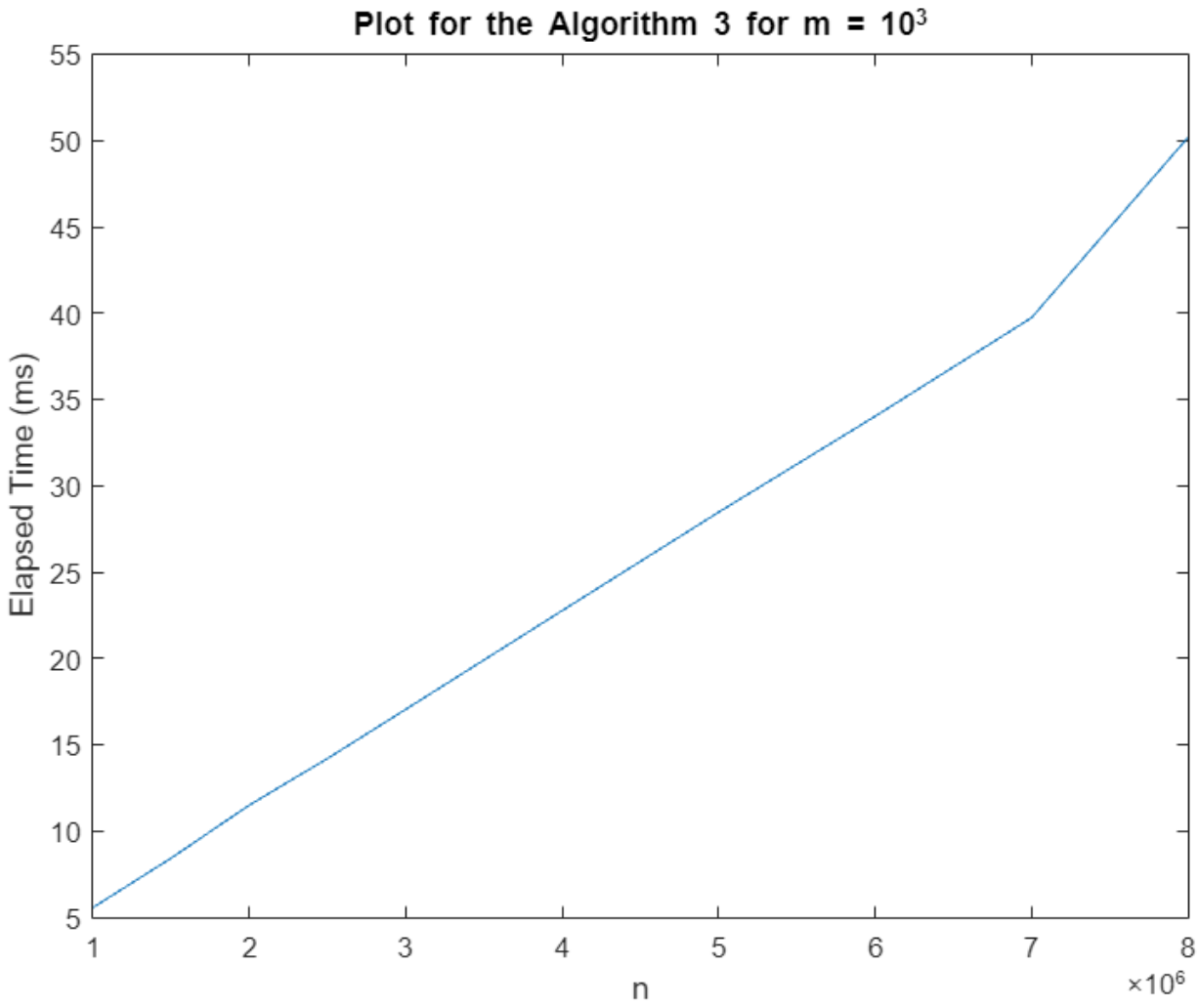
PLOTS

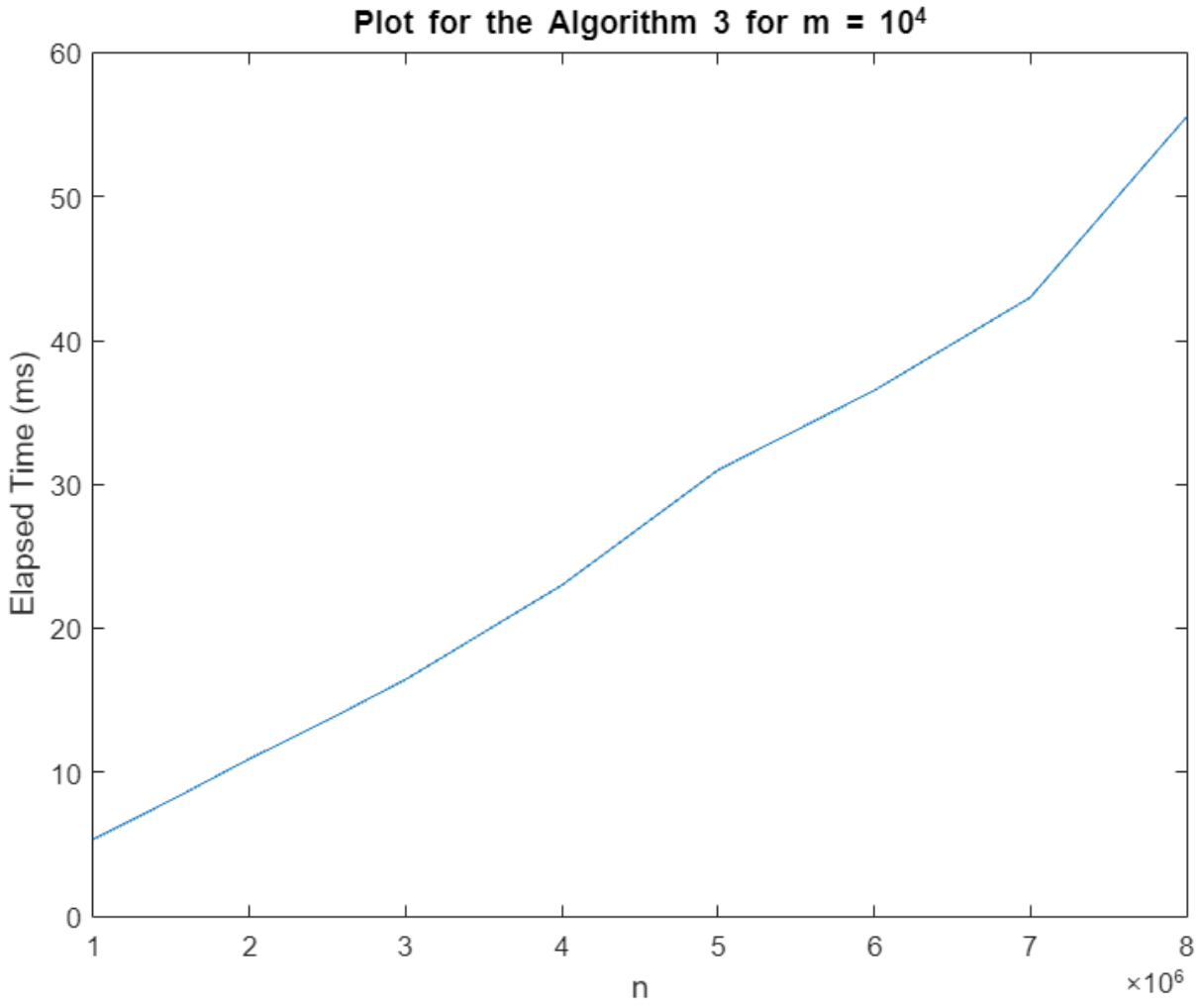

Plot for the Algorithm 1 for m = 10^3
Elapsed Time (ms)
7000
6000
5000
4000
3000
2000
1000
0
1
2
3
4
5
6
7
8
n
×10^6


Plot for the Algorithm 1 for m = 10^4
×10^4
7
6
5
4
3
2
1
0
Elapsed Time (ms)
1
2
3
4
5
6
7
8
n
×10^6


Plot for the Algorithm 2 for m = 10^3
Elapsed Time (ms)
0.35
0.34
0.33
0.32
0.31
0.3
0.29
0.28
0.27
0.26
1
2
3
4
5
6
7
8
n
×10^6


Plot for the Algorithm 2 for m = 10^4
Elapsed Time (ms)
4.2
4
3.8
3.6
3.4
3.2
3
2.8
2.6
1
2
3
4
5
6
7
8
n
×10^6


Plot for the Algorithm 3 for m = 10^3
Elapsed Time (ms)
55
50
45
40
35
30
25
20
15
10
5
1
2
3
4
5
6
7
8
n
×10^6


Plot for the Algorithm 3 for m = 10^4
Elapsed Time (ms)
0
10
20
30
40
50
60
1
2
3
4
5
6
7
8
n
×10^6