Name: Nnamdi Kelvin Ikenna-Obi (nzi0007)
Comp 3270 Programming Assignment

Algorithm1

Step	Cost of each execution	Total # of times executed
1	1	1
2	1	n + 1
3	1	n(n + 1)
4	1	n^2
5	1	n^2(n + 1)
6	6	n^3
7	7	n^2
8	2	1

Multiply col.1 with col.2, add across rows and simplify
$T_1(n)$ = 1 + n+1 + n(n+1) + n2 + n2(n + 1) + 6n3 + 7n2 + 2
= 7n3 + 10n2 + 2n + 4
Big-O Notation of Algorithm 1: O(n3)

Algorithm2

Step	Cost of each execution	Total # of times executed
1	1	1
2	1	n + 1
3	1	n
4	1	n(n + 1)
5	6	n^2
6	7	n^2
7	2	1

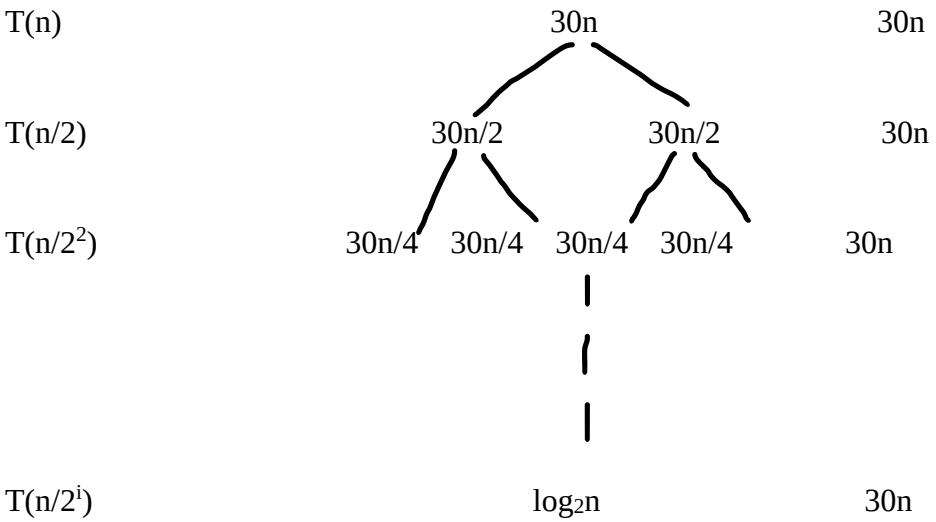
Multiply col.1 with col.2, add across rows and simplify
$T_2(n)$ = 1 + n+1 + n + n(n+1) + 6n2 + 7n2 + 2
= 14n2 + 3n + 4
Big-O Notation of Algorithm 2: O(n2)

Algorithm3

Step	Cost of each execution	Total # of times executed in any single recursive call
1	4	1
2	7	1
Steps executed when the input is a base case: line 1 or line 2		
First recurrence relation: T(n=1 or n=0) = 7 when n = 1 and 4 when n = 0		
3	5	1
4	2	1
5	1	n + 1
6	6	n

7	8	n
8	2	1
9	1	n + 1
10	6	n
11	8	n
12	4	1
13	4	(cost excluding the recursive call)
14	5	(cost excluding the recursive call)
15	14	1
Steps executed when input is NOT a base case: 13 steps		
Second recurrence relation: T(n>1) = 2T(n/2) + 30n + 38		
Simplified second recurrence relation (ignore the constant term): T(n>1) = 2T(n/2) + 30n		

Solve the two recurrence relations using any method (recommended method is the Recursion Tree). Show your work below:



$T_3(n) = 30n \log_2 n + 30n$

Big-O Notation of Algorithm 3: O(n log(n))

Algorithm4

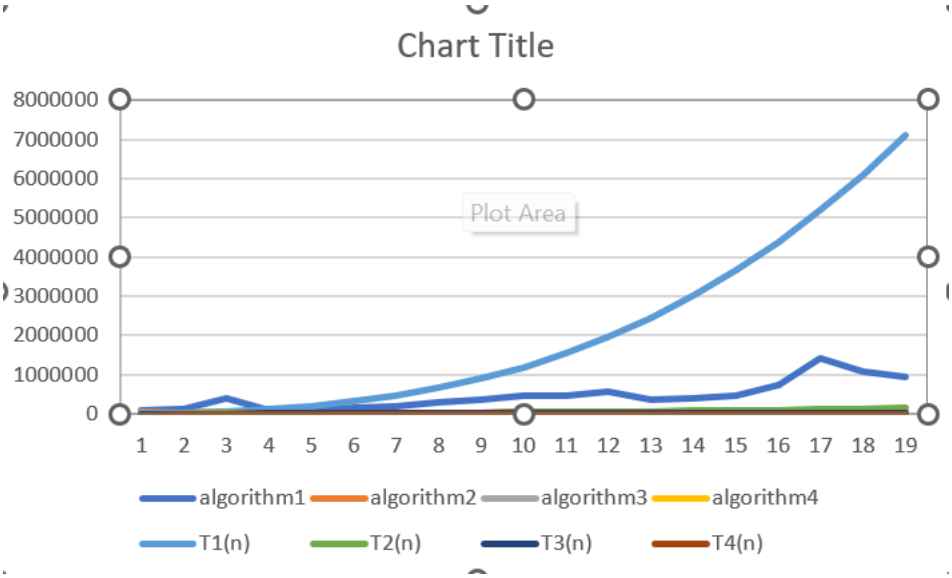
Step	Cost of each execution	Total # of times executed
1	1	1
2	1	1
3	1	n + 1
4	7	n
5	7	n
6	2	1

Multiply col.1 with col.2, add across rows and simplify

$T_4(n) = 1 + 1 + n+1 + 7n + 7n + 2$

$= 15n + 5$

Big-O Notation of Algorithm 4: O(n)



This graph shows the Theoretical analysis vs empirical analysis with respect to n. The predicted time complexity for each algorithm did not match the actual time taken by each algorithm except for algorithm 1, as for algorithm 3 and 4, they seem to have deviated a lot with their respective curvatures and algorithm 2 slightly.