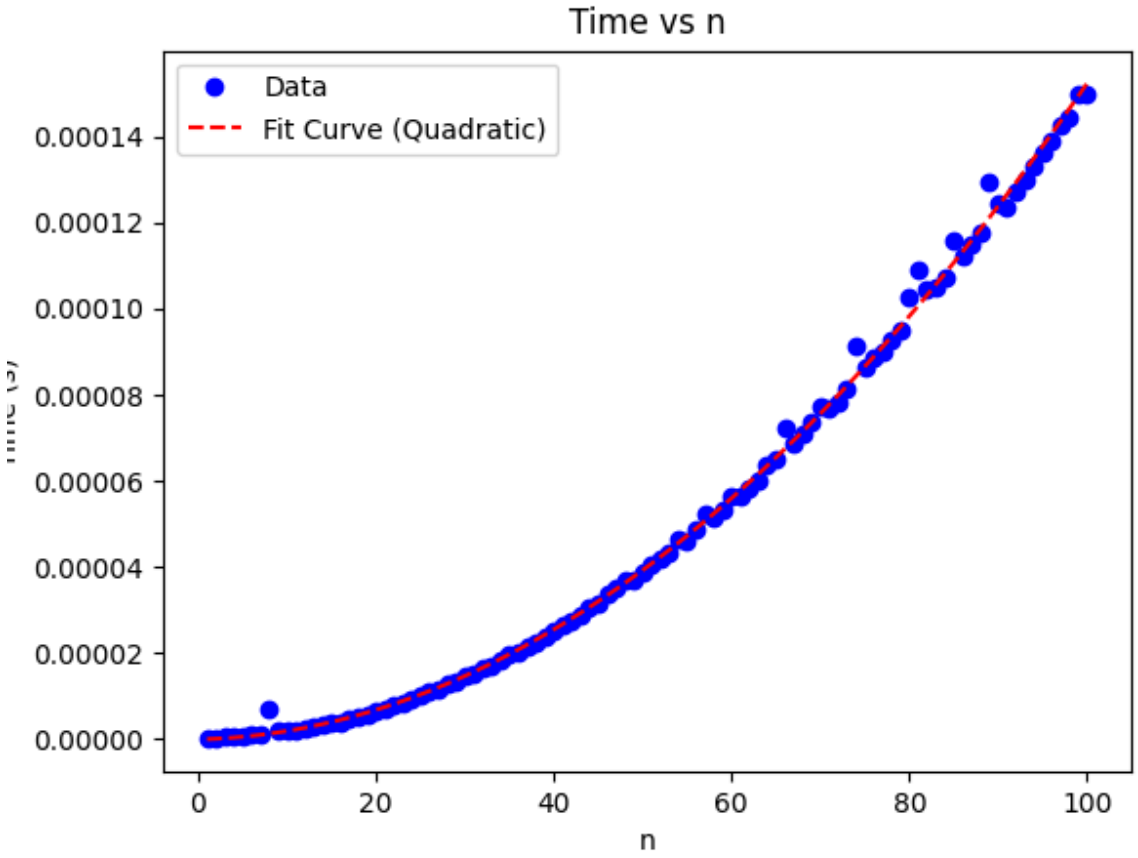
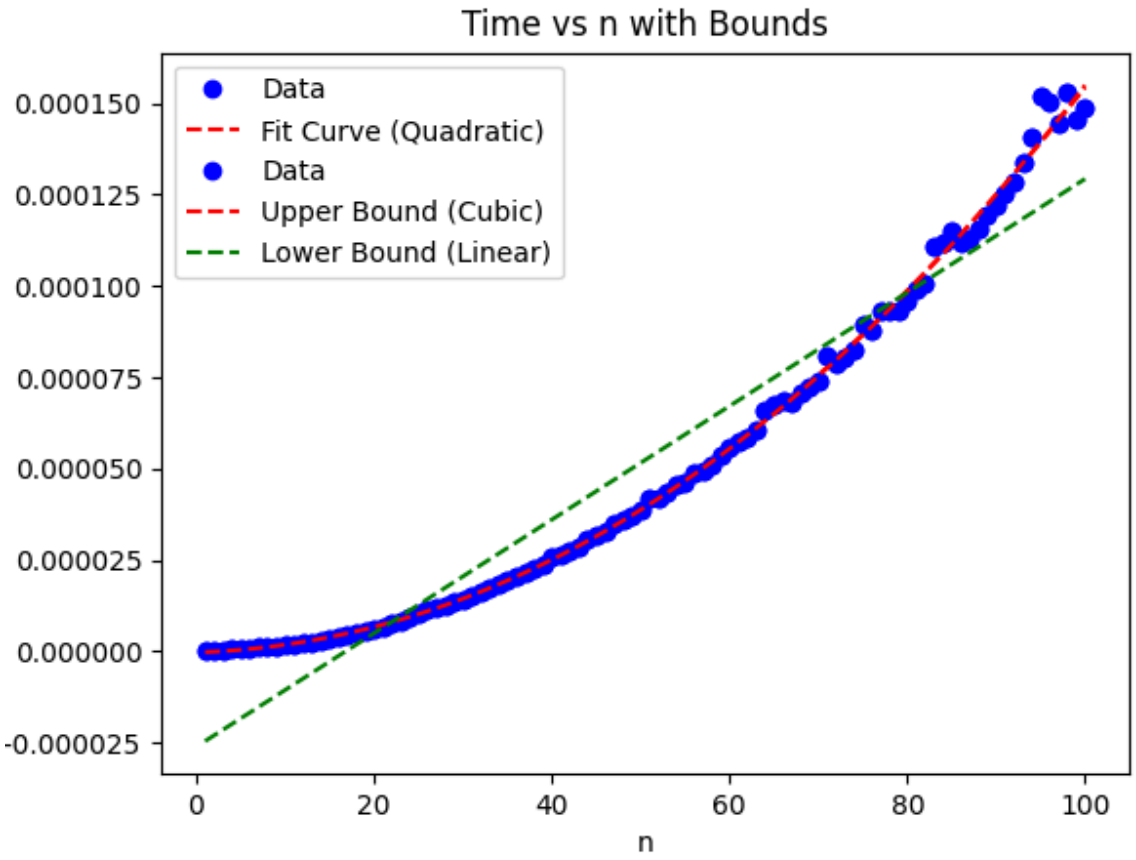
2242-CSE-5311-006-DSGN & ANLY ALGORITHMS

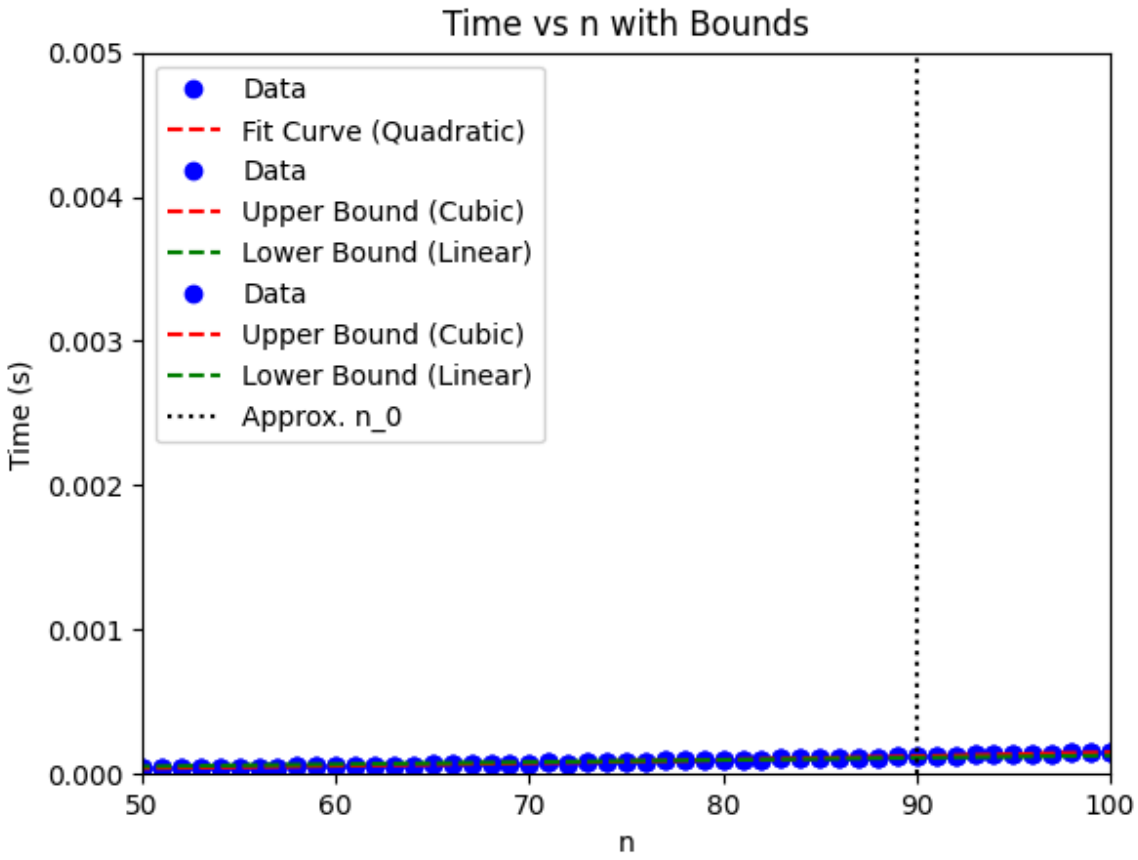
Hands on 3 Assignment : Nagmat Nazarov 1002186972

1. To find the runtime of the algorithm, let's analyze the nested loops. The outer loop runs from 1 to n, and the inner loop also runs from 1 to n. Inside the inner loop, there is a constant number of operations. So, the total number of iterations of the inner loop is n*n=n^2. Within each iteration of the inner loop, there are constant operations. Therefore, the total number of operations inside the inner loop is constant. Let's denote the number of constant operations inside the inner loop as k. Now, the total number of operations for the entire algorithm is given by the product of the number of iterations of the outer loop and the number of operations inside the inner loop: Total Operations : n*n=n^2* k. Therefore, the runtime complexity of the algorithm is O(n^2).
2.


Time vs n
Data
Fit Curve (Quadratic)
0.00014
0.00012
0.00010
0.00008
0.00006
0.00004
0.00002
0.00000
0
20
40
60
80
100
n

3.


Time vs n with Bounds
Data
Fit Curve (Quadratic)
Data
Upper Bound (Cubic)
Lower Bound (Linear)
0.000150
0.000125
0.000100
0.000075
0.000050
0.000025
0.000000
-0.000025
0
20
40
60
80
100
n



4. Yes, modifying the function to include the assignment statement y = i + j inside the nested loops will increase the time it takes for the algorithm to run. This is because with this modification, there's an additional operation being performed inside the inner loop for each iteration. Previously, the inner loop only had a constant number of operations (i.e., incrementing x). Now, with the addition of y = i + j, there's an extra arithmetic operation for each iteration of the inner loop. Therefore, the runtime complexity of the modified function will still be O(n^2), but the constant factor involved in the big-O notation will increase, resulting in longer execution times compared to the original function.

5. Yes, it will affect the results obtained from the timing analysis in #1. When you modify the function to include the additional operation y = i + j, it changes the number of operations executed within each iteration of the inner loop. Consequently, this affects the overall time taken for the function to execute for a given value of n. As a result, the timing results obtained in #1 will reflect the increased execution time due to the additional operation. In other words, the modified function will exhibit longer execution times compared to the original function, impacting the timing results obtained for various values of n.