Question 1:

$T(n) = T(0) + T(n-1) + cn$
$T(n-1) = T(0) + T(n-2) + c(n-1)$

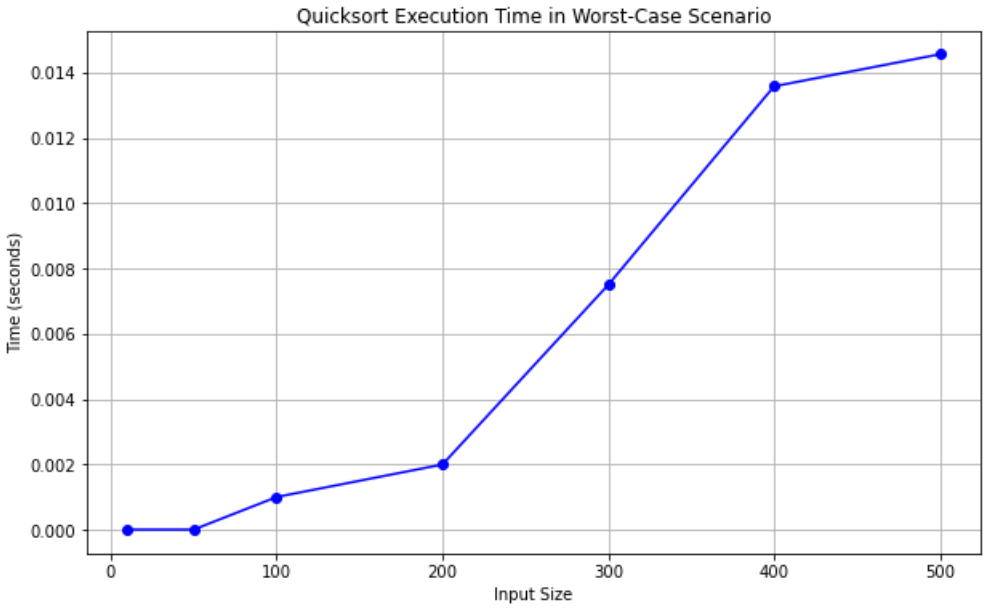
Substitute T(n-1) into equation 1:
$T(n) = 2T(0) + T(n-2) + c(n-1) + cn$

Substitute T(n-2) into equation 3:
$T(n) = 3T(0) + T(n-3) + c(n-2) + c(n-1) + cn$

Continue this process until you reach the base case:
$T(n) = nT(0) + c(n-1) + c(n-2) + ... + 2c + cn$

Simplify:
$T(n) = O(n^2)$

Questions 3 and 4:



Quicksort has an average-case time complexity of O(n log n). However, in the worst-case scenario, it degrades to O(n^2). While the log-log plot helps visualize the growth, deviations might occur due to factors like system resources, optimizations, etc. It is essential to interpret the results in the context of the experimental setup and Python's recursion limits. The experiment highlights the importance of understanding algorithmic behaviour under different scenarios.