1. In quick sort, the worst case occurs when the largest or smallest element is chosen as the pivot. When the largest or smallest element is chosen as the pivot, it causes very unbalanced partitions, causing n-1 elements to be on one side of the partition, with the remaining 1 element (the pivot) to be on the other side of the partition. Each recursive call will only effectively reduce the problem by 1 each time. Everytime the recursive (partitioning) function is called, it will do n-1 comparisons to further partition the array depending on the pivot. If the largest / smallest element is chosen as the pivot, the recursive function will be called for a total of n times, one for each element in the array. The total number of comparisons would then result in (n-1) * n = n^2 - n. So the worst case complexity is **O(n^2).**

2. Using a vector of 16 elements: *[16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1]*

- Choose the pivot = 1.
- Partition the array into 2 sub arrays such that:
    - Sub-array 1 contains elements [1]
    - Sub-array 2 contains elements [16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2]
    - Recursion continues for sub-array 2, but since sub-array 1 only has one element it is already in place.

For sub-array 2 (Recursive call):

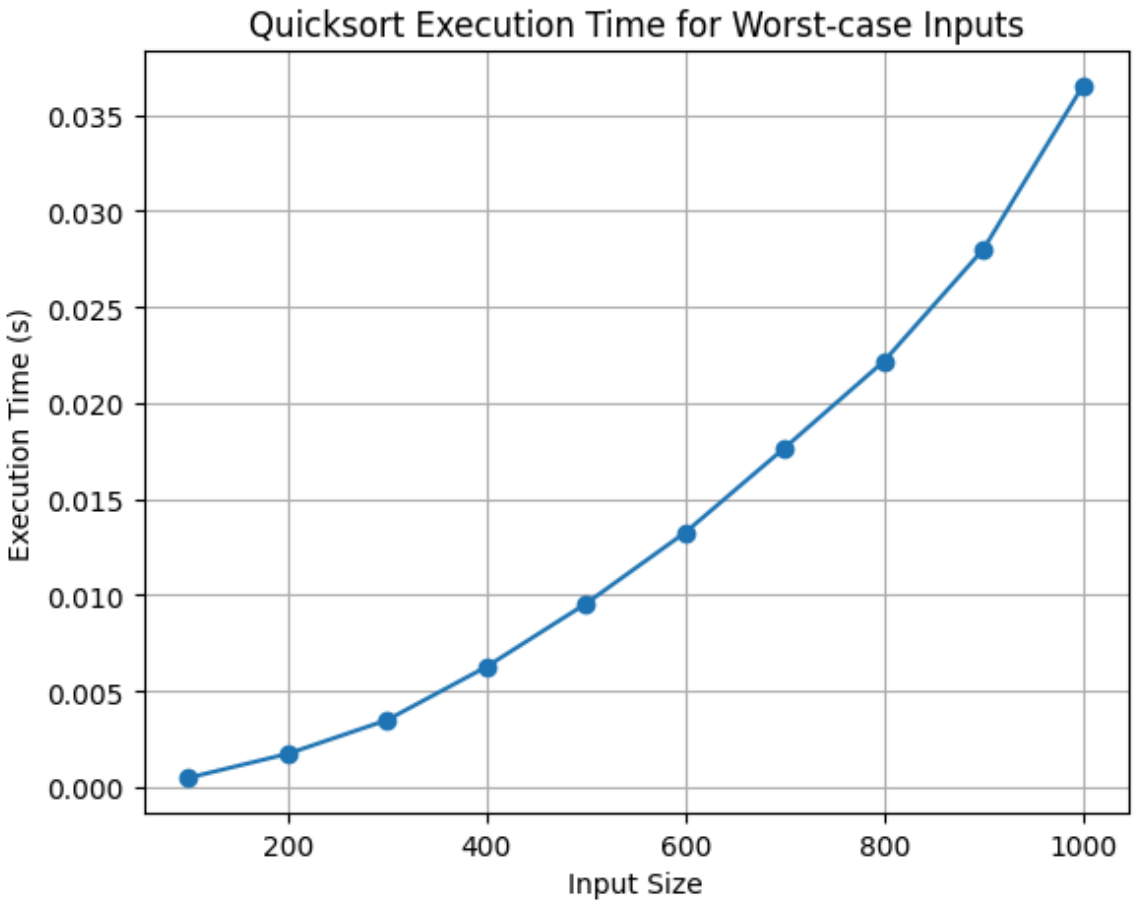
- Choose the pivot = 2.
- Partition the sub array into 2 sub arrays such that:
    - Sub-array 1 contains elements [2]
    - Sub-array 2 contains elements [16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3]
    - Recursion continues for sub-array 2, but since sub-array 1 only has one element it's already in place.

For sub-array 2 (Recursive call):

- Choose the pivot = 3.
- Partition the sub array into 2 sub arrays such that:
    - Sub-array 1 contains elements [3]
    - Sub-array 2 contains elements [16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4]
    - Recursion continues for sub-array 2, but since sub-array 1 only has one element it's already in place.

This is further continued with recursive calls for a total of 16 times until the vector is fully sorted.

4.



Yes, the plot matches the complexity analysis of O(n^2), showing a quadratic relationship.