(i) Number of Comparisons

The number of comparisons in bubble sort depends on the array's length n, and does not vary with the array's initial order (i.e., it's the same for the worst, best, and average cases).

- In the first pass, bubble sort compares the first and second elements, then the second and third, and so on up to the second to last and elements, for a total of n-1 comparisons.
- In the second pass, it makes $n-2$ comparisons.
- In the last pass, it makes only 1 comparison.

Therefore, the total number of comparisons is the sum of the first $n-1$ numbers:

Total comparisons=n(n-1)/2 (From Gauss)

(ii) Average-Case Number of Swaps

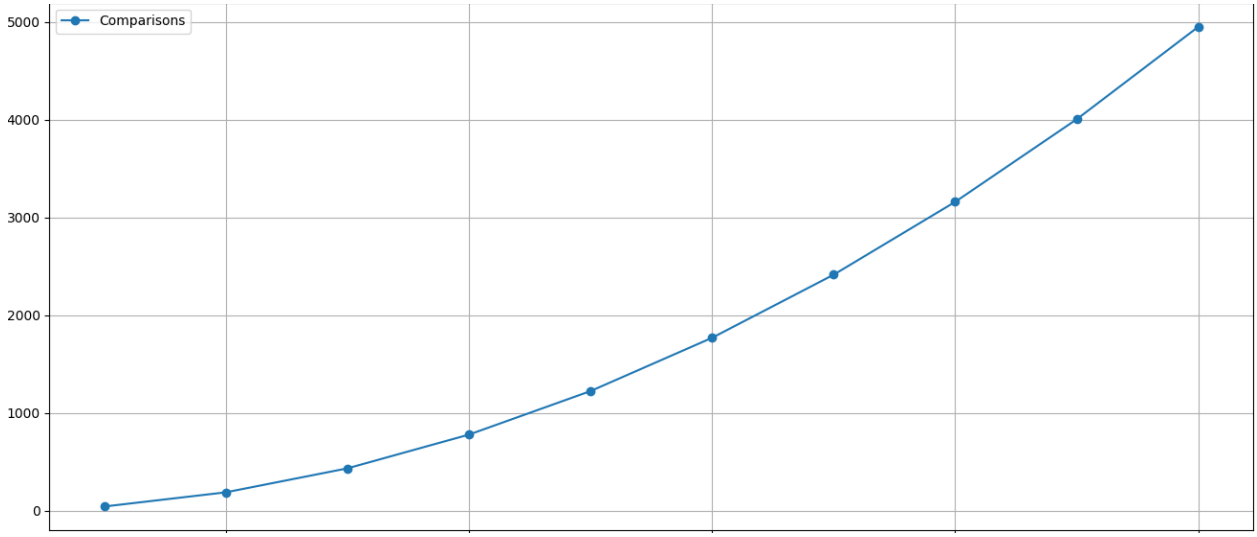
In the worst case (the array is in reverse order), bubble sort performs the maximum number of swaps, which is also n(n-1)/2. In the best case (the array is already sorted), it performs no swaps.

For a random array, the chance that any given pair of elements needs to be swapped is 50 percent. This is because there's an equal chance that a pair is in the correct order or the wrong order.

Using this assumption, we can estimate the average number of swaps as half of the total number of comparisons (since about half the comparisons will result in a swap):

Average-case swaps = n(n-1)/4

Using this we can estimate the behavior to be O(n^2)



It somewhat matches our analysis as it is slightly exponential