Bubble Sort Algorithm.

Bubble Sort is simple comparison-based sorting algorithm. It works by repeatedly swapping abjacent element if they are in wrong order. This process continues until array is fully sorted

Example :- [3, 1, 5, 4, 2]

First pass :- 3, 1, 5, 4, 2

1, 3, 5, 4, 2

1, 3, 4, 5, 2

1, 3, 4, 2, 5

- with first pass the largest element came to end.

Second pass :- 1, 3, 4, 2, 5

1, 3, 2, 4, 5

- with second pass the 2nd largent element came to 2nd last ~~in the~~ position.

3rd pass :- 1, 3, 2, 4, 5

1, 2, 3, 4, 5

∴ Sorted

Space Complexity :-

Space Complexity = O(1) // Constant
// No extra Space required.
i.e Copying the array etc. not required.
aka ~~in~~ inplace shorting algorithms

TIME COMPLEXITY :-

Best Case : O(N) ⇒ Sorted Array

Worst Case : $O(N^2)$ ⇒ Sorted in opposite order.
means array is given in descending order & we have to sort the array in ascending

① Best Case →
i = 0

j j j j
1 . 2 . 3 . 4 . 5
x x x

→ it run for once & we don't have to check it again

Note :- When j never swap for value of i, it means array is sorted.
Hence, you can end the program.

No of Comparision = N-1 = N.

- In time complexity constant is ignored as we only want relationship not exact time (Mathematical function)

Worst Case :-

i=0 — 1st pass

5, 4, 3, 2, 1

4, 5, 3, 2, 1

4, 3, 5, 2, 1

4, 3, 2, 5, 1

4, 3, 2, 1, 5 ⇒ (N-1) Comparision

i=1 — 2nd pass

4, 3, 2, 1, 5 — already sorted

3, 4, 2, 1, 5

3, 2, 4, 1, 5

3, 2, 1, 4, 5 ⇒ N-2 comparision

i=2 — 3rd pass

3, 2, 1, 4, 5 — already sorted

2, 3, 1, 4, 5

2, 1, 3, 4, 5 ⇒ N-3 comparision

i=3 — 4th pass

2, 1, 3, 4, 5 → already sorted

1, 2, 3, 4, 5 ⇒ N-4 comparision

Date
Page

total Comparision = $N-1 + N-2 + N-3 + N-4$

$= 4N - (1+2+3+4)$

$= 4N - \left[\frac{N \times (N+1)}{2}\right]$

$= 4N - \frac{N^2+N}{2}$

$\Rightarrow \frac{7N-N^2}{2}$ ⇒ In time complexity constant & less dominating term are ignored.

So; Time complexity = $O(N^2)$

Stable Sho

- Stable Sorting Algorithms :-

10 20 20 30 10

↓ sort.

10 10 20 20 30

Here ; In original array (black)10 is before 10(red) & this is maintained in sorted array.

- Unstable Sorting Algorithm :-

10 20 20 30 10

↓ sort

10 10 20 20 30

Here ; Order is not mainted. in sorted array.