SORTING AGORITHMS

1. Size of data
2. randomness of data
3. time and space complexity
4. stability

Q. What makes a sorting algorithm stable?

Ans🡪 Stability of a sorting algorithm means in case of equal elements the order of occurrence is maintained

Q. What causes this instability?

Ans🡪Swap of non-adjacent elements causes instability

Q. Why is this order if the elements are same?

Ans🡪 In case of leaderboard if two scores are same then while sorting the first occurrence of the equal score must be kept above.

1. BUBBLE SORT

* slow n2 as worst and n for best
* almost sorted data
* stable (swaps only adjacent elements)

1. INSERTION SORT

* best n worst n2
* works good for small data (approx. 30)
* stable (no swaps)

1. MERGE SORT

* based on divide and conquer algorithm
* suitable for large data
* but takes space n
* n logn best and worst
* no swaps so stable

1. QUICK SORT

* N logn best and average
* worst case O(n2)
* we usually consider it average case because of its randomness
* not a stable algorithm

1. HEAP SORT

* N logn best and worst
* no space complexity
* not maintain relative order so not stable

1. SLECTION SORT

* suitable for small data
* slow n^2 both best and worst
* special feature -> It retains first k smallest elements in 1st k iterations
* un stable because of non-adjacent elements