Result Of Sort

Day : Thursday
Date : 31/03/2022

Quicksort	10	50	100
ascending	5.6667	88.6667	155.333
descending	6.3333	208.3333	105.6667
Random order	7.0	40.333	28.333
Nearly sorted	5.6667	96.333	155.333

Explaination : Divide and conquer, you pick an element as pivot and partition the given Arrays, the partitioning puts everything in place worst case = O(n2) Bestcase = O(nlogn)

It is the fastest compared to another sorts in this ascending and descending in itself four fastest Random is the slowest for higher values values keep increasing, it slow, unstable and in place

mergesort	10	50	100
ascending	13.0	136.333	224.333
descending	9.666	131.0	262.2
Randomorder	16.333	496.333	178.0
Nearly sort	10.333	118.333	357.0

Explaination : Divide and conquer algorithm, divides Array in two halves instead of merge which partitions, then it merge the two sorts. Time complexity O(nLogn) for all worst, average, best in itself descending is fastest second ascending third nearly and fourth Random order. It has a good speed, Not in place stable

QuicksortMedian3	10	50	100
Ascending	6.667	167.666	391.0
descending	7.333	127.0	552.334
RandomOrder	12.333	69.333	620.0
Nearly sort	9.666	86.0	248.333

Explaination:

Quicksort median 3 is an improvement of quicksort, but in my case it is being called 6 times so the time complexity has increased however in real, faster than left most pivot. It is inplace and unstable. Time complexity is O (N log N) worst case is $O(N^2)$. Ascending is fatest, descending on second, nearly sort third and randomorder fourth. Improves chances of getting a good pivot.

RandomQuicksort	10	50	100
Ascending	23.333	198.333	469.0
Descending	25.333	225.666	1210.333
RandomOrder	33.3336	410.333	485.0
Nearly sorted	31.666	460.667	556.0

Explaination: chooses random value and not middle index if it choose middle then would've been faster. Not best case on sorted or reversed data. Does ascending fatest, descending on second, nearly on third and random on fourth. Time complexity same as Quick sort worst case same as well