Bubble sort

Bubble sort is an exchange-based algorithm that compares adjacent values and if the one on the left is greater, it swaps them.

Best: O(n)

Average: O(n^2)

Worst: O(n^2)

Space overhead: O(1)

Stable? Yes!

Bubble sort has very poor performance due to the large amount of data movement.

Heap Sort

Heapsort is a selection-based algorithm, that uses a heap to improve performance by reducing the time required to find and extract the next “smallest” value.

The idea is that we have a 2 pass algorithm:

1. Insert all values into a heap (we will use a max heap for reasons that will become clear)🡪O(nlogn)
2. Repeatedly extract the maximum value from the heap and place it in the output array, which is build from right to left🡪O(nlogn)

Thus, the total run time required is O(nlogn)

No comparison-based sorting algorithm can have a worst case run time better than O(nlogn)

The cool thing about heapsort is that it does it all in place: rather than having an input/output array and a separate heap, they are all held in one array.

Heap sort is the only sorting algorithm with worst case O(nlogn) and contain space overhead.

Best: O(nlogn)

Average: O(nlogn)

Worst: O(nlogn)

Space overhead: O(1)

Stable? No!

Shell sort

Shell sort is an insertion-based sorting algorithm based on insertion sort. The key idea is that insertion sort runs in time proportional to the number of slots a value is away from where it belongs.

Shell sort moves values in large increment to move data faster. Its average case run time is O(n^1.25)