

DSA Prac 1

Performance Analysis

Jakob Wyatt

Bubble Sort	1024	2048	4096
Sorted	0.0009	0.0017	0.0034
Random	0.76	2.71	10.25

Insertion Sort	1024	2048	4096
Sorted	0.0009	0.0019	0.0037
Random	0.45	1.57	5.53

Selection Sort	1024	2048	4096
Sorted	0.19	0.76	3.05
Random	0.34	1.04	3.55

It can be noted that selection sort does not differ massively in performance between the already sorted and randomly ordered test cases. This is because the algorithm has to swap every array element regardless of any preordering, making its best, average, and worst cases identical.

In contrast to this, bubble sort and insertion sort are both much faster on already sorted arrays. The time taken scales linearly as data increases, confirming that their best case time complexity is $O(n)$.

Randomly sorted arrays are the average case for all sorts. With these test cases, run times increase by approximately 3.5x when the input size is doubled. This falls just below the theoretical best case complexity of $O(n^2)$.

This occurs because the time complexity only considers the number of comparisons, and disregards any swaps that occur (which are generally more computationally intensive).

It can also be seen that selection sort performs better than insertion sort, which performs better than bubble sort. This is also due to the number of swaps required.