Sorting Report.

Adrian Jonsson Sjödin

Fall 2022

Introduction

The task in this assignment was to explore some different sorting algorithms to gain a better understanding how different implementations can affect the time it takes to sort a large array.

Task

Implement the three following sorting algorithms: Selection sort, Insert sort and Merge sort. For each of the algorithms explain the run time as a function of the size of the array and state their time complexity using Big O notation. Lastly do some benchmark and compare how the different algorithms compares to each other.

Method

The Selection Sort algorithm was the first to be implemented and is quite straight forward. It takes the first element and then looks through the array to see if there's an element smaller than it. When it has looked through the whole array it swaps the first element with the found minimum element and then goes on to the next element and repeat the process until the end of the array has been reached. This means that it will have a time complexity of $\mathcal{O}(n^2)$ since it will have to take one element at a time (out of n elements) and look through an array of size n to see if there's a smaller element $n-1$ times.

Result

Discussion

The code for this assignment can be found here GitHub