

Data Structures Assignment 6, Sorting Algorithms

Sarah Monthofer, Member, IEEE

*Chapman University & Technology and Department of Electrical and Computer
California, Chapman University, Anaheim 92866 USA*

DOI: 10.1109/JPHOT.2009.XXXXXXX
1943-0655/\$25.00 ©2009 IEEE

Published December 13th, 2019

Abstract: This paper outlines the shortcomings of different Sorting Algorithms, and overviews Empirical Analysis as a whole.

1. Introduction

Throughout this document we will examine 4 sorting algorithms, Bubble Sort, Merge Sort, Selection Sort and Insertion Sort. While all Algorithms are useful at sorting data some are quicker, so we will examine which Algorithms are more effective and what makes them more effective.

2. Algorithm Comparison

Lets begin the most rudimentary Algorithm, Bubble Sort, bubble sort works by switching adjacent elements in ascending order until all elements are sorted. Because Bubble Sort must iterate through every element it's run-time in the best case would be $O(n)$ and in the worst case $O(n^2)$, and is not effective unless used on smaller data sets. Next Insert Sort looks at the current element and places it in it's location in a sorted array until all elements are sorted. Insert Sort has the same time complexity as Bubble Sort, but is far quicker than other Algorithms when the data is partially sorted. The final simple sorting algorithm we made use of was Selection Sort, with a run-time of $O(n^2)$. Selection sort works by dividing the initial data set into a sorted and unsorted sub-arrays, after each iteration the smallest element in the unsorted array is moved to the sorted array. The final Algorithm used in Assignment 6 was Merge Sort, which is most effective for larger data sets due to it's use of the divide and conquer method with a run-time of $O(N * (\log N))$. Merge Sort Works by recursively dividing the data set into halves until no longer possible then merges the smaller lists in sorted order.

3. Conclusions

After Examining the time complexity of each Algorithm we can see every Algorithm is effective but the size of the data set and the order of the data effect the performance of each Algorithm. While Mathematical Analysis can give us an idea on the performance of each Algorithm, there is always a best and worst case, and our mathematical Analysis can never but into perspective the time taken with so many changing variables and human error. From my Experience with Assignment I noticed a negligible difference in time consumption, but still a visible difference. This inference means that as the data set increases in size so while the difference in preformance.
