Assignment 6

Arshia Behzad

December 12, 2019

1 Introduction

This assignment had us use four different sorting algorithms. The ones I used were bubble sort, selection sort, insertion sort, and quick sort. I observed the time it took for each sorting algorithm, while given a large array of data to sort.

2 My Program

2.1 A Few Notes

I decided to create the numbers for the file myself, as you suggested, and so I generate a certain number of random doubles, and add them to a file. This file is then read in and the array is created from there. Finally it is sorted. Creating a lot of random numbers took a lot of time so I also included a time stamp for how long it took to create the random numbers as well.

2.2 Results

This is the ranking of the fastest sorting algorithms from slowest to fastest 1) Bubble Sort, 2) Selection Sort, 3) Insertion Sort, 4) Quick sort

At 500,000 numbers it took bubble sort a few minutes to sort. At around 700,000 numbers it took over ten minutes (so long that I decided to quit the program). At 500,000 numbers it took selection sort around a minute to sort. At 500,000 numbers it took insertion sort 0.0000053 seconds. At 700,000 numbers it took insertion sort 349 seconds. At 500,000 numbers it took quick sort far less than a millisecond to sort. At 700,000 numbers it took quick sort far less than a millisecond to sort.

From these results we can see that between 500,000 and 700,000 numbers is where insertion sort starts to slow down and quick sort clearly pulls away as the fastest sorting algorithm of the 4. I personally was surprised at how efficient the insertion sort was up to this point

3 Sources

https://www.geeksforgeeks.org/quick-sort/ https://stackoverflow.com/questions/997946/how-to-get-current-time-and-date-in-c