Ivy Truong

2/23/2020

Algorithms HW2 Pt2

Time Analysis of Sort Functions for Different Size Inputs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Input Size (below with times in seconds)** | | | | | |
| **Algorithm** | | 1000 | 10000 | 100000 | 250000 | 500000 | 1000000 |
| Insertion Sort | | 0.0313 | 3.4531 | 603 | 6957.047 | 25,160.0 | 78,136.50 |
| Merge Sort | | 0 | 0.0625 | 1.4375 | 8.2813 | 27.9844 | 147.0781 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Time in Minutes** | | | | | |
| Insertion Sort | | 0.000522 | 0.057552 | 10.05 | 115.9508 | 419.3333 | 1302.275 |
| Merge Sort | | 0 | 0.001042 | 0.023958 | 0.138022 | 0.466407 | 2.451302 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Input Size (below the sizes are the number of comparisons)** | | | | | | |
| **Algorithm** | 1000 | 100000 | 100000 | 250000 | 500000 | 1000000 |
| Insertion Sort | 246373 | 24982755 | 2505219319 | 15644804958 | 32239809616 | 61579219832 |
| Merge Sort | 1982 | 19979 | 199962 | 499975 | 999983 | 1999969 |

**Extra Credit:**

Observation: The Insertion sort takes significantly longer than the Merge sort. This is because insertion sort involves shifting up to n number of elements to the right until the values are no longer greater than the current key that is being sorted. In the insertion sort, the more elements are sorted, the longer it takes to sort the next incoming element to be inserted to the sorted array. In the merge sort, the smallest values of two subarrays are compared to each other and placed into a bigger sorted array, which eventually sorts and combines the left and right subarray. Merge sort is faster because of its divide and conquer approach to a problem versus the insertion sort that deals with the whole problem at once.

Graph: