Cemal Türkoğlu – 150140719

Algorithm Homework 1 Report

Q1.

Full Sort

File	Insertion Sort	Merge Sort
hs-set-10k	3 seconds	0-1 seconds
hs-set-100k	490 seconds.	1 seconds
hs-set-1M	4562 seconds	9 seconds

Filter Sort

File	Insertion Sort	Merge Sort
hs-set-10k	1 seconds	0-1 seconds
hs-set-100k	150 seconds.	1 seconds
hs-set-1M	3089 seconds	6 seconds

Q2.

For very small input size, both algorithms works close to each other. While the input size is increasing, the difference of algorithms' performance also increase. Merge sort is working way faster then insertion sort for the larger inputs. For example for the 1 million size input, it takes more than 1 hour for insertion sort but merge sort does it in 9 seconds. So there is a huge performance difference between these algorithms when it comes to large inputs.

Q3.

Rarity and set parameters does not have so much different values, which is similar to type parameter. So for example when we are sorting by type it makes kind of big groups of this parameter. So there will not be too much comparison. However name parameter has a lot of different values in the data. So it means we need to make a lot of 'if' comparison and it would be slower. I think sorting by rarity or set in filter sort would take similar time with sorting by rarity.

Q4.

Stable sorting is related with same values in the list that will be sorted. When there are equal values in the array their order in the orginal list does not changed. Insertion sort and Merge sort are stable sorting algorithms. For the full sort we are sorting first by class after by cost after by name, however with an unstable algorithm for example after sorting by class, when we are in the next step the order by name may change because of unstablity when we are sorting by other parameters.