**CSE 511**

**Assignment I – Empirical Analysis of Popular Sorting Algorithms**

1. **Comparison of Merge Sort and Quick Sort**

I’ve run Merge Sort and Quick sort for 5 different input sizes of: 100, 1000, 10000, 1000000 unsorted integers for 5 times each and took its average in order to get a precise measurement of Execution Time, Comparison Count and Assignment Count.

Table below shows execution time of merge sort and quick sort with different input sizes.



Graph below shows how execution times of merge sort and quick sort changes.

Table below shows the comparison count of Merge Sort and Quick Sort



Table below shows the assignment count of Merge Sort and Quick Sort



As you can see, even if the worst case complexity of Merge Sort is better then Quick Sort Quick Sort performs better since it has less assignment as uses memory very little since it is an in place algorithm and hard disk usage is the bottleneck when using large datasets with limited RAM.

1. **Comparison of Selection Sort and Bubble Sort**

I’ve run Selection Sort and Bubble sort for 5 different input sizes of: 100, 1000, 10000, 1000000 unsorted integers for 5 times each and took its average in order to get a precise measurement of Execution Time, Comparison Count and Assignment Count.

Table below shows execution time of Bubble sort and Selection sort with different input sizes.



Table below shows the comparison count of Selection Sort and Bubble Sort



Table below shows the assignment count of Selection Sort and Bubble Sort



1. **Comparison of Quick Sort and Bubble Sort**

I’ve run Selection Quick Sort and Bubble sort for 5 different input sizes of: 100, 1000, 10000, 1000000 unsorted integers for 5 times each and took its average in order to get a precise measurement of Execution Time, Comparison Count and Assignment Count.

Table below shows execution time of Bubble sort and Quick sort with different input sizes.



Table below shows the comparison count of Quick Sort and Bubble Sort



Table below shows the assignment count of Quick Sort and Bubble Sort



1. **Comparison of Radix Sort and Quick Sort**

I’ve run Selection Quick Sort and Radix sort for 5 different input sizes of: 100, 1000, 10000, 1000000 unsorted integers for 5 times each and took its average in order to get a precise measurement of Execution Time, Comparison Count and Assignment Count.

Table below shows execution time of Radix sort and Quick sort with different input sizes.



Table below shows the comparison count of Quick Sort and Bubble Sort



Table below shows the assignment count of Quick Sort and Bubble Sort

