Group 7

Placing: 14

final_data1 (larger data set):

- Median: 2022

final_data2 (smaller data set):

- Median: 608

Total: 2630

Algorithm: Modified Bucket Sort

- String sorted into ten buckets (arraylists) based on the % of their first four digits.
- Use timsort on each bucket
- Iterate over the buckets, copying their sorted values over the original toSort array.

Notes:

Using other methods (arrayscopy) of copying over data did not speed results as this requires converting the arraylist buckets into arrays.

Runtime

Runtime is dominated by timsort as we use arraylists rather than linked lists like most implementations of bucket sort use. (Linked lists can potentially result in n^2 performance if all values end up in the same bucket)

Worst case: nlog(n) -> timsort on random data

Average case: nlog(n) -> timsort on mostly random data

Best case: n -> Timsort uses modified insertion sort which can have constant runtime on already sorted data

Our Case: Since there is binomial distribution in the data, it should have a good nlog(n) case as there is some amount of sorted data

Memory usage

- n memory for storing the data into the buckets

- n memory for timsort