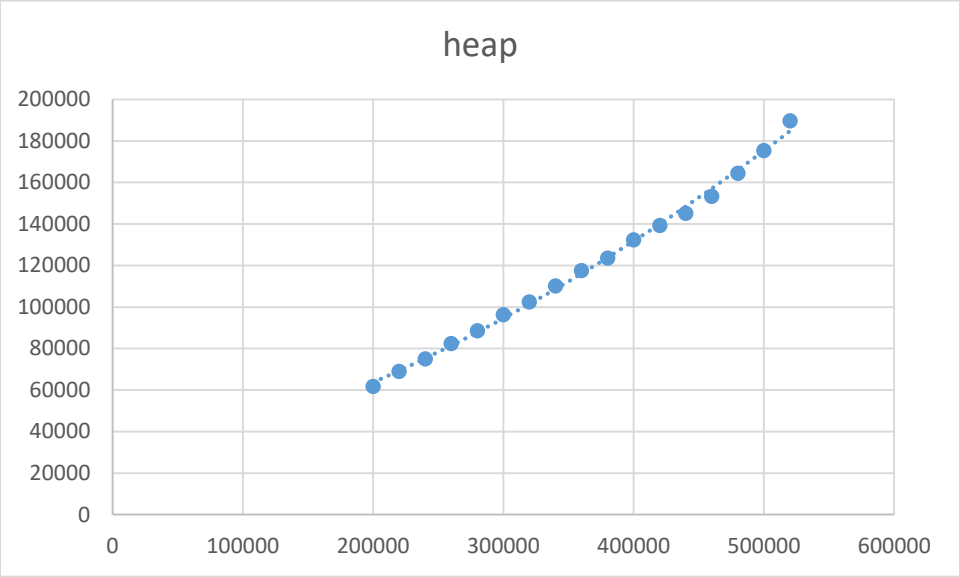
Javier Borja

CSC 17C

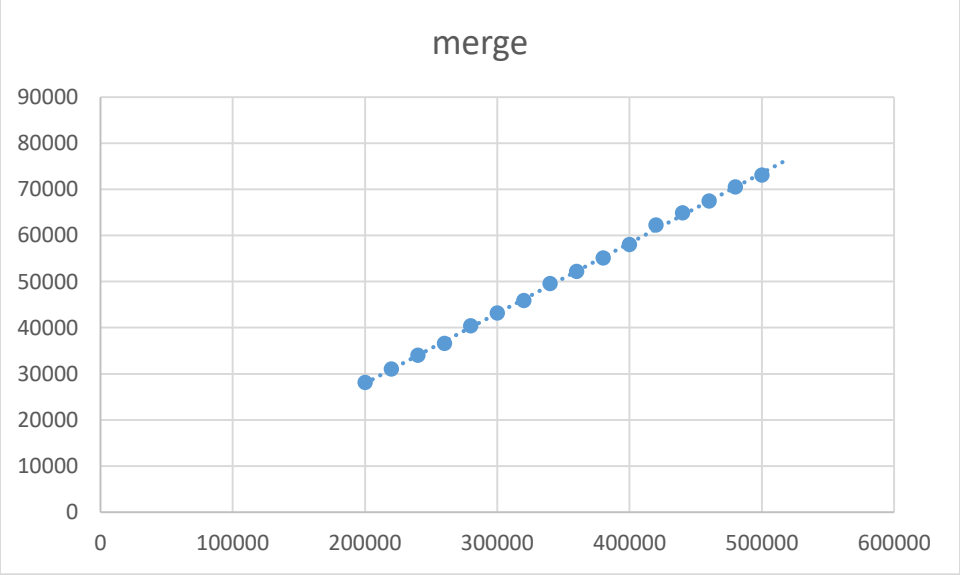
Sorting Comparisons

Proving O( ) characteristics by timing. Included in the files are the code for each type of sort.

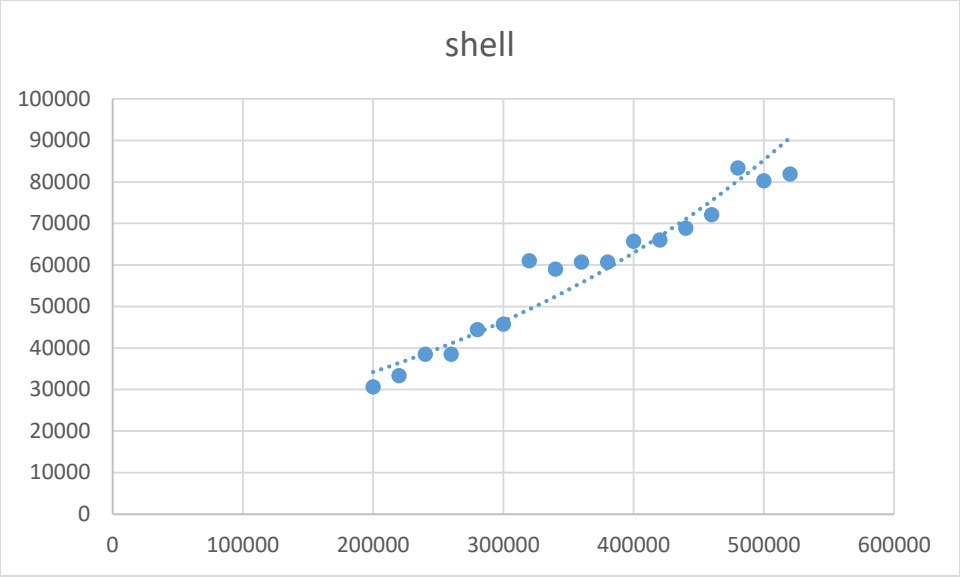
Methodology: X axis is [n] size of randomly filled array of ints. I sorted the array using each different type of sort. Y axis is average time in microseconds (µs).



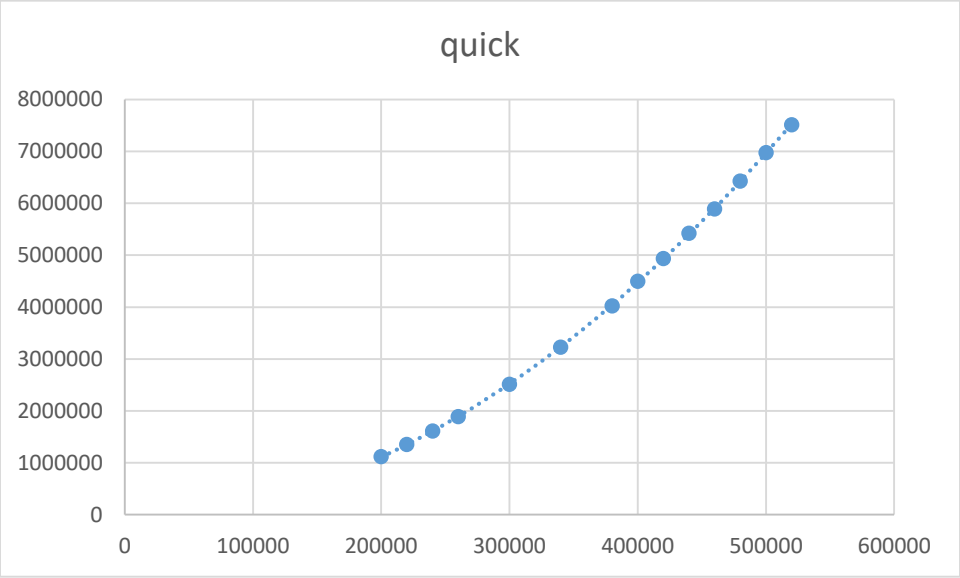
Heapsort: It is almost linear but it is closer to O(n log(n)).



MergeSort: This particular implementation is O(n). Sources claim it is usually O(n log(n)).

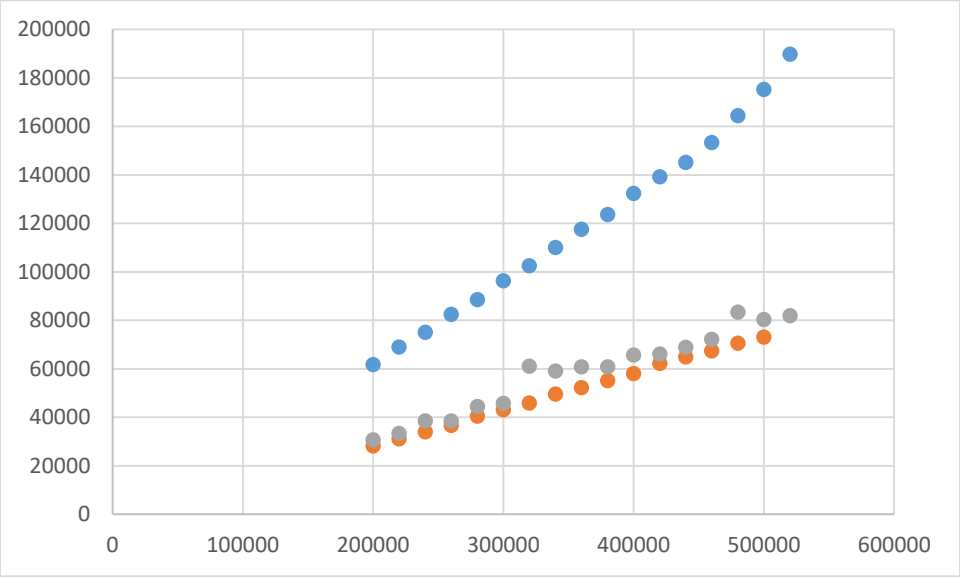


Shellsort: Many sources claim shellsort is O(n(log(n)^2). My timing is similar to it, but it behaves weirdly.



QuickSort: It is O(n^2) and increases rapidly.

Comparison:



Legend: Heap Shell Merge

Quicksort is not pictured because it is O(n^2) and increases too fast in comparison to the others.

Heap is better than most rudimentary sorts that are typically O(n^2), but shell and merge sort are much faster. Merge is the best out of the four in my timing analysis.