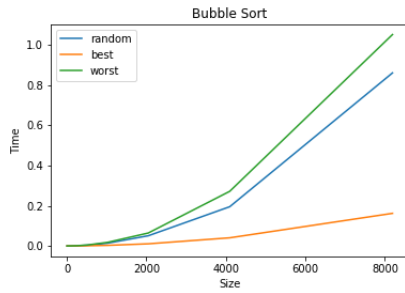# ALGORITHMIC DESIGN - HOMEWORK - SORTING
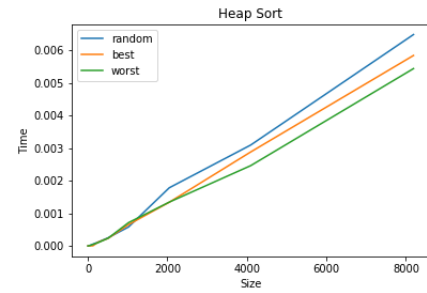
Keerthana C J

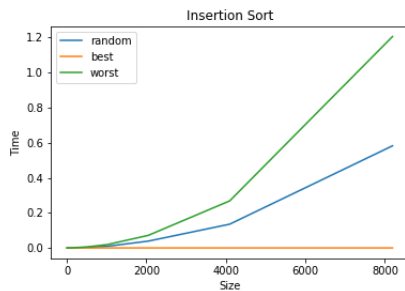## September 2020

1 Sorting

The INSERTION SORT, QUICK SORT, BUBBLE SORT, SELECTION SORT, and HEAP SORT have been implemented and they are compared with their time in the best, worst and a random case. The following figures represent the comparisons.
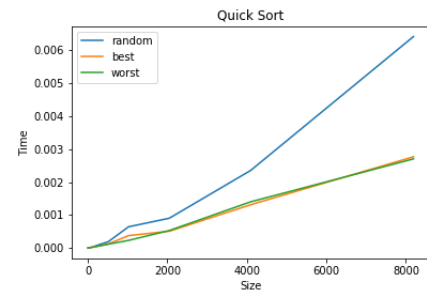

(a) Bubble Sort

(b) Heap Sort

(c) Insertion Sort

(d) Quick Sort

Figure 1: Comparison of the various sorting algorithms

1. HEAP SORT on a array A whose length is n takes time $O(n)$
   Argument: The complexity of building a heap from an array A is $\Theta(n)$.
   The complexity of extracting the minimum value from the heap is $O(logn)$
   In a heap sort, we build the heap from A and use the extract_min for for
   $n-1$ times. Hence, a Heap sort cannot be done in a time of $O(n)$

2. What is the worst case complexity for HEAP SORT?
   Answer: Complexity $= T(build\_heap) + n.T(extract\_min) \leq \Theta(n) + n.O(log_2 n) \leq O(n.log_2 n)$

3. QUICK SORT on an array A whose length is n takes time $O(n^3)$
   Argument: The worst case scenario for a quick sort is when $|S| = 0$ or
   $|G| = 0$, where the complexity is $O(n^2)$. Hence, a quick sort on an array
   taking time $O(n^3)$ is not possible.

4. What is the complexity of Quick Sort?
   Answer: Best case complexity $= \Theta(n.log_2 n)$
   Worst case complexity $= O(n^2)$

5. BUBBLE SORT on an array A whose length is n takes time $\Omega(n)$
   Argument:The best case complexity of Bubble sort is $\Theta(n^2)$. Hence the
   above statement cannot occur.

6. What is the complexity of BUBBLE SORT?
   Answer: Complexity $= \Theta(n^2)$