

Chandler Phillips

Instructor: Maciej Zagrodzki

TA: Jay Luther

CSCI 2270-100, Recitation 101

12/6/20

**Please Note Part A is on a different PDF

Part B Figures:

Figure 1:

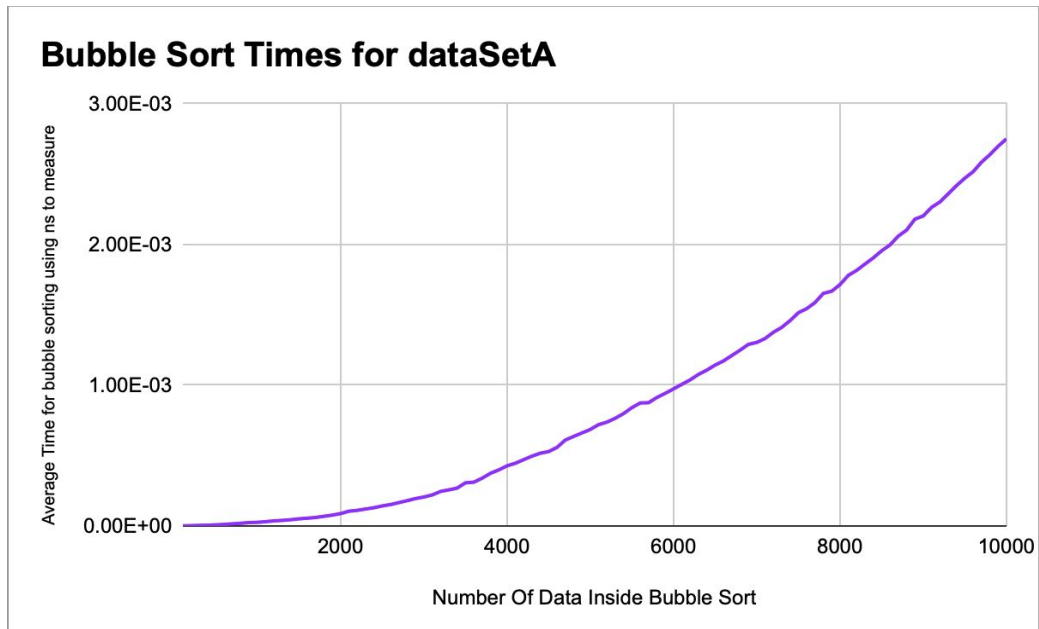
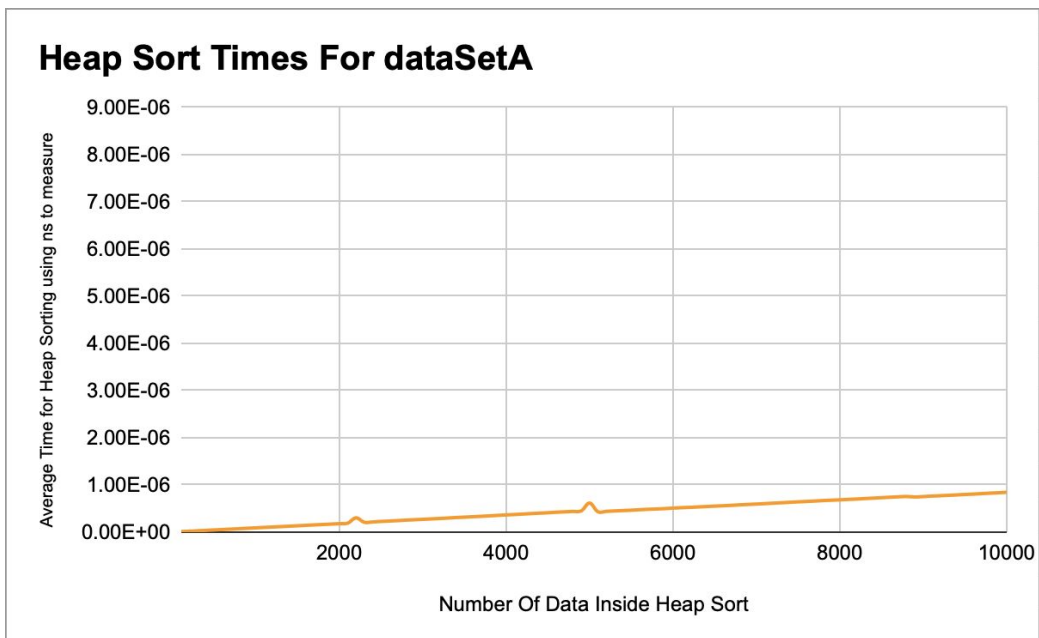


Figure 2:



Part B Analysis

For part B of this project, the Medical Tracker Company wanted us to improve the software application by sorting the patients id's. In order to do this we had to make 2 different sorting methods, a bubble sort and a heap sort, compare the two and see what the better sorting data structure was.

Let's start with the Bubble sort. When using the bubble sort we have to traverse through the whole list swapping elements until they are in the right order, so we would compare elements moving the bigger elements from their position towards the back until we have the data from smallest to largest. Although this method works, it's not too good on time complexity because we have to constantly traverse through the list until we have all the data in the right order, so the bubble sort has a high time complexity, Big O (N^2). Although it does sort the data, it doesn't do it that efficiently as seen in figure 1.

Now let's talk about the heap sort. For the heap sort it stores the data into a tree-like structure, having the smallest values on top and increasing when moving down. Although it is not set up like this, we actually keep the data stored inside of a dynamically allocated array. But since our sorting algorithm uses the tree like structure to sort everything we know that the time complexity of sorting a binary search tree is the same thing, so it has the time complexity of Big O($n \log n$) which is hard to tell based on the graph from the data, but if I was to zoom in on the graph it would be easier to tell.

Comparing the two time complexities we can see that the heap sort is the best option for the data structure to use as our sorting algorithm based on the time complexity. In figure 1 it shows the time complexity for Bubble sorting on our data, and in figure 2 it shows the time complexity for the heap sorting on our data. We can tell based on the range of the time it takes to sort from the graphs that the heap sort is the better choice because the bubble sort time complexity is increasing exponentially.