This project was arguably one of the hardest ones, mainly due to my lack of experience using <chrono>. Other than that, implementing the quicksort algorithm did require me to think a little bit, but once I got a firm grasp on the algorithm and how it works, the whole project came together pretty smoothly.

One of the problems I came across when developing my sorting algorithm was trying to set up the templates correctly. Many of my errors came from incorrect template types, and it caused everything to break. I was lucky to have had experience working with iterators before, and it made working with the vectors a lot easier. I could have used just regular vector bracket indexing, but I learned how to implement sorting algorithms with iterators.

The majority of time I spent on this project was trying to get my proj11_assess.cpp to work. I had a lot of trouble figuring out how std::chrono worked, and I ended up deviating a lot from the basic code template for measuring the time that was posted in the project instructions.

The structure of my proj11_assess was relatively straightforward. I timed each algorithm (100 times*), then found the average time across all trials. I tested the 100- and 10,000- element containers 100 times, however I only tested the 1,000,000 element containers 10 times just to save on program runtime. I set all of the time measurements to be in milliseconds, however I am unsure about the accuracy of these measurements.

For the sorting algorithms, I noticed that the 10,000-element containers were sorted in the same amount of time as (and sometimes even faster than) the 100-element containers. More surprising however is that the search algorithms

produced times that were even smaller than 1 millisecond. I am not too familiar about the speeds of these programs, but if these measurements were to be trusted I am very surprised.

One of the problems I came across was that when I tried to measure a time interval, it always came out to be 0. I eventually got around this by using a steady_clock instead of the system_clock.