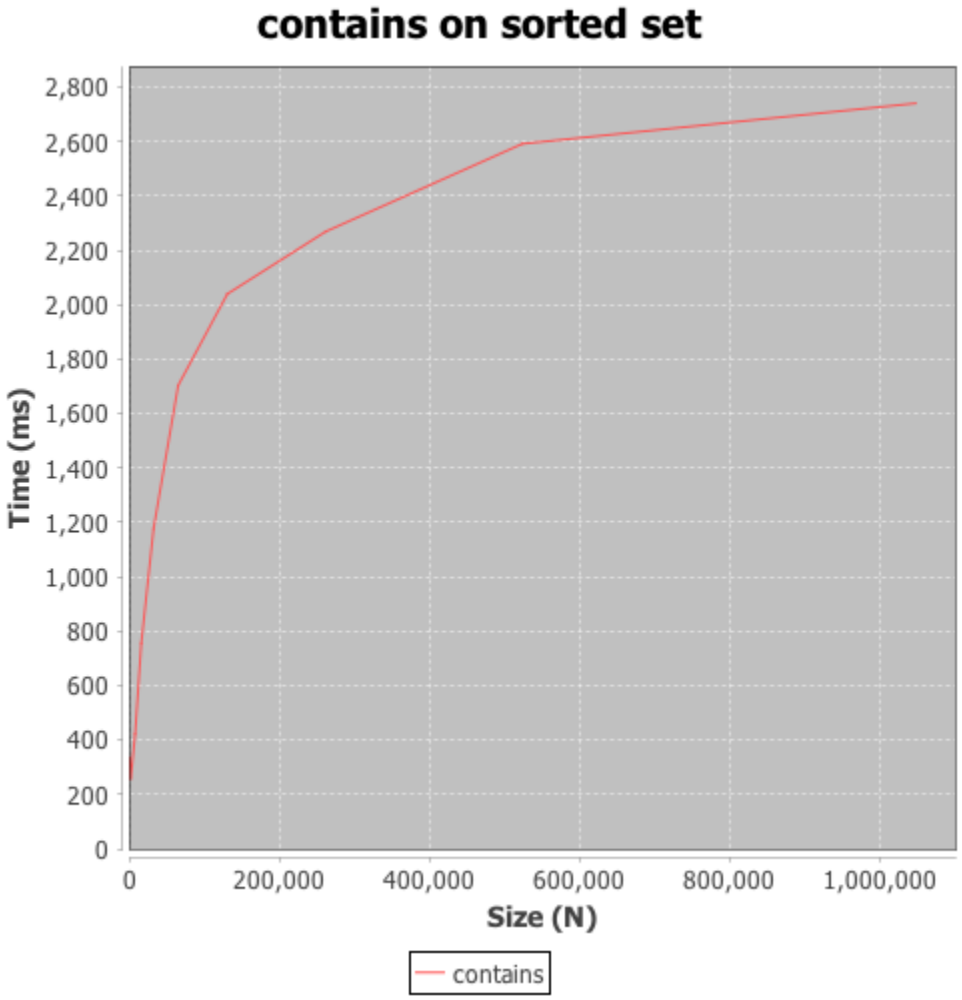
Patrick Ekel partner: Kyle Price

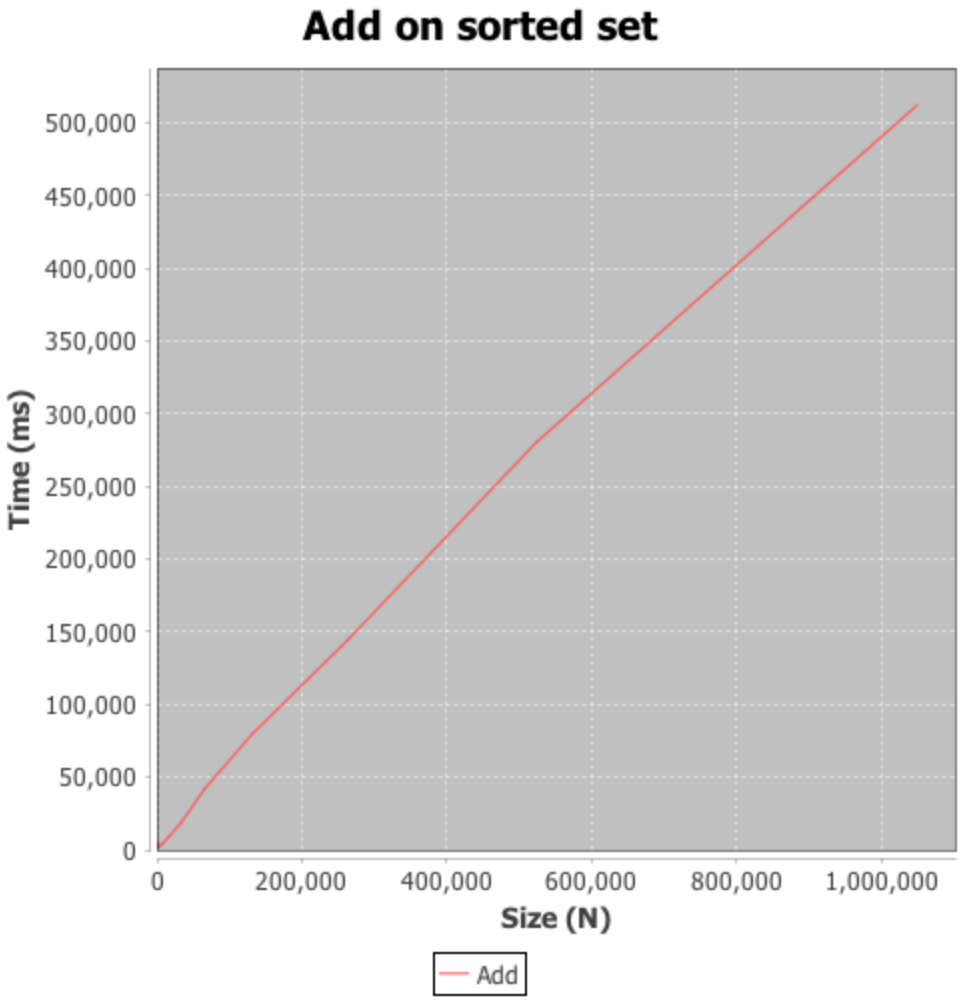
9/14/16 CS 2420

Assignment 3 Analysis

1) My programming partner was Kyle Price. Kyle submitted the source code.
2) We switched roles a few times each time we met to work on the assignment. We met several times throughout the week to work on the assignment, so we ended up switching many times(probably about 15 times). I think this was a good amount of switching, although I wanted to grab the computer from time to time. It is definitely a bit strange working with a partner on a single machine because on one hand I felt like we could have switched more often, but on the other it felt like we were constantly switching. Communicating effectively made it easier to watch another person work on the code, because then I could still share my thoughts.
3) Kyle was a good partner. It is never easy working on large assignments with somebody else and work/school schedules definitely complicate things, but I feel as though we worked well together. I do plan on working with Kyle again.
4) If we had backed the sorted set with a Java List it would have eliminated the need to worry about ensuring the data structure is sized properly, as Java Lists are able to re-size automatically. This would have definitely eliminated a fair amount of work and I think would have allowed us to complete the assignment a bit faster. Lists generally allow duplicate elements so it would still be necessary to search the list before adding a new item to see if it is already in the List to prevent duplicates from being added. List also has an “add” method which allows for a new element to be inserted into the list and the other items in the list are automatically shifted accordingly. Elements in a list can be accessed using the “get” method with the index as the argument, so accessing elements in a list would be very similar. These built in methods would have saved quite a bit time that was spent writing code. But, using a Java List might decrease efficiently in the running of the code. Searching the list for a particular value is likely to be far less efficient as it may rely on a simple linear search. So unless this method were overwritten, using the standard contains method would greatly decrease the efficiently of the program. Overall, it probably would have been more efficient writing the code to use a List, but the code runs more efficiently using an array.
5) I expect the Big-O behavior of the contains method to be Log(N). The contains method searches the ordered array via a binary search. Thus, each time the number of elements in the array doubles, only one more check is needed. We were also taught in class that the Big-O behavior of binary a search is Log(N).



6) As shown in the graph above, the growth rate of the running times does match the big-O behavior I predicted in question 5.



7) The big-O of simply finding the location of where to insert a new element is logN, but to actually insert the new element into the array has a big-O of N as once the index is found, other elements will need to be shifted. This big-O behavior of N is shown in the graph above.
8) I spent about 26 hours on this assignment.