Nicholas Madsen

Prof. Meyer

CS 2420

<center>Assignment 3 Analysis</center>

When you are satisfied that your program is correct, write a brief analysis document. The

analysis document is 30% of your Assignment 3 grade. Ensure that your analysis document

addresses the following.

1. Who is your programming partner? Which of you submitted the source code of your

   program?

   I submitted the assignment and my partner is Michael Cline.

2. How often did you and your programming partner switch roles? Would you have

   preferred to switch less/more often? Why or why not?

   We use Saros to program at the same time. We usually aren't physically programming at the

same moment but we are both watching and typing when either one of us thinks of a way to

solve a problem. I prefer it this way because we both can contribute our thoughts easily and

quickly without having to explain our idea for the other to type it.

3. Evaluate your programming partner. Do you plan to work with this person again?

   I do plan to work with my partner again. We work well together and do a good job coming

   up with solutions together.

4. If you had backed the sorted set with a Java List instead of a basic array, summarize the main points in which your implementation would have differed. Do you expect that using a Java List would have more or less efficient and why? (Consider efficiency both in running time and in program development time.)

I think that this set up works well with a generic array. It is simple and easy to follow. I would think that if we used Java List instead of an array we would be able to take advantage of many of the things already built into the Java List, like its .add function. I think that running time a Java List would be slower because it has an extra layer of things to process to get to the data; whereas with using a basic array you can go straight to what you need.

5. What do you expect the Big-O behavior of MySortedSet's contains method to be and why?

logN because array it is cut in half with each pass through to find where the element is.

6. Plot the running time of MySortedSet's contains method for sets of sizes 100000 to 2000000 by steps of 100000. Use the timing techniques demonstrated in Lab 1. Be sure to choose a large enough value of timesToLoop to get a reasonable average of running times. Include your plot in your analysis document. Does the growth rate of these running times match the Big-oh behavior you predicted in question 5?

I kept getting a very large spikes in the data which I would attribute to either a really bad case scenario or a lag somewhere in the computer running the test (which I think is more likely).

When taking out the very large spikes the graph would be growing with a logN pattern. The time started around 90 nanoseconds and was slowly climbing up toward 118 nanoseconds. I would suspect that like a graph of log the graph of this function would mimic it in time.
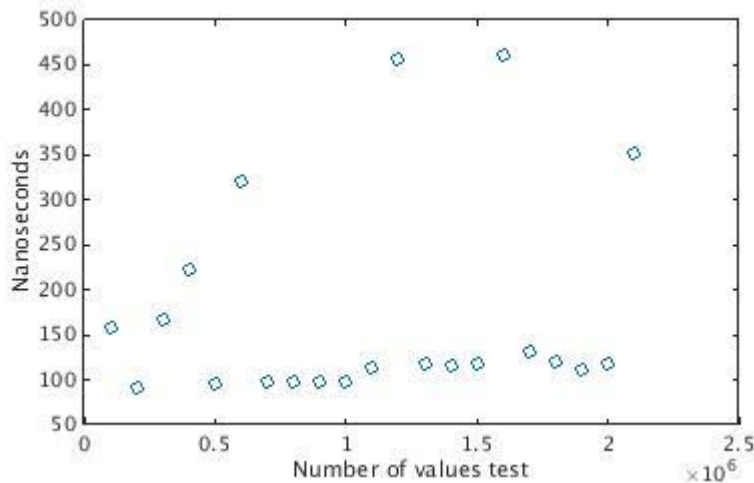
7. Consider your add method. For an element not already contained in the set, how long does it take to locate the correct position at which to insert the element? Create a plot of running times. Pay close attention to the problem size for which you are collecting running times. Beware that if you simply add N items, the size of the sorted set is always changing. A good strategy is to fill a sorted set with N items and time how long it takes to add one additional item. To do this repeatedly (i.e., timesToLoop), remove the item and add it again, being careful not to include the time required to call remove() in your total. In the worst-case, how much time does it take to locate the position to add an element (give your answer using Big-oh)?

NlogN
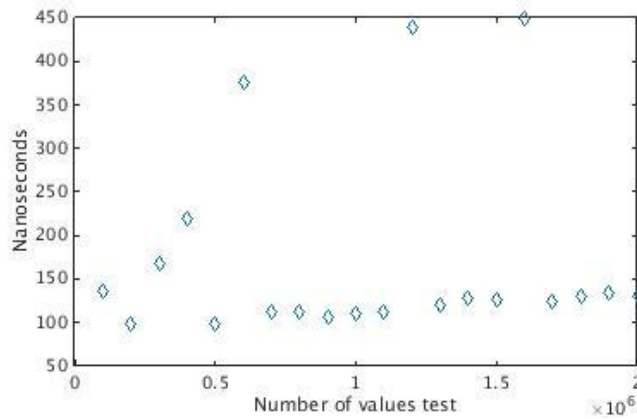
8. How many hours did you spend on this assignment?

About 17 hours

Programming partners are encouraged to collaborate on the answers to these questions. However, each partner must write and submit his/her own solutions.

Upload your document (.pdf only!) to the Assignment 3 page by 11:59pm on February 5.