Brandon Mercado
CS 2420-001
2/5/15

# Analysis Document- Assignment 3

1. Who is your programming partner? Which of you submitted the source code of your program? – Morgan Mercado, Morgan will be the one to submit the zip of our .java files.

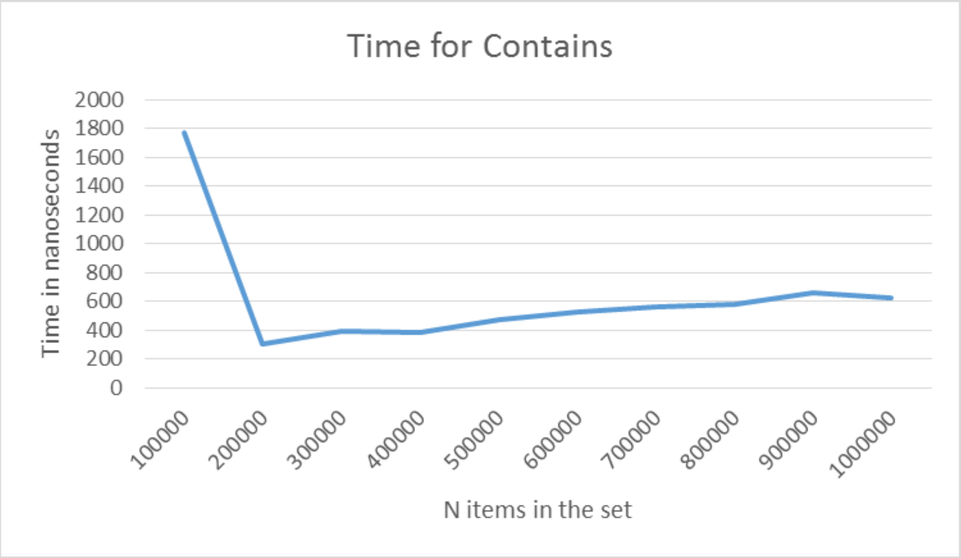
2. How often did you and your programming partner switch roles? Would you have preferred to switch less/more often? Why or why not? – My partner and I switched roles about every 40 min, I think I would prefer to switch more often so that way the perspectives are switched that much more often and better code may come from that.

3. Evaluate your programming partner. Do you plan to work with this person again? – My programming partner is good at java coding and is good at finding problems and coming up with possible solutions. I will probably work with this person again sometime later in the semester.

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.) -

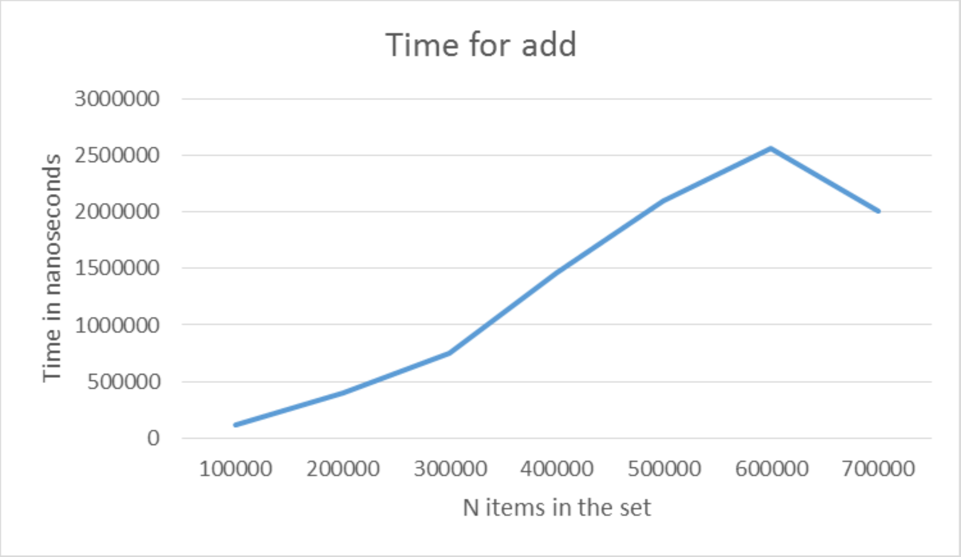
5. What do you expect the Big-O behavior of MySortedSet's contains method to be and why? – I expect the contains method to behave close to logarithmically because we are using a binary search to find where and if MySortedSet contains the object.

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? – The growth rate appears slightly logarithmic just by how the ending appears to be growing slower than it was around 200k-700k, but this is only slightly if at all.



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)?

- O(N log N)



8. How many hours did you spend on this assignment? - approximately 7 hours.