Kayden Thomson
5 February 2015
CS2420
Assignment 3 Analysis

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?

Leonardo Roese, my partner submitted the source code.

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 switched roles about every 20 minutes. I think this was a good amount of time, because I allows us each write a significant amount of code, while being able to switch places frequently enough for both of us to keep our attention on the project.

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

I thought my programming partner was great. I felt that we were at the same level and did an equal amount of work. I plan on working with him again.

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.)

If we would have used Java List we wouldn't need to implement as many methods such as add, remove, etc., because Java Lists already does this for us. But it would have been less efficient because it doesn't use binary search and ours does. We add the items in a sorted manner, but Java List would need to sort the list later.

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

O(logN), we use binary search to find where the item would belong in the set and to see if it's already there.

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?

Graph on next page. Yes, the growth rate matches the run times.

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)?

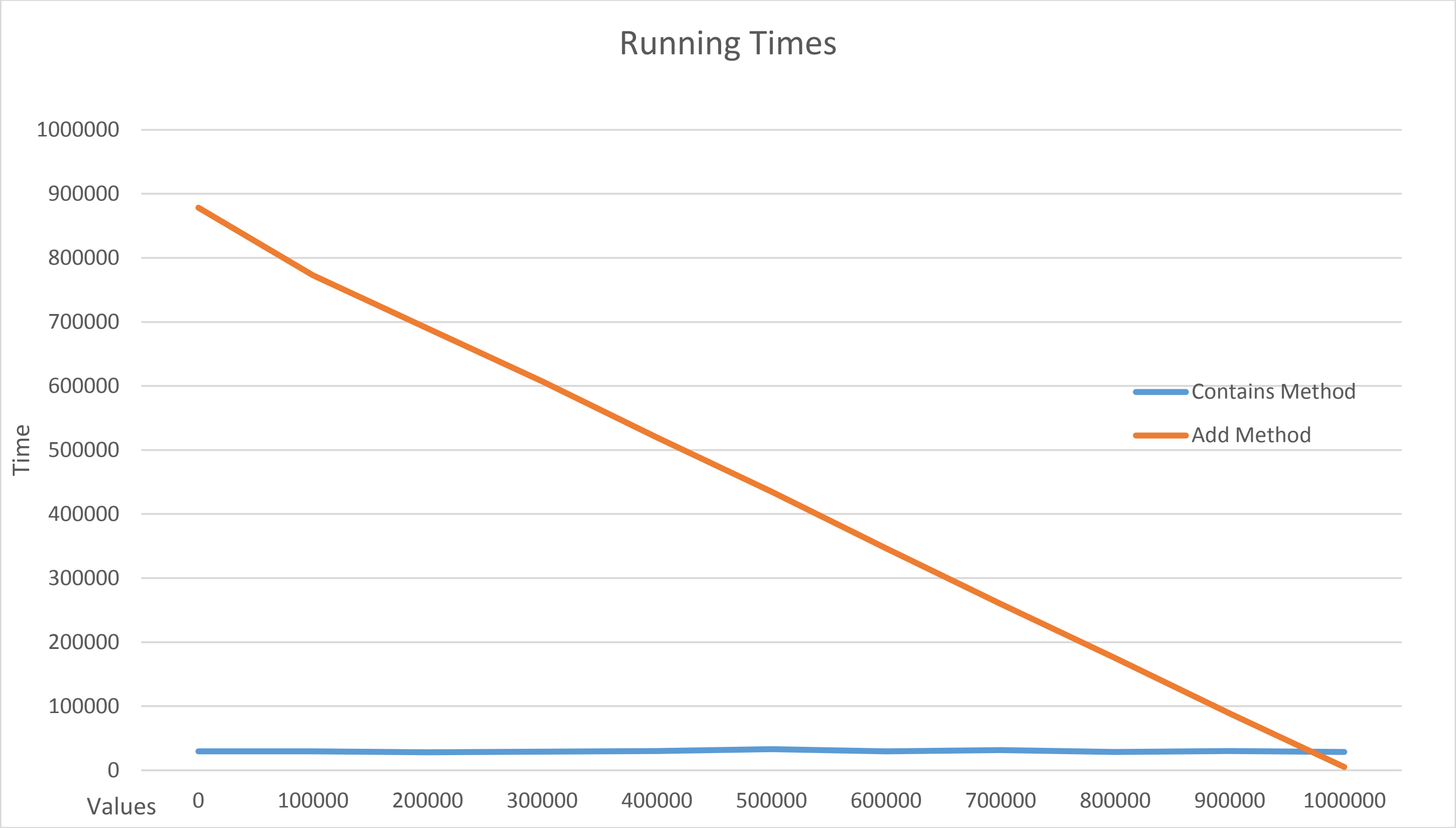
Graph on next page. O(N), due to the iteration. When the object is added, it has to shift all values greater than itself to the right.

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

Around 14 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.


Running Times
Time
Values
Contains Method
Add Method