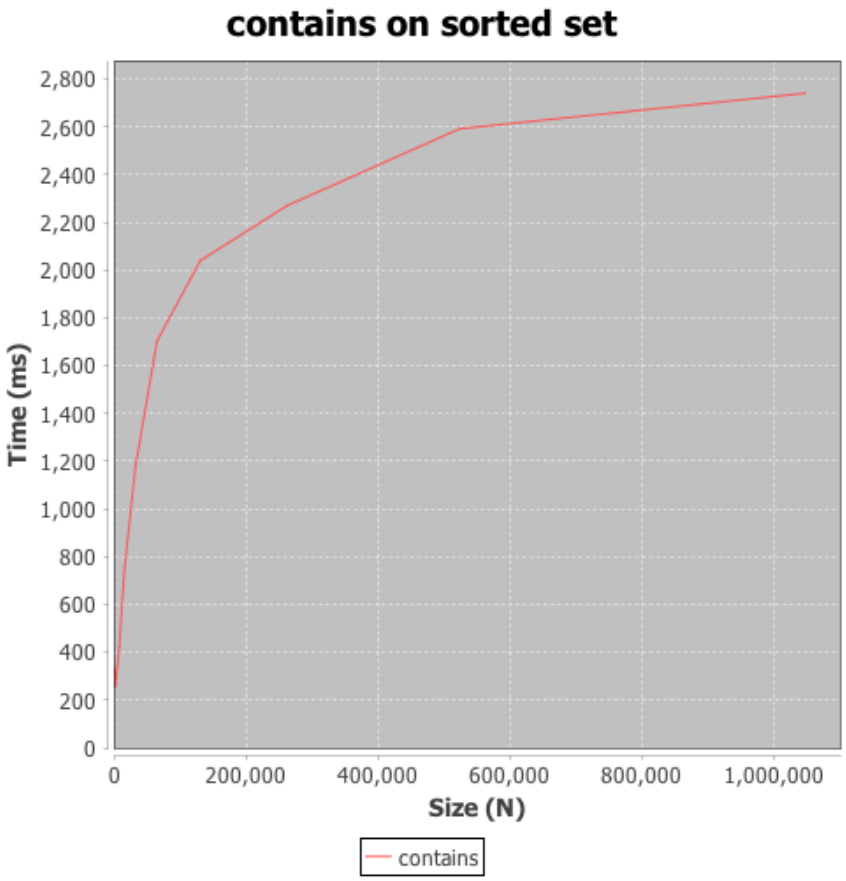
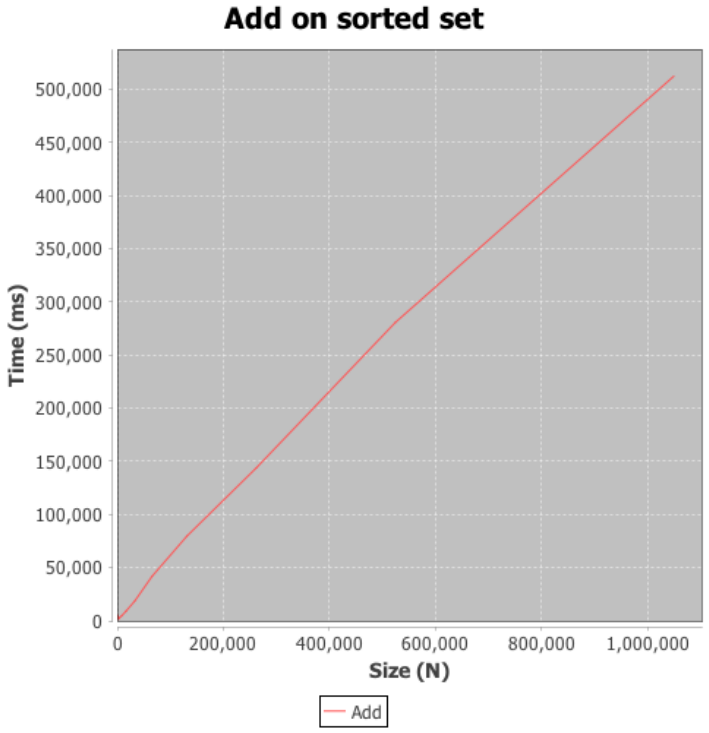
Kyle Price
U101887
Analysis Doc a03

1) **Who is your programming partner? Who submitted the code?** My coding partner is Patrick Ekel. I submitted the source code.
2) **How often did you and you partner switch roles? Would you prefer to switch less/more often? Why or why not?** I would say we'd switch off every hour or so. It varied but that was probably the average. I was pretty happy with how long our work intervals were. It gave enough time to take a break from typing while still giving the driver a good solid experience.
3) **Evaluate your programming partner. Do you plan to work with this person again?** Patrick was a really good partner. We worked really well together. When one of us would struggle, the other swooped in to save the day. And yes, I do plan to work 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.** A lot would have been simplified if that had been the case. We wouldn't have had to write add, remove, or size methods. The array would have grown itself. **Do you expect that using a Java list would have been more or less efficient and why? (Consider efficiency both in running time and in program development time.)** In programming development time, it would have been significantly more efficient. We wouldn't have had to write several of the more complex methods. I have a lot of faith in our program. We tested it thoroughly and the timing is correct. I'm not sure if using a Java List would have made it more efficient at run time but I can't imagine it would have performed any worse.
5) **What do you expect the BIG-O behavior of BinarySearchSet's contains method to be and why?** Binary search works by continually halving the amount of items in the list. This behavior is logarithmic, therefore contains should be O(logN).
6) **Plot the running time of BinarySearchSet's contains method, using the timing techniques demonstrated in Lab 2. Be sure to use a decent iteration count 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-O behavior you predicted in question 5?**



As you can see, the function increases logarithmically. It matches the Big-O behavior I predicated in problem 5.

7) Consider your add method.



As you can see from the graph. Worst case scenario for add is simply O(N). It's a nearly perfect linear function. Contains performs under binary search which is O(logN). Therefore, locating the correct spot to insert the element should happen logarithmically.

8) **How many hours did you spend on this assignment?** Pat and I probably clocked in 20 hours on this assignment. We had major problems with binSearch initially.