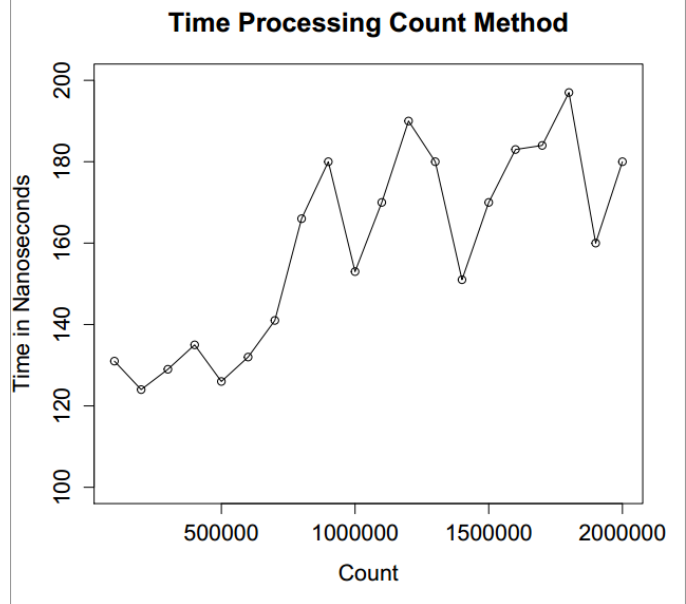
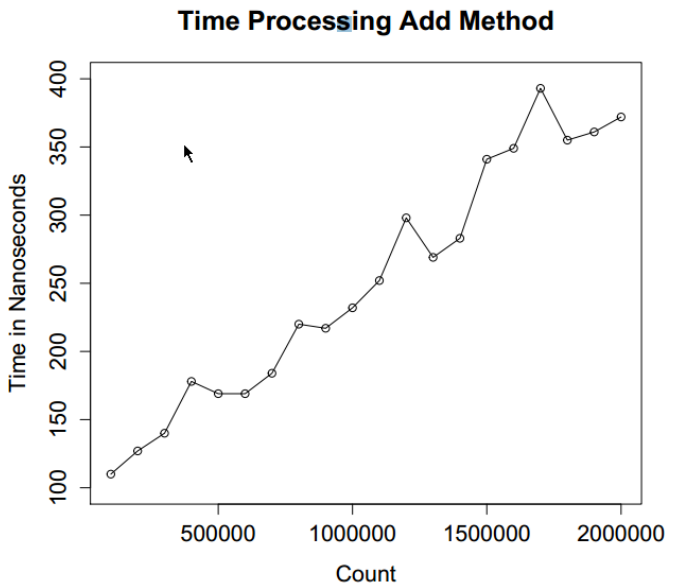


1. My programming partner is Sierra Allred. I am the one who submitted the source code.
2. Not very often, about 4 times. I have no preference. If I am a driver, I get to implement what I'm thinking of and get feedback. If I'm a navigator I get to get my thoughts clear before I Implement them.
3. My programming partner is great. I plan on working with her again.
4. The main differences would be that we would not have to have provided code for the methods but just call the Lists methods. In program development time, the builtin Java list would obviously be more efficient, because it is all written already. In running time, I think ours is more efficient because we use a binary search.
5. I would expect it to be $O(\log(N))$ because it uses a binary search method and the binary search method splits the list in half each time.
6. On the top right is a graph of the running time of our contains method. Yes the graph matches my assumption. It grows pretty slowly. It is consistent.



7. The bottom right figure is a plot of the running time of our add method. I believe the Big-oh complexity of this method is $O(n \log(n))$ because we still have to go through the whole array that exists to check if the object that is being added exists in the array, which is n , and then we have to find where to add it, which is $\log(n)$. Hence $n \cdot \log(n)$



8. We spent 15 hours on this assignment.