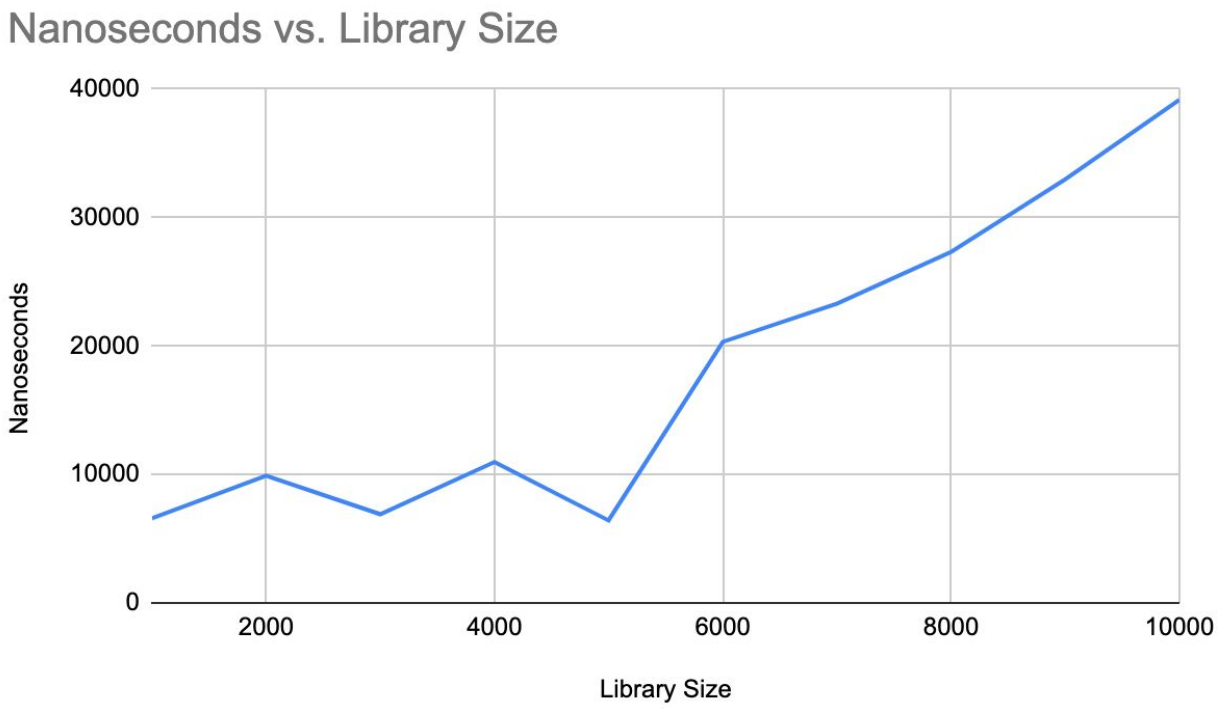
Harry Kim
U1226472
9/9/2020
CS 2420

Analysis Document — Assignment 2

1. I am **not** invoking one of my three exemptions for pair programming due to an extenuating circumstance. My programming partner is Braden Morfin, and I submitted the program to Gradescope.

2. Pair programming experience:
    - I liked being both the driver and the navigator both for different reasons. I enjoyed being the driver because it allowed me to get hands on experience with the actual coding while getting help from the navigator, and I liked being the navigator because it was nice to follow along and learn from my partner while also providing input as we worked on the assignment together.
    - I think my partner and I applied the techniques of pair programming to this assignment very well.
    - Together, my partner and I developed the program for this assignment very efficiently as we were able to finish early which left us time to review and catch any mistakes we might have made.
    - My partner and I spent about two hours each day to complete the assignment for a total of 6 hours.
    - I think I would plan to work with Bradeno again. He's very smart and is a nice person.

3. Comparable is good for classes that sort by an intrinsic, natural order, such as numbers. Comparator is simpler to use but limits how things can be sorted. Comparator is good for classes that sort by what order the programmer wishes. The flexibility Comparator allows comes at a cost of writing more code to sort objects. For sorting something like numbers in order, the Comparable interface would be better because numbers have a natural order. For sorting something by color, the Comparator interface would be better as there is no natural order of colors, and the programmer would have to code for what type of sorting they would like to sort colors. I suppose it could be possible to change the extra features in LibraryGeneric such that Comparable is used instead of Comparator because the methods that use Comparator seem to use objects that can be broken down to numbers such as the orderByTitle() method by ordering the objects lexicographically.



4. As expected, as the library size increases, the average time it takes to execute the program increases.

5. The Big-O behavior of the lookup method is O(N) because the method only loops through the array once to find an instance of a certain object.