Name: Nathaniel Coleman
uID: u0913541

1. This assignment is traditionally done with Pair Programming. Were there times you wish you had a partner to bounce ideas off of? We will ask you to compare your experiences with these first, "single" to assignments to the paired assignments later on in the semester. Please use your answer here for reference later.

The most difficult aspect of this assignment was understanding the code we were given. To have a partner would have possibly made it easier to comprehend. The coding itself however would have been difficult to split between 2 people.

2. Java's built-in classes Comparable and Comparator are both interfaces for doing comparisons among objects. What is the difference in the two interfaces? Give a situation when it is best to use each. Is it possible to change the extra features in LibraryGeneric such that the Comparable interface is used instead of Comparator? Why or why not?

If a class implements comparable, it must define the method compareTo and is generally used only if an object has an obvious way of sorting. If a class implements comparator, it must define the method compare and can be used to define many different ways of sorting an object. For example, to compare two circles would be a good situation to use comparable, since size is an obvious way to sort. On the other hand, comparing fruits would prove less obvious, and would be a good situation to implement comparator. It would be possible to change the extra features in LibraryGeneric such that the Comparable interface is used instead of Comparator because the two interfaces function the same – each compareTo method would be in a different class, which would keep the methods from overriding each other.

3. Comment about the efficiency of your programming time. Did you utilize the time spent on this assignment effectively? How might it be improved?

I think every moment spent trying to think of code to write or understand code given to me is time efficiently spent. Sometimes problem solving time can be minimized by a drawing, but this assignment wasn't the place for that. Troubleshooting always feels like a waste, so perhaps being more willing and able to use the debugger would help that.

4. Reiterate why writing Generic code is important for this course. Phrase your answer in terms of Data Structures and Algorithms.

Generic code is important because data structures can deal with many different kinds of objects. The ability to deal with any kind of object is inherent in generic programming.

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

Bout 7

Upload your document (as a .pdf only!) to the Assignment 2 page by 11:59pm on September 7th.