**CS2261 Complex Matrices Fall 2019**

**Project #5 [100 points] Due: Nov 19th**

---------------------------------------------------------------------------------------------------

In this project, rather than completely implementing your own code, you will be extending some code from our textbook.

This project is designed to let you get some practice using comparators, generics and containers.

**Task 1: [20 points]** Write a Complex class (defined in Exercise 13.17 in our text) to store complex numbers of the form a + bi. You should implement add, subtract, multiply, divie and abs for complex numbers. Your toString method should return a string representation of the number in the form a+bi. It should be *Comparable* and *Cloneable* (deep copy). Provide 3 constructors (Complex(a,b), Complex(a), ie:, just the real part, and Complex(), as well as methods getImag and getReal, as well as setters. If this is all you do in the project, make sure to test it in your main.

**Task 2: [25 points]** For the next part, you should create a ComplexMatrix class by extending the *GenericMatrix* class in section 19.9 of our text (Listing 19.10, GenericMatrix.java) and it should implement *add*, *multiply* and *zero* , toString and any other appropriate methods. Note you will need to modify the GenericMatrix and replace every occurrence of *Number* by *Object* because *Complex* is not a subtype of *Number*. In your main create two matrices (Prompt the user for the size and fill it out randomly) and then display the results after adding them and multiplying them. Make sure that your ComplexMatrix class is *Comparable* of course by implementing the Comparable interface. I want this to compare them by the total magnitude of the real parts.

**Task 3 [35] :** Prompt the user for a number of matrices and their size in rows and columns. Using an ArrayList and a LinkedList, fill both of them up with that number of random matrices. Display them to the screen using your toString. Now I want you to sort them using two different things. Your ArrayList should be sorted by the compareTo method (the one in the Comparable interface). The LinkedList should be sorted using a Comparator, which you need to create. This Comparator object should compare ComplexMatrix objects by the total magnitude of their imaginary parts. Display the ArrayList and then the LinkedList.

**Turnin:**

Turn in your source code in separate files on canvas. Make sure to also turn in UML