CS206 Lab#3: Linked List

In this lab, we will learn about linked list and get ready for assignment 4.

Exercise 1: Design a class `City` that represents a city. It should have instance variables to store the following information. Include appropriate constructor and accessors (getters and setters).

1. name of the city
2. population

Exercise 2: Implement a doubly linked list (DLL) that stores a list of `City`. This need not be a generic DLL. Your DLL should implement the following interface

1. Your DLL should implement the following interface.

```
public interface RabbitLinkedListInterface
{
    int size();
    boolean isEmpty();
    City first();
    City last();
    void addLast(City c);
    void addFirst(City c);
    City removeFirst();
    City removeLast();
    City remove(City r);
}
```

2. Test the above methods to make sure they work properly
3. Override `toString` to print out a list of all stored city names
4. Add the following cities to the list: (Philadelphia, 1,567,442), (New York, 8,550,405), (Houston, 2,296,224), (Chicago, 2,720,546), (Los Angeles, 3, 971,883). Print and make sure they are all there
5. Implement `private void insertBefore(City newCity,Node n)` so that you can insert `newCity` just before some existing node
6. Use `insertBefore` to implement `public void insertSorted(City c)` so that a `City c` is inserted into the list in order by population (lowest first). (You may find it useful to implement a method `private Node firstGreater(City c)` which goes through the existing cities and first the first one with a population greater than the city to be added.
7. Insert all cities using `insertSorted` rather than `addFirst` or `addLast`. Print and make sure they are inserted in the correct order.
8. Modify `insertSorted` to insert in sorted order based on city name

9. Print to show cities in sorted order

When you have completed step 4, send the result to the printer and hand that in.