

Homework #1

In this assignment, you will write a class that implements a contact book entry. For example, my iPhone (and pretty much any smartphone) has a contacts list app that allows you to store information about your friends, colleagues, and businesses. In later assignments, you will have to implement this type of app (as a command line program, of course.) For now, you will just have to implement the building blocks for this app, namely, the Contact class.

Your Contact class should have the following private data:

- The first name of the person
- The last name of the person
- The phone number of the person
- The street address of the person
- The city of the person
- The state of the person

The Contact class should implement the Comparable interface. More on this later.

Of course, you may implement private helper methods if it helps your implementation.

Your class should have the following public methods:

- A constructor that initializes all the fields with information.
- A constructor that initializes only the name and phone number.
- accessor (getter) methods for all of the data members.
- an update method that allows the user to change all information. (They must change all of it).
- An overridden equals() method that can tell if one Contact is the same as another.

It should have the method signature:

```
public boolean equals(Object obj);
```

We will define one Contact as being the same as another contact if the first and last names both match. (Be careful! The parameter may not be a bona fide Contact!)

- An overridden toString() method that creates a printable representation for a Contact object.

It should have the method signature:

```
public String toString();
```

The String should be created in the following form:

<First name> <last name> Phone number: <phone number>
<street address>
<city> , <state>

For example my contact info would look like:

Ari Mermelstein Phone number: (718) 111-1111
2900 Beford Avenue
Brooklyn, NY

- A comparison method that looks like this:

```
public int compareTo(Contact another);
```

We will define this method in the following way:

 If the last name of “another” is lexicographically first, return a positive number.

 If the last name of “another” is lexicographically second, return a negative number.

 If the last names are the same and the first names are also the same, return 0.

 If the last names are the same and the first names are different, use the first names to determine the order.

You must also declare the fact that Contact implements the Comparable interface.

You must also write a main program that tests each of these functions and shows me that you understand how to use the Contact class in a program.

You can choose to write this main program as a stand alone class that sits in the same directory as the Contact class, or you can just make the main program the main program of the Contact class itself. It doesn't matter much to me.

What to submit:

Your Contact.java file
(Your main program class if it is separate).