Lab 4: Linked List

We use a `LinkedList` to keep a list of `Contact` objects (class `Contact` is given in the file) Each contact contains a name, a phone number, and an email address.

Add the following method to the class LinkedList. Do not modify existing methods. Throws clause(s)/try-catch can be added if necessary. JUnit test is provided for each method.

`public int size()` returns the number of elements in the list, excluding the header node.

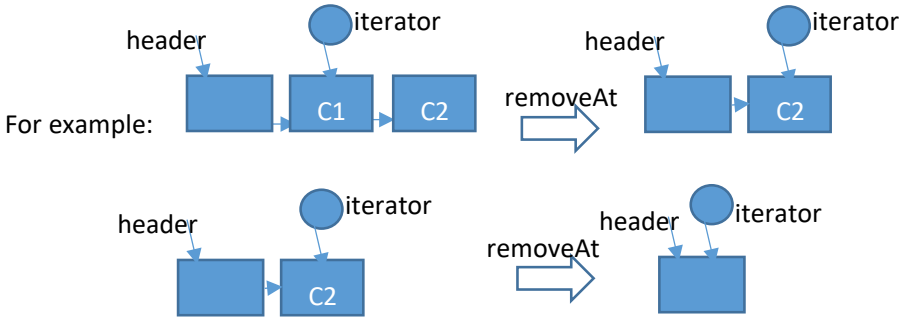
`public void printList()` prints all contacts stored in the list. One contact per line.

`public Iterator findPosition(String name)` returns an iterator pointing to a contact that stores a given name. Otherwise, return null.

`public void add(Object c)` adds a new contact to the list. The contact must be added in such a way that each contact is sorted by name, in alphabetical order. Assume that all existing data in the list are sorted in this way already.

`public Iterator removeAt(Iterator i)` removes the contact pointed by the iterator, i, and then returns an iterator pointing to the next contact. If the removed contact is the last one, return the iterator pointing to the first contact (if there is no first contact, make the iterator point to the header). If it is marking an illegal position that cannot be removed, just return null.

Example of removeAt are shown below:



How to submit:

Make a jar file containing all your source code and submit it through Courseville. Name your file **YourID_Lab04.jar** where YourID is your student ID.