CECS 277 LAB LINKED LIST ITERATOR

OBJECTIVE: Build an Iterator for a collection, and learn how the collection class and the iterator have to collaborate.

INTRODUCTION: Please remember the coding standards here.

The demo code that we went over in lecture is here. In this lab, I want for you to start with that code, and augment it to include a new feature that allows the user to **delete** a node from the linked list at the point in the linked list that the iterator points.

- Remember that if the iterator presently points to the first element in the linked list, you will need to change the head of the linked list to point to the **old second** element in the linked list.
  - If the first element of the linked list is the **only** element in the linked list, then the head of the linked list will end up being null, which always means that the linked list is empty.
- Remember that the iterator has the trail stack to work with.

PROCEDURE:

1. Add a delete method to the Iterator interface.
2. Add an implementation of delete to the LLIter concrete implementation class.

WHAT TO TURN IN:

- Your new and improved version of Iterator.java
- Your new and improved version of LinkedList.java
- Your new and improved version of LLIter.java
- Your test program. Call it LinkedListRunner.java
- Your console output.

SAMPLE OUTPUT:

```
Right after appending some words.
Listing in the forward direction.
Next value: Harry
Next value: loves
Next value: Sally
Next value: very
Next value: much.
And now in reverse.
Previous value: much.
Previous value: very
Previous value: Sally
Previous value: loves
Previous value: Harry
After two inserts:
Listing in the forward direction.
Next value: I
Next value: think
Next value: Harry
Next value: loves
Next value: Sally
Next value: very
Next value: much.
And now in reverse.
Previous value: much.
Previous value: very
Previous value: Sally
Previous value: loves
Previous value: Harry
Previous value: think
Previous value: I
Deleting after 2nd element
What's left:
Listing in the forward direction.
Next value: I
Next value: think
And now in reverse.
Previous value: think
Previous value: I
Deleting the first element
Listing in the forward direction.
Next value: think
And now in reverse.
Previous value: think
Completed satisfactorily.
```