CECS 277 – Lab 9 – List Iterator

List of Words

Create a program that reads in a file "words.txt" and stores them in a LinkedList. Allow the user to choose to add new words to the list, remove words from the list, to display the list, or to quit the program.

Create the following functions:

1. readFile – construct the LinkedList and a ListIterator, read in the file and add each word to the LinkedList. Call moveIter to move the iterator to the correct position to add each word, so the list is always in sorted order. Return the list.
2. moveIter – pass in the ListIterator and the new word. Move the iterator backward or forward from its current position to find the correct location to place the word so the list always stays in sorted order.
3. addWord – pass in the list, create a ListIterator, and call moveIter to move the iterator to the correct position, then add the word to the list.
4. removeWord – pass in the list, then prompt the user to enter the word they would like to remove. Create a ListIterator that starts at the beginning of the list, then move it forward to the correct position, if the word is found, then remove it, if it isn't, then display a message to tell the user that it was not found.
5. printForward – pass in the list, create a ListIterator that starts at the beginning of the list, iterate through the list to display the contents.
6. printReversed – pass in the list, create a ListIterator that starts at the end of the list, iterate through the list to display the contents in reverse order.
7. menu – display the menu and get the user's choice. Return the input.

Menu – Repeat the menu until the user quits.

```
1. Display Words
2. Display Reversed Words
3. Add Word
4. Remove Word
5. Quit
```

Notes:

1. Read in the file when the program starts (happens only once).
2. Use an iterator to move through the list to add the words, one at a time, in sorted order. DO NOT add all the items and then just sort the list. The point of the assignment is to practice using iterators.
3. The iterator should be moving back and forth through the list. You should not reset the iterator to the front every time a new word is added from the file.
4. Check the user's input for the menu. You do not need to error check the strings.

Starting Out -

1. Start with a stripped down version of the readFile function. For now, just read in the file and the words directly to the end of the LinkedList.
2. Create the printForward and printReversed functions.
3. Create the menu function and the main.
4. Create the addWord function, for now, add new words to the end of the list.
5. Create the removeWord function, for now, you can have it iterate through the entire list to search for the word to remove (since it isn't sorted yet).
6. Create the moveIter function. Compare the new word to the word at the iterator's position, then move the iterator to the correct position. Modify readFile/addWord to call moveIter and add the word at that position when moveIter returns.
7. Modify removeWord. Now that it is sorted, make it so that it only iterates up to the correct position of where the word would be, rather than the entire list.