Homework Assignment #6 due Tuesday Fab 13, 2018

Collections

Problem 1:

Build a class called LinkedListRunner with a main method that instantiates a LinkedList<String>. Add the following strings to the linked list:

aaa

bbb

ccc

ddd

eee

fff

ggg

hhh

iii

Build a ListIterator<String> and use it to walk sequentially through the linked list using hasNext and next, printing each string that is encountered. When you have printed all the strings in the list, use the hasPrevious and previous methods to walk backwards through the list. Along the way, examine each string and remove all the strings that begin with a vowel. When you arrive at the beginning of the list, use hasNext and next to go forward again, printing out each string that remains in the linked list.

Problem 2:

Use the class below as the driver for this lab. The class called `LinkedListRunner` with a main method that instantiates a `LinkedList<String>`. Add the following strings to the linked list:

```
import java.util.LinkedList;
import java.util.Stack;
public class StackRunner
{
   public static void main(String[] args)
   {
       LinkedList<String> myLinkedList1 = new LinkedList<String>();
       myLinkedList1.add("aaa");
```

```
        myLinkedList1.add("bbb");
        myLinkedList1.add("ccc");
        myLinkedList1.add("ddd");
        myLinkedList1.add("eee");
        // print the first linked list
        System.out.println("My Linked List 1:");

        // … your code goes here

        Stack<String> myStack = new Stack<String>();
        //Iterate through elements in the linked list (don't remove them), but
        // push all the elements onto the stack

        // … your code goes here

        //Pop all the stack elements off the stack and add them to
        // a new linked list
        // … your code goes here

        //print the second linked list
        System.out.println("My LinkedList 2:");

        // … your code goes here
    }

}
```

How the second list is ordered compared to the first? Why?