CS 0445 Spring 2024
In Class Exercise 3

Introduction:

In Assignment 2 you will be implementing two classes that utilize a circular, doubly linked list. The "base" class for these classes is the class A2LList<T>, which is my conversion of the author's LList<T> class to make it a circular, doubly-linked list. Before completing this exercise be sure to carefully read over Assignment 2 and the code in the file A2LList.java. I have comments there that should help you to better understand the implementation of the circular, doubly linked list.

In this exercise you will complete one of the methods required for the LinkedListPlus<T> class, which is a subclass of A2LList<T>. The method you will complete is the leftShift(num) method, which shifts the leftmost num elements out of the list. The header for the method is shown below:

```
public void leftShift(int num)
```

Consider, for example, a list, L, containing (in order, starting at location 1):

```
Worf   Picard   Data   Crusher   Troi   Riker
```

The call L.leftShift(3);

would result in the list

```
Crusher   Troi   Riker
```

If the argument, num, is greater than or equal to the length of the list, the result should be an empty list. If the argument, num, is <= 0, the list should not be changed.

Note that since the underlying linked list is a circular, doubly-linked list, you must be careful to keep everything connected correctly. Make sure also to handle the special cases mentioned above.

To do this exercise you will need the following files:

A2LList.java (parent class)
LinkedListPlus.java (starter code for LinkedListPlus class)
InClass3.java (main program to test your method)

These files are also available on Canvas.

During lecture we will run the program to see the output.