Double-Linked List

CS 221 Programming Assignment

*“The way to succeed is to double your error rate.”*– Thomas J. Watson

# Objectives

* Create a doubly linked list implementation of the IndexedUnsortedList interface
* Create a fully functional iterator that implements the ListIterator interface
* Use a test suite to ensure correct functionality of a class

# Tasks

For this programming assignment, you will:

* Create a class called IUDoubleLinkedList that implements the IndexedUnsortedList interface. The underlying data structure of your class should use a node-based, doubly linked list.
* Implement a list iterator within your class that supports all [ListIterator](http://docs.oracle.com/javase/8/docs/api/java/util/ListIterator.html) methods, including the add(), remove(), and set() methods.
* Update your ListTester class:
  + Implement list tests for all of the ListIterator methods. [Example tests](https://raw.githubusercontent.com/BoiseState/CS221-Public/master/assignments/DataStructures/files/ListTester_ListIteratorTests.txt) for empty and single-element lists are given to get you started.
  + Implement [change scenarios](https://raw.githubusercontent.com/BoiseState/CS221-Public/master/assignments/DataStructures/files/ListTestingScenarios.txt) resulting from modifications to the list from ListIterator's add(), remove(), and set() methods.
  + Add [ListIterator concurrency tests](https://raw.githubusercontent.com/BoiseState/CS221-Public/master/assignments/DataStructures/files/ListTester_ListIteratorConcurrencyTests.txt). Replace the test\_ListIteratorConcurrency() method with the given method.
  + Modify your test class to run with the IUDoubleLinkedList class. You should comment out any mention of the GoodList, BadList, IUArrayList, and IUSingleLinkedList classes.
* Report on your testing procedures and results in your README file using [this format](https://drive.google.com/file/d/1lkysQn5_6OruZ3de0hSKqixus4MiF0AN/view?usp=sharing).
* Document all the classes and methods in your code with [Javadoc comments](https://docs.google.com/document/d/1ph7t-TelDTuX64KUjBXxY0wxIfe16HvA0lHCct8jIwQ/edit?usp=sharing). Include in-code comments as necessary.

# Files

This project builds on the ListTester begun in previous assignments and uses the same [IndexedUnsortedList.java](https://raw.githubusercontent.com/BoiseState/CS221-Public/master/assignments/DataStructures/files/IndexedUnsortedList.java).

# README

Your plain text README should follow the formatting and content guidelines [given here](https://drive.google.com/file/d/1lkysQn5_6OruZ3de0hSKqixus4MiF0AN/view?usp=sharing), with special attention given to the following:

* Design: describe the roles of and relationships between the interface and classes that make up this program. You had choices about how to implement many methods; why did you choose what you chose?
* Testing: a detailed description of your testing procedure. You should describe how you tested your class and what bugs you found and fixed by doing so.
* Discussion: a discussion of your experience writing this assignment. This could be in the form of a journal.

# Grading

Points will be awarded according to the following breakdown:

| **Tasks** | **Points** |
| --- | --- |
| Documentation - README, Javadocs, comments | 20 |
| ListTester - complete, including ListIterator functionality | 20 |
| IUDoubleLinkedList and ListIterator Functionality | 50 |
| Quality - code formatting, naming conventions, encapsulation, etc. | 10 |

# Required Files

Please submit the following files:

* IUDoubleLinkedList.java
* ListTester.java (updated)
* README
* Any other files required to compile / run your ListTester class using the IUDoubleLinkedList class.

# Submission

Submit all files from the same directory. Do not include any unnecessary files. Use the submission command given on your section's class web page from the directory containing your files.