Overview

This assignment is an **optional** opportunity for extra credit. Within this assignment, you will create a JUnit test suite for **HW01: ArrayList**.

JUnits

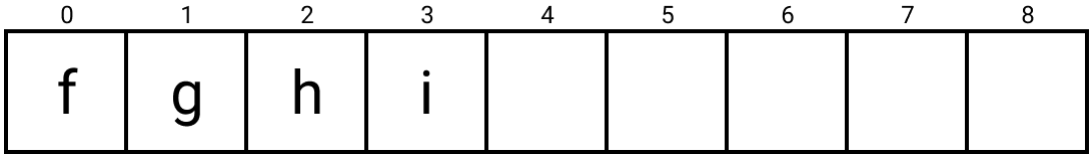
JUnit is a unit testing framework for Java. We use it for the autograder in this course, and it is used in industry. The goal of this assignment is to give you experience writing your own JUnit tests. These skills will be useful for this course and beyond.

Boilerplate

Each JUnit file requires boilerplate code to initialize and set-up your tests. For example, you could create an empty data structure at the start of each unit test to preserve the independence of tests. We have provided `ArrayListExtraCreditTest.java` to you, which includes the necessary boilerplate code.

Initialization

The first step in creating a unit test is to initialize the data structure to your intended "before" state. For example, you may wish to start with this `ArrayList`:



To create this "before" ArrayList, execute commands to add data to the currently empty `ArrayList` at the beginning of the test method:

```
list.addToBack("f");
list.addToBack("g");
list.addToBack("h");
list.addToBack("i");
```

Execution

The next step in a unit test is executing the functionality you are trying to test. For example, if you want to test `addToFront()`, call the method with some data:

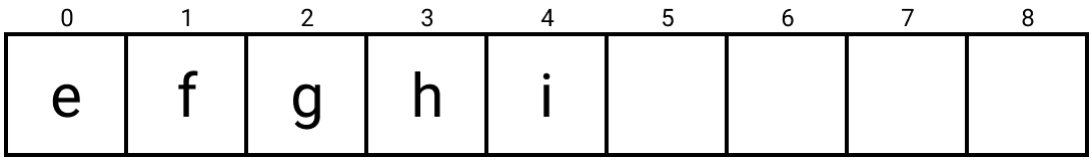
```
list.addToFront("e");
```

Assertion

The final step of a unit test is Assertion, which verifies that the state of the data structure matches what is expected after execution. For this, two main methods are used:

- `assertEquals` - Verifies that two variables are equal to each other using `.equals()`. If the two variables are not equal to each other, the test will fail.
- `assertArrayEquals` - Verifies that two arrays are equal to each other in length and contents. If the two arrays are not equal to each other, the test will fail.

Continuing with our previous example, our `ArrayList` should look like this after execution:



To verify this, we will use `assertArrayEquals()` as follows:

```
assertArrayEquals(new Object[]{"e", "f", "g", "h", "i", null, null, null, null},
   list.getBackingArray());
```

You could also create a separate Array object with the expected data and use that in the assertion:

```
String[] expected = {"e", "f", "g", "h", "i", null, null, null, null};
assertArrayEquals(expected, list.getBackingArray());
```

Additional Resources

If you are looking for more resources on how to write JUnit tests, they can be found on Canvas under `Files > Coding Set > JUnits`. Additionally, the `ArrayListStudentTest.java` file provided with **HW01: ArrayList** is a great place to start if you are looking for more concrete examples of tests.

Deliverables

For this assignment, you should submit the following:

- Your JUnit files and any accompanying helper files.

The files are not required to be named anything specific, but there are a few things that should be avoided to avoid breaking the autograder:

- Do not add `package` statements to your files.
- Do not include `ArrayList.java`. The autograder uses its own implementation based on HW01.
- All code must compile under **JDK 17**.

Grading

For grading, we will make a series of incorrect changes to our implementation of an `ArrayList` and run them against your JUnit tests. Points will be awarded for each change that your tests catch.

The score for your submission will be visible as soon as the autograder completes.

Collaboration Policy

Every student is expected to read, understand and abide by the Georgia Tech Academic Honor Code.

When working on homework assignments, you **may not** directly copy code from any source (other than your own past submissions). You are welcome to collaborate with peers and consult external resources, but you **must** personally write all the code that you submit. **You must list, at the top of each file in your submission, every student with whom you collaborated and every resource you consulted while completing the assignment**.

As this assignment overlaps directly with sharing JUnits for **HW01: ArrayList**, there are some JUnits created by your peers on Piazza. However, you may not copy from anyone else's shared JUnits or the provided `ArrayListStudentTest.java`; all submitted work must be solely your own. Additionally, while AI generated JUnits are allowed on the Piazza post, the use of AI is not permitted for this assignment.