CS-115 Lab Class 5 — 24/02/2020

☐ Task 5.1

Today's lab will help directly with the assignment.

Download the following (possibly familiar) code.

Link.java

In this lab, you will implement a class called `LinkedList.java`. The class should use a generic (`LinkedList<T>`) to constrain the types that can be put into the linked list. This class should have the following attributes and methods that implement a linked list:

- a `head` and `tail` reference to the beginning and end of the linked list
- a constructor that initialises an empty linked list
- `isEmpty()` that returns `true` if there are no elements in the list and false otherwise.

Your linked list should also have the following methods:

- `T elementAt (i)`: Returns the element at position `i`. The first position in the list is 0. If the ith position does not exist, a `NoSuchElementException` is thrown.
- `void insertAt (int i, T e)`: Inserts element e into the ith position of the linked list. If the ith position does not exist, a `NoSuchElementException` is thrown. Note for an empty list, you can insert at position 0. For a list of one element, you can insert at position 1. etc.

Implement this linked list inside `LinkedList.java`. Create a `Main.java` class with a main method to test your linked list. Create link lists of type `String` and `Integer`. Print out the elements to the screen to demonstrate that your methods work.

☐ Task 5.2

In your main method, write the following java code:

```
LinkedList<String> strs = new LinkedList<String> ();
LinkedList<Integer> ints = new LinkedList<Integer> ();

ints.insertAt (0, ``Hi there.'');
```

Please explain why this code does not compile.

☐ Challenge Task 5.3

It doesn't make too much sense to add elements without removing them. Create a method that removes an item from the list.

- `void removeAt (int i)`: Removes the element at the ith position of the linked list. If the ith position does not exist, a `NoSuchElementException` exception is thrown.