CSF202 – Lab Class 3 – Wednesday 07/10/2020

☐ Task 3.1

Model a Queue and a Stack in a UML class Diagram.

You should use three classes: Queue, Stack and Element.

A Queue should support:

- enqueue: given an element, we enqueue it.
- dequeue: remove the front element of the queue (and return it).
- length: get the length of the queue.

A Stack should support:

- push: given an element, we push it on the stack.
- top: inspect the top most element.
- pop: remove the top most element.
- length: get the length of the queue.

(An alternative model of stack, which we do not model, would combine the pop and top operations).

As we are dealing with class diagrams, which describe what classes do and not the how, you do not need to attempt to model the behaviour of operations. To model the behaviour you would use one (or more) of the other UML diagram types, e.g., State diagrams.

For those interested there is actually specific UML notation for template classes / template parameters that can be used to model generics in Java. We will not be using these in this module.

☐ Challenge Task 3.2

Add an **Iterable** interface to your class diagram. This interface should have no attributes and one method:

- **iterator()** which returns an Iterator<T>

Show that your Queue class makes use of this new Iterable interface.