23/2/2017

Niall Morris
1005926m
Advanced Programming
Coursework 2017

The UML diagram in **Fig.1** (overleaf), when paired with the attached Java implementation of said design, constitutes my submission for the Advanced Programming coursework (as detailed in the coursework specification document *APCourseworkTask2017.pdf*).

The program's functionality should accurately match the requirements as specified in the coursework specification document, as well the functionality as discussed in classes.

On launch, the program prompts the user for input of the numbers '1' or '2', in order to choose what type of world to create (either a bounded 'Edge World', or world that allows creatures to reproduce across the edge boundaries by 'wrapping around' ('Wrap World')). Evidence of this can be seen in the screenshot in **Fig. 2**.

The size of the world, and interval at which it should be printed to the console, have been determined by instance variables in the `main` class (`TestWorld.java`). These have been defaulted to both an x-size and y-size of 20; and for the world to be printed to the console every 500ms.

Each species of creature is represented by a different `String` output; this can be set as an instance variable in each of the classes that extend from `Creature.java` (that is, `Species1.java` and `Species2.java`). I have defaulted these to two different emoji characters ('🐊' and '🐙'), in favour of the suggested '1' and '2', as it makes it slightly clearer to distinguish between the different species. The program running with these values can be seen in **Fig. 3**.

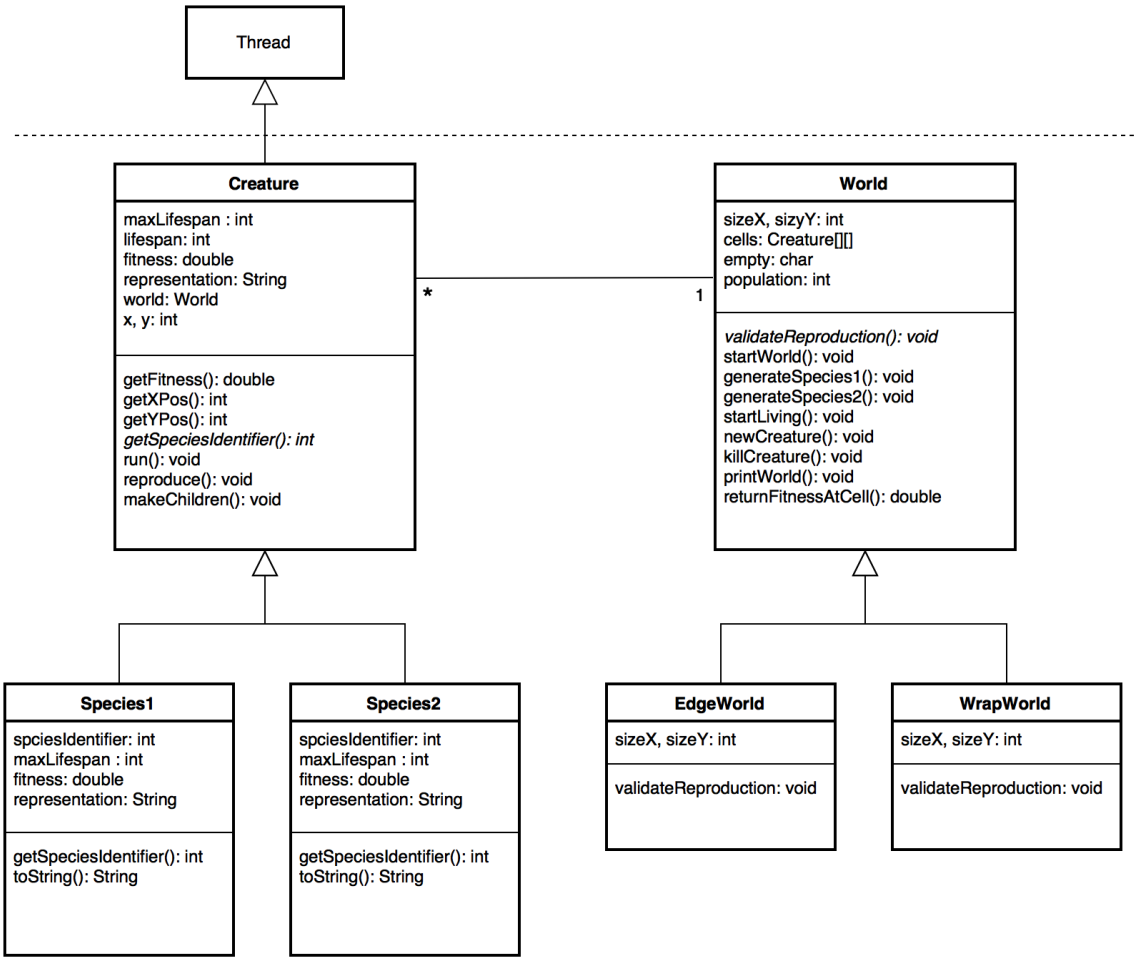

***Fig 1**: Program Design UML diagram*

```
Type '1' for world with edges, or '2' for wrap-around: One
Input not recognised! Please enter either '1' or '2'...
Type '1' for world with edges, or '2' for wrap-around: 1
EDGE WORLD:
```

***Fig 2**: User prompt at program launch*

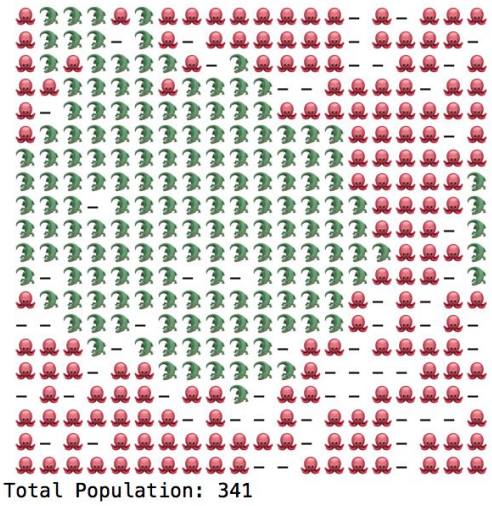

***Fig 3**: Program running with default String representations for Species 1 and Species 2*