World of Robots

Master 1 Génie Physiologique et Informatique

Lab 5-1 : Expertise of the given Turtle project

```
document.getElementById(div).
else if (i==2)
{
    var atpos=inputs[i].indexOf("@");
    var dotpos=inputs[i].lastIndexOf(".");
    if (atpos<1 || dotpos<atpos+2 ||
    document.getElementById('errEmail')
    else
    document.getElementById(div).
}
se if (i==5)
```

Group 14 : Laurène DONG, Marion DOSPITAL, Lauranne LAIR, Thibault NIGGEL, Florian PAILLAS, Ophélie SOCHARD

Wednesday, 11th of October

Some parts of the project present obvious functionalities, while others remain incomplete or merely sketched out. This has enabled us to identify several limitations in the source code. Here's a summary of our analysis of all the classes in the project:

What is done:

A turtle is able to be created and visualized in a canvas. The turtle can rotate 90° and move in the direction it is facing while drawing a line behind it after a movement. The turtle can manage if it is drawing or not. The specific functions 'ralentir' and 'setVitesse' of the 'TortueRapide' class work correctly, as well as the specific function 'Parler' of the ‘TortueSavante’ class. The 'setCouleur' method of the ‘TortueCouleur’ class works well because we can change the color of the trace made by the turtle's movement. Furthermore, if the entered color name is incorrect, the trace will automatically be in the color 'Black'.

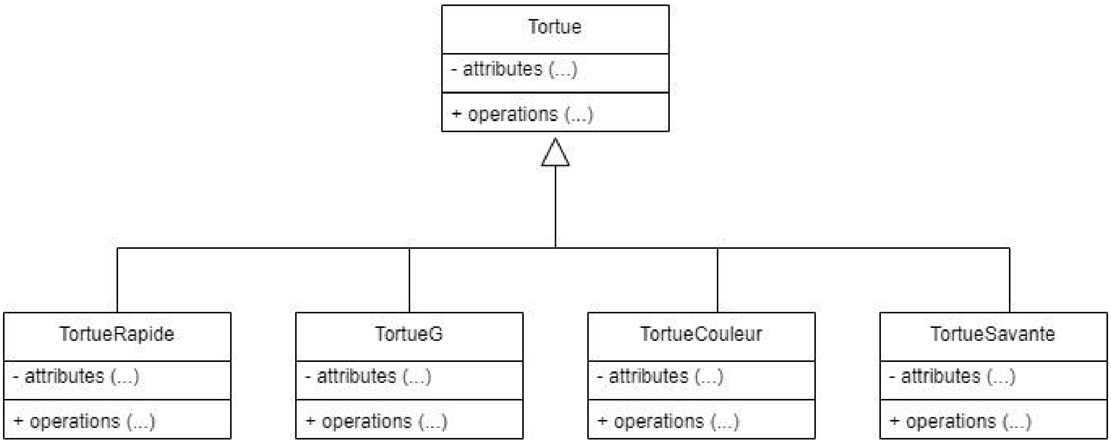
The test classes work correctly because the different turtles perform each line of the tests every time. And thanks to the 'CanvasTortue' and 'Canvas' classes, we can visualize the 'TortueG', 'TortueRapide', and finally, the 'TestCanvasTortue'.

What hasn't been done:

There is no relationship between the "TortueSavante" class and the "CanvasTortue" class, nor between the "TortueCouleur" class and the "CanvasTortue" class. This relationship could help visualize turtles. It might be a good idea to use inheritance for all turtle classes, to avoid code repetition.

Limitations:

The ‘TortueRapide’ class works relatively well, but the forward method with an input parameter refuses to plot the turtle's trajectory even if the plot is true. What's more, the code is rather repetitive, especially in the Turtle classes. It would be advantageous to use inheritance with turtle classes and link the “Tortue” class to the “CanvasTortue” class:


Tortue
- attributes (...)
+ operations (...)
TortueRapide
- attributes (...)
+ operations (...)
TortueG
- attributes (...)
+ operations (...)
TortueCouleur
- attributes (...)
+ operations (...)
TortueSavante
- attributes (...)
+ operations (...)