Assignment 1: PostScript output

Probably you noticed that I finally added Assignment 1 to StudiUM. I put the submission date for **Tuesday** October 3, 20:15 - you will have 2 bonus points for submission before October 1, 20:15 (the original date). The duty is worth 50 points. I leave the bid open until October 4th (with a reasonable penalty for delay).

I added 4 JSON specifications in the test directory on github .

You will also find a tortoise (`EPSTurtle`) to draw and a class `Plotter` that shows how to use the turtle.

```
LSystem lsystem;
// GhostTurtle: tortue bidon sans de
// sortie PostScript sur standard ou
EPSTurtle turtle = new EPSTurtle(new
// maintenant l'en-tête du fichier F
... code pour parser JSON avec turtl
... lsystem.addSymbol, lsystem.setAc
// maintenant plot écrit la reste du
turtle.plot(lsystem, n_iter); // n_i
```

Note that in this solution (see code `EPSTurtle.plot`), the turtle produces the drawing (with `lsystem.tell`), the pseudo-random number generator is reset to its initial state (`lsystem.initRandomGenerator`- v my AbstractLSystem class), and calls `lsystem.getBoundingBox`to add it to the end of file. (BoundingBox is mandatory in an Encapsulated PostScript file.) In principle, EPSTurtle works immediately with your classes.

I produced the drawings with my Plotter class (epstopdf converts x.eps to x.pdf).

```
% java -cp build / classes: lib / json-java.jar lindenmayer.Plotter test /
% epstopdf test / buisson5.eps
```