

L-Système

1 Sujet

Le but de cet exercice est de modéliser un **L-Système** (ou au moins une version simplifiée). Un L-Système est un système de réécriture. On démarre par un premier *mot*, puis on transforme un certain nombre de fois ce mot en suivant des *règles* pour en obtenir un nouveau. Les *règles* indiquent quelles sont les transformations à faire.

Par exemple, on peut avoir le système suivant :

- Mot initial : A ;
- Règles :
 - ' A ' est transformé en ' AB ' ;
 - ' B ' est transformé en ' A '.

En appliquant une première fois les règles sur le mot initial, on obtient le mot AB . En appliquant à nouveau les règles sur le mot AB , on obtient le mot ABA . En appliquant une troisième fois les règles à partir du mot ABA , on obtient le mot $ABAAAB$.

On représente souvent les mots ainsi obtenus sous forme graphique. On attribuera arbitrairement à chaque lettre une fonction (avancer d'une unité, tourner à gauche, ...). On peut par exemple utiliser le framework p5 pour pouvoir visualiser les résultats.

2 Version simplifiée

Dans le cadre de cet exercice, on peut définir un L-système de la manière suivante :

Définition 1. Un **L-Système** simplifié est composé de 2 éléments :

- Un mot initial, appelé **axiome**;
- Un ensemble de **règles**, qui indiquent comment transformer les lettres d'un mot. Quand on n'indique pas comment est modifiée une certaine lettre, cela veut dire qu'elle ne sera pas modifiée.

Pour obtenir une représentation visuelle des mots générés par un L-système, il faut imaginer un petit robot qui se déplace sur une zone de dessin, en laissant une trace là où il passe. Le robot est capable de comprendre certaines lettres, qui correspondent chacune à une instruction bien précise. Pour dessiner un mot, le robot se place quelque part dans la zone de dessin, tourné dans un sens, puis regarde les lettres du mot une à une, en suivant les instructions correspondantes à chaque lettre. Dans un premier temps, le robot sera capable de comprendre 3 lettres :

- F : avancer de 10 pixels;
- $+$: tourner de 90 degrés à gauche;
- $-$: tourner de 90 degrés à droite.

Lorsque le robot lit une lettre qu'il ne connaît pas, il oublie cette lettre et passe à la suivante.

Traditionnellement, ce robot est appelée une *tortue* (Wikipedia).

2.1 Génération des mots

1. Le mot initial sera stocké dans une variable. Quel sera le type de cette variable?

Solution: →

2. Comment peut-on stocker l'ensemble des règles?

Solution: →

3. Comment pourrait-on alors représenter le L-système donné en exemple dans la partie 1?

Solution: →

4. Quel mot obtient-on avec le L-système donné en exemple en appliquant 0, 1, 2, 3 et 4 fois les règles de transformation?

Solution: →

2.2 Dessins associés

5. Écrire une fonction permettant de dessiner un mot, en simulant le déplacement de la tortue. On la fera démarrer tournée vers la droite.
6. Qu'obtient-on en dessinant le mot $F + F$?

Solution: →

7. Qu'obtient-on en dessinant le mot $F + F - F - F + F$?

Solution: →

2.3 Quelques exemples

8. Dessiner une approximation de la courbe de Koch grâce au L-système dont le mot initial est F et les règles sont : $F \rightarrow F + F - F - F + F$.

Solution: \rightarrow

9. Dessiner une approximation de la courbe du dragon avec le L-système dont le mot initial est FX et les règles sont :

- $X \rightarrow X + YF+$;
- $Y \rightarrow -FX - Y$.

Solution: \rightarrow

3 Version un peu moins simplifiée

Dans la partie précédente, la tortue ne pouvait tourner que de 90 degrés. Dans cette partie, le but est d'avoir une tortue qui tourne d'un angle arbitraire. Il faut donc modifier un peu le code de la tortue pour connaître son angle actuel. Initialement, la tortue démarre avec l'angle 0 : elle regarde vers la droite. Avec un angle de π , elle regarde à gauche. Lorsqu'une tortue en position (x, y) est tournée avec un certain angle et avance de n pixels, sa nouvelle position est : $(x + n * \cos(\text{angle}), y + n * \sin(\text{angle}))$.

▲ Les angles sont exprimés en radians, pas en degrés! Pour passer de l'un à l'autre, il suffit de savoir que 180 degré = π radians (Lien).

10. Dessiner une autre version de la courbe de Koch en utilisant une tortue qui tourne de 60 degré, avec le L-système suivant :

- Mot initial : F ;
- Règles :
 - $F \rightarrow F + F - -F + F$.

11. Dessiner le flocon de Koch. La différence avec le L-système précédent est le mot initial : $F - -F - -F - -$.

12. Dessiner un arbre à l'aide du L-système :

- Mot initial : FAB ;
- Règles :
 - $A \rightarrow +FAB++++++F++++++$;
 - $B \rightarrow -FAB++++++F-----$.

L'angle utilisé est 22.5 degrés.

4 Version un peu plus moins simplifiée

On peut rajouter une mémoire à la tortue, en lui permettant de comprendre deux nouveaux symboles : '[' et ']'. Le premier veut dire qu'il faut enregistrer la position actuelle de la tortue (emplacement et angle). Le second veut dire qu'il faut charger la position qui a été enregistrée auparavant.

13. Que va tracer la tortue qui tourne de 90 degrés en lisant le mot $F[F] + F$?

Solution: →

14. Que va tracer la tortue qui tourne de 90 degrés en lisant le mot $F[F][+F] - F$?

Solution: →

Il est aussi tout à fait possible d'empiler les sauvegardes. Lorsque la tortue doit charger une position, elle charge la dernière qui a été enregistrée. Celle-ci disparaît alors de la mémoire.

Par exemple, la tortue qui tourne de 90 degrés produira, en lisant le mot " $\textcolor{red}{F}\textcolor{blue}{F}\textcolor{green}{F} + \textcolor{orange}{F} - F$ " le dessin :

