

# HYBRIDATION SÉQUENTIELLE ALÉATOIRE

C'est un programme qui reprend le principe d'un livre d'image découpé en 3 parties et qui permet d'associer 3 parties de différents animaux ensemble et de manière aléatoire, en appuyant sur une touche du clavier ou en cliquant avec la souris.

```
Accident_Mutation1
1 // variables globales
2 PImage img;
3 PImage img2;
4 PImage img3;
5
6 String[] nomTetes ;
7 String[] nomTorses;
8 String[] nomJambes;
9
```

Ensuite, on définit à quoi correspond chaque variable, pour les blocs textes (ex: nomTetes équivaut au bloc texte dans le dossier DATA nommé «nomTetes.txt»), et pour les images (la première img équivaut aux images contenues dans le bloc texte à partir de sa première ligne [0] ).

Et le cadre dans lequel s'inscrit le programme, de 400 par 600 pixels, ainsi que le couleur de fond qui ici ne sert à rien en fait !

Tout d'abord, des constantes sont initialisées et nommées dans le programme, ce qui permettra de les mettre en forme ensuite.

Ici dans la première série, on a les 3 images tapées de manière distinctes, et ensuite 3 blocs texte qui contiendront l'ensemble des images pour chaque bande.

```
10 void setup(){
11     nomTetes = loadStrings("nomTetes.txt");
12     nomTorses = loadStrings("nomTorses.txt");
13     nomJambes = loadStrings("nomJambes.txt");
14
15     size (400,600);
16     background(#1A1A1A);
17     img = loadImage(nomTetes[0]);
18     img2 = loadImage(nomTorses[0]);
19     img3 = loadImage(nomJambes[0]);
20 }
```

Puis, le «void draw» permet de dessiner les images qui vont s'afficher dans le programme, de les faire exister en entrant les coordonnées de leur apparition par rapport aux dimension de la fenêtre.

(Ici, chaque image est découpée en 400X200 pixels)

```
22
23 void draw() {
24     image(img, 0, 0);
25     image(img2, 0,200);
26     image(img3, 0,400);
27 }
```

```
29 void keyPressed(){
30     int i = floor(random(0,nomTetes.length));
31     img = loadImage(nomTetes[i]);
32     println("nomTetes = " + nomTetes[i]);
33
34     int j = floor(random(0,nomTorses.length));
35     img2 = loadImage(nomTorses[j]);
36     println("nomTorses = " + nomTorses[j]);
37
38     int k = floor(random(0,nomJambes.length));
39     img3 = loadImage(nomJambes[k]);
40     println("nomJambes = " + nomJambes[k]);
41
42 }
```

Enfin, une fonction est ajoutée permettant de faire changer aléatoirement chaque image une fois en appuyant sur une touche du clavier :

Les fonctions i, j et k sont données pour chaque variables devenant aléatoire [ grâce au «random» ]. La formule va sélectionner à chaque fois un fichier présent dans le document txt nommé.



Chose qui marche bien.

L'avantage avec ce programme c'est que l'on peut ajouter des fichiers à mélanger à l'infini. En revanche les raccords ne sont pas toujours élégants, le résultats varie donc en fonctions des pièces associées.



Chose qui marche moins bien.

Le fond gris foncé permet de bien distinguer chaque partie sans interférence avec un fond différent pour chaque image.

Chaque être vivant est le fruit de mutations évolutives, à travers ce programme, j'ai voulu caricaturer la mutation en assemblant les animaux existants entre eux. Ce programme peut servir d'inspiration pour créer de petites chimères ! Le dessin permet de raccorder les parties et de rendre l'association plus fluide.

