

Écosystème

Projet UML

2022

1 But

Le but de ce projet est d'implanter en machine une simulation d'univers cohérent composé de quelques animaux et de quelques végétaux.

On s'attend à un code source qui utilise les concepts présentés en cours, et en particulier de l'héritage.

2 Description globale

L'univers considéré est composé de moutons, de loups, d'herbe et de sels minéraux. Les loups mangent les moutons pour survivre, les moutons mangent l'herbe dans le même but. Lorsqu'un loup ou un mouton meurt de faim ou de vieillesse, il dépose sur la case de sa mort des sels minéraux, qui serviront d'engrais à l'herbe. On considère que les loups ne se mangent pas entre eux, et ne peuvent que mourir de vieillesse. L'univers est dit "mort" lorsqu'il n'y a plus ni loups ni moutons vivants. Un des buts du projet est d'éviter cette situation, en équilibrant soigneusement le nombre initial de loups, moutons, herbe et sels minéraux.

2.1 Univers

Un univers possède $m \cdot n$ cases, et de l'herbe sur chaque case au début de la simulation. Le nombre de moutons et de loups est fixé par l'utilisateur, et les animaux sont placés aléatoirement dans l'univers (avec un animal maximum par case). Votre programme doit savoir gérer les cas limites, par exemple plus d'animaux que de cases dans l'univers...

2.2 Déplacement

Les moutons et les loups se déplacent d'une case par tour. Chaque animal possède donc 9 possibilités de déplacement à chaque tour. Dans un premier temps, vous pourrez implémenter un déplacement aléatoire, avant de penser à des algorithmes plus évolués.

2.3 Fonctions vitales

Un mouton mange toute l'herbe d'une case, et vit 50 tours (s'il n'est pas mangé avant). À sa mort, si elle est naturelle, il dépose des sels minéraux sur la dernière case qu'il a fréquenté. Deux moutons sur des cases adjacentes peuvent se reproduire si leur sexe le permet : ils sont immobiles pendant un tour, puis un nouveau mouton apparaît sur une case adjacente au mouton mère (toutes les cases ne sont pas forcément disponibles... Si le jeune mouton n'a pas de case libre, il ne naît pas).

Enfin, un mouton meurt s'il est mangé par un loup, s'il n'a pas mangé d'herbe pendant 5 tours (il meurt au 6ème), ou s'il est trop vieux. Un loup mange des moutons (un mouton rassasie un loup), et doit en manger un tous les 10 tours. S'il meurt de vieillesse (au bout de 60 tours) ou de faim, il dépose des sels minéraux sur la dernière case qu'il a fréquenté. Deux loups peuvent se reproduire de la même manière que les moutons. Enfin, de l'herbe pousse un tour après que des sels minéraux aient été déposés sur une case dépourvue d'herbe.

3 Interface texte

Pour l'interface texte, on pourra par exemple, représenter l'univers par un tableau 2D, utiliser des lettres ou symboles ASCII pour représenter chaque entité, et afficher une trace textuelle de ce qu'il se passe dans l'univers. Par exemple :

	1	2	3	4	5	6	7	
	+	+	+	+	+	+	+	+
A			S					
	+	+	+	+	+	+	+	+
B				L		S		
	+	+	+	+	+	+	+	+
C					M			
	+	+	+	+	+	+	+	+
D					S			
	+	+	+	+	+	+	+	+
E							M	
	+	+	+	+	+	+	+	+
F								
	+	+	+	+	+	+	+	+
G				L				
	+	+	+	+	+	+	+	+

[C4] Un mouton a servi
de repas a un loup.
[A2, B5, D4] De l'herbe
repousse !
[G3] Un loup meurt de
vieillesse.
[E6] Un mouton meurt de faim.

Tour 800 | Loups : 2 | Moutons : 2

	1	2	3	4	5	6	7	
	+	+	+	+	+	+	+	+
A			H					
	+	+	+	+	+	+	+	+
B						H		
	+	+	+	+	+	+	+	+
C					L			
	+	+	+	+	+	+	+	+
D					H			
	+	+	+	+	+	+	+	+
E							S	
	+	+	+	+	+	+	+	+
F								
	+	+	+	+	+	+	+	+
G				S				
	+	+	+	+	+	+	+	+

Tour 801 | Loups : 1 | Moutons : 0

4 Idées supplémentaires

Les plus avancés d'entre vous peuvent ajouter d'autres fonctions. Voici une liste d'idées (évidemment non exhaustives) :

- Ajoutez la possibilité de sauvegarder/charger un univers existant.
- Ajoutez la possibilité de mettre la simulation en pause (et de la reprendre).
- Modifiez le déplacement des loups/des moutons pour qu'il soit plus intelligent.
- Modifiez votre code pour imposer un rythme de reproduction : par exemple, une louve ne peut faire un petit qu'une fois tous les n tours.
- Trouvez les meilleurs paramètres pour faire durer l'univers le plus longtemps possible !