

Un jeu de la vie en JavaScript

HAI305I (projet JavaScript en monôme ou binôme)

MCC de HAI305I : $\max(\text{écrit}, \text{écrit} \cdot 0.7 + \text{QCM-PHP} \cdot 0.15 + \text{ProjetJS} \cdot 0.15)$

Pierre Pompidor

Le but du projet consiste à réaliser un jeu graphique (de type simulateur) en JavaScript, médiatisé dans un navigateur, et qui permette à chaque démiurge (1 à 4 joueurs en réseau) de créer un couple de créatures herbivores (mâle et femelle), puis une fois la simulation lancée, d'assister à l'évolution de ces créatures et de leurs descendants dans un écosystème où vivront également les créatures créées par les autres joueurs.

Le but du joueur est de préalablement "paramétrer" ses créatures pour quelles puissent prospérer au mieux, le gagnant étant celui qui (après un nombre paramétrable de tours) aura le plus de créatures de son espèce en vie (ou qui aura été le créateur de la dernière créature vivante).

Un joueur « initiateur » créera une partie en définissant le nombre de joueurs (il pourra aussi jouer en solo), le nombre de tours de simulation, et créera un premier couple de créatures. Un joueur suivant ne pourra pas choisir un paramétrage déjà défini. La simulation se lancera automatiquement une fois que les paramètres des espèces auront été établis. Les créatures évolueront dans l'ordre d'entrée de démiurge dans la partie (en réservant par exemple 2 secondes par espèce)..

Toutes les valeurs de paramètres qui vont suivre doivent être externalisées pour pouvoir être facilement modifiées (je ne suis pas sûr que mes propositions conviennent à un bon fonctionnement du jeu).

Chaque joueur doit distribuer 9 points sur trois axes valués de 1 à 5 :

- le taux de reproduction ;
 - la perception (pour repérer à distance les ressources en eau et en nourriture) ;
 - la force (pour conserver/occuper une case convoitée/occupée par une créature d'une espèce différente).
- Au moins 1 point devra être attribué à chaque axe.

Le terrain sera constitué de 13x13 hexagones. Hormis les hexagones tanières, le terrain sera approximativement constitué d'hexagones de type eau (15%), prairie (35%) et rocher (50%) distribués aléatoirement. Hormis la tanière, chaque hexagone ne pourra être occupé que par une seule créature.

Par tour de jeu, une créature peut se déplacer sur n'importe quel hexagone connexe à condition qu'il soit inoccupé, ou s'il est occupé, qu'il le soit par une créature d'une espèce différente et que cette dernière soit plus faible et qu'elle puisse être « rejetée » sur un hexagone connexe inoccupé.

Chaque tanière est au centre d'un bord du terrain. Les créatures d'une même espèce naissent dans leur tanière et ne se reproduisent que dans celle-ci.

L'état physiologique des créatures sera représenté par les taux d'hydratation et de satiété valués de 0 à 10.

Une case eau confère 3 unités d'hydratation par tour.

Une case prairie confère 2 unités de satiété par tour.

Un déplacement d'un hexagone à une autre coûte 1 unité d'hydratation et 0.5 unité de satiété.

Un arrêt sur un hexagone coûte 0.5 d'hydratation et 0.25 de satiété.

Un taux d'hydratation ou de satiété à 0 fait mourir la créature.

Dès que les taux d'hydratation et de satiété sont au moins de 6, la créature revient vers sa tanière pour se reproduire. Si deux créatures de sexes opposés s'y trouvent et n'ont pas enfanté depuis 5 tours, elles donneront naissance à autant de nouvelles créatures (de sexe aléatoire) que leur taux de reproduction.

L'implémentation des spécifications précédentes étant plafonnée à 17/20, une interaction à votre choix entre joueur et écosystème pourrait être implémentée. Par exemple celle qui permette à un joueur (lors du temps d'évolution de ses créatures) d'échanger deux hexagones (pour rapprocher ou éloigner) une case eau ou prairie avec une case rocher !

L'implémentation se fera en JavaScript en utilisant obligatoirement la plateforme Node.js et les bibliothèques D3.js et socket.io. Des projets aux codes similaires ne seront pas notés.