

# Vie Artificielle

## Automates Cellulaires 2D (jeu de la vie, wireworld)

Mise à jour : Janvier 2019

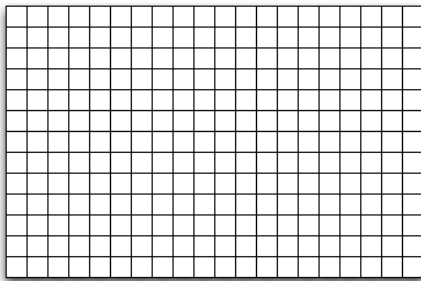
**Mots clés:** automates cellulaires, 1 et 2 dimensions, n états, totalistique.

Une page est disponible sur le site <http://pages.isir.upmc.fr/~bredeche/> - Vous y trouverez des informations diverses (lien vers les sources, les sujets, etc.). En particulier vous trouverez un lien vers les codes sources donnés en annexe.

Automates cellulaires 2D

[sur feuille]

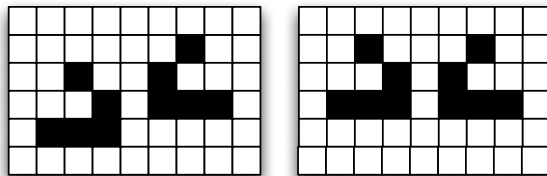
**Question 1. planeurs dans le jeu de la vie**



1. Dans un environnement du type de celui présenté ci-dessus, placez deux planeurs tels qu'ils rentrent en collision dans les itérations qui suivent.

**Question 2. Collision de planeurs**

Soit la configuration suivante:



Exécutez les quatre prochaines étapes du Jeu de la Vie pour ces deux configurations

Automates Cellulaires 2D

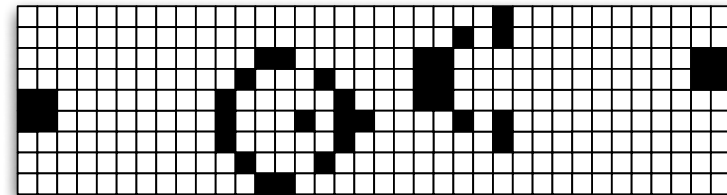
[sur ordinateur]

Vous trouverez sur le site web une archive contenant plusieurs sources permettant de coder un automate cellulaire 2D. Dans l'état actuel, cet automate se contente de mettre à jour l'état d'une cellule en copiant l'état de la cellule qui se trouve au dessus et à gauche d'elle.

**Question 1.** Modifiez ce programme pour implémenter les règles du jeu de la vie. Initialisez votre environnement avec les configurations de planeurs vues dans l'exercice précédent afin de valider votre implémentation.

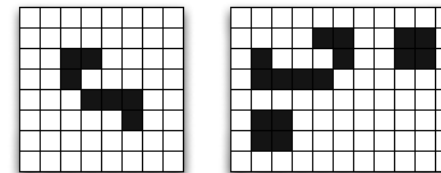
**Question 2.** Testez des configurations au hasard, en faisant varier la probabilité d'initialisation d'une cellule à 1. Menez une étude sommaire pour trouver une valeur basse de probabilité permettant toutefois d'éviter l'extinction.

**Question 3.** Initialisez votre environnement selon l'image ci-dessous.



**Question 4.** L'initialisation ci-dessus vous permet d'obtenir un canon à planeurs. En utilisant ce type de canon, trouver une configuration telle que la rencontre de deux planeurs provoquent leur annulation.

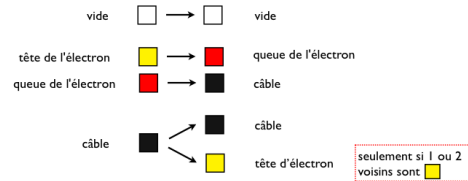
**Question 5.** Les deux configurations ci-dessous montrent des "mangeurs de planeurs". Lorsqu'un planeur arrive sur ces blocs, si l'angle d'approche est correct, il disparaît (il est "mangé"). Trouvez une configuration incluant un planeur tel qu'il sera bien absorbé par chaque mangeur.



**Question 6. [FACULTATIVE, à faire une fois tous les exercices de ce TD terminé]** construisez des portes ET, OU et OU-EXCLUSIF. La motivation? Construire ces portes logiques est un premier pas vers la programmation dans le Jeu de la Vie.

Un article pour aller plus loin, et qui peut vous aider: "Implementation of logical function in the Game Of Life" [JP Rennard, 2002]

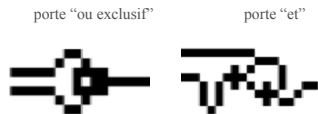
Cet automate cellulaire 2D a été proposé par [Silverman, 1987] et permet (dans une certaine mesure) de simuler le parcours d'électrons sur une surface conductrice. Les règles sont rappelées ci-dessous:



**QUESTION 1.** Programmer les règles de cet automate cellulaire. Vous exécuterez ensuite les trois schémas suivants (il s'agit d'un montage simulant une porte OU):



**QUESTION 2.** A partir des deux environnements initialisés comme suit, exécutez le même processus que précédemment (ie. les 3 cas possibles outre le cas où il n'y a pas d'activité):



**QUESTION 3.** Fabriquez un schéma permettant de simuler le montage "a ou b et a ou c". Ecrivez la table de vérité du schéma.

On souhaite simuler de manière simplifiée (ie. qui *paraît* réaliste sans respecter les lois de la physique) l'écoulement d'un liquide (eau, lave, boue, etc.) sur un espace avec des différences de niveau. Pour cet exercice, on souhaite simuler l'écoulement d'une quantité finie de lave sur les parois d'un volcan. En utilisant ce que vous avez appris, improvisez et implémentez ce système.

Suggestion: vous pouvez utiliser 2 matrices 2D: une matrice "altitude" et une autre "liquide". La première ne sera jamais mise à jour (ie. pas de règles à implémenter) et contiendra les altitudes de chaque case. La seconde contiendra pour chaque case la quantité de liquide présente (un nombre entier entre 0 et ...). La mise à jour pour chaque case de la matrice "liquide" sera faite de manière asynchrone randomisée: on choisit une case au hasard et on la met à jour en fonction de son voisinage immédiat (règle de Moore), puis on passe à une autre case. On utilisera la règle suivante: (1) pour une case donnée, on calcule ( altitude + unités\_de\_liquide\_présentes ) puis on compare avec une case voisine prise au hasard. (2) Si la case voisine choisie est plus haute et contient au moins une unité de liquide, alors on "transfère" cette unité à la case courante.

A vous de trouver un moyen de simuler les différences de viscosité (l'eau est moins visqueuse que la lave). Lorsqu'une case est traitée, on pourra par exemple calculer une probabilité de mise à jour dépendant de la différence d'altitude et/ou de quantité d'eau entre les cases considérées (ex. la vitesse d'écoulement sera la même quelque soit la différence d'altitude pour de l'eau alors qu'elle sera plus différente pour de la lave).