CHAIX Thibault
DELAIRE Romain

UNIVERSITÉ Clermont Auvergne

TP jeu de la vie

Vue d'ensemble

Travail pratique de réalisation d'un jeu de la vie fonctionnel en langage C.

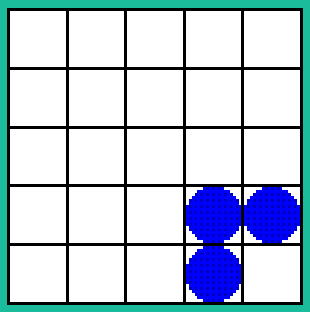
voir les règles ici: Jeu de la vie — Wikipédia (wikipedia.org)

Objectifs

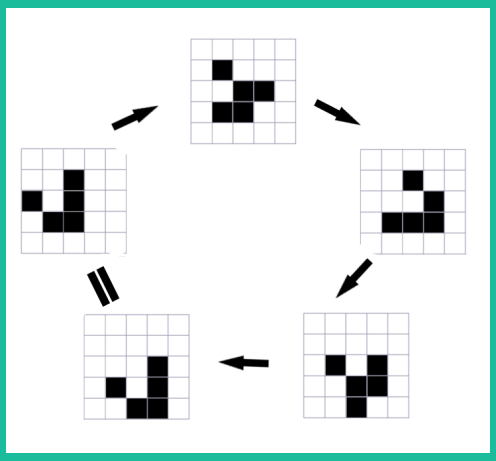
1. **Réalisation de base et 1er test :** Dans un 1er temps nous avons réalisé le jeu de la vie avec la configuration dite du "glider" dans un tableau 10x10.
2. **Implémentation du "Torus Universe" :** Dans un second temps, nous avons modifié ce tableau pour en faire un espace infini à 2 dimensions.
3. **Tests sur différentes tailles de grilles:** modification de la taille de la grille, test et observations.
4. **Implémentation de configurations aléatoires:** Après avoir testé le "glider", que se passerait-il si la grille se remplissait seule de façon aléatoire?

1. Réalisation de base et 1er test

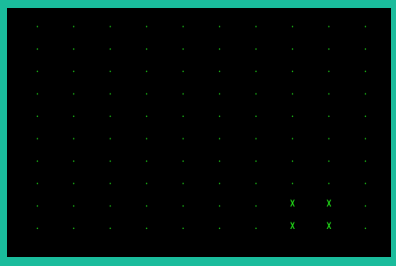
Réalisation d'un jeu de la vie avec la configuration "glider" .



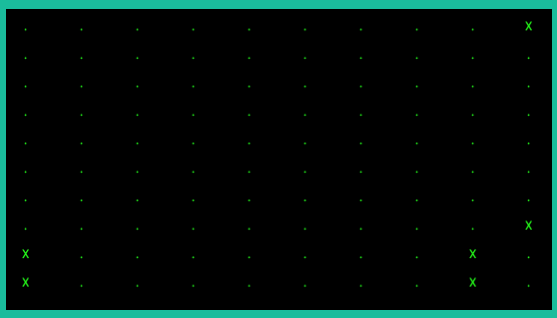
Pour se faire, nous avons créé un tableau 10x10 en y insérant des croix représentant les cellules vivantes aux coordonnées [0][1] , [1][2], [2][0], [2][1] et [2][2] et des "." aux autres cases.
Suite à celà nous avons créé une fonction qui va compter le nombre de voisins de chaque cellule et l'insérer dans un second tableau.
Une seconde fonction va quant à elle tester la parité de chaque cellule (. ou X) pour lui appliquer la règle qui s'y applique en fonction de la parité et du nombre de voisin, pour ensuite remplir la grille avec les nouvelles cellules.



Une fois le programme créé, il ne reste alors plus qu'à le tester: comme prévu ce motif se répète CEPENDANT, la grille étant pour le moment fini, arrivé au bout de cette dernière le motif se fige dans la position ci-contre, formant alors 4 X dans l'angle inférieur-droit de la grille.



2. Implémentation du “Torus universe”



Pour remédier à ce problème nous avons donc utilisé le “Torus Universe”

qui permet de transformer un tableau fini en espace 2D infini

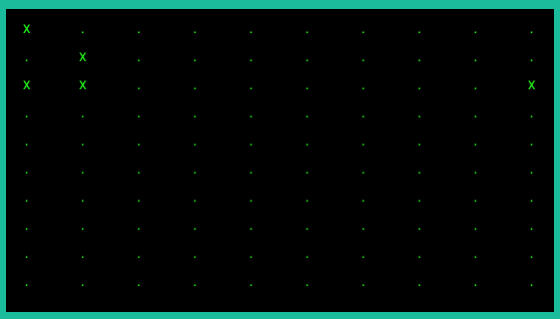
en réunissant les bordures gauche et droite du tableau ainsi que

les bordures haute et basse de ce dernier. Pour cela il suffit

de changer les conditions des test de voisinage pour que

les cellules placées en bordures soient testées avec

les cellules placées sur la bordure opposée.

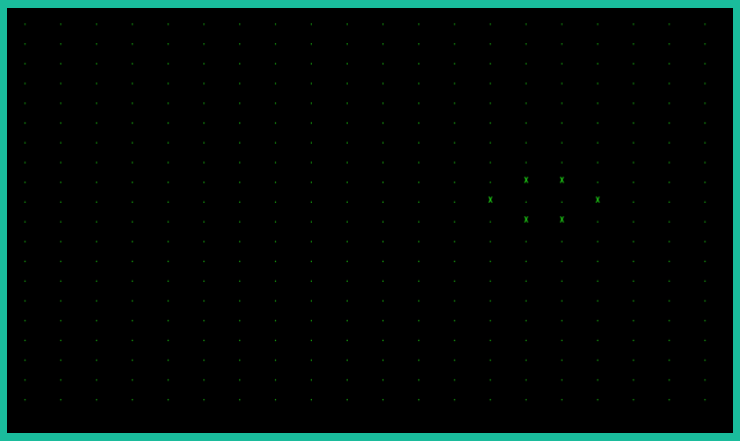


3. Tests sur différentes tailles de grilles

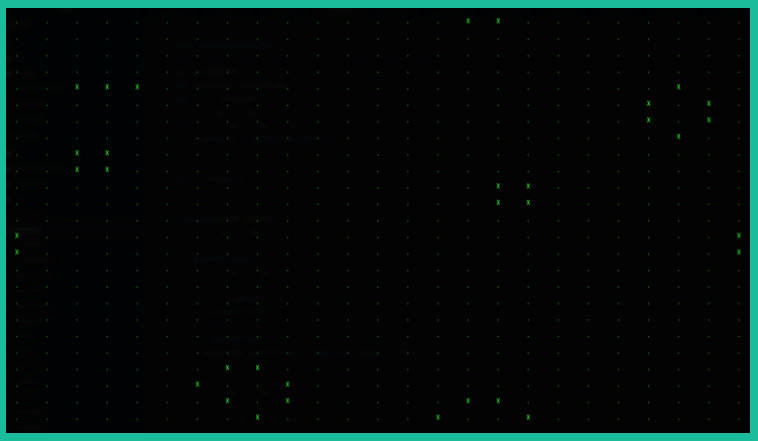
Comme demandé nous avons alors testé notre programme sur une grille de taille 50x50 mais cela n'a pas été très concluant puisque grâce au Torus Universe notre grille 10x10 est en réalité une grille infinie donc la seule différence est le temps pour que le motif fasse le tour de celle-ci.

4. Implémentation de figures aléatoires

Pour conclure ce Tp, nous avons modifié le programme pour que, au lieu de commencer la boucle en configuration “glider”, nous avons laissé le hasard choisir notre état de base en implémentant l’algorithme de Mersenne Twister. Nous avons ainsi pu observer l’évolution de figures n’étant pas choisies pour correspondre à ce jeu. 3 résultats différents sont apparus, une figure stabilisée qui ne se modifie pas de l’état T à l’état T+1. Un motif comme le Glider qui continue de se transformer à chaque état. Le dernier état un peu plus triste est l’extinction de toute la population de façon définitive...



motif fixe



motif en constante évolution