

ITC – TD n°09

Jeu de la vie

Nous continuons notre exploration des automates cellulaires avec le « jeu de la vie » : il s'agit est un automate à deux dimensions très connu, imaginé par le mathématicien John Conway en 1970.

Dans tout ce qui suit, on appelle « monde » un tableau à deux dimensions, représenté en Python par une liste de listes, chaque élément du tableau pouvant prendre les valeurs 0 ou 1.

Fichiers disponibles en ligne

- ☞ des instructions à copier-coller pour des tests dans `tests_fournis.py`
- ☞ les structures pré-établies `test1.txt`, `test2.txt`, `clignotant.txt`, `plaineur.txt`, `ensemble_complexe.txt`, `gosper.txt`

I. Préparation

1 – Écrire une fonction `copie2D` qui prend en paramètre un monde (tableau à deux dimensions représenté par une liste de listes) et en renvoie une copie complètement indépendante.

2 – Écrire une fonction `lire_fichier(nom_fichier)` qui prend en paramètre une chaîne de caractères représentant un nom de fichier contenant des points et des « O », et qui ouvre ce fichier, le lit et en déduit les coordonnées des « O » pour les mettre dans une liste qu'elle renvoie. Par exemple, si on propose un fichier contenant les lignes suivantes :

```
..0.0
0.00
..0
```

la fonction `lire_fichier` doit renvoyer la liste

```
[[0, 2], [0, 4], [1, 0], [1, 2], [1, 3], [2, 2]]
```

On pourra utiliser les fichiers `test1.txt` et `test2.txt` pour tester cette fonction.

3 – Écrire une fonction `initialisation(structure_départ, taille_x, taille_y)` qui prend en paramètres une liste de coordonnées ainsi que deux entiers `taille_x` et `taille_y`, et qui crée puis renvoie un tableau à deux dimensions contenant des cases vivantes aux coordonnées présentes dans la liste fournie, et `taille_x` colonnes vides et `taille_y` lignes vides de part et d'autre. Par exemple, si la structure de départ a une taille 3×4 et que `taille_x` vaut 10 et `taille_y` 20, le tableau à 2D créé sera de taille 23×44 . On pourra utiliser le test fourni pour cette fonction.

À ce stade, on peut déjà utiliser l'ensemble de ces fonctions préparatoires :

```
struct = lire_fichier('gosper.txt')
monde = initialisation(struct, 10, 10)
imshow(monde, cmap='binary', interpolation='none', aspect='equal', vmin=0, vmax=1)
```

doit afficher le résultat ci-contre

