

Architecture Orienté Objet

TP - Jeu de la vie

Sujet :

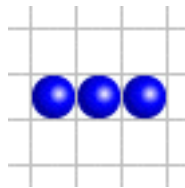
Dans ce TP, vous allez devoir implémenter le jeu de la vie.

Le jeu se déroule sur une grille à deux dimensions, théoriquement infinie (mais de longueur et de largeur finies et plus ou moins grandes dans la pratique), dont les cases — qu'on appelle des « cellules », par analogie avec les cellules vivantes — peuvent prendre deux états distincts : « vivantes » ou « mortes ».

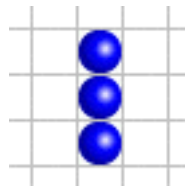
À chaque étape, l'évolution d'une cellule est entièrement déterminée par l'état de ses huit voisines de la façon suivante :

- Une cellule morte possédant exactement trois voisines vivantes devient vivante (elle naît).
- Une cellule vivante possédant deux ou trois voisines vivantes le reste, sinon elle meurt.

Ainsi, la configuration :



donne au tour suivant la configuration :



...qui redonne ensuite la première.

On peut également formuler cette évolution ainsi :

- Si une cellule a exactement trois voisines vivantes, elle est vivante à l'étape suivante.
- Si une cellule a exactement deux voisines vivantes, elle reste dans son état actuel à l'étape suivante.
- Si une cellule a strictement moins de deux ou strictement plus de trois voisines vivantes, elle est morte à l'étape suivante.

Questions :

Récupérer le projet gamoflife.

Il s'agit d'un programme implémentant le jeu de la vie. Cependant, ce programme comporte des erreurs de conception, mais aussi, à l'exécution.

Ré-usiner (refactorer) et corriger le programme afin que son architecture soit correcte et qu'il respecte les spécifications suivantes :

- Le programme devra prendre en entrée un fichier texte contenant la matrice de départ.
 - Sur la première ligne, on aura la taille de la matrice : nb_colonne * nb_ligne
 - Ensuite sur chaque ligne, on aura un «x» pour une cellule morte et un «o» pour une cellule en vie.
- Ensuite, le programme devra être capable de calculer les X (X étant une nombre saisi par l'utilisateur) itérations suivantes et d'enregistrer le résultat dans un fichier.
- Le programme s'exécute à l'infini si X vaut -1. Dans ce cas l'enregistrement se fait lorsqu'on quitte le programme.
- Du plus, le programme permet d'afficher chaque itération dans une fenêtre afin de contrôler l'exécution du programme.

Exemple :

Fichier en entrée :

```
3*3
xxx
ooo
xxx
```

Correspond à une matrice de 3 lignes sur 3 colonnes, dont les cellules de la deuxième ligne sont toutes en vie.

Votre programme devra à la première itération enregistrer le résultat suivant dans un fichier.

```
3*3
xox
xox
xox
```