

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

13/12/2019

# Rapport d'analyse

Le jeu de la vie

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

ESIEE Paris

CLARET MARJOLAINE, ELHUYAR THEO, BESNIER MORGANE

## Objectifs

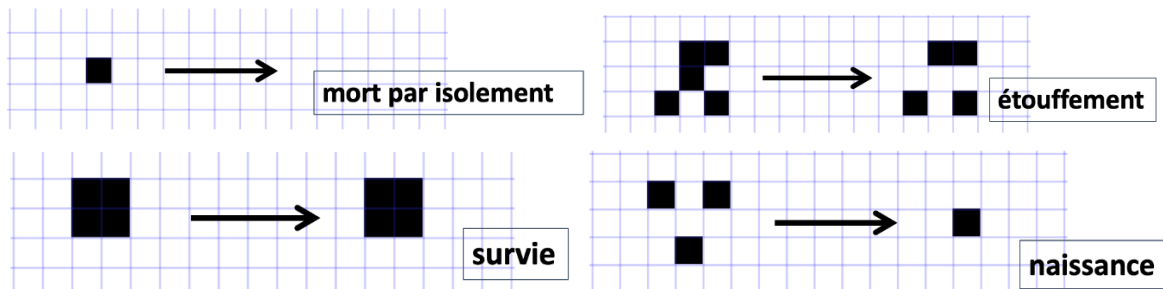
### Contexte

Le jeu de la vie a été créé en 1970, et contrairement à ce que laisse entendre son nom il ne s'agit pas d'un jeu mais d'un automate cellulaire, c'est-à-dire une grille contenant des cellules qui évoluent au cours du temps. Le passage d'un instant  $t$  à  $t+1$  est appelé génération, et l'état de chaque cellule à un instant  $t$  est binaire (morte ou vivante) et dépend de son état à l'instant  $t-1$  et de ses voisines immédiates.



Le joueur peut imposer ses règles d'évolution de l'état des cellules en début de partie, et interrompre le jeu à tout instant. Il positionne manuellement les cellules de l'état initial sur la grille. Les règles par défaut sont simples : une cellule de la grille est entourée de 8 voisines, et évolue selon :

- Mort par isolement : de 0 à 1 cellule(s) voisine(s) vivante(s)
- Mort par étouffement : de 4 à 8 cellules voisines
- Survie : 2 cellules voisines vivantes
- Naissance d'une nouvelle cellule : 3 voisines vivantes



On suit ainsi l'évolution au cours du temps des cellules sur la grille, nous choisirons de représenter les cellules vivantes en noir et les mortes en blanc.

### *Le programme*

Le programme devra être en mesure de permettre à l'utilisateur de choisir la configuration de cellules initiale, les règles décidant de l'évolution de l'état des cellules ou de garder celles par défaut et de sélectionner, si désiré, le motif définissant l'état des cellules. Il devra être possible de quitter le jeu à tout moment.

Une fois le jeu de la vie lancé, l'utilisateur visualisera en temps réel l'évolution des cellules sur la grille, et un compteur de génération permettra de suivre le déroulement. Il devra aussi avoir la possibilité de recommencer la partie, c'est-à-dire de revenir à l'état initial, de mettre le jeu sur pause, d'augmenter la vitesse de passage d'une génération à l'autre et d'implémenter le changement de génération manuellement.

*Interface proposée*

Le jeu de la vie

1. Jouer

2. Aide

3. Quitter

Votre choix : <?>

*Interface de départ*

Choisissez le nombre d'individus :

Coordonnee de l'individu no 1 :

Coordonnee de l'individu no 2 :

	0	1	2	3	4	5	6	7	8	9
A	;	;	;	;	;	;	;	;	;	;
B	;	;	;	;	;	;	;	;	;	;
C	;	;	;	;	;	;	;	;	;	;
D	;	;	;	;	;	;	;	;	;	;
E	;	;	;	;	;	;	;	;	;	;
F	;	;	;	;	;	;	;	;	;	;
G	;	;	;	;	;	;	;	;	;	;
H	;	;	;	;	;	;	;	;	;	;
I	;	;	;	;	;	;	;	;	;	;
J	;	;	;	;	;	;	;	;	;	;

*Mise en place des cellules*

	0	1	2	3	4	5	6	7	8	9
A	;	;	;	;	;	;	;	;	;	;
B	;	;	;	;	;	;	;	;	;	;
C	;	;	;	;	;	;	;	;	;	;
D	;	;	X	X	X	;	;	;	;	;
E	;	;	X	;	X	X	;	;	;	;
F	;	;	;	;	X	X	;	;	;	;
G	;	;	;	;	;	;	;	;	;	;
H	;	;	;	;	;	;	;	;	;	;
I	;	;	;	;	;	;	;	;	;	;
J	;	;	;	;	;	;	;	;	;	;

generation : 4

Appuyez sur Entrer pour passer a la generation suivante

### Évolution des cellules

Jouer

1. Regles par default
2. Regles personnalisées

Votre choix : <?>

### Écran de jeu

Regles personnalisées

Population (entrez un caractere) :

Seuil de mort par isolement (entrez un chiffre) :

Mort par etouffement (entrez un chiffre) :

Seuil de naissance (entrez un chiffre) :

### Personnalisations des règles

Aide

Principe du jeu de la vie :

Le jeu de la vie a été créé par...

On suit l'évolution d'une population...

Une cellule meurt si...

### Interface Aide

Quitter

c'est votre dernier mot ?

*Interface de fermeture du programme*

## *Analyse*

### *Le squelette du programme*

L'ensemble des fonctions du produit consistera en un enchaînement de menus, saisies de réponses et affichages de résultats.

Un squelette (réutilisable) pourra être le suivant :

### **Programme principal**

jeu de la vie

boucle infinie (tant que l'utilisateur ne quitte pas)

introduction

saisie du choix :

choix 1 : jouer

choix 2 : aide

choix 3 : fin

si l'utilisateur choisit de jouer saisie du choix des règles

selon le choix du joueur :

choix 1 : jouer avec les règles par défaut

choix 2 : jouer avec des règles personnalisées

au revoir

**Programme jeu**

introduction (explications...)

boucle infinie (tant que l'utilisateur ne quitte pas ou que le nombre de génération n'est pas atteint)

affichage de la grille de jeu

calcul du nouvel état cellulaire (en fonction de l'état interne et des voisins)

affichage de la grille mise à jour

si l'utilisateur fait une action (pause, remise à zéro, ...)

calcul du nouvel état cellulaire selon l'entrée :

pause : figer la grille (et reprise si demandé)

remise à zéro : reprendre avec l'état initial

augmenter la vitesse : mise à jour du temps nécessaire pour passer d'une génération à l'autre

configuration manuelle : pas d'évolution tant que l'utilisateur ne demande pas à passer à la génération suivante

quitter : fin

affichage de la grille mise à jour

fin

## *Les données*

La représentation des données pourra être la suivante :

**Taille de la grille de jeu** *int*

**Nombre de générations voulues** *int*

**Nombre de générations voulues** *int*

**Matrice/grille de jeu** *array de type int* (par exemple un array, M=[[0][1], [0][0], [1][1], [0][0]] )

D'autres données peuvent être envisagées, comme par exemple les retours de menus :

**JEUV\_GAME** *int* – l'utilisateur a choisi le choix "jouer"

**JEUV\_RULES\_P** *int* – l'utilisateur a choisi le choix "règles personnalisées"

**JEUV\_RULES\_D** *int* – l'utilisateur a choisi le choix "règles par défaut"

**Motif des cellules** *list* (par exemple une liste [photo1, photo2, photo3])

**Nombre de cellules voisines nécessaire pour qu'une nouvelle cellule naisse** *int* (par exemple un dictionnaire 'Naiss' : {3})

**Nombre de cellules voisines nécessaire pour qu'une cellule meurt par isolement** *int* (par exemple un dictionnaire 'Isol' : {1, 2, 4, 5})

**Nombre de cellules voisines nécessaire pour qu'une cellule meurt par étouffement** *int* (par exemple un dictionnaire 'Etouf' : {2, 3, 4})

**Nombre de cellules voisines nécessaire pour qu'une cellule survive** *int* (par exemple un dictionnaire 'Surv' : {2})

**JEUV\_AIDE** *int* – l'utilisateur a choisi le choix "aide"

**JEUV\_QUITTER** *int* – l'utilisateur a choisi le choix "quitter"



## *Les fonctions à écrire*

**Affichage/saisies réutilisables tels quels dans tout projet**

### ***Cpt\_voisin***

Déclaration	Def Cpt_voisin (M, xcell, ycell)
Rôle	Compte les cellules voisines d'une cellule donnée (xcell, ycell) de la matrice ( $0 \leq x \leq 8$ )
Paramètres	Une matrice M, les coordonnées d'une cellule dont on veut connaître le nombre de voisines xcell, ycell
Retour	Nb_voisin le nombre de voisin(s) de la cellule
Appelé par	Evolution
Appelle	Aucun

### ***Placement***

Déclaration	Def Placement (M, coord_new_cell)
Rôle	Place les cellules vivantes initialement choisies par l'utilisateur
Paramètres	Une matrice M, coord_new_cell contenant les coordonnées de chaque cellule qu'a choisi l'utilisateur
Retour	M la matrice M avec les cellules vivantes initiales placées
Appelé par	Joueur
Appelle	Aucun

**Affichages/saisies adaptable à d'autres projets**

### ***Menu\_principal***

Déclaration	Menu_principal ()
Rôle	Affiche le menu principal, 4 propositions (Jouer, Aide, Quitter) et demande une saisie
Paramètres	Aucun
Retour	Indicateur de choix int : <ul style="list-style-type: none"> <li>- MM_GAME -choix « jouer »</li> <li>- MM_AIDE -choix « aide »</li> <li>- MM_QUITTER -choix « quitter »</li> </ul>
Appelé par	Jeu_de_la_vie
Appelle	Aucun

***Affichage\_aide***

Déclaration	Def Affichage_aide()
Rôle	Affiche l'aide
Paramètres	Aucun
Retour	Aucun
Appelé par	Jeu_de_la_vie
Appelle	Aucun

***Affichage\_au\_revoir***

Déclaration	Def Affichage_au_revoir()
Rôle	Affiche un message de fin
Paramètres	Aucun
Retour	Aucun
Appelé par	Jeu_de_la_vie
Appelle	Aucun

***Jeu\_de\_la\_vie***

Déclaration	Def Jeu_de_la_vie ()
Rôle	Programme principal
Paramètres	Aucun
Retour	Aucun
Appelé par	Aucun
Appelle	Affichage_aide, Affichage_au_revoir, Joueur, Menu_principal

***Affichage\_joueur***

Déclaration	Def Affichage_joueur ()
Rôle	Menu de choix des règles (par défaut/personnalisées)
Paramètres	Aucun
Retour	Surv, Naiss, Isol, Etouf, Motif
Appelé par	Joueur
Appelle	Aucun

***Joueur***

Déclaration	Def Joueur ()
Rôle	Une partie du jeu de la vie
Paramètres	Aucun
Retour	Aucun
Appelé par	Jeu_de_la_vie
Appelle	Affichage_joueur, Evolution, Crea_matrice, Affichage_matrice, Placement

**Crea\_matrice**

Déclaration	Def Crea_matrice (longueur, largeur)
Rôle	Crée une matrice à deux dimensions avec un nombre de lignes et de colonnes qui peut varier
Paramètres	Une valeur pour le nombre de lignes (longueur) et le nombre de colonnes (largeur)
Retour	Une matrice M
Appelé par	Joueur
Appelle	Personne

**Affichage\_matrice**

Déclaration	Def Affichage_matrice (M, motif)
Rôle	Affiche la matrice
Paramètres	Une matrice M, motif une liste de motifs (photos) choisis par l'utilisateur pour représenter les cellules vivantes
Retour	Aucun
Appelé par	Evolution
Appelle	Motif

**Affichages/saisies propres au jeu de la vie****Evolution**

Déclaration	Def Evolution(M, Naiss, Isol, Etouf, Surv)
Rôle	Modifie la matrice pour tuer/faire naître des cellules tant que l'on n'a pas atteint le nombre de générations maximales
Paramètres	Une matrice M, le seuil à partir de laquelle une cellule meurt Isol, le seuil à partir de laquelle une cellule meurt Etouf, le seuil à partir de laquelle une cellule naît Naiss, le seuil à partir de laquelle une cellule survie Surv
Retour	M
Appelé par	Personne
Appelle	Affichage_matrice, Cpt_voisin, Mort_isolement, Mort_etouffement, Survie, Naissance

**Motif**

Déclaration	Def Motif (canevas, xcell, ycell, motif)
-------------	--

Rôle	Écrit dans le canevas : place une image choisie par l'utilisateur à la place d'une cellule vivante
Paramètres	Le canevas, xcell et ycell les coordonnées où l'image doit être placée (l'image = une cellule), motif une liste de motif(s) (photos) choisi(s) par l'utilisateur
Retour	Aucun
Appelé par	Affichage_matrice
Appelle	Affichage_matrice, Compt_voisin, Mort_isolement, Mort_etouffement, Survie, Naissance

***Mort\_isolement***

Déclaration	Def Mort_isolement(M,cpt_voisin, Isol)
Rôle	Dire si la cellule meurt par isolement
Paramètres	Une matrice M, un compteur de voisin cpt_voisin, le seuil à partir de laquelle une cellule meurt Isol
Retour	True ou False
Appelé par	Evolution
Appelle	Personne

***Mort\_etouffement***

Déclaration	Def Mort_étouffement(M, cpt_voisin, Etouf)
Rôle	Dire si la cellule meurt par étouffement
Paramètres	Une matrice M, un compteur de voisin cpt_voisin, le seuil à partir de laquelle une cellule meurt Etouf
Retour	True ou False
Appelé par	Evolution
Appelle	Personne

***Survie***

Déclaration	Def Survie(M,cpt_voisin, Surv)
Rôle	Dire si la cellule survie
Paramètres	Une matrice M, un compteur de voisin cpt_voisin, le seuil à partir de laquelle une cellule survie Surv
Retour	True ou False
Appelé par	Evolution
Appelle	Personne

**Naissance**

Déclaration	Def Naissance(M,cpt_voisin, Naiss)
Rôle	Dire si une cellule naît ou si la case reste vide
Paramètres	Une matrice M, un compteur de voisin cpt_voisin, le seuil à partir de laquelle une cellule naît Naiss
Retour	True ou False
Appelé par	Evolution
Appelle	Personne

**Graphe des dépendances**