

# DOCUMENTATION UTILISATEUR

## Game Of Life

### Groupe 6

Date : Décembre 2025

Projet : Game Of Life

Misha DOPRE  
Gabriel BUTAUD  
Elliot CAZAJOUS REY

## Table des matières

1.	Introduction .....	2
1.1	Règles fondamentales .....	2
1.2	fonctionnalités bonus implémentées .....	2
2.	Installation .....	2
2.1	Méthode 2 : Via la release.....	2
2.2	Méthode 2 : Compilation manuelle (Terminal/Bash).....	3
2.3	Méthode 3 : Avec VSCode .....	4
3.	Modes de fonctionnement .....	4
3.1	Mode Console .....	4
3.2	Mode Graphique .....	4
4.	Création de grilles personnalisées .....	5
5.	Vérification avec les tests unitaires .....	5

## 1. Introduction

Le Jeu de la Vie est un automate cellulaire inventé par le mathématicien britannique John Conway. Pour fonctionner, il nécessite une configuration initiale, après cela l'évolution est entièrement déterminée automatiquement.

Le principe du jeu étant donc d'observer les motifs qui se déplacent créés par les cellules, génération après génération.

### 1.1 Règles fondamentales

Chaque cellule possède 8 voisines (verticales, horizontales et diagonales).

A chaque génération :

Etat actuel	Nombre de voisines vivantes	Nouvel état
Morte	Exactement 3	Vivante (naissance)
Vivante	2 ou 3	Vivante (survie)
Vivante	< 2	Morte (sous-population)
Vivante	> 3	Morte (surpopulation)

### 1.2 fonctionnalités bonus implémentées

- **Grille torique** : Les bords de la grille sont connectés. Une cellule au bord droit a pour voisine la cellule correspondante au bord gauche. Idem pour haut/bas.
- **Cellules obstacles** : Des cellules spéciales qui ne changent jamais d'état, permettant de créer des barrières ou des motifs fixes. Les cellules obstacles sont affichées en rouge.

## 2. Installation

### 2.1 Méthode 2 : Via la release

C'est la méthode la plus simple, elle ne nécessite pas de compilation.

Téléchargement :

1. Aller sur la page des releases : <https://github.com/ZiKorT/Game-Of-Life/releases>
2. Télécharger JeuDeLaVie.zip
3. Extraire le fichier ZIP téléchargé dans l'emplacement de votre choix

Lancement du jeu :

Fichier	Action
LauncherGraphique.bat	Ouvre la fenêtre graphique avec la grille <i>input.txt</i>
LauncherConsole.bat	Exécute n itérations et sauvegarde les résultats
LauncherTests.bat	Vérifie que le programme fonctionne correctement

Attention : Ne lancez pas les *.exe* directement, utilisez directement les fichiers *.bat* qui permettent une configuration automatique.

**Fichiers générés automatiquement :**

- *config.txt* : Sauvegarde la vitesse d'exécution entre les sessions
- *input.txt\_out/* : Dossier contenant les résultats du mode console

**2.2 Méthode 2 : Compilation manuelle (Terminal/Bash)****Etape 1 : Cloner le projet**

```
git clone https://github.com/ZiKorT/Game-Of-Life.git
cd Game-Of-Life
```

**Etape 2 : Créer le dossier de construction**

```
mkdir build
cd build
```

**Etape 3 : Configuration CMake**

```
cmake..
```

**Etape 4 : Compilation**

```
cmake --build . --config Release
```

- *--build .* : Construire dans le dossier actuel
- *--config Release* : Version optimisée

**Etape 5 : Exécution**

(Depuis le dossier build)

**Mode graphique**

```
.\Release\Game_Of_Life.exe . \input.txt 1
```

**Mode console**

.\Release\Game\_Of\_Life.exe . \input.txt

### Tests unitaires

.\Release\RunTests.exe

## 2.3 Méthode 3 : Avec VSCode

- Installer les extensions **C/C++** et **CMake Tools**
- File -> Open Folder -> Sélectionner Game-Of-Life
- Ctrl + Shift + P -> CMake: Configure
- Ctrl + Shift + P -> CMake: Build

## 3. Modes de fonctionnement

### 3.1 Mode Console

Le mode console exécute la simulation sans interface graphique et sauvegarde chaque état dans des fichiers.

**Via la release** : Double-cliquer sur *LauncherConsole.bat*

**Via ligne de commande** :

.\Release\Game\_Of\_Life.exe ..\input.txt 0

Résultat (à l'intérieur du dossier *input.txt\_out*) :

iter_0.txt	06/12/2025 16:27	Document texte	1 Ko
iter_1.txt	06/12/2025 16:27	Document texte	1 Ko
iter_2.txt	06/12/2025 16:27	Document texte	1 Ko
iter_3.txt	06/12/2025 16:27	Document texte	1 Ko
iter_4.txt	06/12/2025 16:27	Document texte	1 Ko
iter_5.txt	06/12/2025 16:27	Document texte	1 Ko
iter_6.txt	06/12/2025 16:27	Document texte	1 Ko
iter_7.txt	06/12/2025 16:27	Document texte	1 Ko
iter_8.txt	06/12/2025 16:27	Document texte	1 Ko
iter_9.txt	06/12/2025 16:27	Document texte	1 Ko
iter_10.txt	06/12/2025 16:27	Document texte	1 Ko

### 3.2 Mode Graphique

**Via la release** : Double-cliquer sur *LauncherGraphique.bat*

**Via ligne de commande** :

.\Release\Game\_Of\_Life.exe ..\input.txt 1

Contrôles clavier:

Touche	Action
Espace	Pause / Reprise
Flèche haut	Accélérer (-10ms par itération)
Flèche bas	Ralentir (+10ms par itération)

**Sauvegarde automatique** : La vitesse est sauvegardée dans *config.txt* à la fermeture.

#### 4. Création de grilles personnalisées

Format de fichier :

```
nombre_lignes> <nombre_colonnes>
<valeur_1_1> <valeur_1_2> ... <valeur_1_n>
<valeur_2_1> <valeur_2_2> ... <valeur_2_n>
...
```

Valeurs possibles :

Valeur	Type de cellule
0	Morte
1	Vivante
2	Obstacle (immuable)

Exemple de motifs :

Le Bloc (stable), ne change jamais :

```

4 4
0 0 0 0
0 1 1 0
0 1 1 0
0 0 0 0
```

Grille avec obstacles, barrière infranchissable :

```

7 7
0 0 0 0 0 0 0
0 1 1 1 0 0 0
0 0 0 0 0 0 0
2 2 2 2 2 2 2
0 0 0 0 0 0 0
0 0 0 1 1 1 0
0 0 0 0 0 0 0
```

Fonctionnalités avancées :

#### 5. Vérification avec les tests unitaires

Via la release :

Double-cliquer sur *LauncherUnitTests.bat*

**Via ligne de commande :**

.\Release\RunTests.exe

Sortie attendue :

```

LANCEMENT DES TESTS
[OK] Test Cell (Normal + Obstacle) valide.
[OK] Test FileManager (Lecture 0, 1, 2) valide.
[OK] Test Grid (Torique & Obstacles) valide.
[OK] Test Stabilit® (Block) valide.
TOUT EST VERT SONT VALIDE
```