

15 JANVIER 2021



BLOCUS

PT1.1 2020

ARTHUR DECORBEZ & SHANA LEFEVRE

DUT INFORMATIQUE

Groupe 4 et groupe 3

Table des matières

I] Introduction	2
II] Description et fonctionnalité du programme	2
A. Le mode SOLO	3
B. Le mode VS.....	3
III] Présentation de la structure du programme	4
IV] Explication des données : Exemple d'une partie en cours	6
A. En mode SOLO.....	6
B. En mode VS.....	11
C. Fin de la partie.....	16
V] Conclusion.....	17

I] Introduction

Nous avons réalisé ce projet en binôme pour mi-janvier 2021. Le but est de créer un jeu nommé “Le blocus” à l’aide de la bibliothèque graphique de notre IUT.

Les règles sont simples : ne pas se faire bloquer par le joueur adverse.

Pour cela, les joueurs, tour à tour peuvent déplacer leur pion dans un rayon d’une case puis mettre une croix dans la grille. Cette dernière est générée par l’utilisateur, il choisit la taille souhaitée allant de 3x3 à 9x9.

L’utilisateur peut jouer contre un autre utilisateur ou bien contre un bot.

II] Description et fonctionnalité du programme

Notre jeu reçoit l’utilisateur sur sa page d’accueil. Ce dernier configure la taille de la grille allant de 3x3 à 9x9 puis il définit le mode de jeu : SOLO ou VS.

Pour les deux modes de jeu, on retrouvera ces deux personnages :



Joueur 1 : Wall-e.



Joueur 2 : Eve.

Pour bloquer l’adversaire, les joueurs utiliseront des croix :



Croix de Wall-e.



Croix de Eve.

A. Le mode SOLO

Le joueur 1 (représenté par Wall-e) initialise sa position avant le début de la partie. Le joueur 2 qui est le bot (représenté par Eve) est déjà initialiser.

Pour commencer la partie, le joueur 1 doit faire sa première action : déplacer Wall-e dans une case adjacente qui est son périmètre autorisé, c'est-à-dire, une case en 1 (soit en position x et ou soit en position y).

Puis c'est au tour du bot Eve. On répète ces actions successivement.

La partie se finit dès que l'un des deux joueurs est bloqué.

B. Le mode VS

Le joueur 1 est représenté par Wall-e et le joueur 2 est représenté par Eve. Chaque joueur initialise respectivement son « pion ».

Dès cette étape accomplie, le jeu commence. Le joueur 1 (Wall-e) commence par se déplacer dans une case adjacente qui sera son périmètre autorisé (soit en position x et ou soit en position y) et il place une croix dans la grille. Une fois fait, c'est au tour du joueur 2 de se déplacer dans son périmètre autorisé et d'y placer sa croix. On répète ces actions successivement.

Le jeu se termine quand l'un des deux joueurs est bloqué.

III] Présentation de la structure du programme

Pour la réalisation de notre jeu, nous avons créé plusieurs fonctions.

Tout d'abord la fonction **menu()**. Cette fonction nous permet d'initialiser le menu et de récupérer les données de l'utilisateur : la taille de la grille et le mode de jeu soit le nombre de joueur (SOLO ou VS).

Pour créer et dessiner la grille, nous avons opté pour la création d'une fonction **DessinerGrille()**.

Afin que le fonctionnement de notre jeu soit clair, nous avons créé deux fonctions **RegleJeu_VS()** et **RegleJeu_Solo()** qui respectivement expliquent « comment jouer » et bien utiliser notre programme.

Après que l'utilisateur ait entré le nombre de joueur, si le nombre de joueur est un, alors notre programme appelle la fonction **Bloquer_Solo()** sinon, si le nombre de joueur est deux, alors il appelle la fonction **Bloquer_Vs()**.

La fonction **Bloquer_Solo()** est utilisé tout au long du jeu. Elle permet l'initialisation des deux pions, le jeu, et la vérification des cases. Pour alterner le tour, nous utilisons l'incrémentation d'une variable « number » et des conditions if.

Pour gérer les déplacements de Wall-e (utilisateur joueur 1), nous associons à notre grille un tableau. Pour cela nous avons procédé de la façon suivante :

`_X = 170 et _Y = 150` (`_X` position x cliquée et `_Y` position y cliquée)

Dans notre programme, une case de notre grille est de 80 et 80. Cette valeur est réglable par l'utilisateur : `#define TAILLE_CASE`

Pour récupérer l'indice de notre tableau par rapport à la position cliquée par l'utilisateur, nous gardons la partie entière.

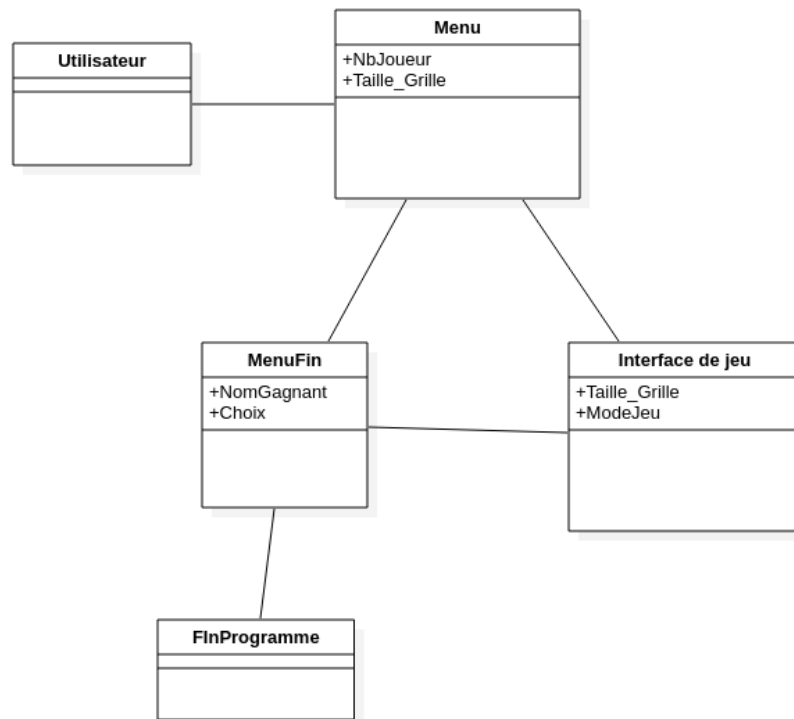
$$\text{Ex : } \frac{_X}{\text{TAILLE_CASE}} = \frac{170}{80} = 2,125 \qquad \frac{_Y}{\text{TAILLE_CASE}} = \frac{150}{80} = 1,875$$

Donc les indices correspondant seront `tab [2][1]`. Nous mettons cette case à 1 pour dire qu'elle est prise.

Pour gérer les déplacements aléatoires de Eve (bot), nous avons opté pour un « random » et des conditions if. Ainsi, cela permet de savoir si posX et posY augmentent ou diminuent d'un, ou bien s'ils gardent la même position. Bien entendu, le pion ne peut pas rester dans la même case.

La fonction **Bloquer_Vs()** est elle aussi utilisé tout au long du jeu si le nombre de joueur est de deux. Elle s'effectue comme la fonction **Bloquer_Vs()** sauf qu'elle n'utilise pas le *random* car les deux joueurs sont des utilisateurs.

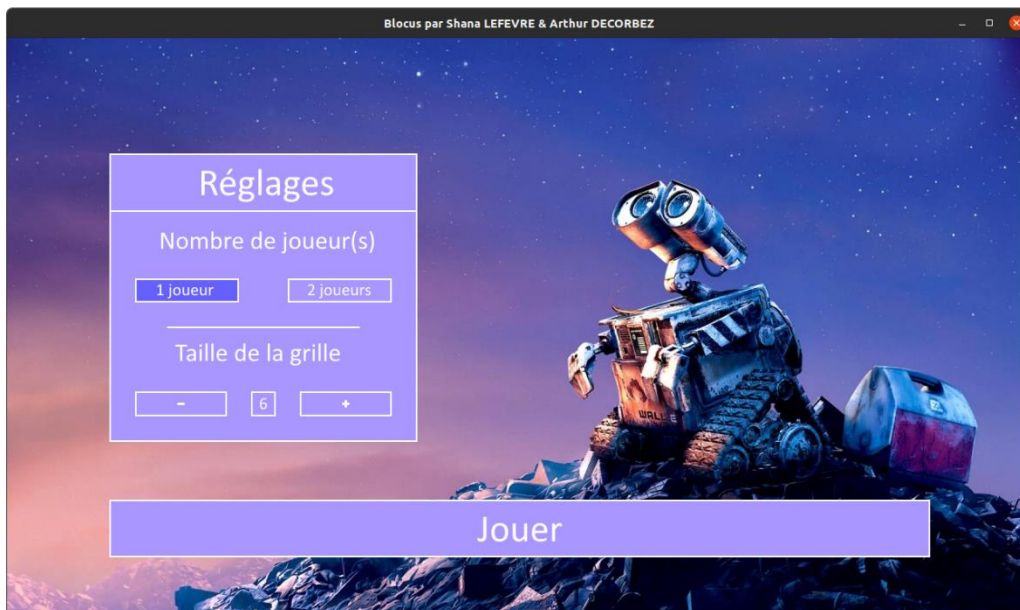
Diagramme de classe du jeu :



IV] Explication des données : Exemple d'une partie en cours

Afin d'être sûr que notre programme fonctionne comme il le faut ou pour bien gérer les problèmes rencontrés, nous utilisons des *printf* pour afficher les différents statuts de notre jeu.

Quand l'utilisateur démarre le jeu, il tombe sur cette fenêtre d'accueil :



Capture d'écran : Fenêtre d'accueil.

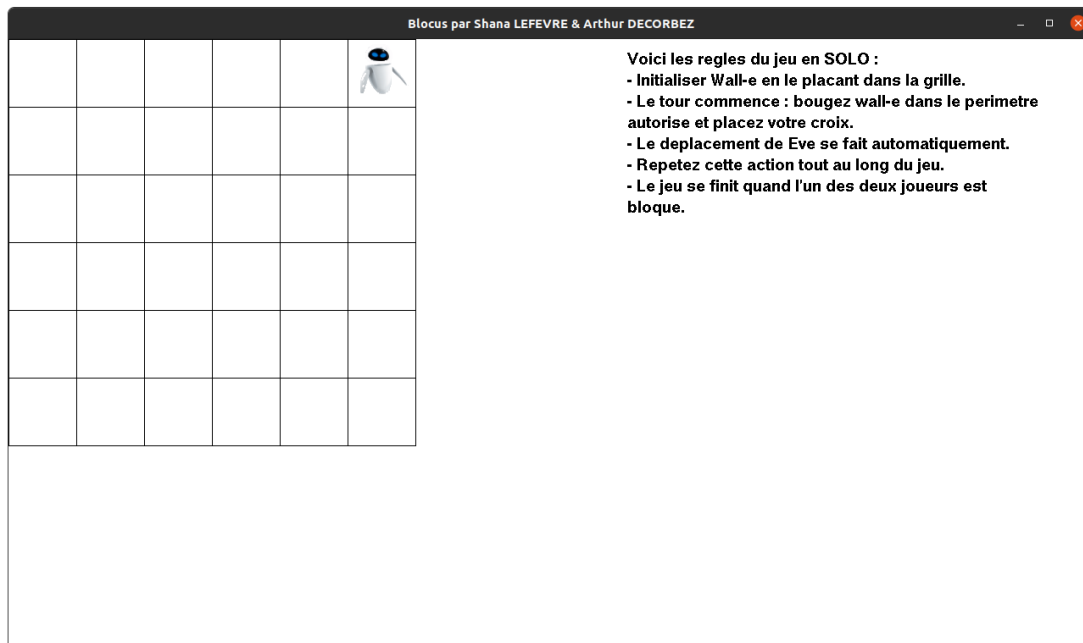
Grace à cette fenêtre, l'utilisateur peut choisir la taille de sa grille ainsi que le nombre de joueur.

A. En mode SOLO

Si l'utilisateur choisit le mode solo, alors on aura cette fenêtre avec la grille de jeu définie dans la précédente fenêtre. Le pion Eve (bot) est initialisé lors de l'initialisation de la grille.

Dans cet exemple, on choisit une grille de taille 6x6.

En mode solo, l'initialisation des coordonnées de Eve (bot) se fait automatiquement lors du lancement du jeu. On voit sur la grille que Eve est initialisée en [5][0].



Capture d'écran : Plateau de jeu.

Nous avons ces données qui apparaissent dans le terminal : le nombre de joueur (*int*), la taille de la grille (*int*) et le mode de jeu (*int*) donc l'appel de la fonction voulue (soit **Bloquer_Vs()** soit **Bloquer_Solo()**). Les informations du terminal nous permettent de vérifier que tout s'est bien passé.

On initialise toutes les cases à 0. Le terminal nous indique que Eve est placé en [5][0].

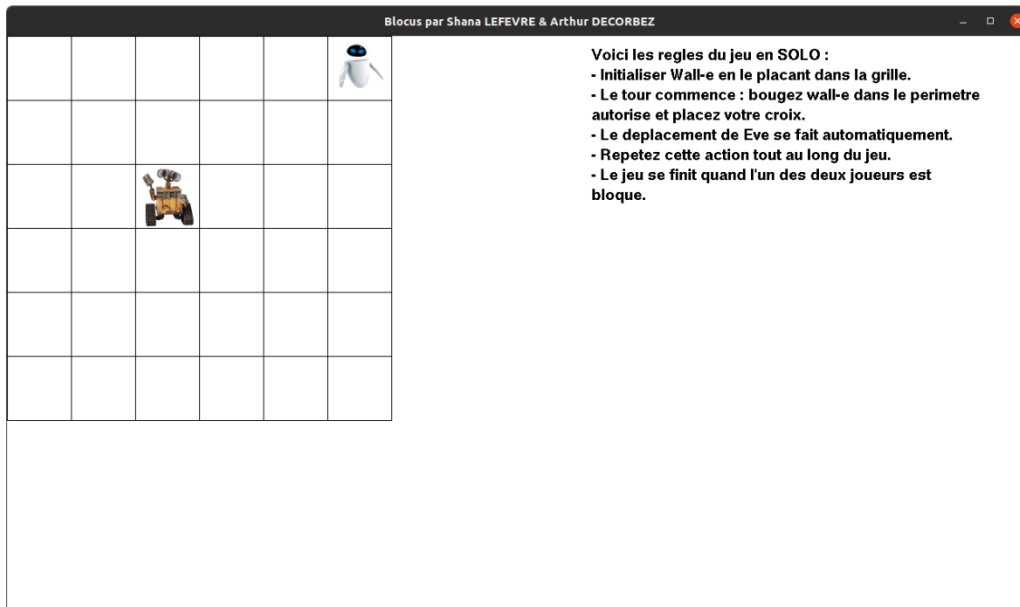
```

Nombre de joueur : 1
Taille de la grille : 6
Initialisation des grilles :
[0][0] = 0
[0][1] = 0
[0][2] = 0
[0][3] = 0
[0][4] = 0
[0][5] = 0
[1][0] = 0
[1][1] = 0
[1][2] = 0
[1][3] = 0
[1][4] = 0
[1][5] = 0
[2][0] = 0
[2][1] = 0
[2][2] = 0
[2][3] = 0
[2][4] = 0
[2][5] = 0
[3][0] = 0
[3][1] = 0
[3][2] = 0
[3][3] = 0
[3][4] = 0
[3][5] = 0
[4][0] = 0
[4][1] = 0
[4][2] = 0
[4][3] = 0
[4][4] = 0
[4][5] = 0
[5][0] = 0
[5][1] = 0
[5][2] = 0
[5][3] = 0
[5][4] = 0
[5][5] = 0
LANCEMENT DU JEU MODE 1 JOUEUR
Coordonnées Joueur Rouge [5][0]

```

Capture d'écran terminal : initialisation de la grille et du pion Eve (bot).

Puis la grille et le pion Eve initialisé, nous décidons de placer notre pion (Wall-e) en [2][2].



Capture d'écran : Plateau de jeu après l'initialisation de Wall-e.

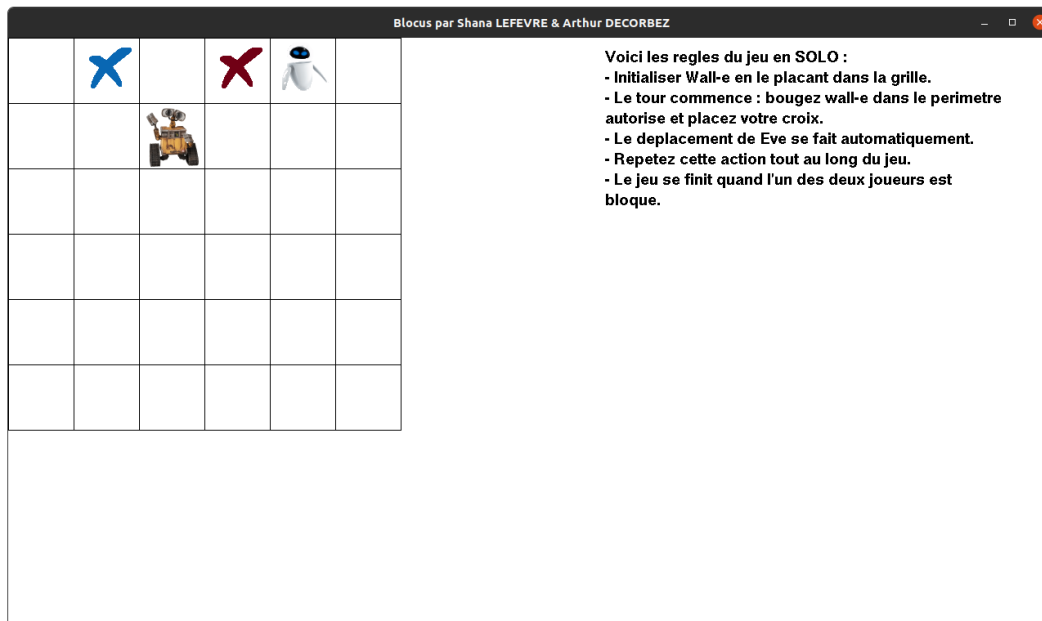
Nous vérifions que tout s'est bien passé. Le terminal nous affiche :

```
Position cliquée [2] [2]  
Initialisation coordonnées joueur B [2][2] OK  
█
```

Capture d'écran terminal : Confirmation de l'initialisation de Wall-e.

Après l'initialisation des pions, la partie commence. C'est le joueur 1 (l'utilisateur Wall-e) qui commence : il se déplace dans son périmètre autorisé, et il place sa croix.

On décide de déplacer de -1y et 0x Wall-e et de mettre la croix en position [3][0]. Dès cette action faite, c'est au tour de Eve. Elle a bougé de -1x et 0y. Elle place sa croix en [1][0].



Capture d'écran : Plateau de jeu de la première action.

Les données du terminal ci-dessous représentent :

- Le «n° clique » qui est le numéro de tour, ce qui permet d'alterner les déplacements entre le joueur 1 et le bot ainsi que les différentes croix,
- La position cliquée par l'utilisateur,
- Les états de la case cliquée,
- Le stockage des coordonnées : gestion des valeurs du tableau.

```
N° clique 2
Position cliquée [2] [1]
Anciennes coordonnées joueur B [2][2] OK
Reinitialisation de la case de Wall-e est ok
Nouvelle position du joueurB [2][1]
Position stocké [2][1]
N° clique 3
Clique croixB en [3][0]
N° clique 4
Anciennes coordonnées R [5][0]
Réinitialisation de la case de Eve est ok
N° clique 5
CroixR en [1][0]
□
```

Capture d'écran terminal : Donnée du premier coup.

Après une succession d'actions, nous avons ce jeu. C'est à Wall-e de mettre sa croix. Nous déciderons que sa croix sera en position [5][0].



Capture d'écran : Plateau de jeu de la dernière action.

Toutes les données sont affichées dans le terminal, ainsi, nous avons un historique des déplacements des pions et la position des croix.

```

CroixR en [2][0]
N° clique 6
Position cliquée [3] [2]
  Anciennes coordonnées joueur B [2][1] OK
  Réinitialisation de la case de Wall-e est ok
  Nouvelle position du joueurB [3][2]
  Position stocké [3][2]
N° clique 7
  Cliquez croixB en [3][1]
N° clique 8
  Anciennes coordonnées R [4][0]
  Réinitialisation de la case de Eve est ok
N° clique 9
  CroixR en [0][0]
N° clique 10
Position cliquée [4] [2]
  Anciennes coordonnées joueur B [3][2] OK
  Réinitialisation de la case de Wall-e est ok
  Nouvelle position du joueurB [4][2]
  Position stocké [4][2]
N° clique 11
  Cliquez croixB en [5][1]
N° clique 12
  Anciennes coordonnées R [5][0]
  Réinitialisation de la case de Eve est ok
N° clique 13
  CroixR en [2][0]
N° clique 14
Position cliquée [4] [1]
  Anciennes coordonnées joueur B [4][2] OK
  Réinitialisation de la case de Wall-e est ok
  Nouvelle position du joueurB [4][1]
  Position stocké [4][1]

```

Capture d'écran terminal : Successions des actions durant la partie.

Après d'avoir placé notre dernière croix, Wall-e a réussi à bloquer Eve (bot).



Capture d'écran : Fenêtre de victoire.

Dès qu'un joueur a perdu, le terminal nous indique « comment il a perdu » ainsi que les informations de la partie : mode de jeu, le nom du vainqueur.

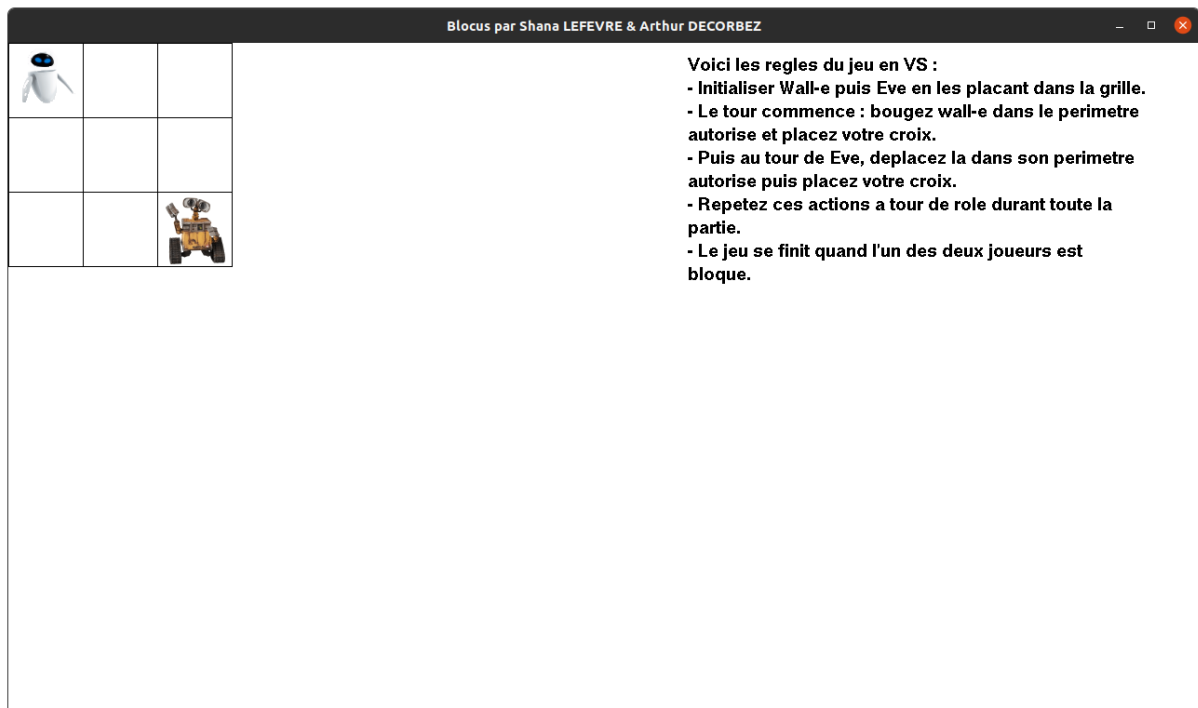
```
Position stocke [4][1]
N° clique 15
Clique croixB en [5][0]
Le joueur R a perdu : blocage haut
valGagnant : 1
modeJeu : 1
Wall-e a vaincu Eve !
█
```

Capture d'écran terminal : Fin du jeu.

B. En mode VS

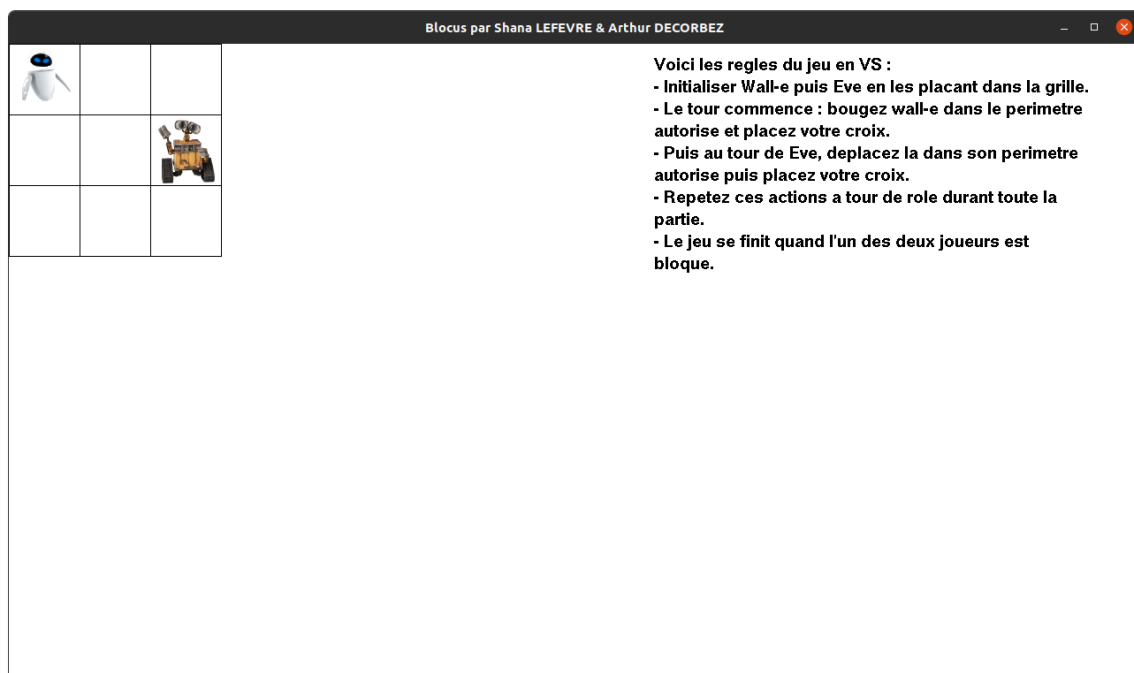
Si l'utilisateur choisit le mode solo, alors on aura cette fenêtre avec la taille de grille définit sur la précédente fenêtre.

On choisit de jouer sur une grille de taille 3x3.



Capture d'écran : Initialisation du pion Wall-e et Eve.

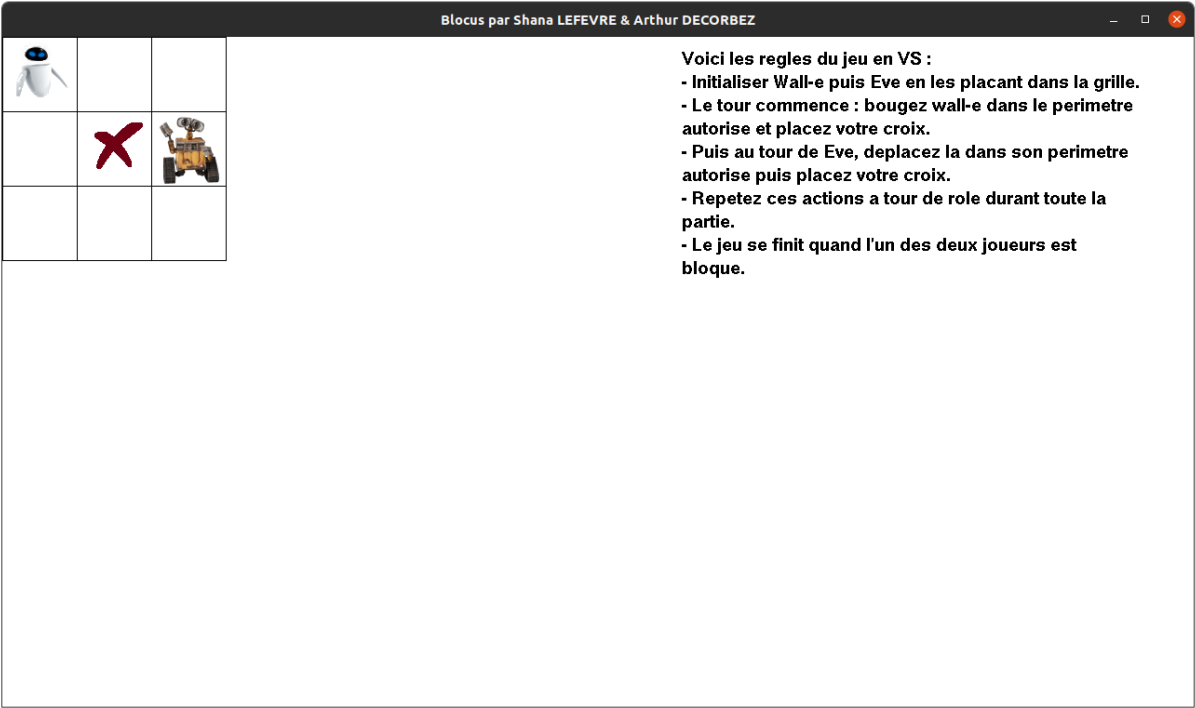
Après d'avoir initialiser les deux pions, Wall-e commence son tour : soit bouger d'une case adjacente et il place sa croix.



Capture d'écran : premier coup de Wall-e.

```
Position cliquée [2] [1] = 0
N° clique 1
Anciennes coordonnées joueur B [2][2] OK
Réinitialisation de la case de Wall-e
Nouvelle position du joueurB [2][1]
Position stocké [2][1]
```

Capture d'écran terminal : Déplacement du pion Wall-e.

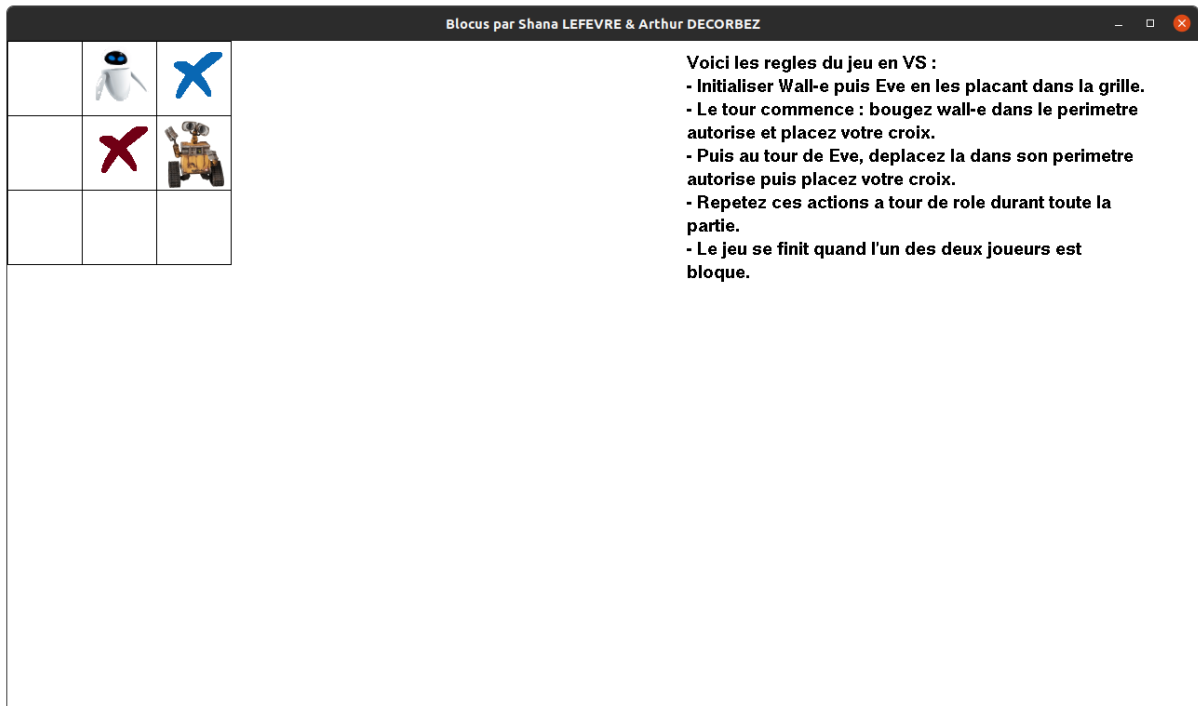


Capture d'écran : Placement de la croix de Wall-e.

```
Position cliquée [1] [1] = 0
N° clique 2
Clique croixB en [1][1]
```

Capture d'écran terminal : Coordonnées de la croix par le joueur 1 Wall-e.

Puis voilà le tour de Eve : déplacement d'une case adjacente et elle place sa croix.



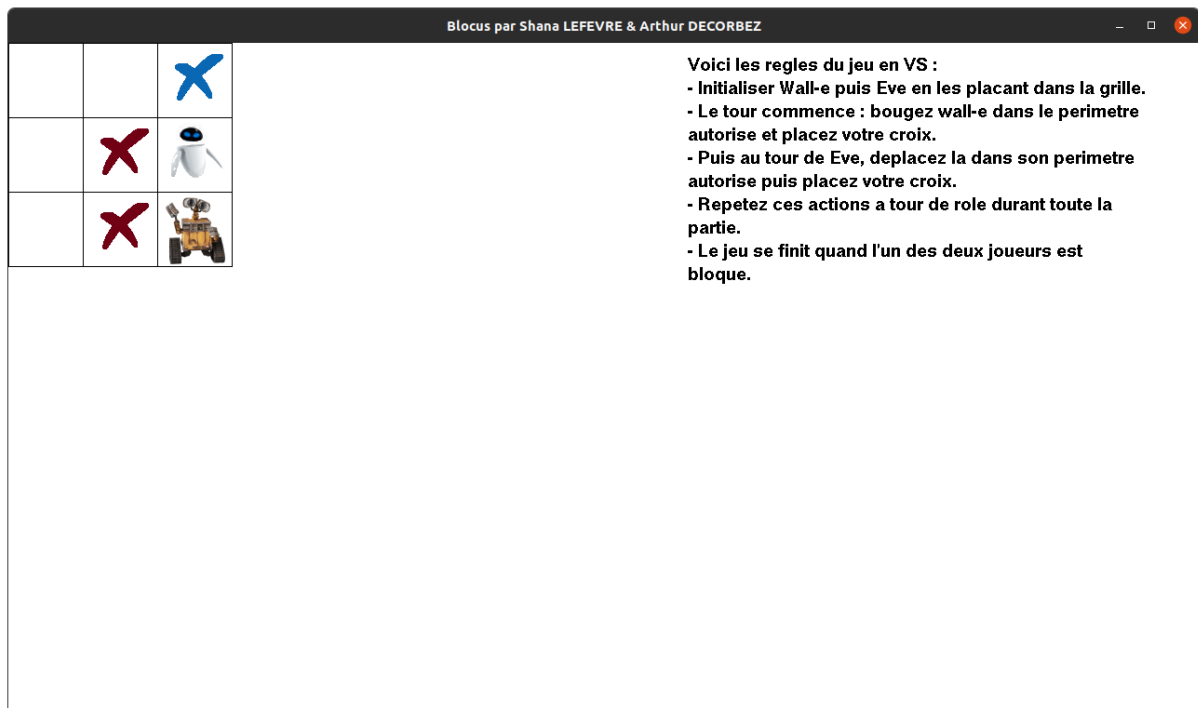
Capture d'écran : premier coup de Eve (déplacement et placement de sa croix).

```
Position cliquée [1] [0] = 0
N° clique 3
Anciennes coordonnées R [0][0]
Réinitialisation de la case de Eve
Clique JoueurR [1][0]
Position stockée [1][0]
Position cliquée [2] [0] = 0
N° clique 4 Clique CroixR en [2][0]
```

Capture d'écran terminal : Coordonnées du pion de Eve et de sa croix. Ici représenté par « joueurR » et « CroixR ».

On répète les tours jusqu'à temps qu'un joueur est bloqué.

Dans la fenêtre ci-dessous, c'est à Eve de placer sa croix. On décide de placer sa croix en [1][0].



Capture d'écran : Dernier tour.

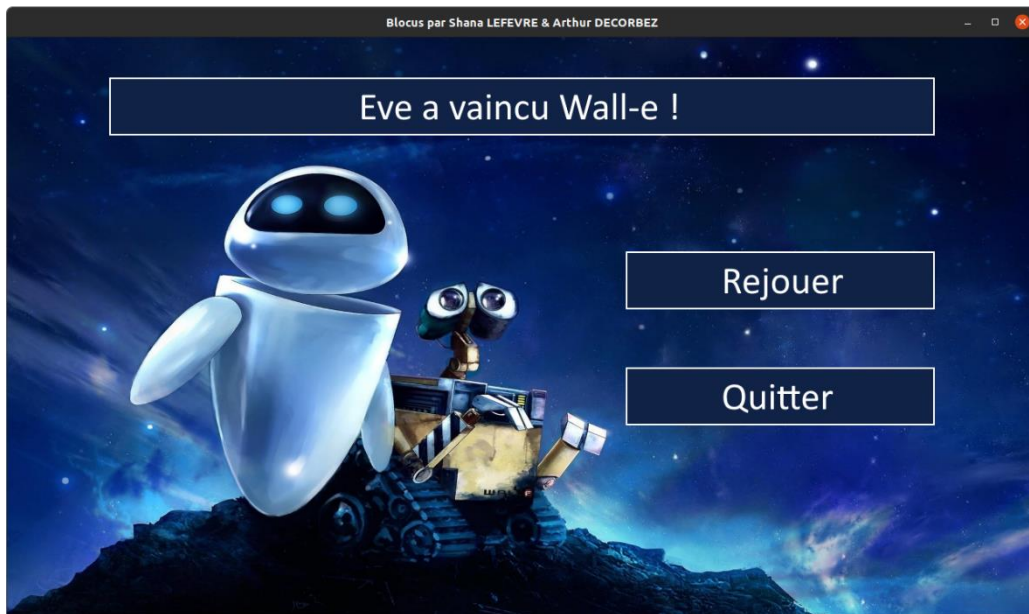
```
N° clique 7
Anciennes coordonnées R [1][0]
Réinitialisation de la case de Eve
Clique JoueurR [2][1]
Position stockée [2][1]
Position cliquée [1] [0] = 0
N° clique 8 Cliquez CroixR en [1][0]
```

Capture d'écran terminal : Coordonnées de Eve et placement de sa croix

Donc, Wall-e est bloqué :

```
N° clique 8 Cliquez CroixR en [1][0]
Le joueur B a perdu : blocage bas droite
valGagnant : 2
modeJeu : 2
Eve a vaincu Wall-e!
```

Capture d'écran terminal : Fin de partie



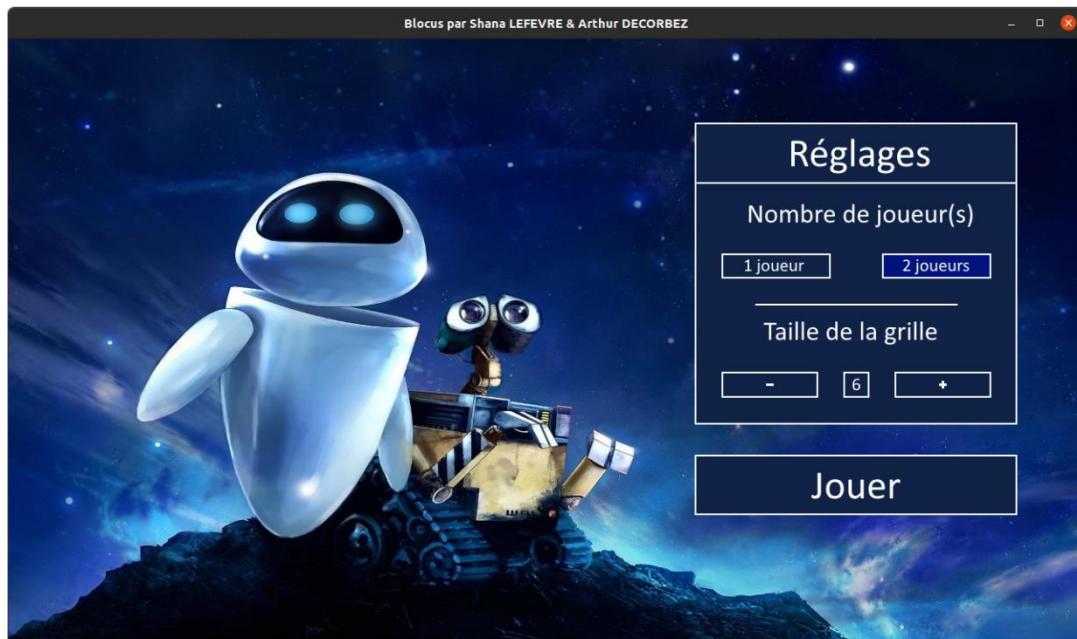
Capture d'écran : Fenêtre de fin.

C. Fin de la partie

Dès qu'un joueur est bloqué, le terminal nous renvoie sur une autre fenêtre qui nous indique le nom du joueur qui a gagné et qui demande aux utilisateurs ce qu'ils veulent faire : rejouer ou bien quitter le programme.

Si l'utilisateur clique sur rejouer, cela l'amène à la fenêtre ci-dessous. Sinon, il quitte le programme et ce dernier se ferme.

Sur la fenêtre ci-dessous, l'utilisateur peut reconfigurer tous les paramètres désirés pour sa prochaine partie, soit le nombre de joueur et la taille de la grille. Le programme est "relancé", c'est le même fonctionnement, le même principe d'utilisation de variables, la même logique.



Capture d'écran : Fenêtre "Rejouer".

V] Conclusion

Arthur :

"Honnêtement, ce projet a été celui que j'ai préféré faire, que ce soit au niveau du cahier des charges ou de mon binôme, j'ai beaucoup aimé travailler sur ce projet, le fait d'avoir déjà un peu d'expérience avec le C aide aussi.

Je suis plutôt fière de ce que nous avons produit, bien que j'aie quelques regrets. Le fait de ne pas avoir pu terminer le jeu à 100% et de ne pas voir put ajouter les features auxquels j'avais pensé, me rend un peu triste, cependant les connaissances ainsi que l'expérience que j'ai acquises durant ce projet me fait relativiser.

Je remercie mon binôme, Shana LEFEVRE, qui m'a supporté durant l'entièreté du projet et qui a également travaillé d'arrache-pied pour pouvoir rendre ce travail."

Shana :

"Ce projet a été une très bonne expérience pour moi. En effet, il m'a permis d'aborder le problème d'une autre façon : dérouler l'algorithme et le problème sous forme papier et fichier texte. Par rapport à mon premier projet en C, j'ai pu découvrir ma capacité à résoudre un problème et coder notre projet. En ce qui concerne le travail en groupe, nous avons réussi à nous répartir nos tâches et ainsi, à partager nos points forts. Je pense, que cela a été très bénéfique pour nous. J'ai beaucoup appris et cela me permettra d'aborder d'autres projets sous un autre œil."