

Compte-Rendu

Du projet d'APL

Par Chloé Trugeon et Mathilde Charrière

Dans le cadre de notre première année en DUT informatique, nous avons pu réaliser un jeu "Blocus" en langage C. Le [sujet](#) consistait à coder un jeu dans lequel l'utilisateur pourrait choisir la taille de la grille et le mode de jeu (un joueur contre l'ordinateur ou deux joueurs) et dont le but serait de bloquer son adversaire en condamnant toutes les cases de la grille autour de lui pour l'empêcher de se déplacer, les déplacements étant seulement d'une case par tour (diagonales comprises). Sur l'écran de fin, le gagnant est affiché, et propose au joueur de rejouer ou non.

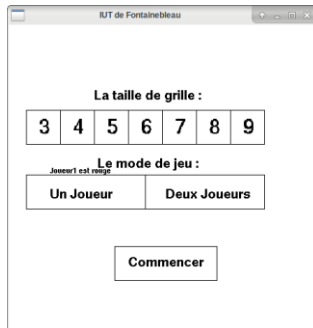
Nous allons donc écrire et illustrer tout ce qui nous semble important sur ce projet dans ce rapport.

Page 1: Introduction et Sommaire

Pages 2: Fonctionnalités

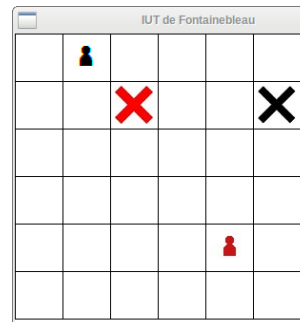
Page X: Conclusion Personnelle

Fonctionnalités

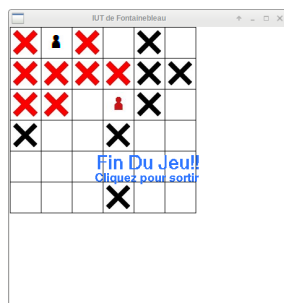


C'est dans cette fenêtre qu'on choisit la taille de grille (de 3 à 9 cases), le mode de jeu (seul contre ORDI, ou deux joueurs). Pour faire sa sélection, il suffit de cliquer sur un des rectangles, et le jeu ne se lance que si une taille et un mode de jeu à été sélectionné.

Ici, le jeu a commencé. Le joueur joue lorsque c'est son tour, se déplace sur une case adjacente, et condamne une case libre du plateau. Le jeu continuera jusqu'à ce qu'un des joueurs à toutes ses cases alentours occupées



Ici, la partie est finie, le pion noir n'est pas capable de déplacer. Un message apparaît donc au centre de la fenêtre pour prévenir le(s) joueur(s) que plus personne ne peut bouger. Il suffit de cliquer une nouvelle fois pour accéder à la suite proposée.



Voici l'écran final, où est affiché le gagnant. On propose aussi aux utilisateurs la possibilité de rejouer une autre partie. La réponse se fait de la même manière qu'au premier écran



Structure :

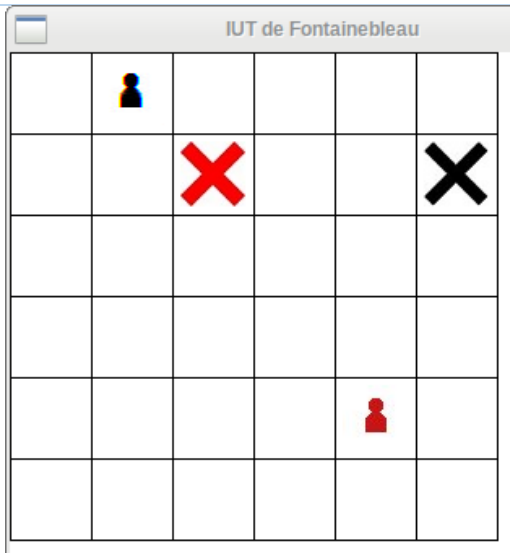
Pour coder ce projet, nous avons d'abord réfléchi par écrit aux différentes fonctions qui nous seraient nécessaires et la façon dont nous pourrions les coder. Cela nous a permis de ne pas foncer et taper des lignes de codes à la suite n'ayant pas vraiment de sens ou de liens entre elle.

Nous avons d'abord commencé par créer un seul fichier, dans lequel nous déclarions toutes les fonctions que nous créions au dessus du main afin de pouvoir les tester. Puis nous les avons séparées du fichier principal pour créer le Makefile. Le programme se présente donc en six fichiers sources plus le fichier main contenant la fonction principale. Les dépendances sont les suivantes :

```
affichage.c : affichage.h  
Deplacement.c : grille.h affichejeu.h deplacement.h  
Affichejeu.c : affichejeu.h  
Finjeu.c finjeu.h  
Grille.c deplacement.h affichejeu.h grille.h  
Selection.c affichage.h selection.h
```

Dans le fichiers affichage.c nous avons regroupé toutes les fonctions relative au design du jeu, tout ce que l'utilisateur peut voir directement, comme le choix de la taille et des modes ou l'affichage du gagnant par exemple. Dans affichejeu.c, il y a également ce que l'utilisateur peut voir à l'écran mais qui appartient au jeu en lui-même, à une partie en cours (le déplacement des joueurs ou l'affichage des croix par exemple). La fichier selection.c regroupe les fonctions correspondant aux choix de l'utilisateur comme le mode ou s'il souhaite ou non refaire une partie. Quant au fichier déplacement.c, il contient les fonctions de déplacements et de joueurs et de l'ordinateur ainsi que la fonction vérifiant qu'il est possible de faire ce déplacement. Nous avons choisis de mettre les fonctions qui vérifient si le joueur est bloqué dans le fichier finjeu.c étant donné qu'un joueur bloqué arrête la partie. Enfin, dans le fichier grille.c nous avons regroupé les fonctions relative à l'état de la grille qui permet de savoir si l'état de chaque case et également de localiser la case dans laquelle se trouve le joueur.

Les données lors d'une partie



Voilà comment représenter les états des cases:

```

0 2 0 0 0 0
0 0 1 0 0 1
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 2 0
0 0 0 0 0 0

```

On a donc la place des joueurs représenté par un 2, celles des croix par un 1 et les libres par 0.

Quand un pion est déplacé, son ancienne place est donc libre, donc le 2 devient 0.

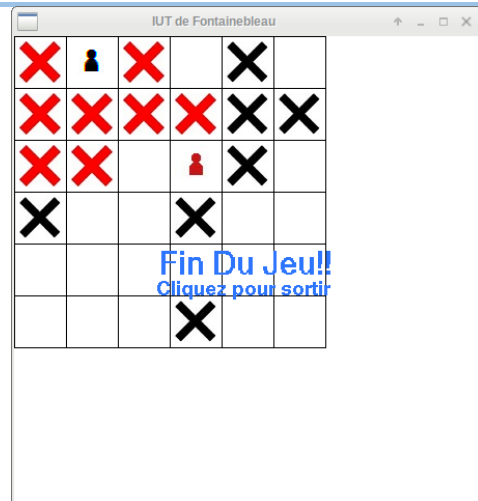
Ici les cases donnent:

```

1 2 1 0 1 0
1 1 1 1 1 1
1 1 0 2 1 0
1 0 0 1 0 0
0 0 0 0 0 0
0 0 0 1 0 0

```

On voit que dans la grille chiffrée, le pion noir est bloqué, ce qui provoque la fin de partie.



Conclusions :

Mathilde: Ce projet m'a permis beaucoup de choses, au niveau de mon organisation et de mes connaissances. Lors de mon ancien projet (pour le Bac), mon groupe avait foncé tête baissée, et on se retrouvait à corriger partout des erreurs répétitives. C'est quelque chose que Chloé et moi avons évité, et nous avons donc codé assez vite. Ce projet m'a aussi permis de comprendre des points clés du cours, notamment les pointeurs, les tableaux et l'utilisation de la bibliothèque graphique. Je suis donc assez satisfaite du travail que nous avons fourni pour l'aboutissement de ce projet.

Sur ce projet: Je pense que ce jeu n'était pas si compliqué à faire,

Chloé : La réalisation de ce projet m'a beaucoup appris et m'a énormément aidée dans la compréhension des notions vues précédemment en cours. Je pensais maîtriser correctement des outils comme les pointeurs ou l'écriture du Makefile par exemple, mais en travaillant sur ce jeu, je me suis rendue compte que ce n'était pas totalement le cas. Grâce à ce projet, j'ai pu appréhender d'une autre manière ces outils et comprendre leur fonctionnement de manière plus précise. J'ai également compris l'importance de la pratique dans l'apprentissage des langages informatiques. Je n'aurais pas été capable de réaliser ce projet seule, et j'ai aimé travailler en binôme, je pense que cela est plus motivant et rassurant. De plus, je trouve que Mathilde et moi formions un bon groupe, que nous nous complétions bien en fonction des choses que chacune avaient comprises et expliquaient à l'autre. Le rendu de notre projet n'est pas parfait, mais je suis plutôt satisfaite de notre travail, nous avons fait de notre mieux pour essayer de réaliser un jeu fonctionnel avec un code lisible.