

Projet du Blocus

Table des matières

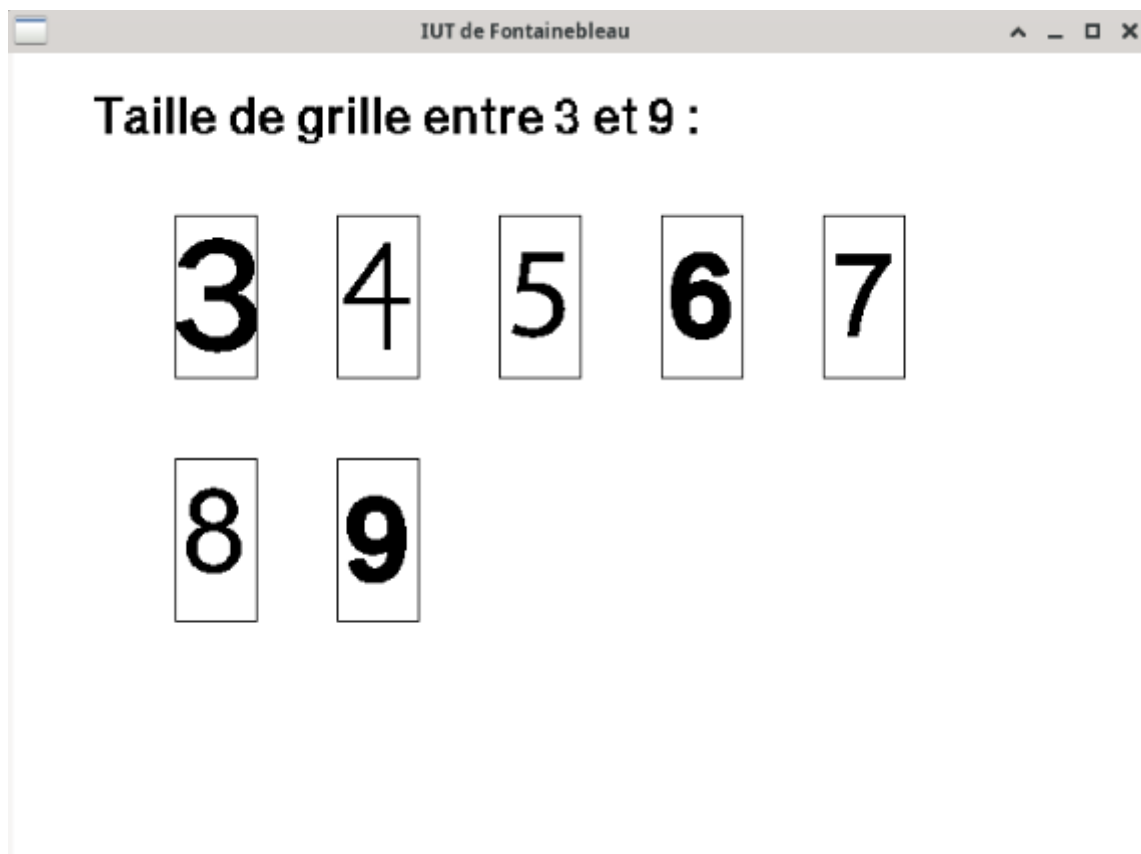
3-4	Présentation du projet
5-11	Présentation des fonctionnalités du programme
12-16	Le Makefile (Structure du programme)
17-18	Explication des données d'une partie
19-20	Conclusions des auteurs

1. Présentation du projet

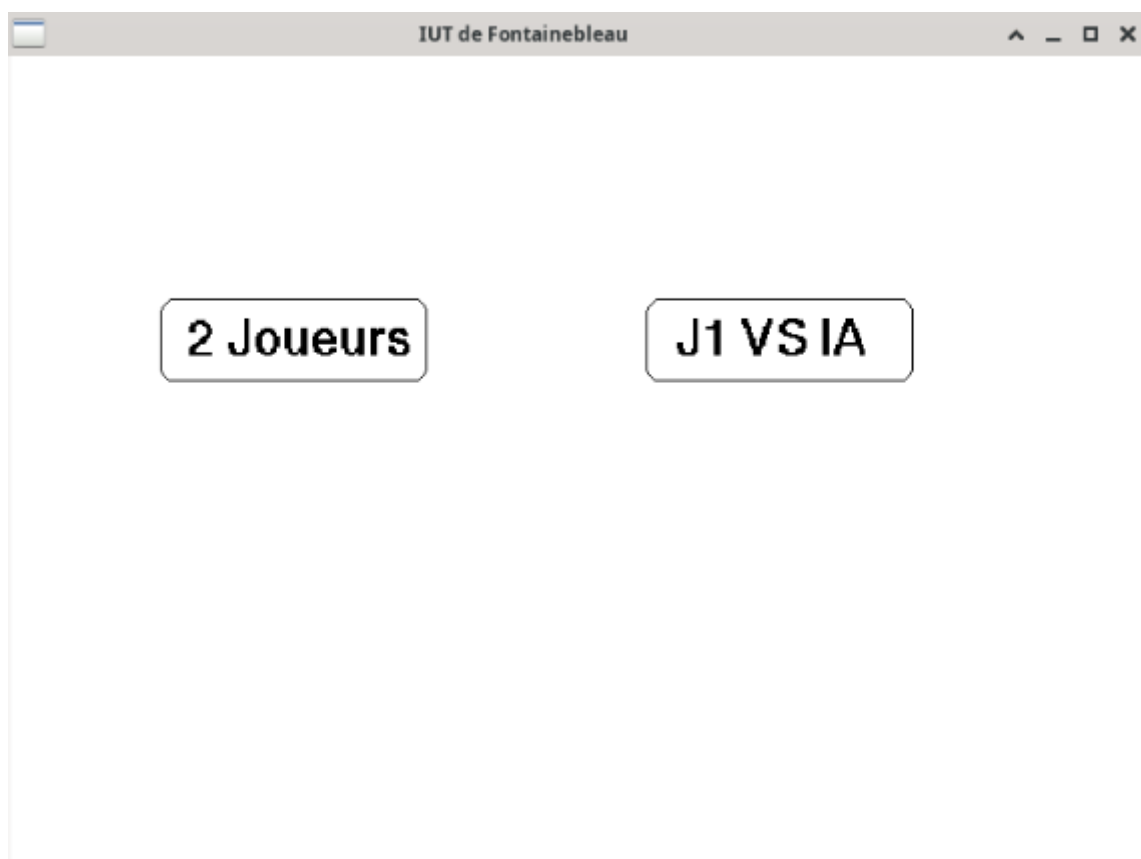
Le projet du Blocus est un jeu réalisé en C89 avec la bibliothèque graphique de l'IUT de Fontainebleau où l'on doit bloquer l'adversaire avec des blocs infranchissable où alors avec le corp de son propre pion. Ce jeu peut se jouer seul contre une IA ou alors contre un autre joueur sur le même ordinateur. Les commandes pour jouer a se projet n'est que la souris. Le déplacement du joueur est limité a une case autour du joueur diagonale compris. Le terrain de jeu est une grille modulable entre 3 X 3 (9 case libre donc partie courte) et 9 X 9 (81 case libre donc partie plus longue). Il y a la possibilité a la fin d'une partie de recommencer une partie ou alors d'arrêter le jeux. Ce projet doit fonctionner avec un Makefile, dont avec la commande « make » on compile les fichiers et la commande « make run » pour lancer le programme. La partie se termine quand un des deux joueurs ne peut plus bouger au début de son tour.

2. Présentation des fonctionnalités du programme

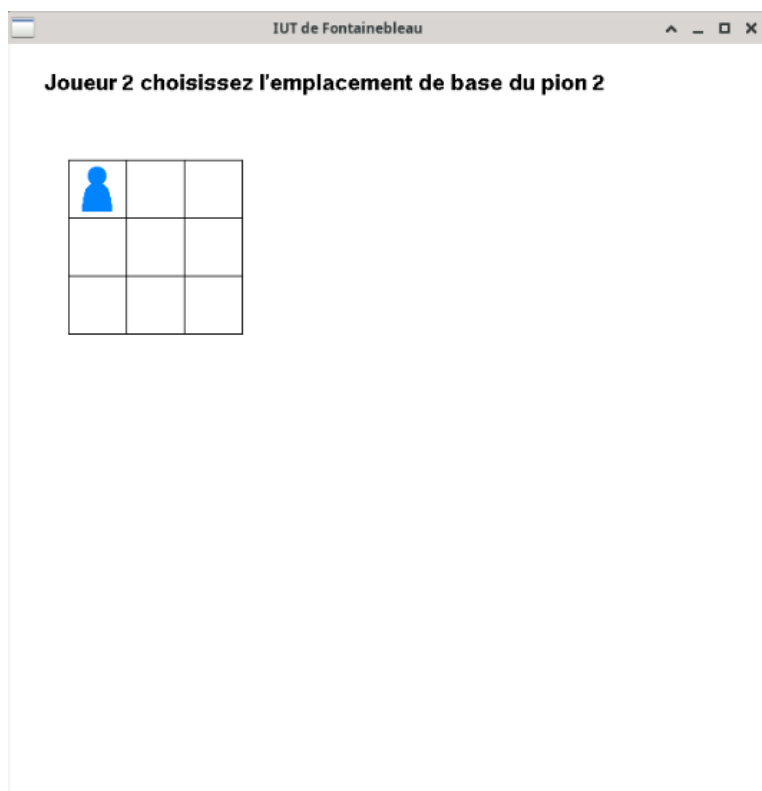
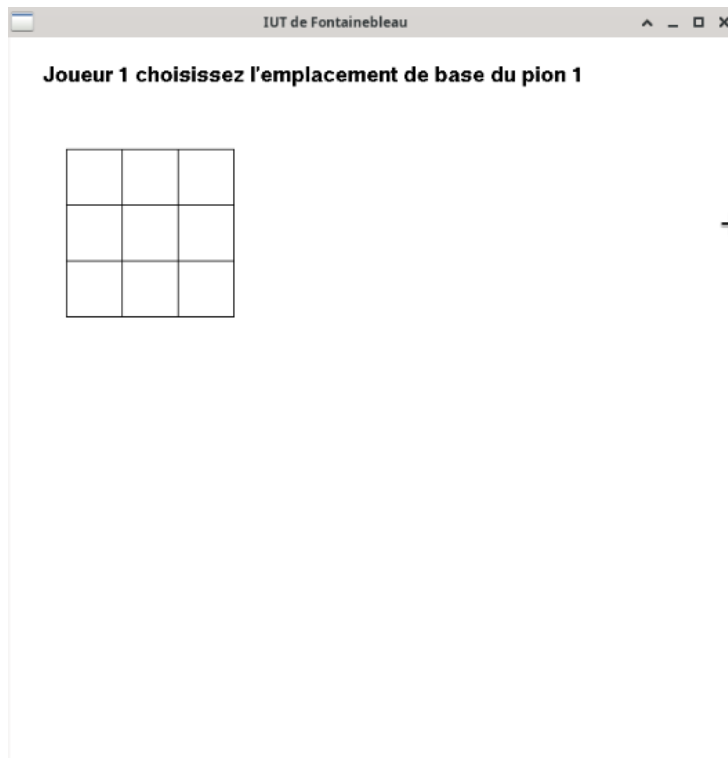
Au lancement du programme, on peut choisir la taille de la grille variant entre 3 et 9 :

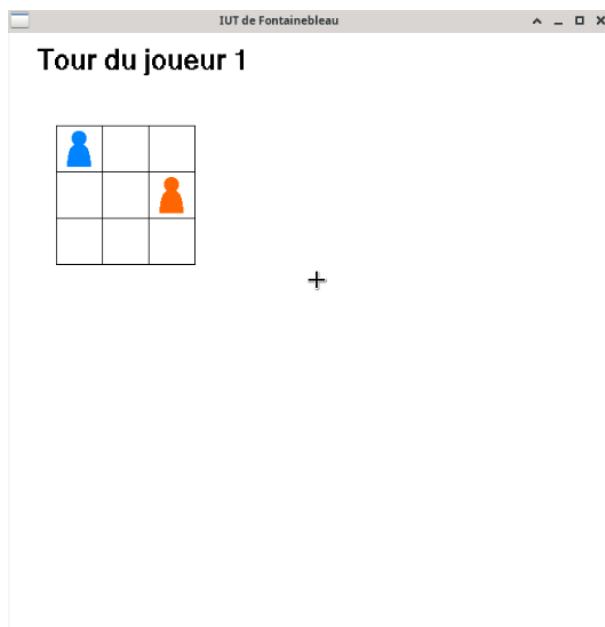


Une fois la taille choisie on a le menu du choix du nombre de joueur (IA ou joueur 2):

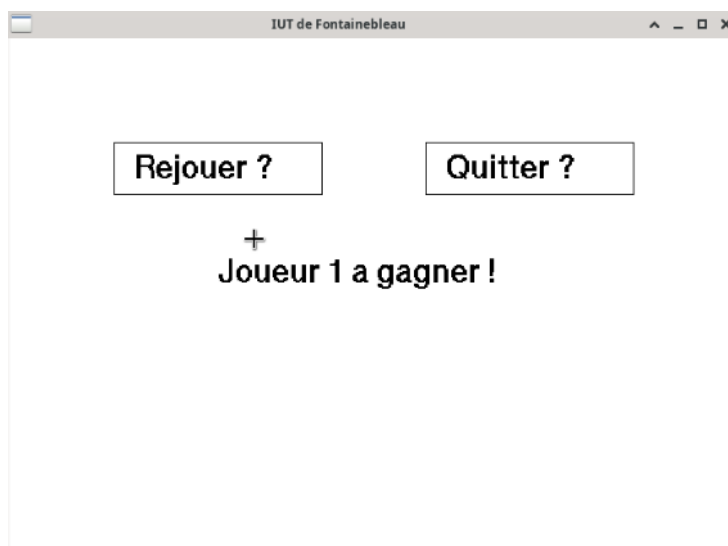
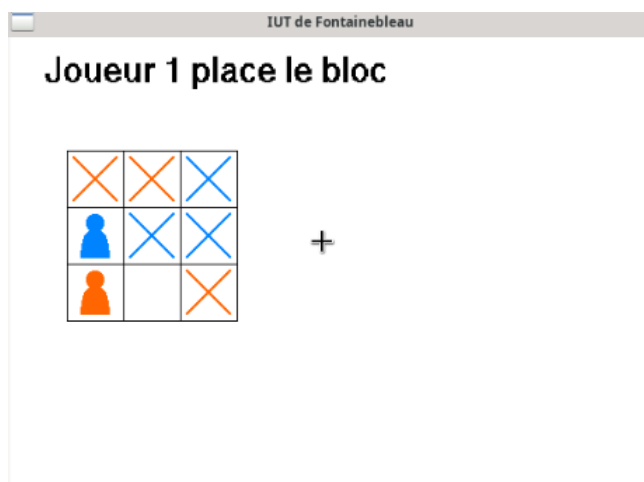


Si le joueur choisit 2 joueur avec une taille de grille de 3 ; le joueur 1 devra placer sur la case qu'il vaudra l'emplacement de base de son pion, puis ensuite se sera au tour du joueur 2 :



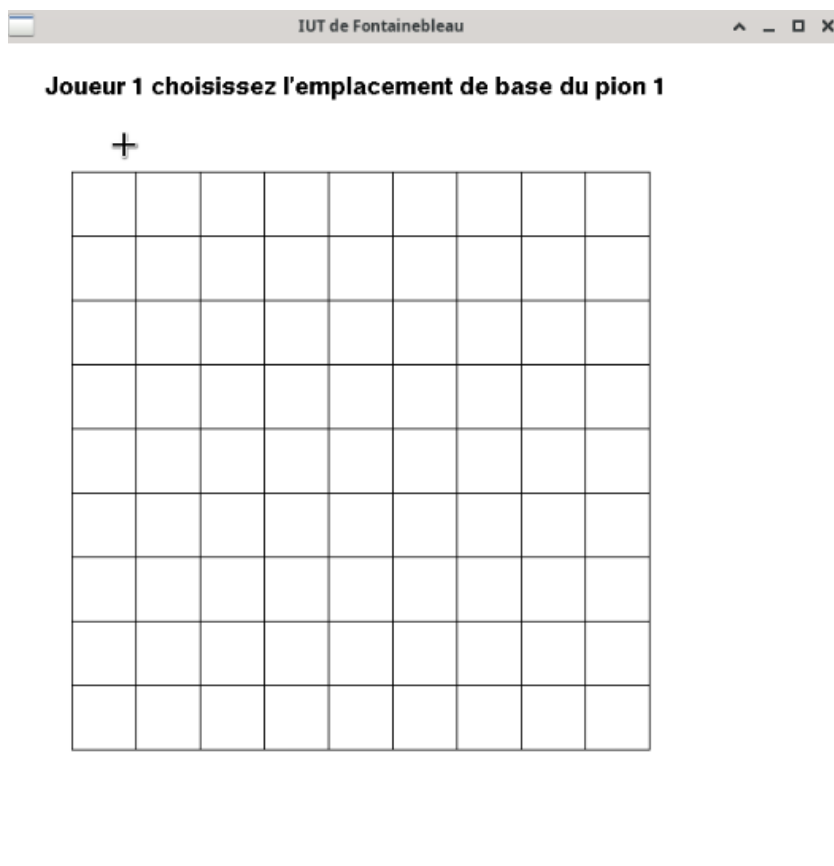
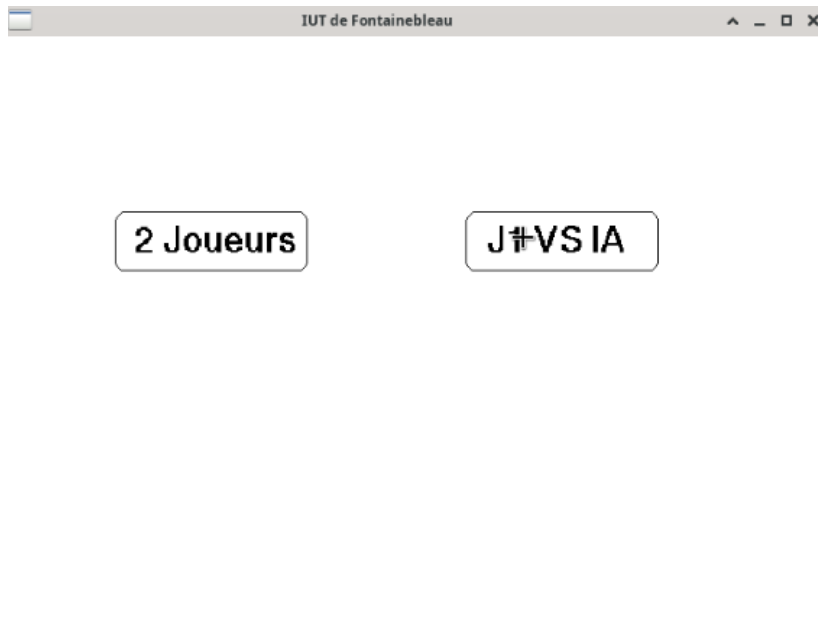


Au moment ou un joueur ne peux plus bouger cela signe la fin de la partie :

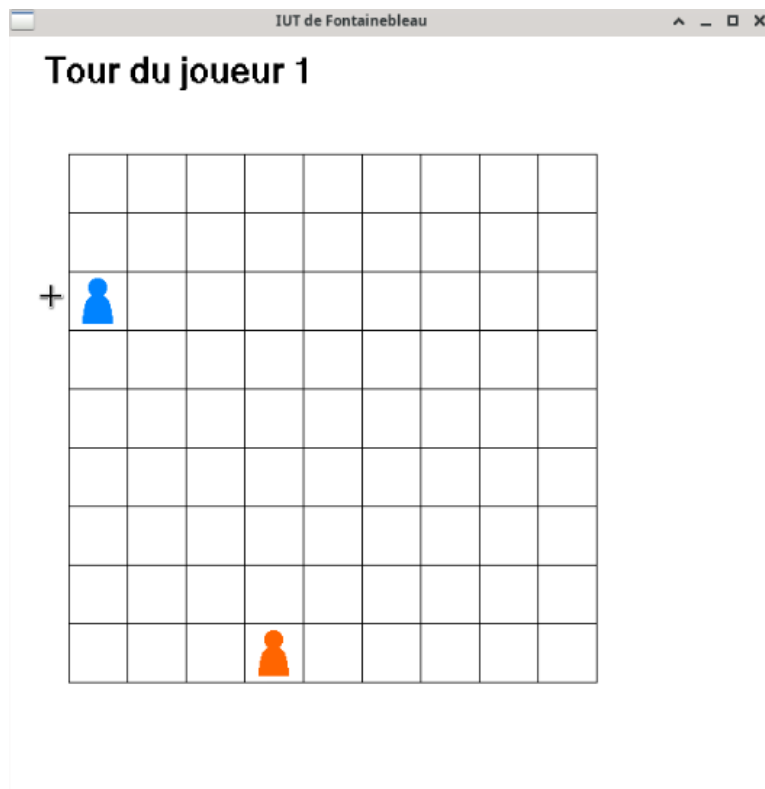


Dans mon cas c'est le joueur 1 qui a gagné, si le joueur décide de quitter le jeu il peut cliquer sur Quitter sinon il peut rejouer.

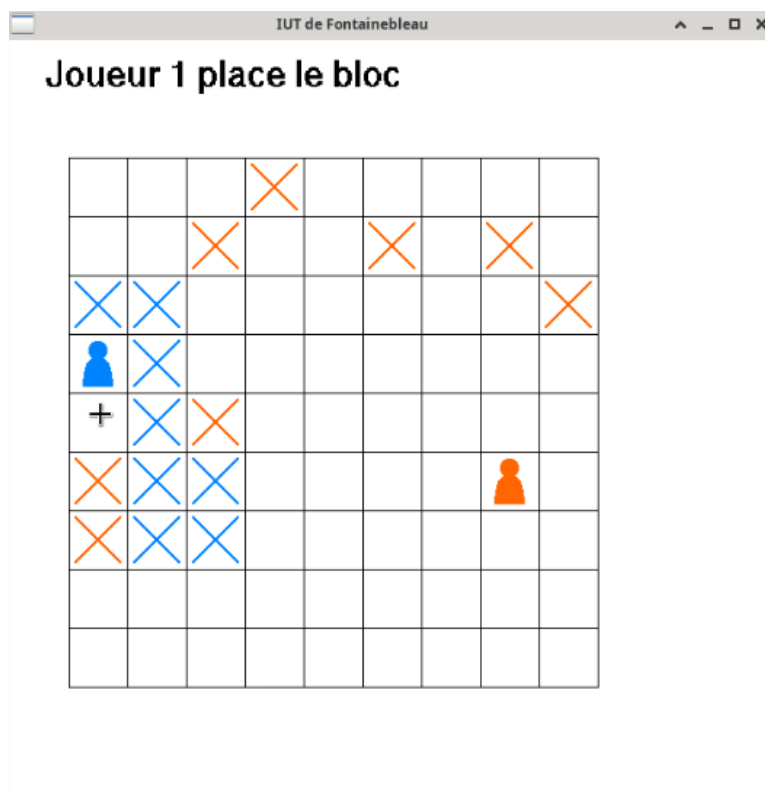
Dans le cas où le joueur aurait voulu une partie avec l'IA on aurait choisi une taille de grille disons 9 puis cliquer sur J1 VS IA :

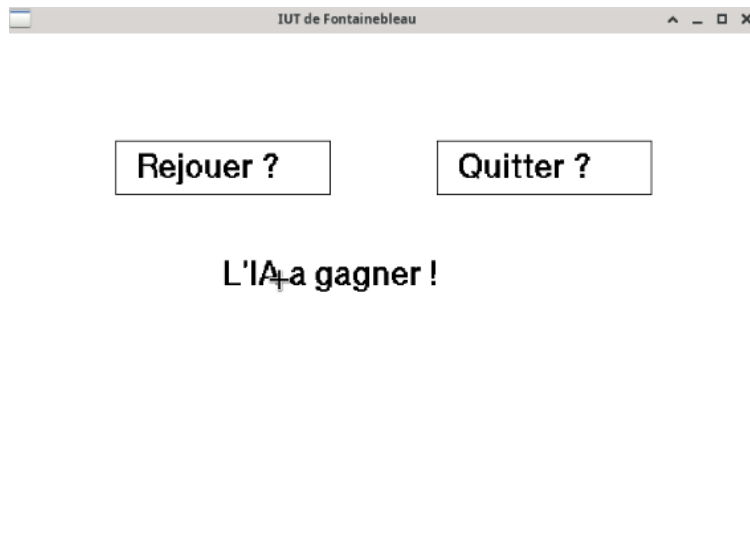


Une fois l'emplacement choisi par le joueur 1 l'IA joue directement après le joueur 1 :



Le programme remplace le comportement du joueur 2 par ses propres tour. Prenons l'exemple de si l'IA gagne :

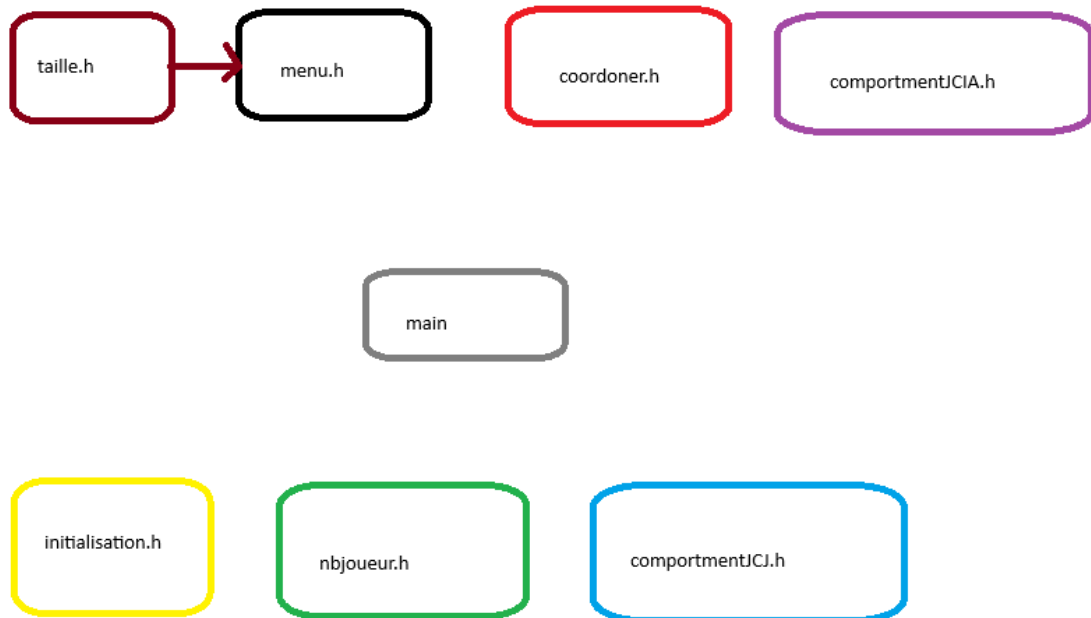




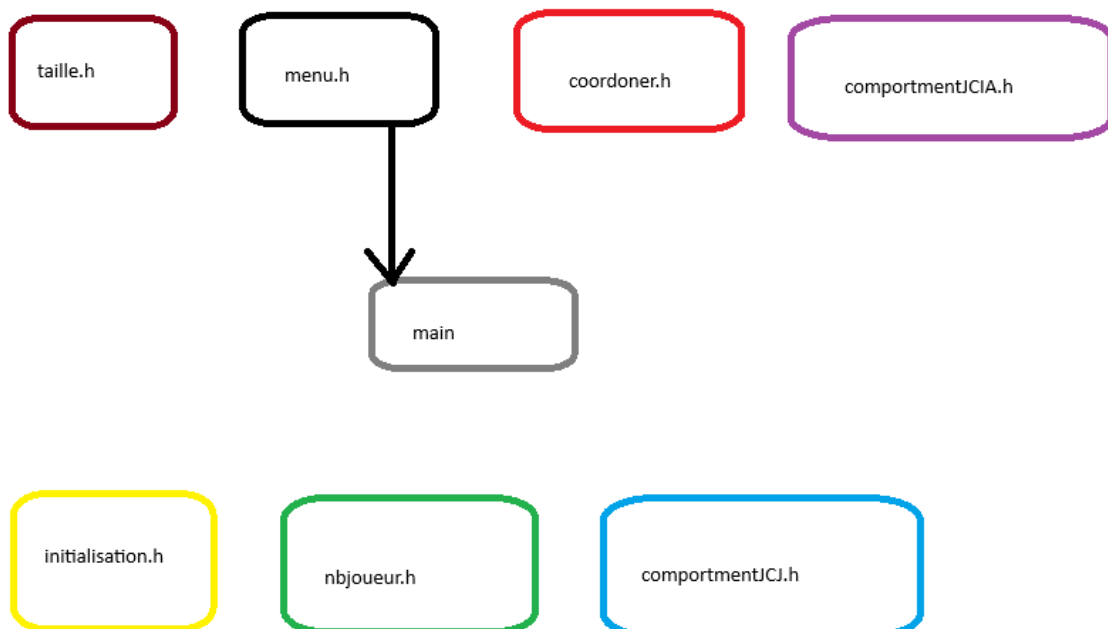
Le message de fin change en fonction de qui gagne (le Joueur 1, le Joueur 2 ou alors l'IA)

3. Le Makefile

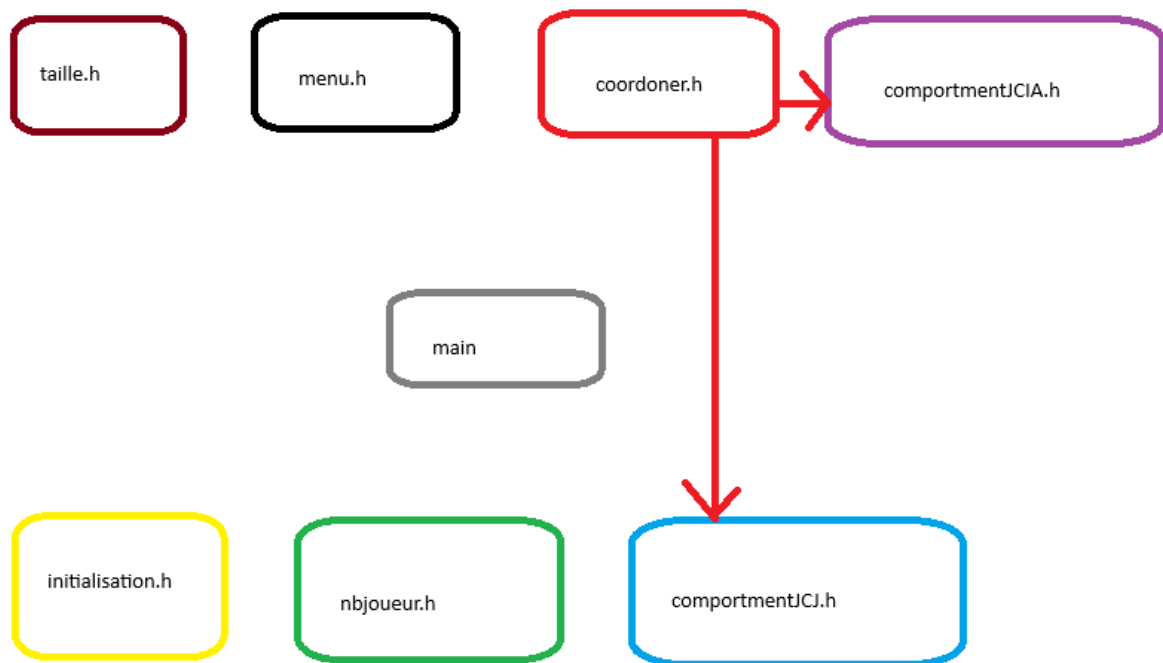
Pour ce projet comme dit précédemment, il a été fait avec un makefile ce qui permet de découper un programme en plusieurs petits programmes ce qui permet de mieux comprendre comment le programme agit ainsi que de pouvoir mieux trouver les parties à modifier si l'on en trouve le besoin. Voici une représentation graphique pour pouvoir visualiser notre makefile :



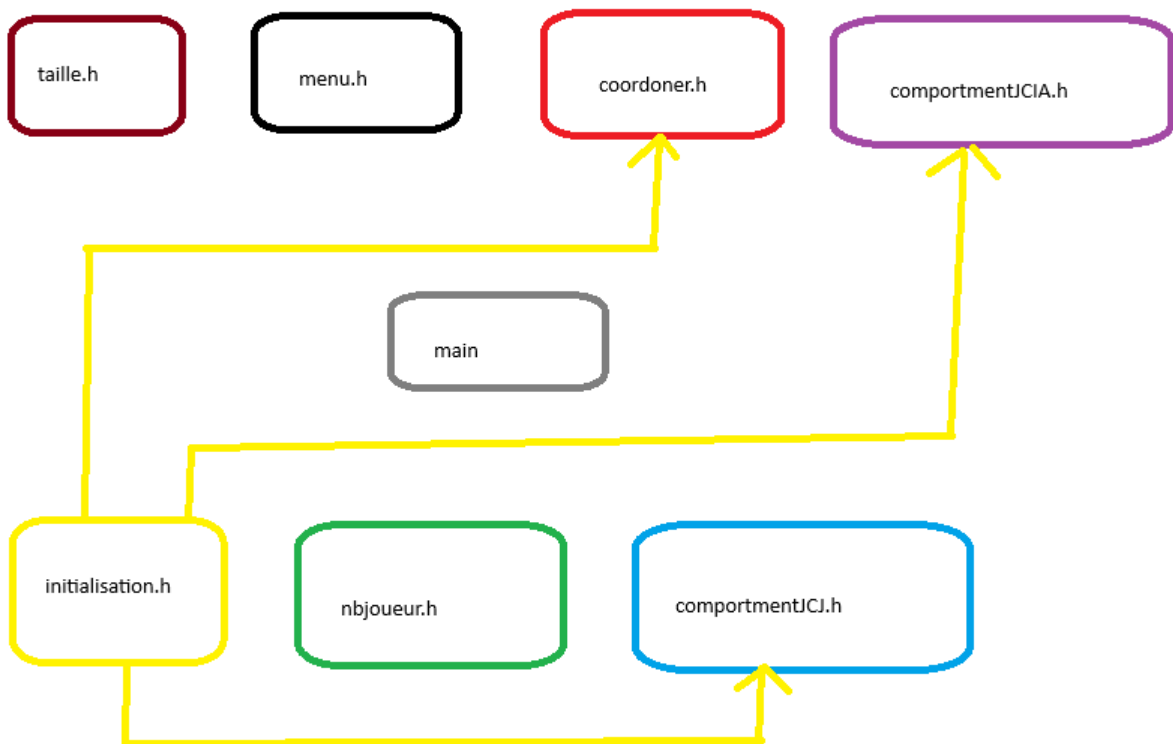
taille.h interagit avec seulement menu.h, car il permet d'afficher les boutons de sélection de la tailles de la grille ainsi que le boîte de collision des boutons.



Menu.h interagit avec le main pour pouvoir afficher le menu.

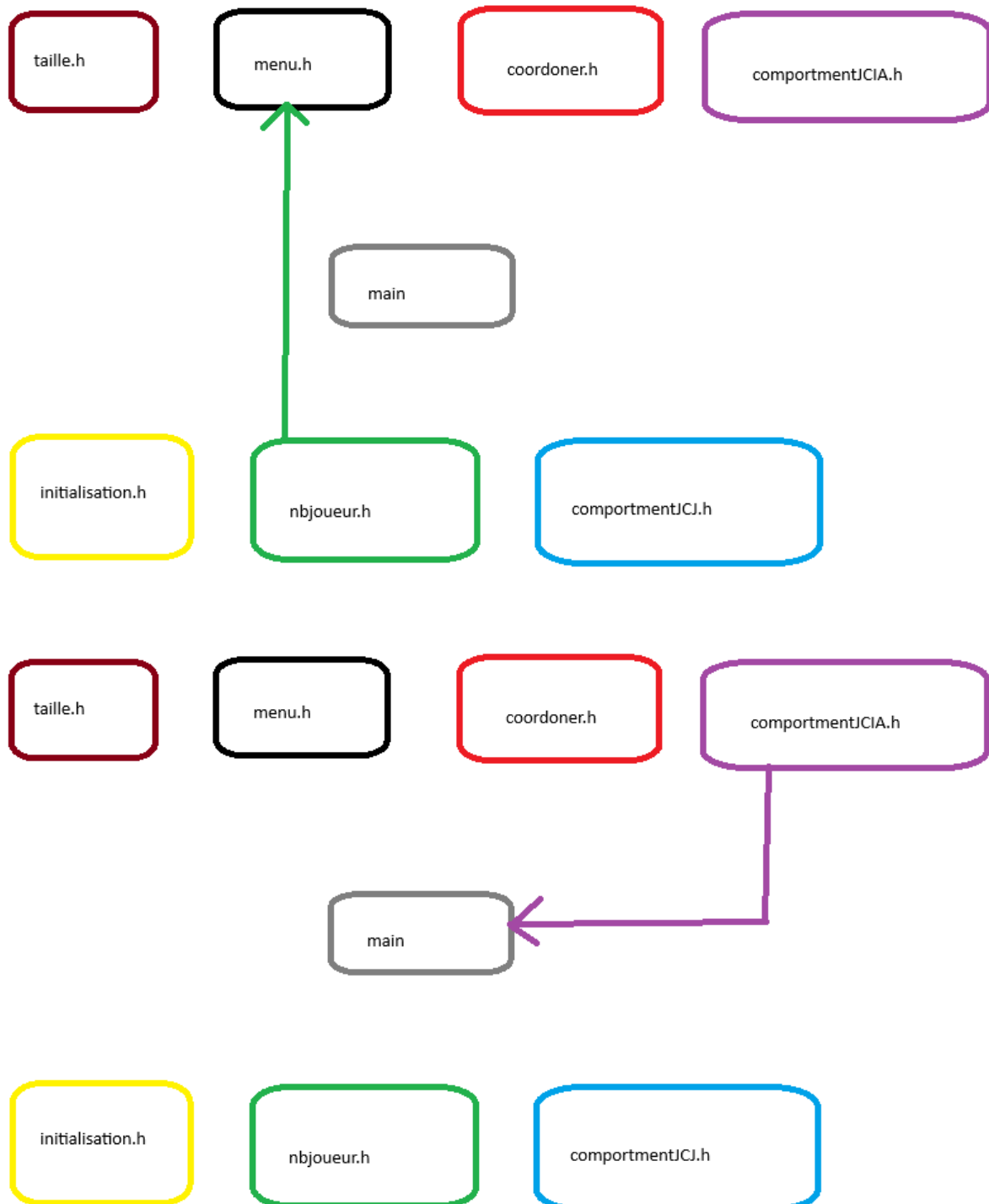


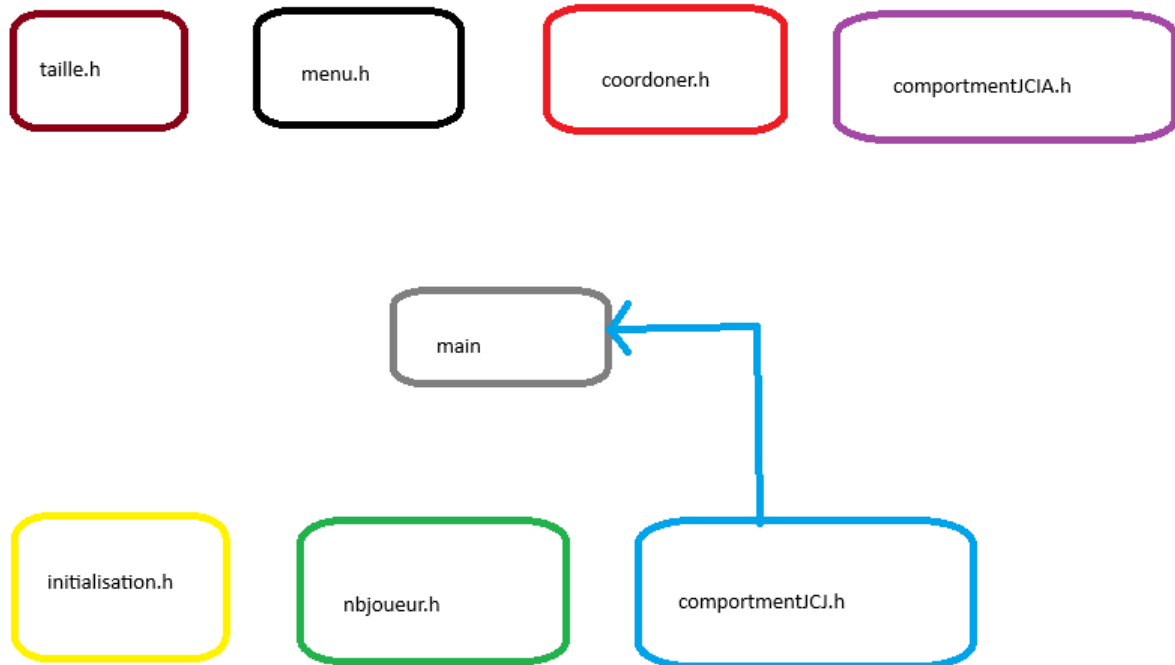
Coordonner.h renvoie à comportementJCJ.h et comportementJCI.h, l'index du tableau correspondant à l'emplacement cliqué par le joueur ainsi que les coordonnées pour afficher les images des personnages et de leurs blocs respectif.



Initialisation.h communique avec coordoner.h, comportementJCJ.h et comportementJCI.h pour leur communiquer les informations de l'initialisation

Voici le reste des graphiques de communication :





Dans notre makefile `main` n'a pas de `.h`, car c'est celui qui centralise les fichiers il ne fait que recevoir les données sans en rendre.

Pour conclure sur notre découpage, il est utile pour bien comprendre quelle partie du code de notre projet fait quoi, même si le nom des fichiers montre déjà leurs fonction ce découpage permet de savoir vers qui le fichier communique.

4. Explication des données d'une partie

Dans notre code pour une partie, prenons une partie joueur contre joueur tout d'abord on crée la grille avec la taille qu'on a choisie on rajoute 2 à la taille pour pouvoir faire une bordure qui n'est pas affichée d'une valeur de 9, en tout premier le joueur 1 entre dans une boucle de déplacement, on vérifie d'abord si l'emplacement autour du joueur est libre si il est libre on continue le programme sinon on retourne le code de victoire 2 comme c'est joueur contre joueur (ça aurait été 3 si c'était l'IA et 1 si c'était au début du tour de l'IA ou le joueur 2 pour dire que c'est le joueur 1 qui a gagné) où une fois que l'utilisateur a cliqué les données sont envoyées dans coordonner.h pour déterminer via le clique l'index sur la grille, si l'emplacement de la grille est égale à 0 se déplacer à cet endroit si et seulement si l'index est égale à l'index de son emplacement autour du personnage donc 8 vérification à faire, on supprime l'ancienne image après on demande au programme coordonner.h les coordonnées pour afficher l'image du joueur au nouveau emplacement. Puis on peut passer à la création du blocs pour le joueur 1 on attend le clique du joueur on vérifie encore avec coordonner.h si la position est libre si c'est libre on met la grille à jour avec la valeur 3. Ensuite c'est au tour du joueur 2 où il se passe la même chose que pour le joueur 1 mais cette fois si c'est la valeur 2 pour le déplacement et la valeur 4 pour la pose du blocs. Une fois que la condition de victoire est validée on retourne vers le main pour afficher le message de fin selon le choix de l'utilisateur on peut retourner faire une partie ou quitter.

5. Conclusions des auteurs

Soraya KHARDI :

Je pense que ce projet a été d'une grande utilité pour mieux comprendre le c89 mais aussi le fonctionnement de l'IA, ça a été un projet plutôt complexe mais plutôt amusant.

Ozvann ABRAHAM :

Le projet du Blocus m'a donné du fil à retordre uniquement pour la communication entre deux fichiers quand je devais faire transiter deux valeurs que retournerais un seul programme, pour moi c'était le plus grand défi de ce projet sinon j'ai adoré faire ce projet il m'a permis de me servir d'un pointeur qu'une seule fois et j'en suis ravie je comprend un peu mieux les pointeurs ; un autre défi se déroule dans l'histoire des gits, avec ce projet j'ai eu une idée pour le projet des premières années qui pourrait être sympathique à réaliser, ce serait une bataille navale. Ce projet a été pour moi une validation des acquis de ce début d'année, grande satisfaction de voir que ce projet-ci a été à 100% réalisé comparé à celui de l'an dernier où l'on n'avait pas fait toutes les fonctionnalités, je suis impatient du projet en Java.