

Othello game

Rapport individuel TERTAKI Anis 12015708

Introduction :

En gros en ce qui concerne mon rôle dans l'implémentation de ce jeu d'abord j'ai proposé l'idée d'utiliser un tableau rempli de valeurs différentes pour représenter l'ensemble du jeu qu'on a transformé après vers un tableau graphique au lieu d'un affichage simple avec la console, j'ai aussi contribué dans l'implémentation de la classe Othello (la classe principale) et la classe Game.

Idée de jeu :

L'idée de jeu c'était d'utiliser un tableau rempli de valeurs différentes pour mettre en place le jeu, chaque case est remplie par une valeur spécifique le 1 pour le noir, 2 pour le blanc, 0 pour une case vide, et - 1 pour une case hors le jeu (hors le tableau affiché). Après on a décidé de l'afficher avec l'interface graphique, et donc pour bouger les pions on change juste les valeurs du tableau et on met à jour l'affichage graphique

La classe Othello :

Dans cette classe j'ai implémenté le constructeur "Othello()" :

- Avec l'aide de Ghiles et Amine qui ont travaillé sur l'interface graphique j'ai pu récupérer les coordonnées de la case cliquée par le joueur noir.
- Je vérifie avec la méthode Legal Est-ce que le pion noir peut se mettre dans cette case sinon on affiche un message d'erreur sur la console.
- On met à jour le tableau en changeant la valeur de la case vers la valeur du pion noir (1) et on met à jour aussi l'affichage graphique.
- La dernière étape dans cette méthode c'est de vérifier Est-ce que le noir peut jouer encore c'est à dire Est-ce qu'il peut mettre des pions encore, on parcourt tout le tableau et si on trouve une seule case qui valide la fonction Legal on met la variable "black_done" à faux.
- Après on donne la main au blanc de jouer avec la méthode "coupBlanc()" implémenté par Nadine et Kamilia.

J'ai mis en place aussi une méthode "initGame()" pour initialiser le jeu c'est à dire remplir chaque case avec une telle valeur pour former la grille initiale du jeu :

- Le noir qui commence toujours.
- Les cases frontières (les cases hors le tableau du jeu) sont remplies avec - 1.
- On remplit les cases qui seront vide avec des zéros.
- Pour les 4 cases de milieu deux on les remplit avec la valeur du blanc 2 et les deux autres par la valeur du noir 1.

La classe Jeu :

C'est la classe qui contient la méthode "Légal" qui est la méthode principale du jeu, on a travaillé dessus moi, Ghiles et Amine. Elle détermine si on peut mettre un tel pion dans telle case (les 3 premier paramètre qu'elle prend en arguments) et elle renverse les pions de l'autres joueurs si le 4ème argument est mis à vrai, elle renvoie un Boolean :

- L'idée c'est de parcourir toutes les directions qui entoure la case en commençant par la première case attenante.
- Si cette case à coté est de couleur différente à celle passé en argument on continue dans cette direction jusqu'à rencontrer un pion de couleur égale à la couleur de tel pion si c'est le cas on renvoie vrai sinon faux.
- Donc d'abord on vérifie si la case est vide pour pouvoir commencer le processus sinon on ne peut rien faire et on renvoie directement faux.
- On rentre dans une boucle qui vérifie chaque direction une par une, on vérifie la première case à coté si elle valide notre idée, on continue dans cette direction en rentrant dans une autre boucle qui vérifie l'existence d'un autre pion de même couleur que celui passé comme argument à la méthode si c'est le cas on met le Boolean de la méthode à vrai et on passe à l'étape de renversements des pions de l'autre joueurs (cette partie c'est Amine et Ghiles qui l'ont complétée) sinon on sort de cette deuxième boucle et on vérifie l'autre direction.

Les problèmes rencontrés :

- Le premier problème c'était l'idée de jeu, l'idée d'utilisation du tableau et surtout le choix des classes et des méthodes à implémenter.
- La coordination entre la partie graphique et la partie console.
- La définition des frontières du tableau mais en rajoutant une case de plus par coté ça a facilité la tâche.
- La méthode "Légal" de la classe Jeu, qui était vraiment difficile à implémenter surtout son idée, elle nous a pris aussi beaucoup de temps.

