

Compte-rendu du jeu Dots and boxes

Quentin SAUVAGE

01 Mars 2017

Chapitre 1

Fonctionnement du programme

Le jeu se lance sur un menu demandant au joueur d'entrer les paramètres de la partie : La taille du plateau, le nombre de joueurs, et la présence ou non de l'Intelligence Artificielle. Une fois que le joueur a validé ces paramètres, le plateau s'affiche, ainsi que le nombre de points marqués par chaque joueur présent. Le joueur 1 joue en premier.

Une fois qu'un joueur totalise plus de la moitié des cases, il est déclaré vainqueur. Le programme demande alors au joueur s'il veut recommencer une partie ou s'il veut quitter le jeu. Dans le premier cas, il pourra alors changer les paramètres de jeu.

Chapitre 2

Structure du programme

Voici le rôle de chacune des classes du programme :

- Constantes est une Interface qui regroupe les constantes utiles au fonctionnement du jeu : Le nombre de colonnes/lignes de tuiles, le nombre de Pixels par case, et la distance entre chaque tuile.
- DemoJeu permet de créer le jeu et de le lancer.
- Fenetre : une JFrame qui contient tous les éléments graphiques du jeu et permet de faire le lien entre ces derniers et la classe Jeu.
- Jeu regroupe la gestion des éléments définissant le jeu, tels qu'un tableau contenant les tuiles, ou un booléen indiquant si le joueur a gagné ou non.
- PanneauFin est un JPanel qui demande au joueur s'il veut refaire une partie ou quitter le jeu.
- PanneauScore est un JPanel situé au-dessus du plateau de jeu, indiquant au joueur le score qu'il possède.
- Plateau est le JPanel principal du programme. Il permet d'afficher de manière harmonieuse les tuiles et couleurs de fond. Il affiche également "GAME OVER" quand le joueur a perdu.
- Tuile gère le comportement de chacune des tuiles du plateau, tout en mettant à jour leur aspect graphique et leurs coordonnées.

Chapitre 3

Fonctionnement de l'IA

3.1 Premier fonctionnement de l'IA

L'IA se déplace aléatoirement parmi les quatre directions possibles jusqu'à ce qu'elle ne puisse plus se déplacer. Elle était censée faire en sorte de toujours choisir la direction permettant de faire fusionner la plus grosse tuile possible, mais cette idée a été abandonnée à cause d'un bug qui bloquait l'IA si une ligne était remplie et qu'elle voulait se déplacer dans la ligne opposée (Par exemple : Si la ligne du bas est pleine, et que l'IA veut se déplacer vers le haut).