

Compte-rendu du Puissance 4

Quentin SAUVAGE

30 Mars 2017

Chapitre 1

Fonctionnement du programme

Le jeu commence avec une fenêtre demandant au joueur d'indiquer la dimension de la grille, le nombre de pions à aligner, les couleurs des joueurs, et l'activation ou non de l'IA/du mode suicide. Après avoir cliqué sur "Valider", le jeu se lance et les joueurs choisissent chacun leur tour la colonne où doit être placé leur pion. Une fois la partie finie, le joueur peut décider de revenir à l'écran de sélection des paramètres. Dans le cas où l'IA est activée, le joueur humain doit cliquer n'importe où sur la grille afin de finir le tour de l'IA.

Chapitre 2

Structure du programme

Voici le rôle de chacune des classes du programme :

Package controler

- AbstractControler définit les méthodes qui devront être utilisées par le Controler pour gérer les actions de l'utilisateur.
- Puissance4Controler hérite des méthodes d'AbstractControler afin de demander au modele de traiter les informations qui lui sont passées, puis de se mettre à jour.

Package initialisation

- Debut contient toutes les méthodes permettant de gérer le panneau de configuration de partie. Une fois la validation faite, il met à jour les attributs de Proprietes.
- Main permet de lancer le jeu.
- Proprietes contient toutes les variables de la partie, telle que la taille de la grille ou les couleurs des joueurs.

Package model

- AbstractModel définit les méthodes et attributs dont ont besoin les classes Puissance4 et IA. Elle implémente Observable.
- IA permet de simuler un joueur humain. Le tour est alors joué sans intervention sur la grille (hormis le clic pour passer le tour). L'IA définit elle-même l'endroit où elle doit jouer. Elle hérite de Puissance4
- Puissance4 hérite de AbstractModel et informe le Controler de son état (rempli, alignement vertical, etc). Elle met également à jour son état et en informe la Vue.

Package observer

- Observable est une interface indiquant à l'AbstractModel les méthodes lui permettant de gérer ses Observers.

- Observer est une interface indiquant à la Vue la méthode qu'elle doit implémenter pour redessiner les éléments graphiques en fonction de l'état de Puissance4.

Package vue

- Pion est un JPanel représentant un pion de la grille. Ils sont gérés par la vue et les pions visibles sont ceux qui ont été joués ainsi qu'un indicateur permettant au joueur actuel de savoir sur quel colonne il s'apprête à jouer (autrement dit, sur quelle colonne la souris se trouve actuellement).

- Vue implémente Observer et permet de gérer les éléments graphiques du jeu, comme la grille ou le JLabel indique que le joueur X doit jouer.

Chapitre 3

Fonctionnement de l'IA

3.1 Premier fonctionnement de l'IA

L'IA parcourt d'abord la grille afin de savoir si elle peut gagner directement en plaçant un pion à une certaine case. Si aucun emplacement ne lui permet de gagner, elle regarde alors si elle peut empêcher le joueur humain de gagner. Dans le cas où il n'y a pas besoin d'empêcher le joueur humain de gagner, elle place son pion dans un emplacement aléatoire, en veillant à ce que ce soit possible. Si le mode suicide est activé, elle génère un entier aléatoirement représentant la colonne sur laquelle elle va jouer, autant de fois qu'il le faut pour déterminer l'endroit où elle doit jouer. Pour déterminer cela, elle vérifie d'abord que l'emplacement ne la fait pas perdre (c'est-à-dire qu'elle n'alignera pas X pions). Si c'est le cas, elle joue à cet endroit, sinon elle ajoute cette position dans une ArrayList. Une fois que l'ArrayList contient (nombre de colonnes - 1) éléments, elle place son pion à la position générée, ce qui signifie qu'elle n'a pas d'autres choix que de perdre.