

Compte-rendu du jeu Picross

Quentin SAUVAGE

26 avril 2017

Chapitre 1

Fonctionnement du programme

Le jeu débute sur une fenêtre demandant au joueur de choisir une grille parmi celles existantes dans le fichier grille, ou d'en créer une lui-même en renseignant au préalable les dimensions de la grille souhaitée. Suivant le choix que le joueur a fait, l'application affiche alors :

- Dans le premier cas :

Une grille divisée en case de n lignes et m colonnes (correspondant au fichier choisi). Chaque ligne/colonne est accompagnée d'une série de nombres indiquant combien de cases noires consécutives sont cachées. Si le joueur est perdu, il peut cliquer sur le bouton solution afin de voir la grille finale. Même si cela n'a pas vraiment d'intérêt, il peut récupérer la grille présente avant l'affichage de la solution en re cliquant sur solution, s'il souhaite finir le niveau.

- Dans le second cas :

Comme pour le premier cas, une grille aux dimensions souhaitées s'affiche. Après avoir fini de dessiner la grille, le joueur renseigne le nom de la grille créée, et clique sur enregistrer pour enregistrer le nouveau fichier.

Lorsque le joueur choisit un niveau, ceux-ci sont affichés sous la forme "Grille" + index du niveau, afin que le joueur ne puisse deviner trop facilement la grille. S'il clique avec le clic gauche de la souris sur la grille, la case se colorie en noir ; si c'est avec le clic droit, la case se colorie en gris. En cliquant sur une case d'une couleur associée au bouton de la souris cliquée, la case reprend sa couleur de départ (blanc). Le joueur a perdu dès qu'il colorie une case en noir alors que celle-ci doit rester blanche. A la fin de la partie, le programme demande au joueur s'il veut recommencer une partie ou quitter.

Chapitre 2

Structure du programme

Voici le rôle de chacune des classes du programme :

Package controler :

- AbstractControler définit les méthodes devant être utilisées par GrilleControler.
- GrilleControler vérifie les actions du joueur en appelant les méthodes adéquates du Model, lorsque le joueur doit deviner une grille.
- CreationControler vérifie les actions du joueur en appelant les méthodes adéquates du Model, lorsque le joueur doit créer une grille.

Package grilles : Toutes les grilles en format txt permettant de récupérer la grille à deviner.

Package initialisation :

- Main permet de lancer l'application.
- Debut permet au joueur de choisir entre deviner une grille ou créer une grille.

Package model :

- AbstractModel définit les méthodes et attributs de Grille.
- CreationModel permet à la classe CreationControler de gérer les clics du joueur en fonction de ceux déjà fait, notamment : redonner à une case sa valeur de départ, ou enregistrer la grille.
- GrilleModel permet à la classe GrilleControler de gérer les clics du joueur en fonction de la grille à deviner, stockée dans GrilleModel.
- Observable est une interface définissant les méthodes permettant au Model de gérer ses Observers.

Package view : - Case représente graphiquement une case de la grille.

- GuiCreation contient tous les éléments graphiques de l'interface "Création de grille".

- GuiGrille contient tous les éléments graphiques de l'interface où le joueur doit deviner une grille..

- Observer définit les méthodes mettant à jour les Objet Observables.

Chapitre 3

Difficultés principales du projet

3.1 Choisir comment représenter les fichiers txt

Au début du projet, j'ai choisi un format qui n'a pas bougé par la suite : Les fichiers des grilles sont représentés de cette manière : La première ligne représente la largeur l de la grille, la deuxième représente sa hauteur h . Ensuite, le fichiers se compose de l lignes et h colonnes. A la colonne x et la ligne y , un entier est écrit, soit 0, soit 1, représentant une case blanche ou noire.

3.2 Accéder aux fichiers txt dans le jar

Il semble que le jar n'arrive pas à accéder aux fichiers grille/nomFichier.txt si celui n'est pas présent dans le même répertoire que le dossier src. J'ai essayé plusieurs solutions, mais aucune ne marchait. Pour que le jeu soit correctement fonctionnel, il faut donc placer le jar dans le même répertoire que src.