



Documents autorisés : une feuille manuscrite non photocopiée.

Jeu de plateau fondé sur des cartes-actions

Ce jeu peut être considéré comme dérivé du jeu de l'oie, avec aussi un parcours à effectuer sur un plateau constitué de cases successives numérotées de 1 au numéro de la dernière case. Dans cette version, le parcours est déterminé par le tirage de cartes-actions figurant dans un *sabot* mélangé au début du jeu (il n'y a pas de dés). Chaque joueur joue à tour de rôle dans un ordre tiré au hasard en début de jeu. Quand son tour vient, il tire la carte située au sommet du sabot, effectue l'action qu'elle indique, puis place la carte tirée au sommet d'un *tas* initialement vide. Le jeu se termine dès qu'un joueur (en l'occurrence le gagnant) atteint la dernière case du plateau (celle dont le numéro est égal au nombre de cases). Les joueurs ont une *position* qui est le numéro de la case dans laquelle ils se situent. Au début du jeu, elle vaut conventionnellement 0 pour indiquer qu'ils ne sont pas encore sur le plateau. Les joueurs peuvent occuper ensemble une même case.

Le jeu est modélisé en utilisant les classes suivantes (on suppose que tous les attributs sont munis des accesseurs nécessaires) :

- Carte : représente les cartes de tous types, dans tous les jeux possibles. Elle n'a pas de propriétés, parce qu'elle peut représenter des cartes de n'importe quel type (cartes à jouer, sept familles, Monopoly...)
- Paquet : représente un paquet de cartes quelconques. Une instance de Paquet doit permettre de manipuler son contenu (ajout, suppression, tri...).
- Joueur : permet de modéliser les joueurs de ce jeu. Un joueur a un *pseudo* et une *position* (le numéro de la case où il se situe).
- LesJoueurs : chaque instance de cette classe gère un groupe de joueurs représenté dans ce devoir par un tableau de type de base Joueur (Joueur[]). Le rang du joueur courant (celui qui a la main pour jouer) est stocké dans l'attribut *courant*.
- Jeu. Une instance de Jeu gère une partie. Elle nécessite un groupe de joueurs représenté par une instance de LesJoueurs et deux paquets de cartes qui constituent le sabot et le tas. Une caractéristique du jeu est aussi le nombre de cases du plateau¹.
- CarteAction : classe dont les instances vont exécuter des actions sur le joueur

¹ Il n'est pas nécessaire de matérialiser le plateau du jeu et les cases, qui n'ont aucune spécificité à part leur numéro qui apparaît comme position des joueurs.

courant. Cette classe contient un attribut d'instance *jeuCourant* qui référence le jeu en cours pour connaître ses caractéristiques (notamment le nombre de cases du plateau). Cette classe abstraite possède une méthode abstraite *execute()* qui représente l'action à effectuer. Des sous-classes de la classe *CarteAction* (*Deplacement*, *DeplacementAlea*, *Retention*...) donnent un corps spécifique à *execute()*, lequel permet d'effectuer l'action particulière correspondant à la sous-classe.

Questions

1. Question de cours : expliquez à quoi sert une lien de généralisation/spécialisation (*Sorte-de*) entre deux classes ? Prenez au moins un exemple.
 2. Comment un tel lien est-il exprimé en langage java ? Écrivez le code correspondant pour le lien entre *Deplacement* et *CarteAction*.
 3. Les liens d'utilisation entre classes sont matérialisés par certains attributs. Dites quelles déclarations d'attributs du diagramme de l'annexe² correspondent à des liens d'utilisation.
- Pour répondre aux questions suivantes, on suppose que tous les attributs décrits dans l'annexe sont dotés de leurs accesseurs.

4. Écrivez :

- a. la méthode d'instance *getJoueurCourant()* de la classe *LesJoueurs* qui restitue une référence à l'instance de *Joueur* qui représente le joueur courant. Ce dernier est celui dont l'indice dans le tableau *jeu* est contenu dans *courant*.
 - b. la méthode d'instance de même nom *getJoueurCourant()* de la classe *Jeu* qui utilise la méthode du a. pour restituer la référence au joueur courant du groupe de joueurs référencé par l'attribut *jeu*. Autrement dit, le joueur courant du jeu est celui du groupe de joueurs de ce jeu.
5. Utilisez les résultats de la question 4 pour écrire la méthode *getSujet()* qui restitue la référence à une instance de *Joueur* qui représente le sujet de l'action, c'est-à-dire le joueur sur lequel elle s'applique. Ce joueur est le joueur courant du jeu.
6. Écrivez le code de la classe *Deplacement* dont l'action consiste à modifier la position du joueur courant en lui affectant la valeur de l'attribut *destination*.
- L'attribut *destination* doit être déclaré selon les normes vues en cours et doit être accompagné de ses accesseurs,
 - Le constructeur de la classe a pour arguments la *destination* en question (une valeur entière) et le *jeu courant* (comme toute carte action),
 - La méthode *execute()* doit réaliser l'action. Utilisez la méthode *getSujet()* vue plus haut pour connaître le joueur concerné par l'action.

² Ceux qui sont représentés dans les cadres qui représentent les classes.

1/2

Annexe

