

Découvrir la programmation orientée objet avec Java



Travaux pratiques

frederic.rallo@univ-cotedazur.fr

TP6: Révision générale

L'objectif des exercices ci-dessous est de mettre en œuvre des éléments statiques que nous avons étudiés en cours.

Exercice 1

Reprenez l'exercice précédent sur les énumérations avec les pokémon (dernière version). Modifiez votre code avec les règles suivantes :

- la classe Battle est une classe abstraite et ne contient que des méthodes statiques
- la classe Referee est une classe abstraite et ne contient que des méthodes statiques

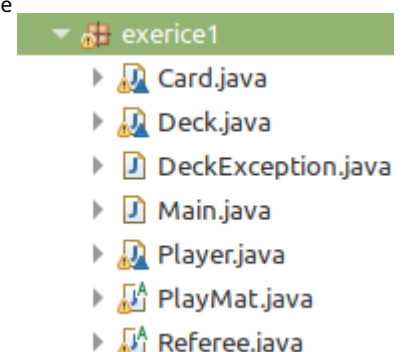
Exercice 2

Vous commencerez par déterminer l'architecture de ce projet (pensez à l'utilisation des interfaces, des enum, des classes abstraites, et des méthodes statiques lorsque cela vous semble opportun).

Créez un projet pour mettre en œuvre un programme d'arbitrage du jeu de la bataille fermée avec des jeux de 32 et 52 cartes classiques contenant en plus 2 Jockers distincts par paquet. Les joueurs jouent avec des cartes piochés dans le paquet commun pouvant provenir de plusieurs paquets de cartes de taille identique. Modéliser en Java les éléments suivants :

DÉBUT DE LA PARTIE

- L'arbitre est informé du nombre de joueurs
- au départ, chaque joueur devra disposer de plusieurs cartes (chaque joueur a le même nombre de cartes).
- L'arbitre distribue les cartes (l'arbitre peut ne pas toutes les distribuer)
- le nombre de paquets de cartes est ajusté en fonction du nombre de joueurs
- L'arbitre détermine l'ordonnancement du jeu
- L'arbitre vérifie pendant la partie le joueur gagnant de chaque manche
- L'arbitre ajoute à la fin de la pile du paquet du gagnant le butin de carte remportées
- Lorsqu'un joueur n'a plus de carte, il quitte le jeu
- L'arbitre arrête la partie lorsqu'il n'y a plus qu'un seul joueur
- Le gagnant de chaque manche est le joueur qui joue le premier au prochain tour



Découvrir la programmation orientée objet avec Java



Travaux pratiques

frederic.rallo@univ-cotedazur.fr

INFORMATIONS TECHNIQUES

- **Le tapis**

- o `//accesseurs nécessaires`
- o `// un joueur pose une carte`
`void add(Card card, Player player)`
- o `//retirer la carte d'un joueur`
`void remove(Card card, Player player)`
- o `//afficher une manche (voir trace) en indiquant dernière carte`
`void displayRound()`
- o `//vider le tapis`
`void clear()`

Découvrir la programmation orientée objet avec Java



Travaux pratiques

frederic.rallo@univ-cotedazur.fr

- **L'arbitre**

```
O //distribuer nbCards du tas (deck) mélangé à une liste de joueurs
void distribute(int nbCards, Deck deck, List<Player> players)

o //déterminer le gagnant d'une manche
Player getRoundWinner()

o //déterminer une liste de joueurs qui font bataille
List<Player> getWarPlayers()

o //donner les cartes remportées d'une manche à un joueurs
void givePlaymateCardsToTheWinner(Player roundWinner)

o //sortir du jeu les joueurs qui n'ont plus de carte
void updatePlayerList(List<Player> players)

o //demander à un joueur d'afficher son jeu (pour suivre sur la trace)
void askPlayerToShowHisCards(Player player)

o //mettre de côté les cartes qui ont fait bataille et les joueurs concernés
//remettent une carte sur le tapis. Les cartes mises de côté seront
//redistribuées à la fin de la manche au gagnant
void movePlayersWarCardToTempoCardlistAndPlayAgain(List<Player> warPlayers, List<Player>
currentPlayers)

o //forcer les cartes des joueurs 0 et 1 pour faire une bataille
void cheat(List<Player> players, Deck deck)
```

- **Les cartes**

```
o // constructeurs
o //accesseurs nécessaires
o //toString pour afficher 9 , AS , V ...
```

- **Le tas (deck)**

```
o // constructeurs avec n paquets de 32 ou 54 cartes (+ 2 Jockers) sinon erreur !
O // 2 constantes DECK52 et DECK32
```

- **Les joueurs**

```
o // constructeurs avec un nom
o // chaque joueur dispose d'une main de x cartes

o Card popCard() // retourne une carte de sa main
o void pushCard(Card card) // ajoute une carte à sa main
```

Découvrir la programmation orientée objet avec Java



Travaux pratiques

frederic.rallo@univ-cotedazur.fr

• Le Main (les valeurs sont à titre indicatif)

```

créer le tas avec 1 paquet de 54 cartes
créer 4 joueurs
L'arbitre distribue 4 cartes par joueurs
L'arbitre triche et fait en sorte que 2 joueurs fassent une bataille
tant qu'il reste des joueurs
    placer sur le tapis 1 carte de chaque joueur
    afficher le tapis
    L'arbitre vérifie s'il y a bataille
    tant qu'il y a bataille
        mettre de côté les cartes qui ont fait bataille
        afficher le tapis
        L'arbitre vérifie s'il y a bataille
    L'arbitre détermine le gagnant de la manche
    L'arbitre donne les cartes gagnées au gagnant
    L'arbitre vérifie si un joueur a perdu (-> il ne joue plus)
    L'arbitre demande à chaque joueur d'afficher sa main (pour trace)
L'arbitre demande à chaque joueur d'afficher sa main (pour trace) félicite le grand gagnant !
  
```

DÉROULEMENT DU JEU

attention : une partie peut être longue :/

```

[D , 10 , 9 , 7 , 3 , 4 , R , 7 , R , R , 3 , V , V , 7 , 7 , 5 , 4 , 2 , 6 , 8 , 9 ,
6 , 8 , JOCKER , 10 , 5 , JOCKER , 6 , AS , AS , 2 , V , 8 , D , 8 , 5 , AS , 6 ]
  
```

```

-----
Mélanie play D --> hand cards are: Mélanie --> [2 , 3 , 10 ]
Juliette play 9 --> hand cards are: Juliette --> [3 , 10 , 9 ]
Tristan play 4 --> hand cards are: Tristan --> [D , V , R ]
Safa play 4 --> hand cards are: Safa --> [2 , AS , 5 ]
REFEREE says: >>>>>>>>> getWarPlayers <<<<<<<<<<< beetween [Safa, Tristan]
Tristan play D --> hand cards are: Tristan --> [V , R ]
Mélanie play D --> hand cards are: Mélanie --> [2 , 3 , 10 ]
Juliette play 9 --> hand cards are: Juliette --> [3 , 10 , 9 ]
Safa play 2 --> hand cards are: Safa --> [AS , 5 ]
REFEREE says: >>>>>>>>> getWarPlayers <<<<<<<<<<< beetween [Tristan, Mélanie]
Tristan play V --> hand cards are: Tristan --> [R ]
Juliette play 9 --> hand cards are: Juliette --> [3 , 10 , 9 ]
Mélanie play 2 --> hand cards are: Mélanie --> [3 , 10 ]
Safa play 2 --> hand cards are: Safa --> [AS , 5 ]
REFEREE says: >>>>>>>>> getWarPlayers <<<<<<<<<<< beetween [Safa, Mélanie]
Safa play AS --> hand cards are: Safa --> [5 ]
Tristan play V --> hand cards are: Tristan --> [R ]
Juliette play 9 --> hand cards are: Juliette --> [3 , 10 , 9 ]
Mélanie play 3 --> hand cards are: Mélanie --> [10 ]
Round winners is: Safa
Safa --> [5 , AS , V , 9 , 3 , 4 , 4 , D , D , 2 , 2 ]
Tristan --> [R ]
Mélanie --> [10 ]
Juliette --> [3 , 10 , 9 ]
-----
  
```

```

Tristan play R (last card) --> hand cards are: Tristan --> []
  
```

Découvrir la programmation orientée objet avec Java



Travaux pratiques

frederic.rallo@univ-cotedazur.fr

```
Mélanie play 10 (last card) --> hand cards are: Mélanie --> []
Safa play 5 --> hand cards are: Safa --> [AS , V , 9 , 3 , 4 , 4 , D , D , 2 , 2 ]
Juliette play 3 --> hand cards are: Juliette --> [10 , 9 ]
Round winners is: Tristan
REFEREE says: Mélanie game over !
Safa --> [AS , V , 9 , 3 , 4 , 4 , D , D , 2 , 2 ]
Tristan --> [R , 10 , 5 , 3 ]
Juliette --> [10 , 9 ]
-----
```

Découvrir la programmation orientée objet avec Java



Travaux pratiques

frederic.rallo@univ-cotedazur.fr

```

Safa play AS    --> hand cards are: Safa --> [V , 9 , 3 , 4 , 4 , D , D , 2 , 2 ]
Tristan play R  --> hand cards are: Tristan --> [10 , 5 , 3 ]
Juliette play 10 --> hand cards are: Juliette --> [9 ]
Round winners is: Safa
Safa --> [V , 9 , 3 , 4 , 4 , D , D , 2 , 2 , AS , R , 10 ]
Tristan --> [10 , 5 , 3 ]
Juliette --> [9 ]
-----

Safa play V     --> hand cards are: Safa --> [9 , 3 , 4 , 4 , D , D , 2 , 2 , AS , R , 10 ]
Tristan play 10 --> hand cards are: Tristan --> [5 , 3 ]
Juliette play 9  (last card) --> hand cards are: Juliette --> []
Round winners is: Safa
REFEREE says: Juliette game over !
Safa --> [9 , 3 , 4 , 4 , D , D , 2 , 2 , AS , R , 10 , V , 10 , 9 ]
Tristan --> [5 , 3 ]
-----

Safa play 9     --> hand cards are: Safa --> [3 , 4 , 4 , D , D , 2 , 2 , AS , R , 10 , V , 10 , 9 ]
Tristan play 5  --> hand cards are: Tristan --> [3 ]
Round winners is: Safa
Safa --> [3 , 4 , 4 , D , D , 2 , 2 , AS , R , 10 , V , 10 , 9 , 9 , 5 ]
Tristan --> [3 ]
-----

Tristan play 3  (last card) --> hand cards are: Tristan --> []
Safa play 3     --> hand cards are: Safa --> [4 , 4 , D , D , 2 , 2 , AS , R , 10 , V , 10 , 9 , 9 , 5 ]
REFEREE says: >>>>>>>>> getWarPlayers <<<<<<<<<<< beetween [Safa, Tristan]
REFEREE says: Tristan is game over
Safa play 4     --> hand cards are: Safa --> [4 , D , D , 2 , 2 , AS , R , 10 , V , 10 , 9 , 9 , 5 ]
Round winners is: Safa
Safa --> [4 , D , D , 2 , 2 , AS , R , 10 , V , 10 , 9 , 9 , 5 , 4 , 3 , 3 ]
-----

REFEREE says: congratulation Safa
  
```