

Nom :

Prénom :

Promo :

SUGGESTIONS

DE

RÉPONSES

Documents et supports de cours autorisés,
calculatrices et téléphones interdits.Répondre sur l'énoncé
en soignant la présentation.

On s'intéresse au jeu de société *Poison*, dont un récapitulatif des règles est donné en annexe.

- Question 1** En plus des cartes et des chaudrons sur lesquels le sujet s'attarde plus en détail, quels concepts du jeu vous semblent importants à représenter par des classes spécifiques ?
3 points
- Pour chacun, décrivez succinctement quelles responsabilités en termes de traitements ou de données vous lui attribueriez.

Jeu :

- déroulement de phases de jeu (donne, manches, score final...)
- règles globales (exclusivité des couleurs dans les chaudrons...)
- connaissance des joueurs/chaudrons/cartes existant(e)s

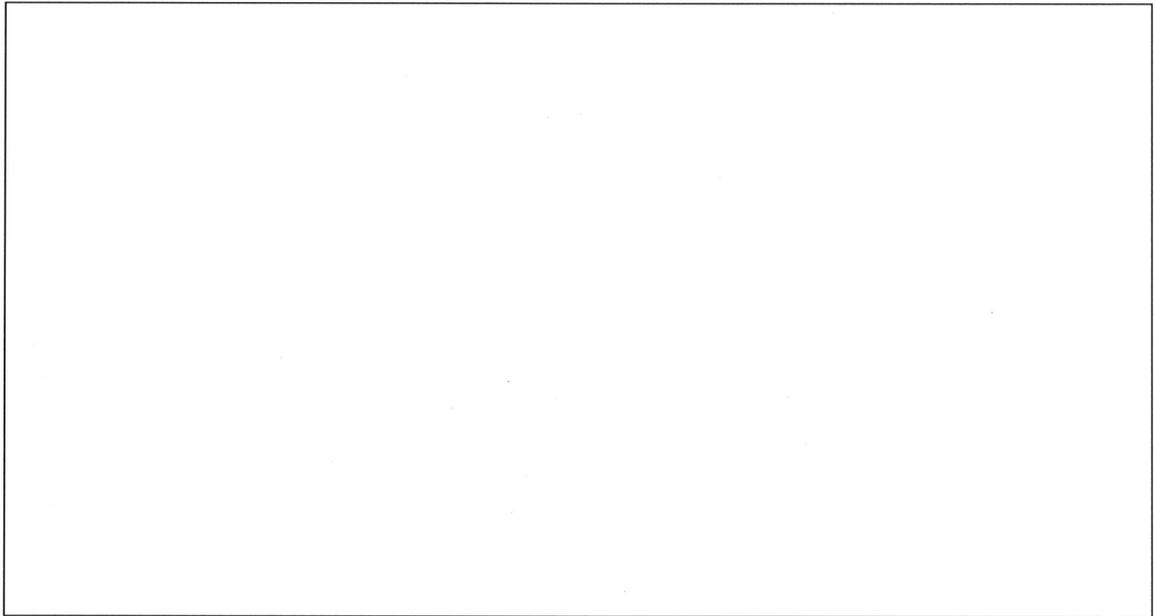
Joueur :

- cartes ramassées
- main
- décision/stratégie de jeu (en demandant à un utilisateur, ou à une intelligence artificielle...)

On modélise les cartes du jeu par le type énuméré et l'interface suivants :

```
enum Color { RED, BLUE, PURPLE, NONE; }
interface Card {
    Color color();
    boolean isPotion(); // potion -> true, poison -> false
    boolean mixesWith(Color another); // règles 3 et 4
    int faceValue(); // valeur pour le remplissage des chaudrons
    int points(); // valeur lors du comptage des points
}
```

Question 2 3 points Donnez le code Java de la classe `PoisonCard`, implémentant l'interface `Card`, et représentant exclusivement les cartes Poison.



Question 3 3 points Donnez le code Java de la classe `PotionCard`, implémentant l'interface `Card` pour représenter les cartes Potion (à l'exclusion des poisons). Définissez un constructeur pertinent.



On s'intéresse maintenant au fonctionnement des chaudrons, dont voici un cahier des charges détaillé : un chaudron contient une liste de cartes, et est capable d'en déterminer plusieurs propriétés :

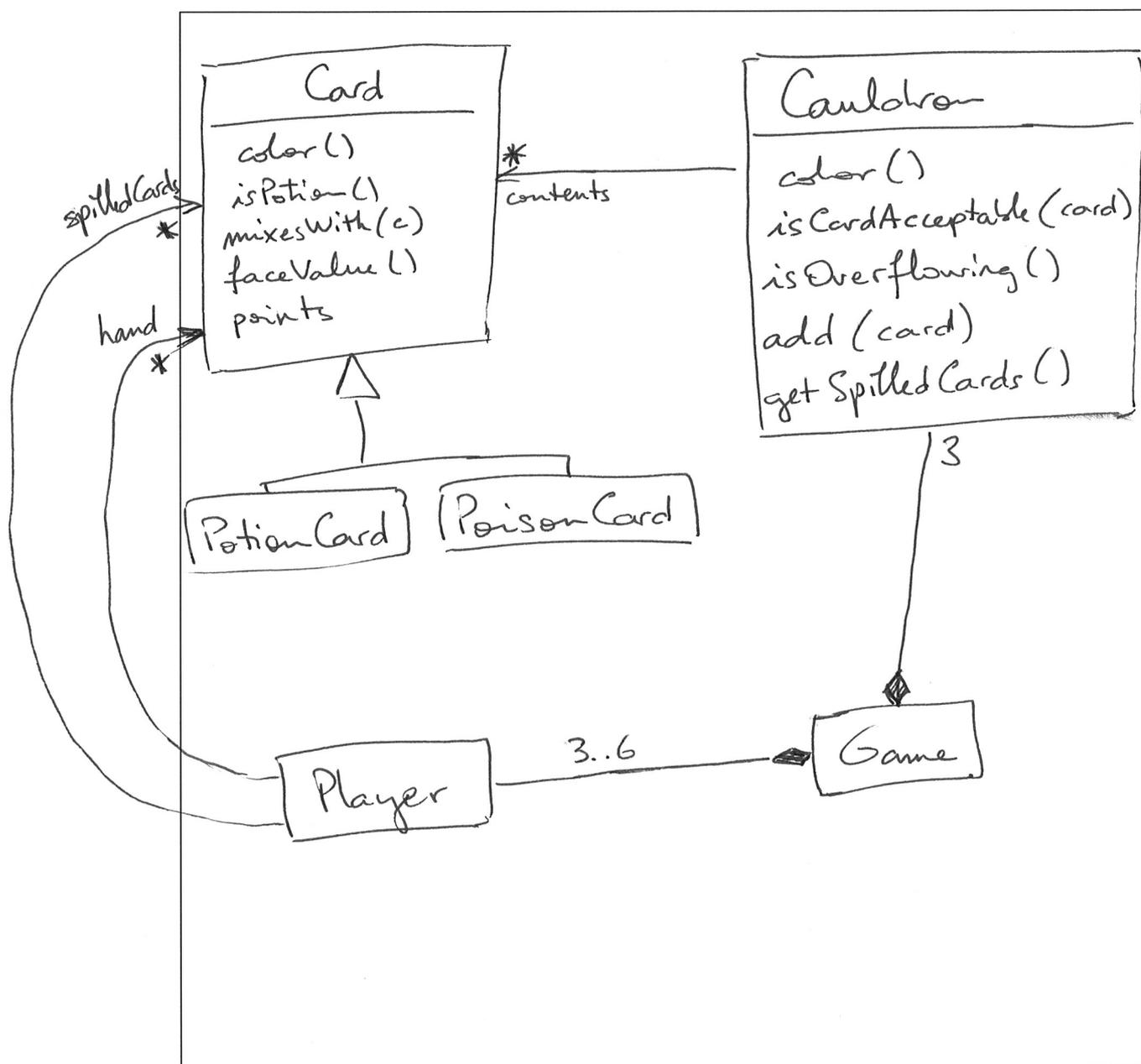
1. la couleur attribuée au chaudron par son contenu (règles 3 et 5),
2. si il est possible de jouer une carte donnée dans ce chaudron (règles 3 et 4),
3. si le chaudron est en train de déborder (règle 6).

D'autre part un chaudron propose deux traitements changeant son état :

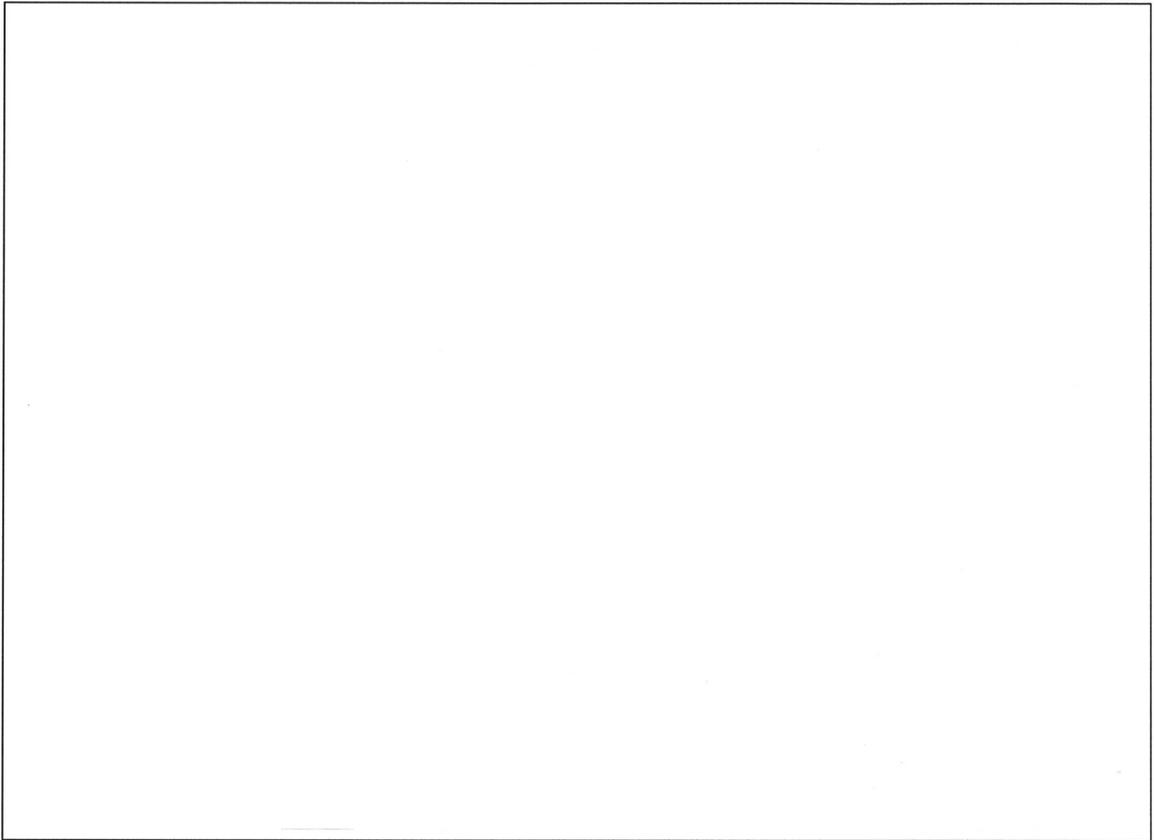
- l'ajout effectif d'une carte potion ou poison au chaudron (règle 2),
- pour un chaudron qui déborde, en extraire la liste de cartes que le joueur doit récupérer (règle 6)

Question 4 Décrivez par un diagramme de classes UML la classe **Cauldron** décrite ci-dessus, ainsi que son lien avec **Card**. La notation favorisera la clarté des noms choisis pour les variables d'instance, méthodes, arguments...

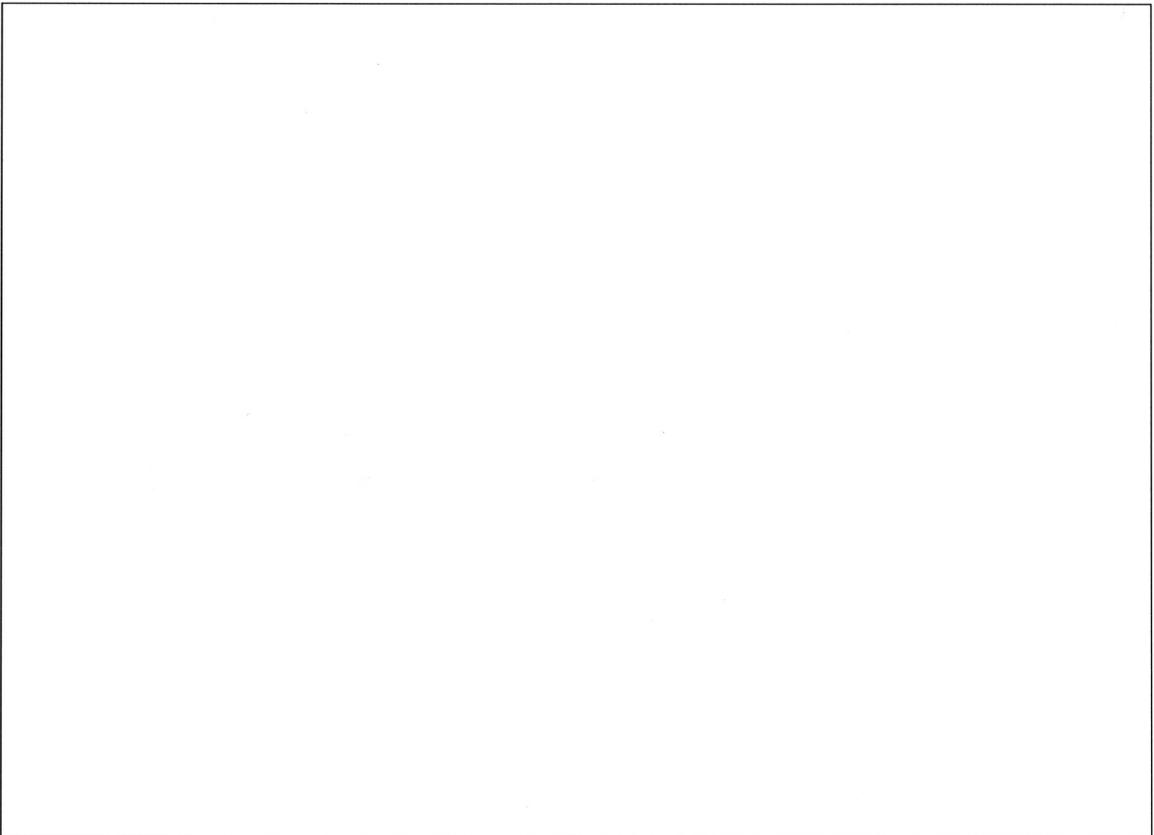
5 points



Question 5 Implémentez en Java la méthode déterminant la couleur d'un chaudron (règles 3 et 5).
3 points



Question 6 Implémentez la méthode déterminant si le chaudron est en train de déborder (règle 6).
3 points



Annexe : le jeu de cartes Poison (R. Knizia, Amigo Spiele)

Matériel de jeu Dans un laboratoire, trois chaudrons de breuvages magiques bouillonnent. Les joueurs y jettent leurs ingrédients, au risque de faire déborder les chaudrons.

Hormis trois emplacements sur la table de jeu, représentant les trois chaudrons, le jeu se joue de 3 à 6 joueurs, avec un paquet de 50 cartes :

- 14 cartes *Potion* de chaque couleur rouge, violette, bleue : deux de valeur 4, et trois de chaque valeur 1, 2, 5, et 7
- 8 cartes *Poison* (toutes de valeur 4)

Déroulement d'une manche

1. Le donneur mélange toutes les cartes et les distribue une à une. Selon le nombre de joueurs, certains recevront une carte de plus que d'autres. Distribution et tour de jeu se font en sens horaire, en commençant à gauche du donneur.
2. Chacun leur tour, les joueurs jouent l'une des cartes de leur main, face visible, dans l'un des chaudrons.
3. On ne peut pas mélanger les potions. La première carte *Potion* jouée dans un chaudron détermine donc la couleur des cartes pouvant être jouées ensuite dans ce chaudron. De plus, on doit toujours pouvoir jouer une couleur ; par conséquent, une même couleur ne peut être présente dans plusieurs chaudrons simultanément.
4. Les cartes *Poison* ne comptent pas comme une couleur : elles peuvent être jetées dans n'importe quel chaudron, quelle que soit la couleur des cartes déjà présentes.
5. Un chaudron vide ou ne contenant que du poison n'a pas (plus) de couleur attirée.
6. Un chaudron déborde quand la somme des valeurs des cartes qu'il contient *dépasse* 13, potions et poisons compris. Le joueur ayant provoqué le débordement récupère alors le contenu du chaudron et place les cartes faces cachées devant lui, sauf la carte qu'il vient de jouer, qui reste dans le chaudron.

Fin de manche et comptage des points

7. La manche prend fin quand tous les joueurs ont épuisé leurs mains. Les cartes restant alors dans les chaudrons n'attribuent de points à personne.
8. Les joueurs révèlent les cartes qu'ils ont fait déborder des chaudrons ; chaque carte *Potion* vaut un point négatif (les valeurs faciales des cartes ne servent qu'à remplir les chaudrons).
9. Pour chaque couleur, si un joueur a *strictement* plus de cartes *Potion* de cette couleur que chacun de ses adversaires, il n'en compte pas les points.
10. Les cartes *Poison* comptent toujours et valent *deux* points négatifs chacune.
11. Le vainqueur est le joueur ayant accumulé le moins de points négatifs à l'issue de la partie.

Nombre de manches

12. Pour quatre à six joueurs, chaque joueur donne une fois (une manche par joueur).
13. À trois joueurs, la partie se joue en six manches, chaque joueur étant donc donneur deux fois. De plus, le donneur distribue pour quatre joueurs, et on met de côté la main supplémentaire pour la durée de la manche.