

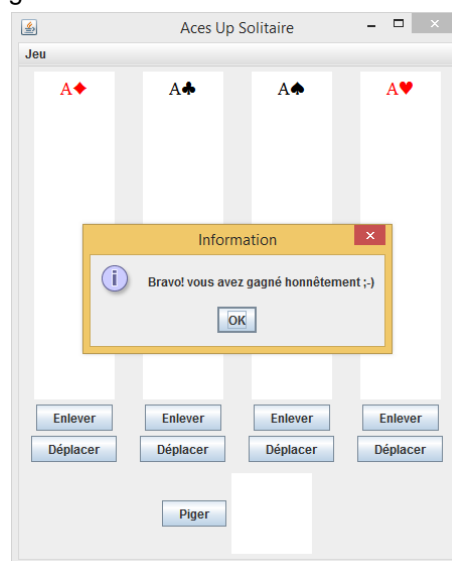
<p align="center">Cours 420-202-RE Traitement de données orienté objet Hiver 2021 Cégep Limoilou Département d'Informatique</p>	<p align="center">Tp 4 partie 1 de 2 (1 semaine pour cette partie) 12 % pour les 2 parties Aces Up Solitaire</p>
--	---

Objectifs :

- Définir et utiliser des structures de données dynamiques comme des piles et des listes.
- Définir et utiliser de la persistance de données.
- Travailler en équipe de 2 personnes.

Version finale du jeu à faire :

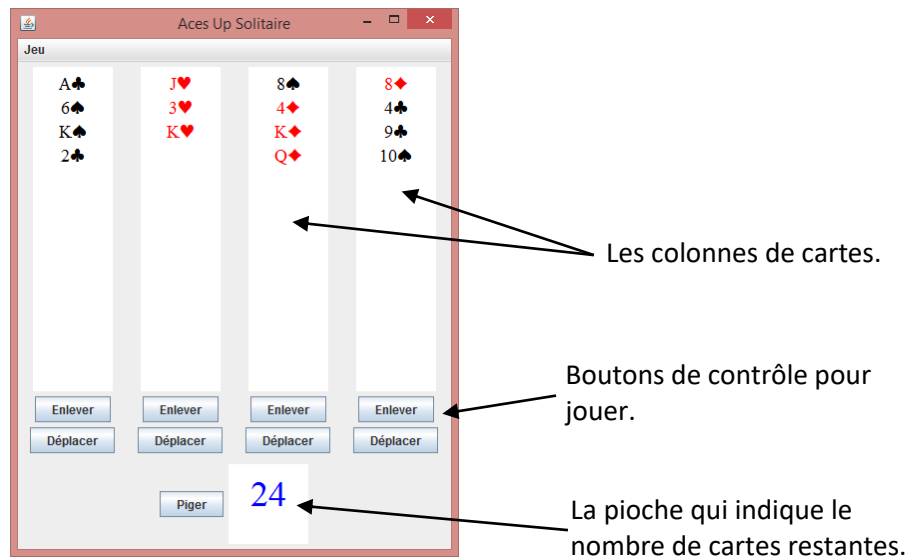
À partir de ce document et du projet fourni (code fourni), réalisez un jeu de cartes de type solitaire le « Aces Up Solitaire ». Il s'agit d'un jeu comportant quelques règles simples utilisant un paquet de cartes standard. Le but est d'utiliser toutes les cartes du paquet en terminant la partie avec seulement les 4 as restants dans les colonnes du jeu tel qu'illustré sur la figure suivante.



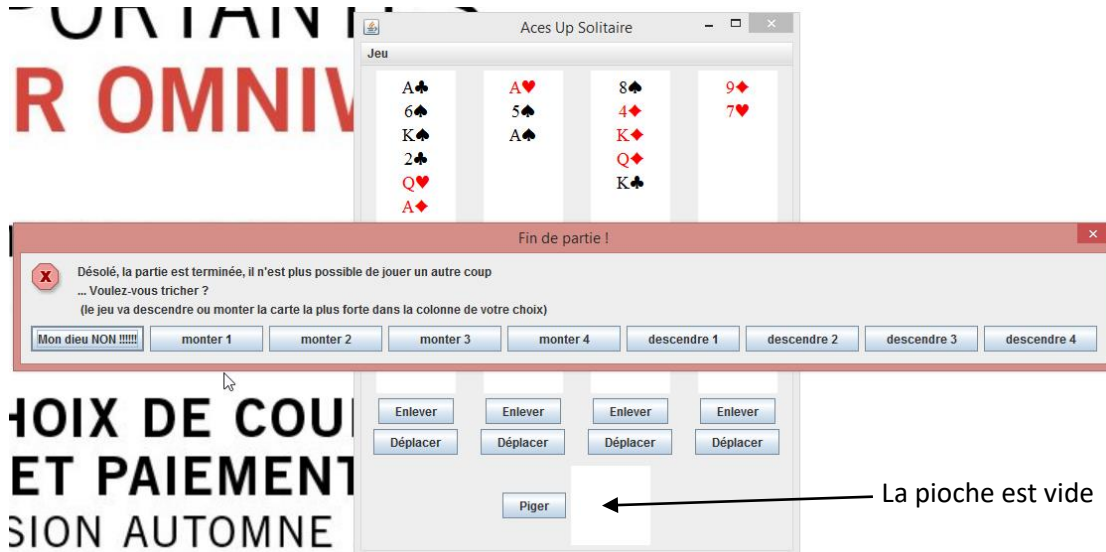
Règles du jeu :

- On commence la partie en mettant un paquet de cartes brassé, faces non-visibles, dans la pioche et on pige 4 cartes dans cette pioche, que l'on distribue dans les 4 colonnes de cartes.
- Pour pouvoir enlever une carte, il faut retrouver une carte de même sorte, mais plus forte (en termes de sa valeur) que la carte qu'on veut enlever dans le bas d'une autre colonne. (étant donné que l'as est, dans ce jeu, la carte la plus forte, il n'est donc pas possible d'enlever un as dans le bas d'une colonne). Autrement dit, dès que deux cartes ou plus des 4 cartes du bas des colonnes sont de la même sorte, on peut écarter les plus basses et garder la plus forte (en utilisant le bouton « Enlever » ou le clic gauche de la souris sur la colonne ciblée).
- Quand une colonne est vide, on peut y déplacer une carte d'une des autres colonnes (en utilisant le bouton « Déplacer » ou le clic droit de la souris sur la colonne source).
- Si plus aucune carte ne peut être enlevée ou déplacée et qu'il reste des cartes dans la pioche, on épuise la pioche, 4 cartes à la fois, que l'on place au bas de chacune des colonnes (on peut piger 4 cartes en utilisant le bouton « Piger » ou le clic sur la roulette de la souris).
- Le but du jeu est de se retrouver seulement avec les 4 as dans chacune des colonnes lorsque la pioche est vide.
- Si la Pioche est vide et qu'il n'y a plus de jeux possibles (qu'il y a une couleur (sorte) différente dans chacune des colonnes), la partie est perdue.

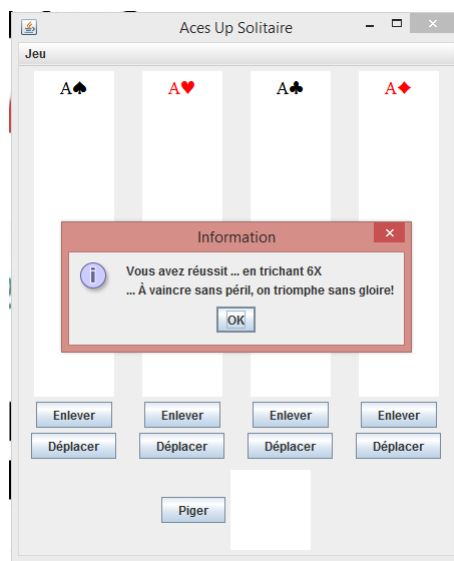
Situation après quelques coups :



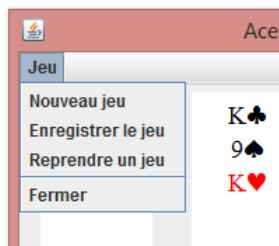
Une partie nulle, mais on peut tricher :



Si on termine en trichant :



Les options du menu « Jeu » :



Matériel disponible pour cette partie:

Vous allez créer une variante du jeu de patience « Aces Up Solitaire » à partir des éléments de base suivants :

- Le code source de départ sur le réseau;
- La JavaDoc fournie pour les classes NoeudPile, Pile, Carte, PaquetDeCartes et Pioche;
- Les règlements du jeu ci-dessus;
- Un lien Web pour jouer à une variante du jeu¹:o) : <http://www.novelgames.com/en/spgames/acesup/>
- Le diagramme de classes disponible à la fin du présent document peut vous aider à mieux comprendre.

Travail à faire dans cette partie :

Pendant la prochaine semaine on veut que vous réalisiez le travail suivant:

- Créez le code des classes NoeudPile, Pile, Carte, PaquetDeCartes et Pioche à partir du code source et de la JavaDoc fournis;
- Transférez et complétez correctement la JavaDoc de ces classes;
- Faites les tests unitaires de ces classes en utilisant JUnit

Points importants pour vous aider:

- Prenez le temps de comprendre le travail qui est à faire dans cette 1^{re} partie, pour ne pas développer plus que demandé;
- Mettez votre projet en UTF-8 dans « Éclipse » avant de copier le code fourni;
- Complétez vos tests au fur et à mesure de votre développement;

Remise :

Il n'y a pas de remise pour cette partie. La remise se fera lorsque la partie 2 sera terminée.

Diagramme de classes pour cette partie : (page suivante)

¹ N'oubliez pas que ce travail doit vous apprendre à programmer et non à jouer aux cartes ! 😊

