

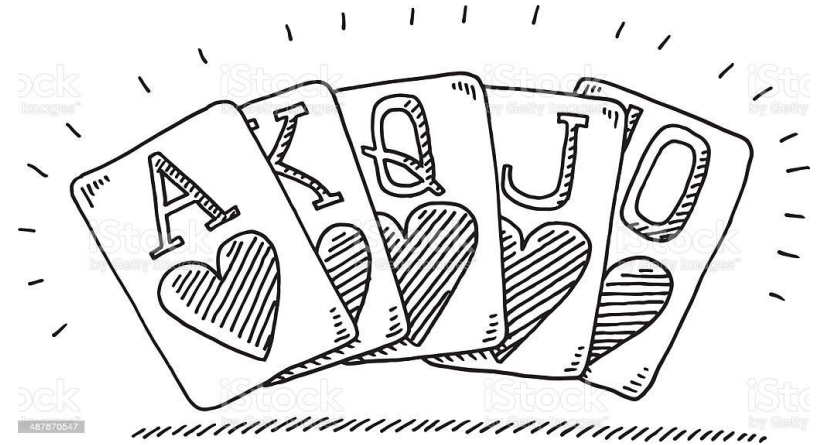
Poker - équipe GD5



Différences par rapport au TD1

- Création de classes enum pour Suit, Rank et Rankings
- Création d'un deck de 52 cartes, créé pour les tests random, pour éviter de rentrer une carte en double et pour plus de réalisme.
- Création d'un classement des joueurs après la partie.
- Création d'un game setting avec nombre de joueurs, de cartes et mode de piochage aléatoire ou non
- Ajout d'une classe Combinaison permettant de détecter la meilleure combinaison d'une main à l'aide des deux tableaux (suit et rank) et du nombre consécutif de cartes dans une main.
- Tests JUnit sur nos 5 classes principales.

Démonstration



Confiance



- Parties en confiance:

- Les slices du TD1
- Nos divers ajouts après le TD1 (input, comparaison, affichage...)

- Parties pas en confiance:

- Tests Junit
- La méthode ajoutée pour se caler sur l'exemple d'entrée dans la console, un string avec tout la main (dans la classe InputReader).

Tests



Qualité

- Parties de bonne qualité:

- La classe Card
- La classe Combinaison
- La classe deck



- Parties de mauvaise qualité:

- Les Tests
- Classe comparator -> on peut réduire le nombre de classes de 2 en les incluant dans Hand et Card.

Ce que nous a apporté ce projet

- Un bon découpage du projet en slices est très important pour un bon déroulement du projet et pour ne pas avancer à l'aveugle
- De solides bases en Java
- Une introduction aux tests JUnit
- Apprentissage de l'outil de github très utile pour l'avancée d'un projet en groupe, malgré des débuts difficiles.



Les améliorations ou fonctionnalités à ajouter

- Ajouts de règles du poker tel que:
 - Apparition de jetons de mise
 - Passage au Texas Hold'em
- Une interface graphique pour un meilleur confort de jeu
- Ajout de probabilités de victoire pour aider le joueur
- Dans l'absolu pouvoir jouer sur un réseau local ?



Petit hors projet : IHM (bonus)

- Démonstration avec interface graphique utilisant java swing



Merci pour votre attention !

Place aux

