

# Document de Première Étape

PICHARD Silvestre, DANET Luca

## ONITAMA



pioche : Cobra

main rouge : Boeuf      Tigre

PPRPP

.....

.....

.....

PPRPP

main bleue : Dragon      Mante

Au tour du joueur Bleu

Joueur Bleu choisissez une pièce à jouer

## DESCRIPTION DU PROJET

Dans *Onitama*, chaque joueur incarne un maître d'arts martiaux accompagné de ses disciples venus affronter un adversaire. Le jeu se déroule sur un terrain 5x5 et chaque joueur possède 5 pions : 1 maître (ou Roi) et 4 disciples (ou Pions).

Contrairement aux échecs, le mouvement des pièces n'est pas régi par leur nature (Roi ou Pion) mais par des cartes. Onitama comporte 16 cartes uniques, chacune permettant un certain nombre de déplacements. Ces cartes sont nommées en référence à des animaux, et les mouvements qu'elles proposent permettent d'élaborer des stratégies plus ou moins agressives. En début de partie, 5 cartes sont tirées au hasard : seules ces cartes seront utilisées pendant cette partie. Chaque joueur a 2 cartes dans sa main, la dernière est mise entre les 2 joueurs (pioche).

A chaque tour, le joueur choisit la pièce qu'il souhaite jouer, et la carte qu'il souhaite utiliser. Une fois la pièce déplacée en accord avec les mouvements proposés par cette carte, cette dernière est mise dans la pioche et remplace la carte qui y était rangée. Le joueur récupère donc la carte qui était défaussée, et aura de nouveau deux cartes au prochain tour.

L'équipe bleue est toujours la première à jouer.

La partie peut être gagnée de deux manières :

1. En mangeant le Roi adverse.
2. En atteignant l'autel adverse. "L'autel" désigne la case d'où commence le Roi.

## DÉROULEMENT D'UNE PARTIE "IN GAME"

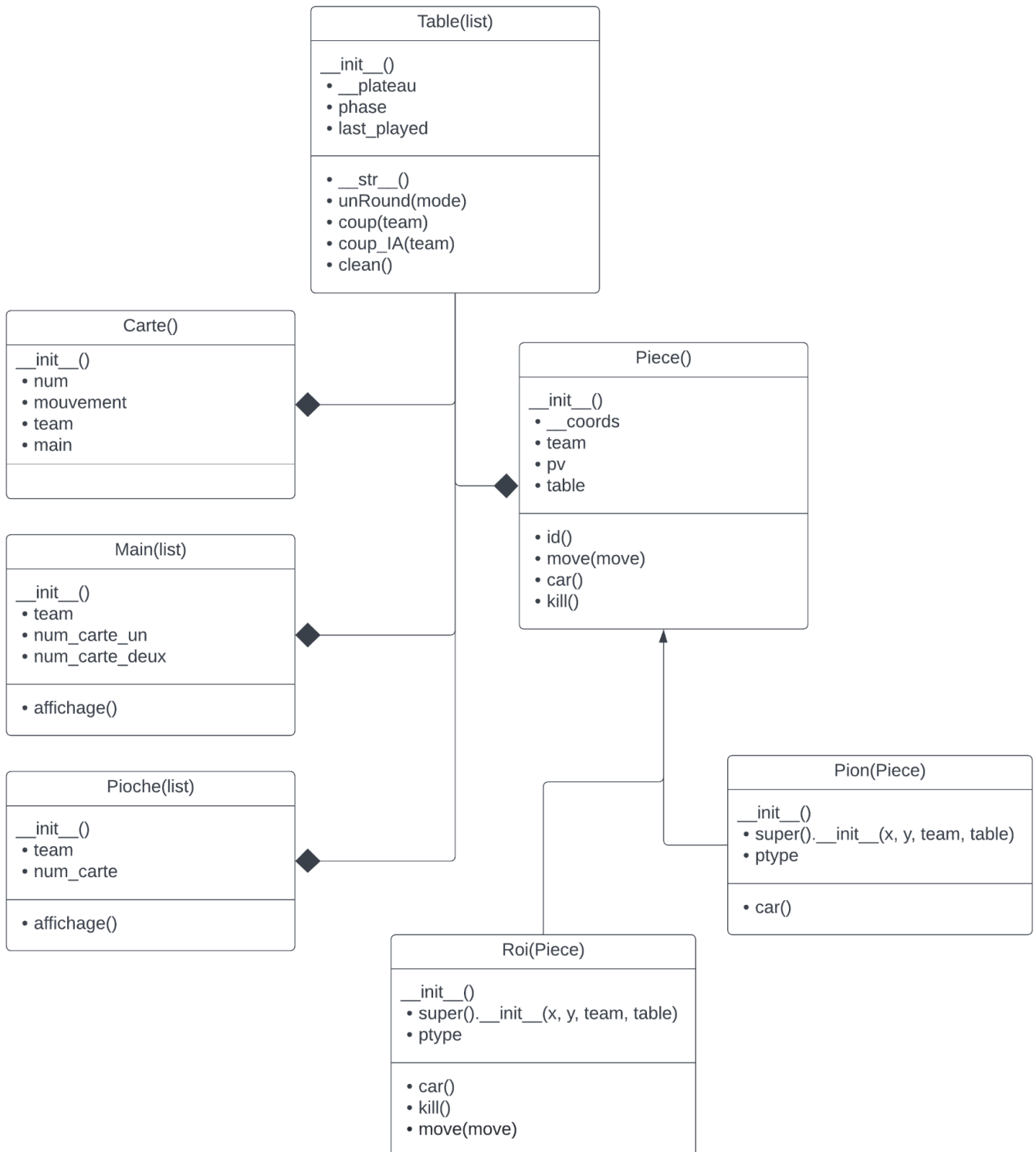
A l'exécution du main de **"table.py"**, les instructions sont données dans la fenêtre d'exécution. Il faudra ensuite entrer les données demandées :

- Mode de jeu : **"PVP" ou "PVE"**

*En mode "PVE", le joueur affronte une IA "naïve", qui va jouer aléatoirement selon les cartes de sa main et les pièces de sa couleur. Elle peut essayer de jouer un coup interdit, ce qui lui fera perdre son tour. Attention : la taille réduite du plateau la rend dangereuse malgré tout. Le mode PVP (deux joueurs) n'est pas encore fonctionnel.*

- Couleur : **"Bleu" ou "Rouge"**
- Coordonnées de la pièce : **"i", puis "j"**. Numéro de ligne (entre 0 et 4), puis (après avoir validé le numéro de ligne), Numéro de colonne (entre 0 et 4).
- Carte : **"gauche" ou "droite"**. Se référer à l'affichage de la table de jeu. Deux cartes sont dans votre main, affichées l'une après l'autre (et donc la première à gauche, la seconde à droite).
- Mouvement : **"n"**. Se référer aux propositions faites, selon la carte choisie.

## CLASSES DÉFINIES



## FONCTION PRINCIPALES

Les fonctions principales sont *str*, *unRound* et *coup de Table*, *move de Piece* et *choix\_carte* de *carte.py*.

- **\_\_str\_\_()** : Affiche le plateau de jeu sous forme de matrice. Les pièces sont représentées par une lettre (R pour les Rois, P pour les Pions. Les cases vides sont représentées par un point. L'équipe Bleue commence en bas, face à l'équipe rouge.
- **unRound(mode)** : Gère le déroulé d'un tour complet (un coup de chaque joueur). Prends en compte le mode de jeu (PVP ou PVE) et l'équipe. Engage le coup du joueur dont c'est le tour.
- **coup(team)** : Gère le déroulé du coup d'un joueur (humain. Une fonction particulière définit le coup de l'IA). Prends en argument l'équipe du joueur dont c'est le tour. La fonction demande en input les coordonnées de la pièce que le joueur souhaite jouer, puis (si une pièce est trouvée) la carte que le joueur souhaite utiliser. La fonction propose ensuite les mouvements possibles, et engage le mouvement de la pièce via la fonction *move()*.
- **move(move)** : Prends en argument les "coordonnées" du mouvement (les composantes du vecteur déplacement). Vérifie également que le joueur ne cherche pas à superposer deux de ses pièces, "mange" une pièce adverse au besoin. Enfin, la fonction déplace la pièce dans la case voulue (si possible). Si un mouvement impossible est tenté, le joueur perd son tour.
  - La sous-classe *Roi* possède sa propre fonction *move()*. Celle-ci possède un module supplémentaire qui vérifie si le Roi est placé sur l'Autel adverse, auquel cas la partie est gagnée.
- **choix\_carte()** : Pioche dans le dictionnaire des 16 cartes d'Onitama 5 cartes qui seront utilisées dans cette partie.

## **TESTS**

### **Test des cartes :**

- Vérifie l'attribution des numéros aux cartes
- Vérifie l'attribution des équipes aux cartes
- Vérifie l'attribution des noms aux cartes
- Vérifie l'attribution des mouvements possibles aux cartes
- Vérifie l'attribution des mains aux cartes

### **Test des pièces :**

- Vérifie l'attribution des coordonnées aux pièces
- Vérifie l'attribution des équipes aux pièces
- Vérifie le fonctionnement du déplacement des pièces
- Vérifie l'attribution des points de vie aux pièces
- Vérifie le fonctionnement de la mort des pièces

### **Test des Rois :**

- Vérifie l'attribution des coordonnées aux Rois
- Vérifie l'attribution des équipes aux Rois
- Vérifie le fonctionnement du déplacement des Rois
- Vérifie l'attribution des points de vie aux Rois
- Vérifie le fonctionnement de la mort des Rois
- Vérifie la condition de victoire sur la mort du Roi

### **Test des Pions :**

- Vérifie l'attribution des coordonnées aux Pions
- Vérifie l'attribution des équipes aux Pions
- Vérifie le fonctionnement du déplacement des Pions
- Vérifie l'attribution des points de vie aux Pions
- Vérifie le fonctionnement de la mort des Pions

## ANNEXES

### Cartes du jeu :

Les fonctionnalités particulières de la Grenouille, de l'Autruche et de la Tortue ne sont pas encore implémentées.

