

## EXERCICE 4 (4 points)

Cet exercice porte sur les structures de données (programmation objet).

Simon souhaite créer en Python le jeu de cartes « la bataille » pour deux joueurs. Les questions qui suivent demandent de reprogrammer quelques fonctions du jeu. On rappelle ici les règles du jeu de la bataille :

### Préparation

- Distribuer toutes les cartes aux deux joueurs.
- Les joueurs ne prennent pas connaissance de leurs cartes et les laissent en tas face cachée devant eux.

### Déroulement

- A chaque tour, chaque joueur dévoile la carte du haut de son tas.
- Le joueur qui présente la carte ayant la plus haute valeur emporte les deux cartes qu'il place sous son tas.
- **Les valeurs des cartes sont** : dans l'ordre de la plus forte à la plus faible : As, Roi, Dame, Valet, 10, 9, 8, 7, 6, 5, 4, 3 et 2 (la plus faible)

Si deux cartes sont de même valeur, il y a "bataille".

- Chaque joueur pose alors une carte face cachée, suivie d'une carte face visible sur la carte dévoilée précédemment.
- On recommence l'opération s'il y a de nouveau une bataille sinon, le joueur ayant la valeur la plus forte emporte tout le tas.

Lorsque l'un des joueurs **possède toutes les cartes du jeu**, la partie s'arrête et ce dernier gagne.

Pour cela Simon crée une classe Python `Carte`. Chaque instance de la classe a deux attributs : un pour sa valeur et un pour sa couleur. Il donne au valet la valeur 11, à la dame la valeur 12, au roi la valeur 13 et à l'as la valeur 14. La couleur est une chaîne de caractères : "trefle", "carreau", "coeur" ou "pique".

1. Simon a écrit la classe Python `Carte` suivante, ayant deux attributs `valeur` et `couleur`, et dont le constructeur prend deux arguments : `val` et `coul`.
  - a. Recopier et compléter les  des lignes 3 et 4 ci-dessous.

```
1. class Carte:
2.     def __init__(self, val, coul):
3.         valeur = 
4.          = coul
```

b. Parmi les propositions ci-dessous quelle instruction permet de créer l'objet « 7 de cœur » sous le nom `c7` ?

- `c7.__init__(self, 7, "cœur")`
- `c7 = Carte(self, 7, "cœur")`
- `c7 = Carte(7, "cœur")`
- `from Carte import 7, "cœur"`

2. On souhaite créer le jeu de cartes. Pour cela, on écrit une fonction `initialiser()` :

- sans paramètre
- qui renvoie une liste de 52 objets de la classe `Carte` représentant les 52 cartes du jeu.

Voici une proposition de code. Recopier et compléter les lignes suivantes pour que la fonction réponde à la demande :

```
def initialiser() :  
    jeu = []  
    for c in ["cœur", "carreau", "trefle", "pique"] : # couleur carte  
        for v in range(...) : # valeur carte  
            carte_cree = ...  
            jeu.append(carte_cree)  
    return jeu
```

3. On rappelle que dans une partie de bataille, les deux joueurs tirent chacun une carte du dessus de leur tas, et celui qui tire la carte la plus forte remporte les deux cartes et les place en dessous de son tas.

Parmi les structures linéaires de données suivantes : Tableau, File, Pile, quelle est celle qui modélise le mieux un tas de cartes dans ce jeu de la bataille ? Justifier votre choix.

4. Écrire une fonction `comparer(cartel, carte2)` qui prend en paramètres deux objets de la classe `Carte`. Cette fonction renvoie :

- 0 si la force des deux cartes est identique,
- 1 si la carte `cartel` est strictement plus forte que `carte2`
- -1 si la carte `carte2` est strictement plus forte que `cartel`