

Uno en ligne de commande



NOM et Prénom

I - Questions

- 1) Donnez les propriétés de la **classe UnoCarte** :

Rép : `self.Valeur, self.Couleur, self.EstPlus4, self.EstPlus2, self.EstJoker` _____

- 2) Citez une méthode de la **classe UnoJeu** qui **renvoie** un résultat :

Rép : `DerniereCarteVisible` _____

- 3) Citez une méthode de la **classe UnoJeu** qui ne fait que **modifier** l'environnement de l'instance de classe :

Rép : `J1_Joue` _____

- 4) Complétez la **fonction** `__init__` de la **classe UnoCarte**

- 5) Complétez la **fonction** `__init__` de la **classe UnoJeu** pour qu'elle :

- lance la procédure `CreerJeu()` de son instance de classe.
- crée deux propriétés `self.J1` et `self.J2` qui contiendront le jeu des joueurs 1 et 2, tous les deux créés à partir de la méthode `InitialiserJoueur()`.

- 6) Complétez la méthode `CreerJeu()` pour qu'elle **initialise un jeu** contenant toutes les cartes données dans la partie I (sauf les cartes "changement de sens").

- 7) A quoi sert l'instruction : `random.shuffle(self.CartesCachees)` ?

Rép : Permet de mélanger aléatoirement les éléments de la liste `CartesCachees`. _____

- 8) Complétez la fonction `MontrerJeuJ1()` et `MontrerJeuJ2()` pour qu'elles donnent le même résultat que l'exemple ci-dessus.

- 9) Compléter la méthode `DerniereCarteVisible()`.

- 10) Compléter la méthode `MontrerDerniereCarteVisible()`. On pourra bien sûr utiliser la méthode `DerniereCarteVisible()`.

- 11) Compléter l'en-tête de la méthode `PeutJouer()`. Celle-ci est liée à l'instance de classe, et comprend un paramètre `c` qui est la carte choisie par le joueur.

- 12) Compléter la **fonction** `J1_Joue` pour réaliser les instructions données en commentaires. On pourra s'inspirer de l'exemple ci-dessus pour écrire les messages de retour (XXX pose la carte YYY...). On complétera de même `J2_Joue`.

- 13) Ce jeu est proposé pour deux joueurs, mais le jeu de Uno devient encore plus intéressant à 3, 4 ou 5 joueurs... Proposez (en expliquant votre démarche ci-dessous) une amélioration à ce programme qui permettra de passer sur un jeu avec plus de joueurs.

Rép : On pourra créer un nouvel attribut dans le constructeur de UnoJeu qu'on nommera

« nb_joueur » qui serait un input où l'on demande à l'utilisateur combien de joueurs il souhaite.

Ensuite toujours dans le constructeur, on va faire une boucle de nb_joueur pour créer les joueurs qu'on mettra dans une liste. Puis dans toutes les méthodes qui concernent le joueur 1 ou 2, on va les changer pour que les méthodes concernent le joueur qu'on aura choisi dans la liste. Exemple : on va passer de « def MontrerJeuJ1(self) » à « def MontrerJeuJ1(self, num_joueur) » et num_joueur sera un élément de la liste. Exemple, on a 4 joueurs qu'on aura mis dans une liste, pour faire jouer le premier joueur, num_joueur sera égal à 0, et dans la méthode « Joue » on changera la valeur de num_joueur afin de passer au joueur suivant. _____

II - Barème

Noms des fichiers remis	Date
	05/12/2022

Difficultés rencontrées
/

Avez-vous apprécié travailler sur ce projet ? Avez-vous d'autres idées ?
Oui on aime bien utiliser la P.O.O et faire des jeux tels que le Jeu des allumettes ou encore ce jeu de uno utilisant le cours sur la P.O.O .

Élément de notation	Barème	Note
Réponses aux questions	7	
Fonction ()	3	
Les différentes fonctions à compléter	3	
Les différentes méthodes à compléter	3	
Le fichier <i>à fournir</i>	4	

TOTAL	20	
--------------	-----------	--