

EXAMEN PRATIQUE #1

Le jeu du Morpion se joue sur une grille carrée de 3x3 cases. Deux joueurs s'affrontent. Ils doivent remplir, chacun à leur tour, une case de la grille avec le symbole qui leur est attribué : 0 ou X.

Le gagnant est celui qui arrive à aligner trois symboles identiques, horizontalement, verticalement ou en diagonale.

Vous allez réaliser une version simplifiée de ce jeu : seules les lignes horizontales complétées seront prises en compte pour déterminer le gagnant.

EXEMPLE D'EXÉCUTION :

Bienvenue au jeu du morpion

Joueur 1 - Symbole : x | Joueur 2 - Symbole : o

Plateau initial :

.	.	.	.
.	.	.	.
.	.	.	.

Début de la partie :

Joueur 1, quelle ligne ? **2**

Joueur 1, quelle colonne ? **2**

.	.	.	.
.	x	.	.
.	.	.	.

Joueur 2, quelle ligne ? **2**

Joueur 2, quelle colonne ? **2**

Cette case est déjà occupée, rejouez

Joueur 2, quelle ligne ? **1**

Joueur 2, quelle colonne ? **2**

.	o	.	.
.	x	.	.
.	.	.	.

Joueur 1, quelle ligne ? **4**

Veuillez entrer un nombre entre 1 et 3

Joueur 1, quelle ligne ? **2**

Joueur 1, quelle colonne ? **1**

.	o	.	.
x	x	.	.
.	.	.	.

Joueur 2, quelle ligne ? **1**

Joueur 2, quelle colonne ? **1**

o	o	.	.
x	x	.	.
.	.	.	.

Joueur 1, quelle ligne ? **2**

Joueur 1, quelle colonne ? **3**

o	o	.	.
x	x	x	.
.	.	.	.

Le gagnant est le joueur 1 qui a joué avec le symbole x

Le premier joueur se voit attribuer le symbole « x ». Le second joueur se voit attribuer le symbole « o ». Les cases disponibles dans le plateau de jeu sont représentées par le symbole « . ».

Au démarrage du jeu, un plateau de jeu est créé. Ce plateau est représenté par une liste 2D.

A tour de rôle, les deux joueurs doivent entrer les valeurs représentant la ligne et la colonne pour lesquelles ils désirent ajouter leur symbole.

Si la case n'est pas libre, ils en seront avertis et auront la possibilité de recommencer. Attention, seules les valeurs comprises entre 1 et 3 sont autorisées.

Dès qu'un joueur a réalisé une ligne **horizontale** complète, la partie se termine et le gagnant est affiché.

Si aucun des joueurs ne gagne et que le plateau est complet, le message suivant est affiché : **Il n'y a plus de cases de libre - Game Over !**

PHASE DE RÉALISATION

Dans un répertoire nommé `eval_janvier`, téléchargez le fichier nommé `morpion.py` se trouvant sur eLearn.

Différentes fonctions sont nécessaires à la réalisation du jeu et seront écrites dans ce fichier. A vous de les implémenter en tenant compte des signatures données ci-dessous.

Attention, si la docstring n'est pas donnée, vous devez l'écrire.

def initialiser_plateau(nbre_lignes):

```
"""
    Crée une liste 2D carrée (nombre de lignes = nombre de colonnes). Chaque
    élément de la liste contient le symbole '.'
    :param nbre_lignes: int, nombre de lignes
    :return: list, liste 2D dans laquelle chaque élément contient le symbole '.'
"""
```

⇒ Cette fonction permet de créer le plateau de jeu

def affichage_message_bienvenue():

⇒ Cette fonction permet d'afficher le message de bienvenue. A vous de l'implémenter en tenant compte du modèle d'exécution

def convertir_en_texte(matrice):

```
"""
    Convertit une liste 2D en texte - chaque élément est entouré par le symbole | et
    les éléments sont affichés 3 par 3 sur une ligne
    :param matrice: list, liste 2D
    :return: str, matrice convertie en texte
"""
```

⇒ Cette fonction permet de convertir la matrice représentant le plateau de jeu en une chaîne de caractères.

⇒ *Exemple :*

Pour la matrice:

```
[
    [0,0,0,0],
    [x,x,x],
    [.,.,.]
]
```

cette fonction produira le texte :

```
| o | o | . |
| x | x | x |
| . | . | . |
```

def afficher(matrice):

⇒ Cette fonction permet d'afficher une matrice convertie en chaîne de caractères à l'écran.

def obtenir_symbole(num_joueur):

```
"""
Retourne le symbole correspondant au numéro de joueur "num_joueur" ou le symbole
de la case vide.
Le paramètre num_joueur peut être soit 0, soit 1, soit 2.
:param num_joueur: int, numéro du joueur
:return: string, symbole 'x' représentant le joueur 1 - symbole 'o'
représentant le joueur 2, symbole '.' représentant une case vide. None,
dans les cas contraires
"""
```

⇒ A différents moments de l'exécution du programme, vous aurez besoin de récupérer le symbole du joueur actif ou le symbole de la case vide.

⇒ Cette fonction permet de récupérer le symbole correspondant du joueur ou de la case vide.

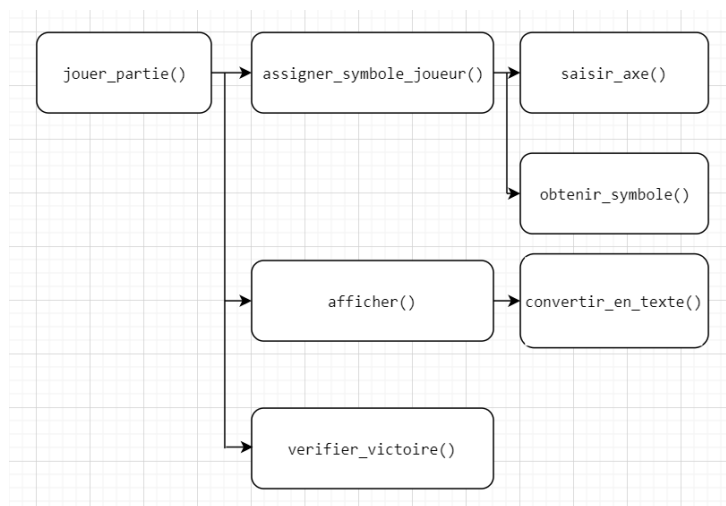
Exemple :

symbole(1) renverra x

symbole(2) renverra o

symbole(0) renverra .

Interaction avec le joueur



```
def saisir_axe(num_joueur, axe):
```

- ⇒ Cette fonction invite l'utilisateur à entrer soit le numéro de la colonne, soit le numéro de la ligne.
Exemple : **Joueur 1, quelle ligne ?** Les valeurs **1** et **ligne** dépendront des arguments donnés lors de l'appel de la fonction.

Exemple :

saisir_axe(2, colonne) donnera le message suivant : **Joueur 2, quelle colonne ?**

saisir_axe(2, ligne) donnera le message suivant : **Joueur 2, quelle ligne ?**

- ⇒ Seules les valeurs comprises entre 1 et 3 sont autorisées.
⇒ La fonction renverra le numéro valide

```
def assigner_symbole_joueur(plateau, num_joueur)
```

```
    """
    Assigner le symbole de l'utilisateur à un élément de la liste 2D
    :param plateau: list, liste 2D représentant le plateau de jeu
    :param num_joueur: int, numéro du joueur actif
    :return: /
    """
```

- ⇒ Cette fonction récupère le numéro de la ligne et le numéro de la colonne valides, encodés par l'utilisateur (via la fonction `saisir_axe()`).
Ces 2 valeurs représentent la case de la grille.

Si la case est déjà occupée par le symbole d'un des joueurs, le message « **Cette case est déjà occupée, rejouez** » apparaît et l'utilisateur doit refaire une saisie. Dans le cas contraire, le symbole du joueur (récupéré via `obtenir_symbole()`) est attribué à la case.

```
def verifier_victoire(plateau):
```

```
    """
    Vérifie si une ligne horizontale est complète
    :param plateau: liste 2D
    :return: numéro de joueur ou 0 si pas de gagnant
    """
```

- ⇒ Cette fonction permet de vérifier s'il y a un gagnant. Dès qu'une ligne comprend 3 fois le même symbole, le joueur a gagné.

[INFO]

`list.count(x)` Renvoie le nombre d'éléments ayant la valeur `x` dans la liste.

[BONUS]

Chaque appel à la fonction `count(...)` entraîne un parcours de toute la liste. Ce parcours est répété plusieurs fois, en fonction des symboles à rechercher. Implémentez une solution qui n'utilise pas cette fonction et qui effectue le moins d'opérations possible

def jouer_partie(plateau):

```
"""
    Cette fonction gère la boucle de jeu jusqu'à ce qu'un gagnant soit trouvé ou
    que la plateau soit complet.
    :param plateau: list, liste 2D représentant le plateau de jeu
    :return: int, numéro du gagnant ou 0 pour le game over.
"""
```

⇒ Cette fonction représente la fonction principale du programme. Cette fonction a comme objectif de renvoyer soit le numéro du gagnant, soit 0 représentant le GAME OVER.

⇒ Les principales étapes de cette fonction sont

- L'assignation du symbole du joueur à la case choisie (via `assigner_symbole_joueur()`)
- l'affichage du plateau de jeu
- la vérification d'un gagnant

⇒ La fonction renverra soit le numéro du gagnant obtenu par la fonction (`vérifier_victoire()`) soit 0 en cas de game over

[SUGGESTION]

Utiliser des expressions régulières afin de sécuriser les saisies clavier

Veillez à respecter les bonnes pratiques de programmation ainsi que les conventions établies en laboratoire (mise en forme du code, commentaires, noms adaptés pour les variables, respect de la casse ...).

DÉPÔT VIA E-LEARNING

Créez une archive au format ZIP (pas .RAR, .7z ou autre) contenant votre fichier `morpion.py`.

L'archive doit porter le nom `SECSY_JANV_GRx_NOM_PRENOM`, où x doit être remplacé par votre numéro de groupe et où NOM_PRENOM doit être remplacé par vos nom et prénom (par exemple, `SECSY_JANV_GR3_LORQUET_C`).

[IMPORTANT] Le non-respect de l'une des consignes entraînera une pénalité de points.

L'accès aux ressources suivantes est strictement interdit :

- les notes de cours et/ou syllabus ;
- les ressources disponibles sur E-Learn ;
- vos documents personnels (qu'ils soient physiques ou virtuels) ;
- vos projets de laboratoires ;
- toute ressource internet autre que ***docs.python.org*** ;
- tout moyen de communication (qu'il soit vocal, écrit ou numérique).