

TP04

L'objectif de ce TP est de vous faire pratiquer l'utilisation des listes et affiner vos connaissances avec les fonctions.

1) Traitement de liste

Une liste de chiffre vous est transmise et on vous demande de réaliser quelques traitements sur celle-ci (dans le même ordre que l'affichage) :

- Afficher la liste (`affichage_liste(liste)`)
- Afficher la liste avec les indices (`affichage_liste_colonne()`)
- Afficher la liste avec les valeurs multipliées par 3 (`multiple_liste()`)
- Afficher le plus petit chiffre de la liste (`plus_petit_chiffre()`)
 - SANS UTILISER `min()`
- Afficher la somme des chiffres impairs (`somme_impair()`)

Exemple de sortie :

```
8, 2, 4, 6, 9, 1, 3
-----
Indices :  0   1   2   3   4   5   6
Valeurs : [ 8 | 2 | 4 | 6 | 9 | 1 | 3 ]
-----
24, 6, 12, 18, 27, 3, 9
-----
Plus petit chiffre : 1
-----
Somme impairs = 13
-----
```

2) **Mastermind**

Programme simulant un jeu de Mastermind. L'objectif du jeu est de trouver la combinaison secrète de couleur en un maximum de 12 essais.

Les couleurs possibles sont Rouge, Vert, Bleu, Orange, Jaune, Brun.

La combinaison est d'une taille de 4 couleurs qui ne sont pas forcément uniques.

Exemple : Rouge, Vert, Rouge, Bleu

Le joueur doit alors entrer 4 couleurs pour tenter de deviner cette combinaison secrète (qui est générée aléatoirement par votre code). Si une couleur est exactement à la même position que la suggestion du joueur, le programme va montrer un pion Noir. Si une couleur est présente dans la combinaison mais pas à la bonne place, le programme va montrer un pion Blanc.

Exemple :

Combinaison secrète : Rouge, Vert, Rouge, Bleu

Suggestion du joueur : Jaune, Rouge, Vert, Bleu

Réponse du programme : Noir, Blanc, Blanc

On peut remarquer que les pions Noirs sont toujours affichés en premier, puis les pions Blancs. Il n'y a donc pas de lien entre la position des pions Noirs/Blancs et la position des couleurs. Un pion vaut pour un des 4 choix uniquement. S'il y a deux fois la même couleur à la mauvaise position, il faut 2 pions blancs. Faites bien attention à tester tous les cas possibles !

Le jeu continue tant que l'utilisateur n'a pas trouvé la combinaison cachée ou dépasse le nombre d'essais.

Fonctions

- `generer_combinaison()`
 - Fonction qui va générer une combinaison aléatoire selon les couleurs définies.
- `jeu(combinaison_cachee: list)`
 - Fonction qui va gérer le déroulement du jeu tant que le nombre d'erreurs n'est pas atteint ou que le joueur n'a pas gagné.
- `suggestion_joueur()`
 - Fonction qui va demander à l'utilisateur des couleurs et composer puis retourner la liste des couleurs choisies.
 - Tant que le joueur n'entre pas une couleur correcte, le programme redemande des couleurs.
- `comparaison(suggestion_joueur: list, combinaison_cachee: list)`
 - Fonction qui va comparer les deux listes et retourner une liste contenant les pions Noirs/Blancs correspondants

Exemple de sortie

* C'est la combinaison secrète, vous pouvez l'afficher, c'est plus simple pour les tests, sa présence n'est pas un critère.

```
['Jaune', 'Orange', 'Bleu', 'Bleu'] *
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Vert
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Brun
['Bleu', 'Bleu', 'Vert', 'Brun']
['Blanc', 'Blanc']
Il vous reste 11 essais.
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Jaune
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Vert
['Bleu', 'Jaune', 'Bleu', 'Vert']
['Noir', 'Blanc', 'Blanc']
Il vous reste 10 essais.
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) :
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) :
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : test
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Jaune
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Vert
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Orange
['Bleu', 'Jaune', 'Vert', 'Orange']
['Blanc', 'Blanc', 'Blanc']
Il vous reste 9 essais.
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Jaune
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Orange
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Bleu
['Jaune', 'Orange', 'Bleu', 'Bleu']
['Noir', 'Noir', 'Noir', 'Noir']
Vous avez trouvé ! Bravo !

Process finished with exit code 0
```

Exemple de défaite (je vous épargne les 11 essais précédents)

```
Il vous reste 1 essais.  
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Brun  
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Brun  
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Brun  
Veuillez choisir une couleur (Rouge, Vert, Bleu, Orange, Jaune, Brun) : Brun  
['Brun', 'Brun', 'Brun', 'Brun']  
['Noir']  
Il vous reste 0 essais.  
Vous avez perdu ! La solution était : ['Jaune', 'Brun', 'Rouge', 'Rouge']  
  
Process finished with exit code 0
```