

▼ Chapitre 6: Fonctions - TP corrigé

▼ Exercice 1: Nombres premiers

Un nombre premier est un entier positif (>0) qui admet exactement deux diviseurs: 1 et lui-même. Ainsi, 1 n'est pas premier car il n'a qu'un seul diviseur entier positif. Zéro n'est pas premier non plus car il est divisible par tous les entiers positifs. Mais, 2, 3, 5, 7, 11, 13, 17, 19,..., sont tous premiers car leurs seuls diviseurs sont 1 et eux-mêmes. Au contraire, par exemple, 8 n'est pas premier (on dit qu'il est composé), car

$$8 = 2 \times 4 \text{ et } 8 = 1 \times 8.$$

Dans cet exercice, écrivez les définitions des fonctions suivantes:

1- *is_prime(p)* cette fonction prend un entier positif comme paramètre renvoi True si ce nombre est premier ou False s'il ne l'est pas.

Aide: pour tester si un nombre p est divisible par m, vous pouvez utiliser cette condition: $p \% m == 0$. Si vous trouvez un nombre m différent de 1 et de p pour lequel le reste est zéro, alors p n'est pas premier, comme m en est un diviseur.

- Appellez la fonction *is_prime* pour vérifier si ce nombre est premier.

➤ CORRECTION

[Afficher le code](#)

→ Entrer un nombre entier : 11
11 est un nombre premier.

▼ Exercice 2 : Saisies contrôlées

1- Définir la fonction *saisiePos* demandant la saisie d'un entier strictement positif et répétant la saisie jusqu'à ce que ce nombre soit correct. La valeur saisie par l'utilisateur sera retournée par la fonction.

➤ CORRECTION

[Afficher le code](#)

→ Saisir un nombre strictement positif
-4
Saisir un nombre strictement positif

```
-8
Saisir un nombre strictement positif
0
Saisir un nombre strictement positif
11
Le nombre saisi est : 11
```

▼ Question 2 :

- Écrire un programme demandant la saisie (contrôlée à l'aide de la fonction précédente) de deux nombres positifs et afficher leur somme.

➤ CORRECTION

[Afficher le code](#)

```
→ Saisir un nombre strictement positif
3
Saisir un nombre strictement positif
6
Somme des 2 nombres = 9
```

▼ Exercice 3 : Calcul de x^n quand n est un entier

- Définir la fonction `puissancePos` qui prend en paramètre un nombre x et un entier **positif** n et retourne la valeur x^n . Le calcul se fera sans utiliser l'opérateur `**`.
- **Remarque :** La fonction retournera 0 si l'entier passé en paramètre n'est pas positif.

➤ CORRECTION

[Afficher le code](#)

```
→ Entrez un nombre : 2
Entrez l'exposant : 5
2 élevé à la puissance 5 est : 32
```

▼ Question 2 :

- Définir la fonction `puissance` qui prend en paramètre un entier n et retourne la valeur x^n . La fonction prend en compte les cas où n est positif ou négatif, en **utilisant** la fonction `puissance_pos`

Rappel :

$$x^{-n} = \frac{1}{x^n}$$

> CORRECTION

[Afficher le code](#)

→ 8
0.125

▼ Exercice 4 : Affichage d'une liste

- Écrire une fonction qui **affiche** un tableau (liste)

> CORRECTION

[Afficher le code](#)

→ ◆ Affichage sans boucle :
[10, 20, 30, 40, 50]

=====

◆ Affichage avec for :
10
20
30
40
50

=====

◆ Affichage avec while :
10
20
30
40
50

=====

◆ Affichage avec for + range :
10
20
30
40
50

=====

◆ Affichage avec enumerate (indice + valeur) :

```
Indice 0 : 10
Indice 1 : 20
Indice 2 : 30
Indice 3 : 40
Indice 4 : 50
```

▼ Exercice 5 : Inverser une liste

- Écrire une fonction qui affiche une liste à l'envers de différentes façons, i.e. en commençant par la fin de liste

> CORRECTION

[Afficher le code](#)

```
→ Liste originale : [10, 20, 30, 40, 50]
Inversion en place : [50, 40, 30, 20, 10]
Inversion avec slicing : [50, 40, 30, 20, 10]
Liste originale après slicing : [10, 20, 30, 40, 50]
Inversion avec reversed() : [50, 40, 30, 20, 10]
Liste originale après reversed() : [10, 20, 30, 40, 50]
Inversion avec boucle for : [50, 40, 30, 20, 10]
Liste originale après boucle : [10, 20, 30, 40, 50]
```

▼ Exercice 6: Moyenne d'un tableau

- Écrire une fonction qui calcule la moyenne d'un tableau de nombres sans utiliser la fonction mean. On veillera à traiter le cas du tableau vide.

> CORRECTION

[Afficher le code](#)

```
→ Moyenne de nombres1 : 30.0
Erreur : le tableau est vide !
Moyenne de nombres2 : None
```

▼ Exercice 7 : Saisie d'un tableau

- Écrire une fonction qui stocke dans un tableau des entiers compris entre 0 et 20 saisis par l'utilisateur. La saisie s'arrête au premier nombre invalide et le tableau est renvoyé.
- Calculer la moyenne des valeurs du tableau grâce à la fonction de la question précédente.

> CORRECTION

[Afficher le code](#)

→ entrez un nombre compris entre 0 et 20: 5
entrez un nombre compris entre 0 et 20: 10
entrez un nombre compris entre 0 et 20: 15
entrez un nombre compris entre 0 et 20: 20
entrez un nombre compris entre 0 et 20: 25
moyenne: 12.5

✓ Exercice 8 : fusionne deux listes

- Écrire une fonction qui fusionne deux listes et retourne une nouvelle liste contenant tous les éléments des deux listes.

> CORRECTION

[Afficher le code](#)

→ Fusion avec '+': [1, 2, 3, 4, 5, 6]
Fusion avec `extend()` : [1, 2, 3, 4, 5, 6]
Fusion avec boucle `for`: [1, 2, 3, 4, 5, 6]

Profs. Echine K. / SEDKI A.
