

Exercice 1

Chacune des deux fonctions comporte une erreur. Les corriger.

```
def carre(x):
    return x**2
```

```
print(carre(1.7))
```

```
def quotient(x,y)
    return x/y
```

```
print(quotient(1.7,2.5))
```

Exercice 2

1. Écrire une fonction f qui renvoie la valeur $x^2 + 3x - 4$.
2. Compléter le programme pour qu'il affiche la représentation graphique de cette fonction sur $[-3; 3]$ en rouge avec la grille.

Exercice 3

1. Écrire une fonction f telle que

$$f(x) = \begin{cases} 2x + 5 & \text{si } x > 1 \\ -x + 8 & \text{sinon} \end{cases}$$

2. Bonus : Compléter le programme pour qu'il affiche la représentation graphique de cette fonction sur $[-3; 3]$ en rouge avec la grille.
Si vous procédez comme dans l'exercice 2, vous observerez une erreur. En effet, x est une liste donc Python a du mal à comparer une liste au nombre 1!
Essayer de contourner le problème.

Exercice 4

Écrire trois fonctions effectuant les calculs suivants :

1. la puissance 5 d'un entier donné en paramètre
2. le produit de deux entiers donnés en paramètre moins leur somme
3. le produit de trois entiers donnés en paramètre au carré

Exercice 5

Écrire une fonction "somme(n)" qui renvoie la somme de tous les entiers de 0 à n .

Exercice 6

Écrire une fonction :

1. "maxim(a,b,c)" qui renvoie le maximum de trois nombres a, b, c .
2. "som1(a,b)" qui renvoie la somme des carrés des entiers compris (au sens large) entre a et b
3. "fact(n)" qui renvoie le produit des entiers naturels de 1 à n
4. "som2(n,p)" qui renvoie $\sum_{i=1}^n \sum_{j=i}^p j^i$ (vous pourrez vérifier que $\text{somme}(6,6)=60277$ et $\text{somme}(5,8)=70676$)

Exercice 7

Algorithme de Syracuse

Pour tout nombre entier n non nul, s'il est pair on le divise par 2, sinon on le multiplie par 3 et on ajoute 1.

En répétant ce processus, on constate que l'on aboutit toujours au cycle 4-2-1.

Écrire une fonction "syracuse(n)" qui affiche la suite des nombres de n jusqu'à 1 et renvoie le nombre d'itérations effectuées.