

1 Entrée / Sorties & Structures conditionnelles

1 Exercice 1

1. L'instruction booléenne ' **True or False** ' sous Python renvoie : _____?
2. Pour convertir la chaîne "-32.7" en grandeur numérique, on utilise : _____?
3. Si $a = 17$, $b = 23$ et $c = 9$ alors la condition logique ' $b \geq c$ **or** $\text{not}(a > a)$ ' est : _____?
4. Combien vaut la variable **f** à la fin de ce programme : _____?

```
a=7
b=False
c=16
a=c<a
c=b==a
f=c!=a
```

2 Exercice 2

Écrire un programme permettant de lire la température exprimée en degré Celsius de la console, la convertir en Fahrenheit et affiche le résultat.

N.B : La formule utilisée pour la conversion est la suivante :

$$fahrenheit = \frac{9}{5} \times celsius + 32.$$

3 Exercice 3

Écrire un programme permettant de calculer la distance entre deux points $A(x_A, y_A)$ et $B(x_B, y_B)$ dont les coordonnées sont saisies au clavier.

4 Exercice 4

Les habitants de paris paient l'impôt selon les règles suivantes :

- les hommes de plus de 20 ans paient l'impôt
- les femmes paient l'impôt si elles ont entre 18 et 35 ans
- les autres ne paient pas d'impôt

Le programme demandera donc l'âge et le sexe (*M* ou *F*) du parisien, et se prononcera donc ensuite sur le fait que l'habitant est imposable.

Exemple d'exécution :

```
>>> donner l'âge : 25
>>> donner le sexe : M
>>> le citoyen est imposable
```

5 Exercice 5

Écrire un programme qui fait la résolution des équations du premier degré de la forme :

$$aX + b = 0, \quad a, b \in \mathbb{R}.$$

6 Exercice 6

Écrire un programme qui fait la résolution dans \mathbb{R} des équations du second degré de la forme :

$$aX^2 + bX + c = 0, \quad a \in \mathbb{R}^* \text{ et } b, c \in \mathbb{R}.$$

2 Boucle While

7 Exercice 7

Écrire un programme qui calcule le reste de la division entière d'un entier positif A sur un entier strictement positif B entrés par l'utilisateur sans utiliser les deux opérateurs `//`: division et `%`: modulo".

8 Exercice 8

Écrire un programme qui calcule le **PGCD** (Plus Grand Diviseur Commun) de deux nombres entiers strictement positif a et b entrés par l'utilisateur.

9 Exercice 9

Calculez le nombre lu à rebours d'un nombre positif entré au clavier.

Exemple :

```
>>> donner un nombre : 12345
>>> 54321
```

10 Exercice 10

(Simulation d'un jeu de hasard)

Un jeu de hasard se déroule de la façon suivante: On paie 2 euros pour jouer puis on lance 2 dés non truqués. Si le joueur obtient un double, il récupère sa mise et reçoit la somme des points marqués. Sinon il ne reçoit rien et perd sa mise.

- ① Écrire un programme pour simuler ce jeu une seule fois.

N.B : On peut utiliser la fonction **random.randint(1,6)** du module **random** pour générer un entier aléatoire dans $[1, 6]$ de la façon suivante :

```
import random
x=random.randint(1,6) # x contient maintenant un entier entre 1 et 6.
```

- ② Écrire un programme qui permet de compter le nombre de fois qu'on gagne parmi n tentatives (n étant saisie par l'utilisateur).
- ③ Conseilleriez-vous ce jeu?

11 Exercice 11

(Facteurs premiers)

Écrire un programme qui décompose un nombre entier en un produit de facteurs premiers.

Exemple :

```
>>> donner un entier : 8
>>> n est positif, suite du programme
>>> facteur trouvé: 2
>>> facteur trouvé: 2
>>> facteur trouvé: 2
>>> Fin du programme
```

3 Boucle For

12 Exercice 12

Écrire un programme qui lit n nombres réels et affiche leur moyenne.

13 Exercice 13

- ① Écrire un programme qui détermine si un nombre entier > 2 entré par l'utilisateur est premier ou non.
- ② Afficher tous les nombres premiers qui sont inférieurs ou égal à 100.

14 Exercice 14

- ① Écrire un programme qui détermine si un entier > 2 saisi par l'utilisateur est parfait ou non.
N.B : Un nombre est parfait si égal à la somme de ses diviseurs stricte (sauf lui-même).
 $N = 6$ est parfait car $6 = 1 + 2 + 3$.
- ② Afficher tous les nombres parfaits qui sont inférieurs ou égal à 100.

15 Exercice 15

Écrire un programme qui lit n nombres entiers et détermine quel est le max de ces nombres.

16 Exercice 16

Écrire un programme qui lit n nombres entiers et détermine est ce que ces entiers ont été entrés triés dans un ordre croissant.

4 Fonctions

17 Exercice 17

- ① Écrire une fonction **MINI(a,b)** et une fonction **MAXI(a,b)** qui déterminent le minimum et le maximum de deux nombres réels.
- ② Écrire un programme se servant des fonctions *MINI* et *MAXI* pour déterminer le minimum et le maximum de quatre nombres réels entrés au clavier.

18 Exercice 18

Écrire une fonction qui permet d'afficher la table de valeurs d'une fonction f . Par exemple si $f(x) = \sin(x) + \ln(x) - \sqrt{x}$, Où x est un entier compris entre 1 et 10.

19 Exercice 19

Écrire une fonction qui compte le nombre de voyelles (minuscules, sans accents) dans une chaîne de caractères u passée en argument :

```
>>> Donner une chaîne : dut idsd 1 est essaouira
>>> le nombre de voyelle est : 9
```

20 Exercice 20

En utilisant une boucle `while`, écrire la fonction **logarithme_binaire(a)**, qui accepte comme paramètre un réel strictement positif et qui donne comme résultat :

$$\min\{n \text{ entier naturel}, 2^n > a\}.$$

21 Exercice 21

- ① Écrire une fonction **premier(n)** qui renvoie *True* si un entier naturel n est premier ou *False* sinon.
- ② En utilisant une boucle `while`, écrire une fonction **plus_petit(n)** qui, à un entier naturel $n > 2$, associe son plus petit diviseur premier.

22 Exercice 22

un palindrome est un mot ou groupe de mots qui peut se lire indifféremment de gauche à droite ou de droite à gauche (ex. radar, rotor, kayak, Roma Amor).

Écrire une fonction **def estPalindrome (chaîne)** qui retourne vrai si la chaîne est un palindrome.

23 Exercice 23

La valeur de π peut être calculée en utilisant la série suivante:

$$m(i) = 4\left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{(-1)^{i+1}}{2i-1}\right) = 4 \sum_{k=1}^i \frac{(-1)^{k+1}}{2k-1}$$

Écrire une fonction qui renvoie $m(i)$ pour un i donné et écrire un programme de test qui affiche le tableau suivant :

```
i   m(i)
-----
1      4.0
101    3.1514934010709914
201    3.1465677471829556
301    3.1449149035588526
401    3.144086415298761
501    3.143588659585789
601    3.143256545948974
701    3.1430191863875865
801    3.142841092554028
901    3.1427025311614294
```