

Devoir Surveillé n°1 d'Informatique

Samedi 11 Octobre 2025 (*Durée : 2 heures 30 minutes*)

Exercice 1

1) Donner l'écriture décimale des entiers naturels représentés par :

$$(a) \overline{11011}^2 \quad (b) \overline{123}^8 \quad (c) \overline{FA}^{16}$$

2) Convertir l'entier naturel 75 en base :

$$(a) 2 \quad (b) 8 \quad (c) 16$$

3) Sans passer par leurs écritures décimales, convertir :

$$(a) \overline{111010}^2 \text{ en base } 16 \quad (b) \overline{EDF}^{16} \text{ en base } 2$$

Exercice 2

On souhaite échanger les contenus de deux variables a et b , déjà présentes en mémoire.

1) Parmi les codes suivants, lesquels répondent au problème ? Justifier les réponses en utilisant, pour chaque code, un tableau d'exécution donnant l'état des variables à chaque étape.

$$A) \begin{array}{|l} c=a \\ b=c \\ a=b \end{array}$$

$$B) \begin{array}{|l} c=a \\ a=b \\ b=c \end{array}$$

$$C) \begin{array}{|l} a=a+b \\ b=a-b \\ a=a-b \end{array}$$

$$D) \begin{array}{|l} a=a+b \\ b=a-b \\ a=a+b \end{array}$$

2) Proposez un code différent répondant au problème.

Exercice 3

Le but de cet exercice est d'écrire un programme qui, partant d'une durée donnée en secondes, l'exprime en jours, heures, minutes et secondes.

Exemple : 95 381 secondes correspondent à 1 jour 2 heures 29 minutes 41 secondes.

1) Que fait le programme suivant ?

```
d=eval(input('durée en secondes : '))
m=d//60
s=d%60
```

2) Écrire alors le programme complet qui demande une durée en secondes et l'affiche en jours, heures, minutes et secondes

Exercice 4

On considère l'algorithme suivant :

```
Demander à l'utilisateur un réel  $x > 1$ 
SI  $x > 1$ 
ALORS
     $N \leftarrow 0$ 
    TANT QUE  $x^N < 10^6$ 
         $N \leftarrow N + 1$ 
    Afficher N
SINON
    Afficher "x doit dépasser 1"
```

- 1) Que se passe-t-il lorsque l'utilisateur fournit pour valeur de x :
(a) 10 (b) 0 (c) 20
- 2) En termes simples, que fait donc cet algorithme ? À quel problème répond-il ?
- 3) Traduire cet algorithme en langage Python.

Exercice 5

Écrire un programme qui :

- ❶ demande un réel strictement positif ;
- ❷ réitère la demande jusqu'à ce que ce soit le cas ;
- ❸ affiche alors le logarithme népérien de ce nombre.

Exercice 6

On considère le programme suivant :

```
n=int(input())
for i in range(n):
    if i%2==0:
        carre=i**2
    else:
        carre=0
    total=total+carre
print(total)
```

- 1) Ce code contient plusieurs erreurs. Trouvez-les et corrigez le programme.
- 2) Que fait donc ce programme ?

Exercice 7

Soit L une liste non vide de nombres réels. On considère le code Python suivant :

```
m=L.pop()
while len(L)>0:
    if m<L.pop():
        m=L.pop()
print(m)
```

1) Que se passe-t-il dans les cas suivants ? Vous justifierez **précisément** vos réponses.

(a) $L=[1, 3, 2]$ (b) $L=[1, 2]$ (c) $L=[1, 3, 2, 4]$ (d) $L=[2, 1]$

Attention à bien simuler l'exécution du programme pas à pas, ce n'est pas si simple.

2) On aimerait que notre programme affiche le maximum de la liste L , quelle que soit la taille de cette dernière. Modifier le code en conséquence.

Il s'agit de corriger ce qui cloche ici, pas de changer radicalement la méthode de calcul !

Exercice 8

Anatole lance un dé cubique jusqu'à ce qu'il obtienne un 6. On souhaite conserver dans une variable `Lancers` tous les résultats des lancers, y compris le 6 qui clôt l'expérience.

- 1) Quel type de variable va-t-on utiliser pour cela ?
- 2) Écrire un programme qui permet de saisir un à un les résultats (il va réitérer des demandes jusqu'à ce que l'on rentre un 6) et qui affiche la variable `Lancers`.
- 3) On note N_1, N_2, N_3, N_4, N_5 les nombres de 1, 2, 3, 4, 5 obtenus à l'issue des lancers. Compléter le programme pour qu'il les calcule et affiche la liste $[N_1, N_2, \dots, N_5]$.

Rappels pour les manipulations de listes :

- ❶ `L = []` crée une liste vide nommée `L`
- ❷ `L.pop()` supprime le dernier élément de la liste `L` et le renvoie
- ❸ `L.append(x)` ajoute l'élément `x` à la fin de la liste `L`