
Série N° 1 (TD & TP)

Instructions itératives et Tableaux

Partie I : Instructions itératives

Exercice 1

Ecrire un algorithme qui permet de lire trois entiers a, b et c et détermine le maximum et le minimum des trois nombres.

Traduire cet algorithme en Python.

Exercice 2

Ecrire un algorithme et le programme en Python qui permet de calculer :

- La puissance x^n telle que x est un réel et n un entier positif.
- La factorielle d'un entier positif.
- La somme : $\sum_{i=0}^n i^m$

Exercice 3

Ecrire les algorithmes et les programmes en Python qui permettent de calculer la valeur des expressions suivantes :

- 1- $A = (1 + 2) \times (1 + 2 + 3) \times (1 + 2 + 3 + 4) \times \dots \times (1 + 2 + 3 + \dots + (N - 2) + (N - 1) + N, (N \geq 2)$
- 2- $B = 1 + \frac{1}{1+2} + \frac{1}{1+2+3} + \dots + \frac{1}{1+2+3+\dots+N}, (N \geq 2)$

Exercice 4

Ecrire un algorithme qui permet de déterminer tous les diviseurs d'un entier N saisi au clavier.

Traduire cet algorithme en Python.

Exercice 5

Ecrire un algorithme qui demande un entier positif puis vérifier si ce nombre est premier ou non.

Traduire cet algorithme en Python.

Exercice 6

Ecrire un algorithme qui permet de calculer la somme de tous les nombres impairs entre deux valeurs N et M ($1 \leq N < M$).

Traduire cet algorithme en Python.

Exercice 7

Ecrire un algorithme qui demande à l'utilisateur les longueurs des côtés d'un triangle (les longueurs sont des entiers positifs) et précise si ce triangle est rectangle ou non.

Traduire cet algorithme en Python.

Exercice 8

Ecrire un algorithme qui permet de convertir un nombre binaire N en décimal (Les valeurs 0 et 1 saisies par l'utilisateur sont stockées sous forme de caractères '0' et '1', et le mot binaire N est stocké dans une chaîne de caractères).

Exemple d'exécution :

Conversion binaire en décimal

Donner le nombre de bits : 8

Entrer le bit numéro 7 : 1

Entrer le bit numéro 6 : 0

Entrer le bit numéro 5 : 1

Entrer le bit numéro 4 : 1

Entrer le bit numéro 3 : 1

Entrer le bit numéro 2 : 0

Entrer le bit numéro 1 : 1

Entrer le bit numéro 0 : 0

N=10111010 Sa valeur décimale est : 186

Ecrire un programme en Python qui permet de convertir un nombre binaire N en décimal.

Exercice 9

Ecrire un algorithme qui permet la saisie d'un entier n, le calcul et l'affichage du n^{ième} terme de la suite (U_n) définie comme suit :

$$\begin{cases} U_0 = 2 \\ U_1 = 3 \\ U_{n+2} = \frac{2}{3}U_{n+1} - \frac{1}{4}U_n \end{cases}$$

Traduire cet algorithme en Python.

Exercice 10

La suite de Syracuse est définie selon une condition de parité comme suit :

$$u_0 \in \mathbb{N}^* \quad u_{n+1} = \begin{cases} \frac{1}{2}u_n & \text{si } u_n \text{ est paire} \\ 3u_n + 1 & \text{si } u_n \text{ est impaire} \end{cases}$$

La « conjecture tchèque » énonce que pour toutes les valeurs initiales $u_0 \in \mathbb{N}^*$ il existe un rang n pour lequel $u_n = 1$. Par exemple, si $u_0 = 6$ alors $u_8 = 1$.

n	0	1	2	3	4	5	6	7	8	9	10	...
u_n	6	3	10	5	16	8	4	2	1	4	2

Ecrire un algorithme qui demande à l'utilisateur la saisie de la valeur initiale u_0 et qui détermine et affiche la plus petite valeur de n vérifiant $u_n = 1$.

Traduire cet algorithme en Python.

Exercice 11

Pour approcher la limite hypothétique de la suite :

$$\begin{cases} U_0 = a \\ U_{n+1} = \frac{1 + U_n}{1 + 2U_n} \end{cases}$$

Trouver un algorithme qui calcule la valeur de la suite U_n tel que $|U_n - U_{n-1}| < \varepsilon$ où $\varepsilon = 10^{-6}$, (a est un réel strictement positif saisi au clavier).

Traduire cet algorithme en Python.

Exercice 12

Soit la suite $(X_n)_{n \in \mathbb{N}}$ suivante :

$$\begin{cases} X_0 = A \\ X_n = \left(X_{n-1} + \frac{A}{X_{n-1}} \right) / 2 \quad n \geq 1 \end{cases}$$

A est un nombre réel strictement positif.

Ecrire un algorithme qui demande à l'utilisateur la saisie de la valeur de A puis calcule et affiche la valeur de la suite (X_n) (Le point d'arrêt des itérations est $|X_n - X_{n-1}| < 10^{-9}$).

Implémenter cet algorithme en langage Python. Que calcule cet algorithme ?