Algorithmique et programmation Python

Atelier 3 : Boucles

Exercice 1

Ecrire un programme permettant calculer la somme des N premiers entiers : Somme ← 0 + 1 + … + N. N est donné par l'utilisateur

Exercice 2

Refaire l'exercice 1 pour calculer la somme des entiers compris entre N et M. N et M sont des entiers saisis au clavier

Exercice 3

Ecrire un programme permettant calculer le factoriel de N : Fact ← 1 * 2 * … * N. N est donné par l'utilisateur

Exercice 4

Ecrire un programme permettant de calculer la somme de tous les nombres impairs entre deux valeurs N et M.

Exercice 5

Ecrire un programme permettant de calculer la somme suivante :

$$S = \sum_{i=1}^{i=n} i^2$$

Exercice 6

Ecrire un programme permettant de calculer la somme suivante :

$$S = \sum_{i=1}^{i=n} (-1)^i \, (i^2 + i)$$

Exercice 7

Ecrire un programme permettant de calculer la somme suivante :

$$\frac{1}{1-x} = 1 + x + x2 + \cdots + xn$$

Exercice 8

Calculer une approximation de π en utilisant par exemple les deux résultats classiques :

$$\frac{\pi^2}{6} = \sum_{n=1}^{\infty} \frac{1}{n^2} \quad et \quad \frac{\pi}{2} = \sum_{n=1}^{\infty} \frac{4n^2}{4n^2-1}$$

Exercice 9 (Facultatif)

Pour illustrer les capacités de base de Python, nous proposons de calculer de manière numérique la valeur de $I = \int_a^b f(x)dx$, en utilisant trois méthodes classiques :

- la méthode des rectangles :$I \approx \sum_{i=0}^{n-1}(x_{i+1} - x_i) f\left(\frac{x_i + x_{i+1}}{2}\right)$
- la méthode des trapèzes : $I \approx h\left[\frac{f(a)+f(b)}{2} + \sum_{i=1}^{n-1} f(x_i)\right]$
- la méthode de Simpson : $I \approx \frac{h}{6}\left[f(a) + f(b) + 4\sum_{i=0}^{n-1} f(x_{2i+1}) + 2\sum_{i=1}^{n-1} f(x_{2i})\right]$ avec $h = \frac{b-a}{n}$ et $x_k = a + k\frac{h}{2}$

et où $(x_0 \cdots x_n)$ est une subdivision régulière de l'intervalle $[a, b]$ de pas h