

Boucles Exercices

- 1) Ecrire un programme qui calcule la somme des N premiers termes positifs.
- 2) Ecrire un programme qui calcule la somme des N premiers termes positifs impaires.
- 3) Ecrire un programme qui calcule la somme des N premiers termes positifs pairs non multiple de 3.
- 4) Ecrire un programme qui calcule la somme $1/2 + 1/4 + 1/8 + \dots + 1/2^n$ ($n \in \mathbb{N}^*$).
- 5) Ecrire un programme qui donne le nombre N tel que la somme $1+1/2-1/3+1/4-1/5+\dots\pm 1/N$ soit proche de 1,33 (avec erreur de $1/100$ près).
- 6) Ecrire un programme qui calcule la somme $1+x+x^2+\dots+x^n$ (x réel et n entier).
- 7) Un nombre est dit *parfait* s'il est égal à la somme de ses diviseurs, excepté lui-même. Ecrire un programme qui détermine si un nombre entier donné est parfait ou non.
- 8) Calculer pour une valeur X donnée du type réel la valeur numérique d'un polynôme de degré n :
$$P(X) = A_n X^n + A_{n-1} X^{n-1} + \dots + A_1 X + A_0$$
Les valeurs de n , des coefficients A_n, \dots, A_0 et de X seront entrées au clavier.
Utiliser le schéma de **Horner** qui évite les opérations d'exponentiation lors du calcul:

9) Nous nous intéressons à une suite de nombres entiers terminée par la saisie d'un zéro qui ne fait pas partie de la suite.

Exemple : 1, 5, 2, 8, 9, 3, 4, 0.

Ecrire un programme permettant de réaliser les tâches suivantes :

- a. Détermination du plus grand élément de la suite,
- b. Détermination du plus petit élément de la suite,
- c. Calcul de la somme des éléments de la suite,
- d. Calcul de la moyenne des éléments de la suite,
- e. Calcul de la somme des éléments pairs de la suite,

N.B. : Prévoir un affichage à l'écran de tous les résultats calculés.

10) En mathématiques, un **nombre presque parfait** (quelquefois appelé aussi *nombre légèrement déficient*) est un entier naturel n , tel que la somme de tous les diviseurs stricts de n est égale à $n - 1$, autrement dit : $\sigma(n) = n - 1$ où $\sigma(n)$ est la somme des diviseurs entiers positifs de n , avec n non compris.

Les premières valeurs sont : 1, 2, 4, 8, ...

Ecrire un programme C qui affiche tous les nombres **presque parfaits** compris entre les entiers **a** et **b** ainsi leurs nombres.

11) En mathématiques, un **nombre abondant** est un nombre entier naturel n qui est strictement inférieur à la somme de ses diviseurs stricts, autrement dit : $n < \sigma(n)$ où $\sigma(n)$ est la somme des diviseurs entiers positifs de n non compris.

Les premières valeurs sont : 12, 18, 20, 24, 30, 36, ...

Ecrire un programme C qui affiche tous les nombres **abondants** compris entre les entiers **a** et **b** ainsi leurs nombres.