



TD3 : Algorithmique 1

Exercice 1

Ecrire un algorithme qui lit un entier positif n puis affiche tous ses diviseurs.

Exercice 2

Ecrire un algorithme avec trois versions qui lit un entier positif n puis calcule et affiche son factoriel selon la formule $n! = 1 \times 2 \times \dots \times n$.

- Pour...Faire
- Tant que... Faire
- Répéter ... Tant Que...

Exercice 3

Ecrire un algorithme permettant de:

- Lire un nombre fini de notes comprises entre 0 et 20.
- Afficher la meilleure note, la mauvaise note et la moyenne de toutes les notes.

Exercice 4

Calculer a^b avec a réel et b entier par multiplication successives.

Exercice 5

Ecrire un algorithme qui lit un entier positif et vérifie si ce nombre est premier ou non.

Remarque : un nombre premier n'est divisible que par 1 et par lui-même.

Exercice 6

Ecrire un algorithme qui lit deux entiers positifs A et B puis calcule et affiche leur PGCD en utilisant la méthode suivante:

- Si $A=B$; $\text{PGCD}(A, B)=A$
- Si $A>B$; $\text{PGCD}(A, B)=\text{PGCD}(A-B, B)$
- Si $A<B$; $\text{PGCD}(A, B)=\text{PGCD}(A, B-A)$

Exemple : $\text{PGCD}(18, 45) = \text{PGCD}(18, 27) = (\text{PGCD}(18, 9) = \text{PGCD}(9, 9) = 9$

Exercice 7

Ecrire un algorithme qui calcule le PPCM (Plus Petit Commun Multiple) de deux entiers positifs A et B en utilisant la méthode suivante:

- Permuter, si nécessaire, les données de façon à ranger dans A le plus grand des Deux entiers;
- Chercher le plus petit multiple de A qui est aussi multiple de B.

Exemple : $\text{PPCM}(6, 8) = \text{PPCM}(8, 6) = 24$.

Exercice 8

Ecrire un algorithme qui calcule et affiche les 10 premiers termes de la suite de Fibonacci.

La suite de Fibonacci est définie par:

- $F_0 = 1$
- $F_1 = 1$
- $F_n = F_{n-2} + F_{n-1}$ pour $n > 1$.

Exercice 9

Ecrire un algorithme qui calcule la somme harmonique $S = 1 + 1/2 + \dots + 1/N$ avec N un entier positif lu à partir du clavier.

Exemple : Pour $N=3$, $S = 1 + 1/2 + 1/3 = 1.83$

Exercice 10

Parmi tous les entiers supérieurs à 1, seuls quatre peuvent être représentés par la somme des cubes de leurs chiffres.

A titre d'exemple, $153 = 1^3 + 5^3 + 3^3$ est un nombre cubique.

Ecrire un algorithme permettant de déterminer les trois autres.

Note: les 4 nombres sont compris entre 150 et 410.

Exercice 11

Un nombre parfait est un nombre présentant la particularité d'être égal à la somme de tous ses diviseurs, excepté lui-même.

Le premier nombre parfait est $6 = 3 + 2 + 1$.

Ecrire un algorithme qui affiche tous les nombres parfaits inférieurs à 1000.