



**Série d'exercices N° 05**  
*Les Procédures et Les Fonctions*

**Exercice 1 :**

Écrire une fonction qui prend un nombre réel  $x$  en paramètre, puis retourne  $f(x)$  tel que :

$$f(x) = \frac{\sqrt{x^2 - 2\sin(x+4)}}{x^3}$$

**Exercice 2 :**

Écrire une fonction qui permet de calculer la multiplication de deux nombres A et B entiers en utilisant l'addition

**Exercice 3 :**

Écrire une procédure ValeurPair(M,N) qui prend deux paramètres M et N et d'afficher toutes les valeurs paires entre M et N si  $M \leq N$

**Exercice 4 :**

Écrire une fonction estPair(N) qui vérifie si un nombre N passé en paramètre est pair ou non

**Exercice 5 :**

Écrire une fonction max(a,b) qui retourne le maximum entre les deux paramètres a et b

**Exercice 6 :**

Écrire une fonction min(a,b) qui retourne le minimum entre les deux paramètres a et b

**Exercice 7 :**

Écrire une procédure swap(a,b) qui échange les valeur des deux paramètres a et b

**Exercice 8 :**

Écrire une fonction partieEntiere(x) qui retourne la partie entière d'un nombre positif

**Exercice 9 :**

Écrire une procédure TableMultiplication(N) qui affiche le tableau de multiplication d'un entier positif N passé en paramètre

**Exercice 10 :**

Écrire une fonction fact(N) qui calcule la factoriel d'un nombre N positif

**Exercice 11 :**

Écrire une fonction qui prend en paramètre un nombre entier positif et retourne son image miroir. Exemple le nombre est 3524, on doit afficher 4253.

**Exercice 12 :**

On appelle "palindrome" un mot ou une phrase qui se lit de la même façon à l'endroit comme à l'envers, sans tenir compte des espaces. Exemple : le mot "ABCBA" est un palindrome.

Écrire une fonction estPalindrome(ch) permettant de vérifier si une chaîne de caractères CH est un palindrome.

**Exercice 13 :**

Écrire une fonction qui détermine si un nombre est premier ou non (un nombre premier n'est divisible que par 1 et par lui-même)

**Exercice 14 :**

Écrire une fonction qui détermine si un nombre N est parfait ou non ( un nombre est parfait s'il est égale à la somme de ses diviseurs)

**Exercice 15 :**

Écrire une fonction `estBissextile(A)` qui prend en paramètre une année A. puis vérifie si cette année est bissextile

**Exercice 16 :**

Écrire une fonction `dateValide(J,M,A)` qui prend en paramètres le jour J, le mois M et l'année A. Puis vérifie si la date est valide

**Exercice 17 :**

Écrire un programme qui demande à l'utilisateur de saisir N réels stockés dans un tableau. Écrivez ensuite les fonctions suivantes :

- Une fonction permettant de remplir les éléments du tableau.
- Une fonction permettant de trier les éléments du tableau par ordre croissant.
- Une fonction permettant de trier les éléments du tableau par ordre décroissant.
- Une fonction permettant d'afficher les éléments du tableau.
- Tester toutes ces fonctions dans un programme principal

**Exercice 18 :**

Écrire une fonction qui calcule la distance de Hamming entre deux mots entrés par l'utilisateur. (La distance de Hamming entre deux mots de même longueur est le nombre d'endroits où les lettres sont différentes)

**Exercice 19 :**

Écrire une fonction `nbr_occurence(ch,e)` qui reçoit en arguments une chaîne de caractères ch et un caractère e, la fonction retourne le nombre d'occurrences du caractère e dans la chaîne ch.

**Exercice 20 :**

Écrire une fonction `pos_occurence(ch,e)` qui reçoit en argument une chaîne de caractères ch et un caractère e, la fonction retourne l'indice positif de la première occurrence du caractère e s'il existe dans la chaîne ch. Sinon elle retourne -1

**Exercice 21 :**

Écrire une fonction `cmp_chaine(ch1,ch2)` qui reçoit en argument deux chaînes de caractères ch1 et ch2, la fonction compare ch1 et ch2 et retourne :

- 1 si  $ch1 > ch2$
- 0 si  $ch1 = ch2$
- -1 si  $ch1 < ch2$

**Exercice 22 :**

Écrire une fonction `chaine_chiffres(ch)` qui reçoit en argument une chaîne de caractères ch, la fonction retourne True si la chaîne ch contient seulement des caractères chiffres sinon elle retourne False.

**Exercice 23 :**

Écrire une fonction `supp_espace(ch)` qui reçoit en argument une chaîne de caractères `ch`, la fonction retourne la chaîne `ch` après avoir supprimé tous les caractères espaces s'ils existent au début de cette chaîne.

**Exercice 24 :**

Écrire une fonction `alphabetique(ch)` qui reçoit en argument une chaîne de caractères `ch`, la fonction retourne la chaîne `ch` après avoir supprimé tous les caractères qui ne sont pas des caractères alphabétiques dans cette chaîne.

**Exercice 25 :**

Écrire une fonction qui convertit une chaîne de caractères, en minuscule puis en majuscule

**Exercice 26 :**

Écrire une fonction permettant de supprimer les espaces superflus (plusieurs espaces successifs) dans une chaîne de caractères.

**Exercice 27 :**

Écrire une fonction qui retourne la liste de tous les nombres premiers inférieurs à un nombre `N` passé en paramètre

**Exercice 28 :**

Écrire une fonction prend en paramètres deux mots `ch1` et `ch2` et qui retourne tous les caractères qui apparaissent dans les deux chaînes sans redondance.

**Exercice 29 :**

Écrire une fonction qui détermine si une adresse ip passé en paramètre et valide ou non :  
exemples :

- "192.168.1.1" et valide
- "392.168.1.1" et n'est pas valide valide
- "192.168.1" et n'est pas valide valide

**Exercice 30 :**

Écrire une fonction qui retourne le masque de sous réseau pour un préfix réseau passé en paramètre  
exemples : `mask(16)` retourne "255.255.0.0"

**Exercice 31 :**

Écrire une fonction ou procédure qui calcule le PGCD de deux entiers strictement positifs.