

SÉRIE N° 1 : PYTHON -RAPPEL

Fonctions/ Listes/chânes de caractères/Matrices/POO/Fichiers

Exercice 1 :

Si p est le demi périmètre d'un triangle de côtés a , b et c alors l'aire du triangle est donnée par :

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

Écrire la fonction **Aire** qui prend en paramètre quatre nombres p, a, b, c et qui retourne l'aire d'un triangle.

Exercice 2 :

Écrire 2 fonctions :

- **mini(a,b)** qui renvoie le minimum entre a et b
- **maxi(a,b)** qui renvoie le maximum entre a et b

Exercice 3 :

Sur l'intervalle $[-1, 0]$, la fonction f est égale à la fonction affine $x \rightarrow x+1$ et sur l'intervalle $[0, 2]$, f est égale à la fonction affine $x \rightarrow 1-x/2$

Ecrire la fonction **f** qui prend en argument un nombre x et qui retourne $f(x)$

Exercice 4 : Paradoxe des anniversaires

La probabilité p_n qu'au moins deux étudiants d'une classe de n étudiants aient leur anniversaire le même jour de l'année est donnée par la formule :

$$p_n = 1 - \prod_{k=1}^{n-1} \left(1 - \frac{k}{365}\right)$$

Écrire une fonction **probabilite(n)** qui calcule et retourne p_n .

Exercice 5 : (4 points) Les coefficients binomiaux (CNC MP2020)

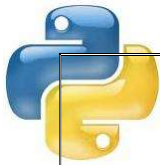
Un coefficient binomial est défini pour deux entiers positifs n et k tels que $n \geq k$. C'est le nombre de parties de k éléments dans un ensemble de n éléments. On le note : $\binom{n}{k}$, et sa valeur est calculée par la formule suivante :

$$\binom{n}{k} = \frac{n * (n-1) * (n-2) * \dots * (n-(k-1))}{k!}$$

Q1- Écrire la fonction **fact(p)** qui reçoit en paramètre un entier positif p , et qui retourne la valeur de **factorielle** p : $p! = 1 * 2 * 3 * \dots * (p-1) * p$.

NB : La fonction **fact(0)** retourne 1

Q2- Écrire la fonction **produit(n,k)** qui reçoit en paramètres deux entiers positifs n et k tels que $n \geq k$, et qui retourne la valeur du produit : $n * (n-1) * (n-2) * \dots * (n-(k-1))$



Q3- Écrire la fonction **binomial(n,k)** qui reçoit en paramètres deux entiers positifs **n** et **k** tels que $n \geq k$, et qui retourne la valeur du coefficient binomial $\binom{n}{k}$

Exemple : La fonction **binomial(6,3)** retourne le nombre **20**

Q4- Écrire la fonction **liste_binomiaux(n)** qui reçoit en paramètre un entier positif **n**, et qui retourne la liste des coefficients binomiaux $\binom{n}{k}$ tel que : $k = 0, 1, 2, 3, \dots, n$

Exemple : La fonction **liste_binomiaux(6)** retourne la liste [1, 6, 15, 20, 15, 6, 1]

Exercice 6 :

Écrire une fonction **somme(L)** qui calcule et retourne la somme des éléments de la liste L (si cette liste est vide, le résultat renvoyé devra être 0).

Exercice 7 :

Écrire une fonction **moyenne(L)** qui calcule et retourne la moyenne des éléments de la liste L (cette liste ne devra pas être vide).

Exercice 8 :

Écrire une fonction **variance(L)** qui calcule et retourne la variance de la liste L définie par :

$$\text{variance}(L) = \frac{1}{n} \sum_{k=0}^{n-1} (L[k] - m)^2$$

où n est la longueur de la liste L .

Exercice 9 :

Ecrire une fonction **smin** qui retourne le caractère minimal d'une chaîne de caractères passée en paramètre.

Ex : **smin**(«hello») -> «e»

Exercice 10 :

Ecrire une fonction **smax** qui retourne le caractère maximal d'une chaîne de caractères passée en paramètre. (**smax**)

Ex : **smax**(«hello») -> «o»

Exercice 11 :

Ecrire une fonction **wcount** qui prend en paramètre une chaîne de caractères et retourne le nombre de mots de cette chaîne.

Ex : **wcount**('beauty is a mind thinking way') -> 6

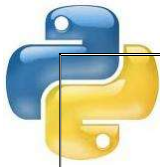
Exercice 12 :

Ecrire une fonction **stri** qui prend en paramètre une chaîne de caractères et retourne une nouvelle chaîne de caractères **MAJSUCULE** triée par ordre alphabétique.

Ex : **stri**(«Top») -> «OPT»

Exercice 13 :

Réaliser une fonction **genereListe** qui renvoie une liste de 100 nombres entiers tirés au hasard dans l'intervalle [0,99] (bornes incluses).

**Exercice 14 :**

Réaliser une fonction **combienNonPrésents** qui reçoit en paramètre une liste V d'entiers et renvoie combien d'entiers entre 0 et 99 (inclus) qui ne sont pas présents dans la liste V

Exercice 15 :

Ecrire une fonction **Recherche(L,x)** qui étant donné une liste L et un objet x renvoie True si x est un élément de la liste et False sinon

EXERCICE 16:

Ecrire une fonction **réursive Recherche(L,x)** qui prend en paramètre une liste de nombres L et un nombre x et qui permet de tester l'existence de x dans la liste

Exercice 17 :Tri_selection

Écrire une fonction **Tri_selection** qui prend en argument une liste d'entiers L et permet de trier les éléments de la liste L en ordre croissant en utilisant l'algorithme de tri par sélection :

- Trouver le plus petit élément et le mettre au début de la liste
- Trouver le 2e plus petit et le mettre en seconde position
- Trouver le 3e plus petit élément et le mettre à la 3e place,
- ...

Rq :Ne pas utiliser une autre liste intermédiaire

EXERCICE 18 : Palindrome :

Un palindrome est un mot dont l'ordre des lettres reste le même si on le lit de gauche à droite ou de droite à gauche.

Par exemple, « ressasser »

Ecrire une fonction **Palindrome** qui prend en paramètre une chaîne de caractères et qui retourne **True** si la chaîne est un palindrome et **False** sinon.

Exercice 19 :

Écrire une fonction réursive **palindrome_rec(mot)** qui prend comme argument une chaîne de caractères **mot** et retourne **True** si cette chaîne est un palindrome, False sinon

EXERCICE 20 :

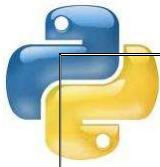
Ecrire une fonction **SupprimeMaj** qui prend en paramètre une chaîne de caractères **s** et qui supprime les caractères majuscules de la chaîne

Par exemple :

- s=" Mon Cpge !"
- L'appel de la fonction **SupprimeMaj (s)** retourne la chaîne " on pge !"

Exercice 21 :

Écrire une fonction **somme_matrices** qui prend en paramètre deux matrices A et B de mêmes dimensions et qui retourne la somme de A et de B

**Exercice 22:**

Ecrire la fonction **transpose(M)** qui prend en paramètre une matrice M et qui retourne la transposée de la matrice M.

Exercice 23: nombre de chiffres d'un entier

On rappelle que le quotient de la division euclidienne d'un entier n par 10 donne le nombre de dizaines de cet entier. Le quotient de la division euclidienne de n=5478 par 10 est par exemple 547.

En déduire une fonction **NbChiffres(n)** prenant en paramètre un entier naturel n (écrit en décimal) et retournant le nombre de chiffres de cet entier n en base 10. Cette fonction sera définie récursivement, en langage python.

Exercice 24:

Écrire une fonction récursive **binaire(n)** qui calcule la décomposition binaire de tout nombre entier n et renvoie le résultat sous forme de liste de 0 et de 1

Exemple : L'appel de la fonction **binaire(13)** retourne la liste **[1,1,0,1]**

Exercice 25:

Ecrire la fonction **copier3(NomFichier1, NomFichier2)** qui prend en paramètre les noms de deux fichiers et permet de copier le contenu du fichier **NomFichier1** dans **NomFichier2** en transformant chaque caractère en majuscule

Exercice 26:

1. Définissez la classe point dont les caractéristiques sont :
 - les coordonnées cartésiennes x et y (dont x est l'abscisse et y est l'ordonnée)
 - Les méthodes sont :
 - `__init__(self, absc, ord)` qui permet de créer une instance de classe point
 - `Get_abscisse(self)` qui retourne la valeur de x
 - `Get_ordonne(self)` qui retourne la valeur de y
 - `Distance(self, autre)` qui calcule la distance entre le point actuel et le point autre et retourne le résultat