SÉRIE N°6 : COMPLEXITÉ ALGORITHMIQUE (SUITE)

Exercice 1 :

Un tableau X est trié par ordre croissant si x[i] <= x[i+1] pour tout i

1) Ecrire une fonction itérative **Tab_Ord(X)** qui prend en paramètre un tableau X et qui retourne True si le tableau est trié en ordre croissant et False sinon.
2) Estimer sa complexité
3) Ecrire la version récursive de la question (1)
4) Estimer sa complexité

Exercice 2 : Médian d'une liste de nombres (Extrait CNC 2019 MP)

1) Écrire la fonction **grands(L,*x*)** qui reçoit en paramètres une liste de nombres **L**, et un élément ***x*** de **L**. La fonction renvoie le nombre d'éléments de **L** qui sont supérieurs strictement à ***x***.
2) Déterminer la complexité de la fonction **grands (L, *x*)**, et justifier votre réponse.
3) Écrire la fonction **petits(L,*x*)** qui reçoit en paramètres une liste de nombres **L**, et un élément ***x*** de **L**. La fonction renvoie le nombre d'éléments de **L** qui sont inférieurs strictement à ***x***.

L est une liste de taille **n** qui contient des nombres, et **m** un élément de **L**. L'élément **m** est un **médian** de **L**, si les deux conditions suivantes sont vérifiées :
• Le nombre d'éléments de **L**, qui sont supérieurs strictement à **m**, est inférieur ou égale à **n/2**
• Le nombre d'éléments de **L**, qui sont inférieurs strictement à **m**, est inférieur ou égale à **n/2**

Exemple : On considère la liste **L = [** 25 , **12** , 6 , 17, 3 , 10 , 20 , **12** , 15 , 38 **]**, de taille **n=10**.
L'élément **12** est un médian de **L**, car :
• **3** éléments de **L** sont supérieurs strictement à **12**, et **3 ≤ n/2** ;
• **5** éléments de **L** sont inférieurs strictement à **12**, et **5 ≤ n/2**.

4) Écrire la fonction **median(L)** qui reçoit en paramètre une liste de nombres **L** non vide, et qui renvoie un élément médian de la liste **L**.
5) Déterminer la complexité de la fonction median(L), et justifier votre réponse.

Exercice 3 :

Pour convertir un nombre entier positif N de la base décimale à la base binaire, il faut opérer par des divisions successives du nombre N par 2. Les restes des divisions constituent la représentation binaire.

1) Ecrire une fonction récursive « Binaire » permettant d'imprimer à l'écran la représentation binaire d'un nombre N
2) Donner une formule récurrente exprimant sa complexité en nombre de divisions. Estimer cette complexité

Exercice 4 : (Extrait CNC 2019 PSI)

1) Écrire la fonction **factoriel(k)** qui reçoit en paramètre un entier positif **k** et qui renvoie la valeur du factoriel de **k** : **k! = 1 * 2 * 3 * … * k**.

Exemples :
• La fonction **factoriel (5)** renvoie le nombre : **120 = 1 * 2 * 3 * 4 * 5**
• La fonction **factoriel (0)** renvoie le nombre : **1**

2) Déterminer la complexité de la fonction **factoriel (k)**, et justifier votre réponse
3) Écrire la fonction **som_fact(L)** qui reçoit en paramètre une liste **L** de nombres entiers positifs. La fonction renvoie la somme des factoriels des éléments de **L**.
Exemple :
L = [5, 3, 0, 6, 1]
La fonction **som_fact (L)** renvoie la valeur de la somme : **5! + 3! + 0! + 6! + 1!**
4) Déterminer la complexité de la fonction **som_fact (L)**, et justifier votre réponse.

Exercice 5 :

Etant donné un tableau X composé de N éléments entiers. On voudrait déterminer son maximum par un programme récursif basé sur le paradigme« **diviser pour régner** » :

1) En considérant que le maximum est le plus grand entre le dernier terme et le maximum des (n-1) premiers termes.

- Ecrire la fonction Max(X)
- Estimer sa complexité.

2) En considérant que le maximum est le plus grand entre les maximums des deux moitiés du tableau. Estimer sa complexité.

- Ecrire la fonction Max(X)
- Estimer sa complexité.

Exercice 6 : (Extrait CNC 2019 TSI)

1) Écrire la fonction somme (L) qui reçoit en paramètre une liste L de nombres entiers, et qui retourne la somme des éléments de **L**.
Exemple :
La fonction **somme ([7, 0 , -1, 5 , -3])** renvoie la valeur **8 = 7 + 0 + (-1) + 5 + (-3)**
2) Déterminer la complexité de la fonction **somme (L)**, et justifier votre réponse.
3) Écrire la fonction **list_puissances (L, p)** qui reçoit en paramètres une liste **L** de nombres entiers, et un entier **p** strictement positif. La fonction renvoie une nouvelle liste qui contient les éléments de **L** élevés chacun à la puissance **p**.
Exemple :
La fonction **list_puissances ([7, 0 , -1, 5 , -3] , 2)** renvoie la liste **[49 , 0 , 1 , 25 , 9]**
4) Écrire la fonction **som_puiss (L, p)** qui reçoit en paramètres une liste **L** de nombres entiers, et un entier **p** strictement positif. La fonction renvoie la somme des éléments de **L** élevés chacun à la puissance **p**.
Exemple :
La fonction **som_puiss ([7, 0 , -1, 5 , -3], 2)** renvoie le nombre **84 =** $7^2 + 0^2 + (-1)^2 + 5^2 + (-3)^2$
5) Déterminer la complexité de la fonction **som_puiss (L, p)**, et justifier votre réponse.