Année Universitaire : 2025/ 2026
Module : Algorithmique & Programmation avancés
Filière : GSEIR-3/SICS-3
Responsable : Pr. Khalid EL ASNAOUI

TP5

Objectifs
Consolider vos compétences en algorithme. Notion de complexité

NB : On vous demande d'écrire en langage Java des programmes des tous ces Exercices

Exercice 1 :

Soient ***x*** un nombre réel et ***n*** un entier positif. On veut calculer x^n
1. Ecrire un algorithme itératif.
2. Ecrire un algorithme Récursif.

Écrivez deux algorithmes (itératif er récursif) qui calculent le factoriel de ***n*** :

$$n\ !\ =\ 1\times 2\times 3\times\ \ldots\ \times(n-1)\times\ n.$$

Le programme demandera à l'utilisateur d'entrer la valeur de ***n***.

Exercice 2 :

Un tableau ***A*** contenant n éléments qui sont redondants. On veut supprimer tous les éléments dupliqués de ***A***.
1. Ecrire un algorithme permettant de résoudre ce problème en utilisant deux boucles.
2. Supposons que le tableau ***A*** est trié dans l'ordre croissant. Ecrire un algorithme en utilisant qu'une seule boucle.

Exercice 3 :

Soit ***A*** un tableau trié de $n \geq 2$ éléments sans aucun élément dupliqué. On cherche deux éléments ***x*** et ***y*** dans A tels que $\mathbf{x + y = 0}$.
1. Ecrire un algorithme en utilisant deux boucles.
2. Ecrire un algorithme en utilisant la recherche dichotomique.
3. Ecrire un algorithme en utilisant qu'une seule boucle.

Exercice 4 :

Soit ***A*** un tableau de ***n*** éléments. On suppose que les éléments de ***A*** sont booléens (0 ou 1). On cherche à trier ce tableau.
1. Ecrire un algorithme en utilisant qu'une seule boucle.

Exercice 5 :

On suppose que l'on a un tableau $A\ [1...n]$ ($n \geq 3$) avec les propriétés suivantes :
$A\ [2] \leq A\ [1]$ et $A[n-1] \leq A[n]$. On dit qu'un élément $A[i]$ est un minimum local, s'il est inférieure ou égal à ces deux voisins, ou plus formellement, si $A[i] \leq A[i-1]$et $A[i] \leq A[i+1]$.
1. Déterminer les minimums locaux de tableau ci-dessous :

A = {9,7,7,2,1,3,7,5,4,7,3,3,4,8,6,9}

2. Un algorithme qui détermine un minimum local d'un tableau ***A [1...n]*** en utilisant qu'une seule boucle.
3. Un algorithme qui détermine un minimum local d'un tableau ***A[1...n]*** en utilisant la recherche dichotomique.

Exercice 6 :

On considère le problème du calcul de la suite de Fibonacci définie par :

$$F_0 = F_1 = 1 \text{ et } F_n = F_{n-1} + F_{n-2}, \text{ pour } n \geq 2.$$

1. Écrire un algorithme récursif ***Fibo1(n)*** qui calcule F_n
2. Écrire un algorithme itératif ***Fibo2(n)*** qui calcule F_n

Exercice 7 :

1. Ecrire un algorithme récursif résolvant la somme des ***N*** premiers entiers.
2. Ecrire un algorithme récursif pour déterminer le plus grand élément d'un tableau.

Exercice 8 :

1. Que font ces deux algorithmes

```
Fonction  test (y, z: Entier) : Entier
Var  x: Entier;
Début
x:= 0;
        Tant Que(z > 0) Faire
        x:= x+ y ×(z% 2);
        y:= 2 × y;
        z:= z/ 2; //La partie inferieure de z/2
        FinTQ
Retourner(x);
Fin
```

```
Procédure  test1(x, y: Entier)
Var  aux: Entier;
Début
    aux:= x;
    x:= y;
    y:= aux;
Fin
```