ISIMM	**ANNEE UNIVERSITAIRE :** 2022-2023
SECTION : ING1 INFO	**ENSEIGNANT :** SAKKA ROUIS TAOUFIK
MATIERE : CONCEPTION ET ANALYSE D'ALGORITHMES	

TD3

EXERCICE 1 : Triangle de Pascal

On veut calculer les coefficients binomiaux $C_n^k = \binom{n}{k} = \frac{n!}{k!(n-k)!}$. Rappelons les propriétés suivantes :

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k} \; pour \; 0 < k < n$$

$$\binom{n}{n} = 1 \; et \; \binom{n}{0} = 1$$

Question 1.1

- Donner un algorithme récursif du calcul de $\binom{n}{k}$
- Evaluer sa complexité.

Question 1.2

- Ecrire l'algorithme qui retourne $\binom{n}{k}$ en utilisant la technique de la programmation dynamique.
- Evaluer sa complexité.

EXERCICE 2 :

1. Ecrire une fonction itérative puissanceIterative (a, n) qui permet de calculer a^n.

Rq. En utilisant seulement les opérateurs simples (+, −, ∗, /)

2. Ecrire une fonction récursive puissanceRecursive (a, n) qui permet de calculer a^n.

3. Comparer la complexité asymptotique de chacune des fonctions proposées. Quelle est la fonction la plus performante (du point de vue complexité de calcul).

4. Supposant qu'on peut écrire la fonction puissance de la manière suivante :

$a^n = a^{n \text{ div } 2} \times a^{n \text{ div } 2}$ si n est pair, sinon $a^n = (a^{n \text{ div } 2} \times a^{n \text{ div } 2} \times a)$

$a^{n/2} = a^{n/4} \times a^{n/4}$

…………

$a^1 = a$

$a^0 = 1$

Proposer une fonction puissanceDdynamique (a, n) en utilisant le principe de la programmation dynamique.

5. Quelle est la complexité asymptotique de cette nouvelle fonction.

EXERCICE 3 : Multiplications chaînées de matrices.

On veut calculer le produit de matrices $M = M_1.M_2 \ldots.M_n$. Multiplier une matrice pxq, par une matrice qxr en utilisant la méthode standard nécessite p.q.r produit scalaire.

Question 3.1 :
Considérons 4 matrices A : 20×5, B : 5×100, C : 100×8, D : 8×30. On veut calculer le produit A B C D. En fonction des parenthétisations, le nombre de produits varie.
Déterminer le nombre de produits pour calculer ABCD, si on utilise les parenthétisations suivantes : ((AB)C)D ou (A(BC))D
L'objectif est de concevoir un algorithme de meilleure parenthétisation qui permet de minimiser le nombre de produits scalaires.
Nous noterons que la matrice M_i est de dimension $d_{i-1} \times d_i$.
Définissons le nombre minimal de produits scalaires nécessaires pour évaluer le produit des matrices $M_i \times M_{i+1} \ldots M_{j-1} \times M_j$ par c(i, j).
Question 3.2 : Écrire une formule de récurrence pour calculer c(i, j).
Question 3.3 : Ecrire un algorithme utilisant la programmation dynamique