

TP - Programmation

Série de TP N°2 – Tableaux à deux dimensions - Matrices

Exercice N°01 : Algorithme → Programme C

Soit l'algorithme suivant :

Algorithme Matrice ;

Variables

A : Tableau [0..99, 0..99] de réel;

i, j, N : entier;

S, M : réel;

Début

// Entrées

Ecrire("Donner la taille de la matrice carrée A : ");

Lire(N);

Ecrire("Donner les composantes de la matrice A:");

Pour i ← 0 à N-1 **faire**

Pour j ← 0 à N-1 **faire**

 Lire(A[i,j]);

FinPour;

FinPour;

// Traitements

S ← 0;

Pour i ← 0 à N-1 **faire**

 S ← S+A[i,i] ;

FinPour;

M ← S/N ;

// Sorties

Ecrire("S=", S, "M=", M);

Fin.

Questions :

1- Traduire l'algorithme en Programme C.

2- Compiler et exécuter le programme pour :

N = 3 et

$$A = \begin{bmatrix} 2 & 4 & -4 \\ 4 & 8 & 1 \\ 3.5 & 9 & 4 \end{bmatrix}$$

3- Dérouler le programme pour les valeurs de N et A ci-dessus ?

4- Déduire ce que fait le programme ?

5- Ré-écrire le programme en remplaçant la boucle *Pour* par la boucle *Tantque* dans la partie des entrées.

6- Ré-écrire le programme en remplaçant la boucle *Pour* par la boucle *Répéter* dans la partie des entrées.

Exercice N°02 : Transposée d'une matrice

Écrire un algorithme/programme C qui permet de calculer la matrice B transposée d'une matrice réelle A d'ordre N x M.

Exercice N°03 : Somme de deux matrices

Écrire un algorithme/programme C qui permet de réaliser la somme de deux matrices réelles A et B d'ordre N x M.

Exercice N°04 : Matrice symétrique

Soit A une matrice carrée de taille N x N et de type réel. Écrire un algorithme/programme C qui permet de vérifier si la matrice A est symétrique.

Rappel : Une matrice A est symétrique si $A[i, j] = A[j, i]$ pour tout i et j .

Exercice N°05 : Produit d'une matrice par un vecteur

Soit A une matrice de type réel et d'ordre N x M. Écrire un algorithme/programme PASCAL qui permet de calculer le produit de la matrice A par un vecteur V de type réel et de taille M.

TP - Programmation

Série de TP N°2 – Exercices supplémentaires

Exercice Sup-01 : Somme, Moyenne et Produit des éléments d'une matrice

Soit une matrice A réelle d'ordre $N \times M$.

1. Écrire un algorithme/programme C qui calcule la somme et la moyenne des éléments de la matrice A.
2. Écrire un algorithme/programme C qui permet de calculer la somme de chaque ligne et le produit de chaque colonne.

Exercice Sup-02 : Produit de deux matrices

Soit A et B deux matrices carrées d'ordre N.

Écrire un algorithme/programme C qui permet de calculer le produit de A et B.

Exercice Sup-03 : La recherche d'une valeur dans une matrice

Soit M une matrice de type réel de taille $N \times M$.

Écrire un algorithme/programme C qui permet de rechercher si une valeur réelle X existe ou non dans la matrice M. Dans le cas où X existe dans M, on affiche aussi sa position (numéro de ligne et de colonne).

Exercice Sup-04 : Le Min et le Max dans une matrice et leurs positions

Soit A une matrice réelle d'ordre $N \times M$.

1. Écrire un algorithme/programme C qui permet de rechercher le plus petit élément dans la matrice A ainsi que sa position.
2. Écrire un algorithme/programme C qui permet de rechercher le plus grand élément dans la matrice A ainsi que sa position.

Exercice Sup-05 :

Soit A une matrice carrée de $N \times N$ composantes entières et V un vecteur de N composantes entières (avec $3 \leq N \leq 8$) :

Partie A : Écrire un programme en langage C qui permet de :

1. Trouver le nombre des éléments pairs dans le vecteur V.
2. Trouver la plus grande valeur paire dans le vecteur V et sa position.

N.B : La première valeur du vecteur V est considérée comme élément pair.

Partie B : Modifier le programme afin de :

1. Calculer le nombre d'éléments non nuls de l'anti-diagonale de la matrice A.
2. Remplacer la première ligne de la matrice A par les éléments du vecteur V.

N.B : L'entrée N doit être contrôlée.