

Chapitre VI : TABLEAUX

Introduction

Un tableau est une structure de données homogène composée de cellules dont chacune d'elle peut prendre une valeur. Les cellules du tableau sont numérotées à partir de 1 de la gauche vers la droite. Tout tableau est caractérisé par :

- **Son nom** : il représente l'identificateur du tableau
- **Son type de valeur** : il détermine les valeurs compatibles aux cellules du tableau. Le type de valeur peut être élémentaire ou composé.
- **Sa dimension** : elle détermine la nature du tableau. Si la dimension est égale à 1, on parle de vecteur, à 2, on parle de matrice et à 3, il s'agit d'Espace. Dans ce chapitre nous étudierons les vecteurs et les matrices.
- **Sa taille** : elle représente le nombre de cellules du tableau.

I. Les Vecteurs

1. Définition

Un vecteur est un tableau à une dimension c'est-à-dire un tableau à une ligne avec plusieurs colonnes. Chacune de ces colonnes représente une cellule et peut contenir une valeur.

2. Déclaration

Syntaxe 1

Const Taille = valeur

Type nomVecteur = tableau[1..Taille] TypeValeur

var nomVariableVecteur : nomVecteur

Syntaxe 2

Const Taille = valeur

var nomVecteur : tableau[1..Taille] TypeValeur

3. Manipulation des cellules d'un vecteur

Pour manipuler un vecteur, il faut accéder à ses valeurs. L'accès se fait de façon linéaire en passant par la position de la cellule concernée ($T[i]$ signifie la valeur de la cellule à la position i du vecteur T).

4. Création de vecteur

Créer un vecteur revient à le remplir. Pour cela, il faut parcourir toutes les cellules et donner à chacune d'elle une valeur.

Exercice d'application

Ecrire un programme qui permet de remplir un tableau d'entiers de 75 cellules, d'afficher le contenu du tableau, de déterminer la moyenne des nombres pairs du tableau, le % de présence des nombres positifs du tableau et d'afficher les résultats.

5. Les opérations applicables aux vecteurs

Il est possible d'appliquer plusieurs opérations à des valeurs d'un vecteur. Parmi ces opérations on peut noter le décalage cyclique, le tri, l'union, la fusion etc.

Application

Ecrire un programme qui permet de remplir un tableau d'entiers de 50 cellules, d'afficher le contenu du tableau, de décaler de façon cyclique chaque valeur du tableau d'une position vers la gauche et d'afficher le tableau après décalage.

II. Les Matrices

1. Définition

Une Matrice est un tableau à deux dimensions c'est-à-dire un tableau à plusieurs lignes et plusieurs colonnes dont chaque ligne se comporte comme un vecteur d'où la notion vecteur de vecteurs. Si le nombre de lignes de la matrice est égal à son nombre de colonnes alors on parle de matrice carrée sinon la matrice est quelconque. Certaines opérations ne peuvent se faire qu'avec des matrices carrées (Ex : la **diagonalisation**)

2. Déclaration

Syntaxe 1 :

```
Const NLigne = valeur1  
Const NColonne = valeur2  
Type nomMatrice = tableau[1..NLigne,1..NColonne] TypeValeur  
Var nomVariableMatrice : nomMatrice
```

Syntaxe 2 :

```
Const NLigne = valeur1  
Const NColonne = valeur2  
Var nomVariableMatrice : tableau[1..NLigne,1..Ncolonne] TypeValeur
```

3. Manipulation d'une matrice

Chaque valeur d'une matrice se trouve à l'intersection d'une ligne et d'une colonne. Soit i l'indice de parcours des lignes et j des colonnes. La notation $mat[i,j]$ ou $Mat[i][j]$ permet d'accéder à la valeur qui est à l'intersection de la ligne i et colonne j de la matrice Mat .

4. Création de la matrice

Pour créer une matrice, il faut la remplir. Pour cela il faut déclarer 2 variables qui vont servir d'indice de parcours pour les lignes et les colonnes.

Exercice d'application 1

Ecrire un programme qui permet de remplir une matrice d'entiers de 10 lignes et 15 colonnes, d'afficher le contenu de la matrice, de déterminer le minimum, le maximum, la moyenne des nombres pairs de la matrice et d'afficher les résultats.

Exercice d'application 2

Ecrire un programme qui permet de remplir une matrice carrée d'entiers d'ordre 9, d'afficher le contenu de la matrice ainsi que pour chaque ligne la somme de ses valeurs.

5. Les opérations applicables aux matrices

Il est possible d'appliquer plusieurs opérations à des valeurs d'une matrice. Parmi ces opérations on peut noter les opérations scalaires, la somme de matrice, la transposée de matrice, le produit matriciel etc.

Exercice d'application

Ecrire un programme qui permet de remplir une matrice d'entiers de 10 lignes et 15 colonnes, d'afficher le contenu ainsi que le minimum des maxima des lignes (minMax) et le maximum des minimas des lignes(maxMin) de la matrice.

III. Tableaux d'enregistrement

1. Définition

Un tableau d'enregistrement est un tableau dans lequel chaque cellule contient un enregistrement. Chaque cellule sera décomposée en sous-cellules et le nombre des sous cellules est égale au nombre de champs de l'enregistrement

2. Déclaration

a. Syntaxe pour un vecteur d'enregistrements

Const Taille = valeur

Type nomEnregistrement = structure

Debut

Champ(s) : type(s)

Fin

Type nomVecteur = tableau [1..Taille] nomEnregistrement

var nomVariableVecteur : nomVecteur

b. Syntaxe pour une matrice d'enregistrements

Const NLigne = valeur1

Const NColonne = valeur2

Type nomEnregistrement = structure

Debut

Champ(s) : type(s)

Fin

Type nomMatrice = tableau [1..NLigne, 1..NColonne] nomEnregistrement

var nomVariableMatrice : nomMatrice

3. Manipulation des cellules d'un tableau d'enregistrement

Pour manipuler les cellules d'un tableau d'enregistrement, il faut parcourir le tableau avec une boucle si c'est un vecteur et 2 si c'est une matrice.

Exemple : Soit T un vecteur d'enregistrements, i l'indice de parcours et x un champ. Pour accéder à la valeur du champ x de la cellule i du tableau T , on note $T[i].x$. Par contre, si Mat est une matrice, i indice de parcours des lignes, j indice de parcours des colonnes et x un champ, la notation $Mat[i,j].x$ ou $Mat[i][j].x$ permet d'accéder à la valeur du champ x se trouvant à l'intersection de la ligne i et la colonne j de la matrice Mat .

4. Création d'un tableau d'enregistrement

Pour créer un tableau d'enregistrement, il faut le parcourir et le remplir. Si c'est un vecteur il faut un indice et si c'est une matrice, il en faut 2.

Application

Ecrire un programme qui permet de remplir un tableau de 150 Produits, d'afficher les données de chaque produit, de déterminer le nombre de produits de catégorie alimentaire, le montant total en stock de tous les produits, et d'afficher les résultats. Un produit est caractérisé par son code, son nom, sa catégorie, son prix unitaire et sa quantité.

5. Les opérations applicables aux tableaux d'enregistrement

Il est possible d'appliquer plusieurs types d'opérations à des valeurs d'un tableau d'enregistrement. Parmi ceux-ci, on peut noter les opérations classiques mais également les opérations relatives aux types de données.

Application

Ecrire un programme qui permet de remplir une matrice d'étudiants de 10 lignes et 15 colonnes, d'afficher le contenu de la matrice, de déterminer la moyenne générale des étudiants de la matrice ainsi que l'étudiant qui a la plus grande moyenne et d'afficher les résultats.

Remarque :

Soit une matrice carrée d'ordre N . i l'indice de parcours des lignes et j l'indice de parcours des colonnes. Le nombre d'éléments se trouvant sur la diagonale principale ou sur celle secondaire est égal à l'ordre N . Si l'ordre d'une matrice carrée est un nombre impair c'est-à-dire s'il vérifie la condition $N \bmod 2 = 1$ alors on aura un élément se trouvant sur les deux diagonales et constituera l'intersection de celles-ci.

	Diagonale Principale	Diagonale Secondaire
Sur	$i = j$	$i+j = N+1$
Au-Dessus	$i < j$	$i+j < N+1$
En Dessous	$i > j$	$i+j > N+1$
Sur (Si $N \bmod 2 = 1$)	$i = j$ et $i+j = N+1$	