

# Chapitre 6

## LISTES ET MATRICES

Les listes et les matrices permettent de stocker plusieurs valeurs dans une même variable. Nous allons voir comment les créer et les utiliser.

### I Les listes

#### I.1 Créer une liste

$$\text{liste} = [ a_1, a_2, \dots, a_n ]$$

où  $a_1, a_2, \dots, a_n$  sont les valeurs à stocker dans la liste.

**Exemple.**

```
L1 = [ 3 , 2.1 , 5 ]
```

#### I.2 Liste d'entiers consécutifs

$$L = [ n : m ]$$

**Exemple.** Définir les listes  $L_2 = [0,1,2,3,4]$  et  $L_3 = [3,4,5,6,7,8]$

#### I.3 Manipulations de listes

On considère une liste appelée  $L$ .

Opération	Syntaxe Scilab
Élément numéro $k$ de $L$	$L(k)$
Modifier l'élément numéro $k$	$L(k) = \text{valeur}$
Longueur de $L$	$\text{length}(L)$
Ajouter un élément $x$ à la fin de $L$	$L = [ L , x ]$

**Exemple.**

1. Définir la liste  $L_3 = [4, 7, 8, 2, 1, 0, 6]$
2. Changer le troisième élément (le 8 donc) en un 10.
3. Ajouter un 4 à la fin de  $L_3$ .

```
-->
-->
-->
```

**Exemple.** Définir, à l'aide d'une boucle, la liste  $L = [1, 2, 4, 8, 16, \dots, 2^{20}]$ .

1. Méthode 1 : on part d'une liste vide et on ajoute un à un les éléments/

```
L = [] // liste vide
for k = ... : ...

end
disp(L)
```

2. Méthode 2 : on part d'une liste de taille voulue et on modifie un à un les éléments.

```
L = [ 1 : ... ] // liste à la bonne taille
// on pourrait aussi partir d'une liste composée de 0
// (voir plus loin)
for k = ... : ...

end
disp(L)
```

## II Les matrices et tableaux de valeurs

Avec Scilab, nous pouvons facilement créer des matrices qui permettent de stocker plusieurs valeurs sous la forme d'un tableau rectangulaire. Remarquons que les listes sont en fait des matrices à une ligne, donc ce que nous allons voir ici permet également de compléter la partie sur les listes.

### II.1 Créer une matrice

La matrice  $A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & a_{2p} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{np} \end{pmatrix}$  peut être définie ainsi :

```
A = [ a11, a12 , ... , a1p ; a21, a22 , ... , a2p ; ... ; an1, an2 , ... , anp ]
```

**Exemple.** Définir la matrice  $A = \begin{pmatrix} 1 & 4 \\ 2 & 7 \\ -1 & 5 \end{pmatrix}$

## II.2 Matrices particulières

- Matrice nulle de taille  $n \times p$  : `zeros(n,p)`
- Matrice identité de taille  $n \times n$  : `eye(n,n)`
- Matrice aléatoire de taille  $n \times p$  : `rand(n,p)`  
Les éléments de cette matrice sont des nombres aléatoire de  $[0, 1]$ .

## II.3 Manipulation de matrices

On considère une matrice appelée  $M$ .

Opération	Syntaxe Scilab
Coefficient $(i, j)$ de $M$	$M(i, j)$
Modifier le coefficient $(i, j)$	$M(i, j) = \text{valeur}$
Ligne $i$ de $M$	$M(i, :)$
Colonne $j$ de $M$	$M(:, j)$
Taille de $M$	$n, p = \text{size}(M)$
Rassembler deux matrices	$[M1, M2]$ (côte à côte) $[M1 ; M2]$ (l'une en dessous de l'autre)

**Exemple.**

1. Définir les matrices  $M1 = \begin{pmatrix} 1 & 2 & 3 \\ 8 & 5 & 6 \end{pmatrix}$  et  $M2 = \begin{pmatrix} 7 & 1 \\ -1 & 4 \end{pmatrix}$
2. Changer le 8 de la matrice  $M1$  en un 4.
3. Rassembler  $M1$  et  $M2$  en une seule matrice, appelée  $M3$ .
4. Afficher la deuxième ligne de  $M1$ .

```
--> M1 =
--> M2 =
-->
-->
-->
```

## II.4 Calculs sur les matrices

Les opérations classiques +, \*, -, ^ sont pour Scilab des **opérations matricielles**.

### a) Opérations matricielles ( = opérations normales)

Action	Syntaxe Scilab
Somme	A + B
Produit matriciel	A * B
Puissance matricielle	A ^ n
Inverse	inv(A)
Tranposée	A'

### b) Opérations terme à terme : .\* ./ .^

On va aussi avoir besoin de réaliser des opérations terme à terme sur des listes ou tableaux de valeurs.

Soient T1 = [a11, a12, ..., anp] et T2 [b11, b12, ..., bnp] deux tableaux (ou listes) de même taille.

Remarque : pour les combinaisons linéaires, l'opération classique est déjà terme à terme. Nous les rappelons quand même ici.

Opération	Syntaxe Scilab	Résultat
Addition terme à terme	T1 + T2	[ a11 + b11, ..., anp + bnp ]
Ajout d'un nombre	T1 + c	[ a11 + c, ..., anp + c ]
Multiplication par un nombre	c * T1	[ c * a11, ..., c * anp ]
Multiplication terme à terme	T1 .* T2	[ a11 * b11, ..., anp * bnp ]
Division terme à terme	T1 ./ T2	[ a11 / b11, ..., anp / bnp ]
Puissance terme à terme	T1 .^ k	[ a11^k, ..., anp^k ]
	k .^ T1	[ k^a11, ..., k^anp ]
	T1 .^ T2	[ a11^b11, ..., anp^bnp ]
Fonctions usuelles	exp(T1), log(T1), ...	[ exp(a11), ..., exp(anp) ]
Somme des termes	sum(T1)	
Produit des termes	prod(T1)	

### Application directe : calculer une somme / un produit en une ligne

#### Exemple.

1. Définir la liste [1<sup>3</sup>, 2<sup>3</sup>, 3<sup>3</sup>, ..., 100<sup>3</sup>].

2. En déduire, en une ligne  $S = \sum_{k=1}^{100} k^3$ .

```
-->
--> S =
```

**Exemple.** Définir en une ligne  $n! = \prod_{k=1}^n k$  (on suppose que n est donné).

```
-->
```