

Cours : Outils de programmation pour les mathématiques

29/01/2023



Programme

- Maîtrise du logiciel Matlab,
- Exemples d'application.

Maîtrise du logiciel matlab

Matlab

- Généralités
- Introduction a Matlab
- Interface matlab
- Listes de quelques commandes d'environnement
- Exemple

Maîtrise du logiciel matlab

La programmation sous matlab

- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations

Maîtrise du logiciel matlab

Les fonctions et le graphisme sous matlab

- Les fonctions sous matlab
- Les graphismes sous matlab

Exemple d'application

calculs
algébriques/ les
matrices

- Introduction
- Les listes
- Les vecteurs
- Les matrices

Maîtrise du logiciel matlab

Matlab

- Généralités
- Introduction a Matlab
- Interface matlab
- Listes de quelques commandes d'environnement
- Exemple

chapitre 01: Maîtrise du logiciel matlab

Généralités

- La programmation: Programmer signifie écrire des lignes de code selon un langage de programmation défini et ce pour réaliser des « programmes informatiques ». Les programmes sont des suites d'instructions qui permettent à l'ordinateur d'effectuer des actions dans des buts connues.
- Les langages de programmation: Un langage de programmation est un langage compris par l'être humain et ce pour faciliter l'interaction avec l'ordinateur qui à son tour utilise une succession de 0 et 1 appelé langage machine(langage binaire).
- Les logiciels: Un logiciel est un ensemble de programmes informatiques qui permettent à un système informatique d'assurer une fonction particulière. Généralement les logiciels possèdent une interface graphique pour faciliter les interactions avec l'utilisateur.

Maîtrise du logiciel Matlab

Matlab

- Généralités
- Introduction a Matlab
- Interface matlab
- Listes de quelques commandes d'environnement
- Exemple

chapitre 01: Maîtrise du logiciel matlab

Introduction

- Matlab est un logiciel de calcul
- Conçu pour répondre à votre manière de réfléchir et de travailler.
 - MATLAB associe un environnement de bureau, conçu pour l'analyse par itération et les processus de conception avec un langage de programmation permettant d'exprimer directement les mathématiques sous forme de tableaux et de matrices.

chapitre 01: Maîtrise du logiciel Matlab

Le langage de programmation Matlab:

- Est une abréviation du terme matrix laboratory qui signifie un atelier de calculs à base matricielle il a été conçu par le mathématicien (Cleve Moler vers la fin des années 1970) à partir des bibliothèques Fortran et ce pour permettre à ses étudiants de pouvoir utiliser des bibliothèques de Fortran sans connaître le Fortran.
- MATLAB est utilisé à des fins de calcul numérique, il permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en oeuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran,etc.

chapitre 01: Maîtrise du logiciel matlab

Fonctionnalités



Analyse de données

Explorer, modéliser et visualiser des données



Graphiques Visualiser et explorer vos données



Développement d'algorithmes

Concevez des algorithmes pour des applications PC et embarquées



Création d'applications

Créer des applications web et desktop



Interopérabilité avec d'autres langages

Utilisez MATLAB avec Python, C/C++, Fortran, Java et d'autres langages



Hardware Connectez MATLAB au hardware



Calcul parallèle Exécutez des calculs à grande échelle sur des ordinateurs multi-cœurs, GPU, clusters, grilles et clouds



Déploiement sur PC et sur le web Partagez vos programmes MATLAB



Cloud Computing Exécutez des applications dans des clouds comme le Cloud MathWorks ou les clouds publics comme AWS et Azure

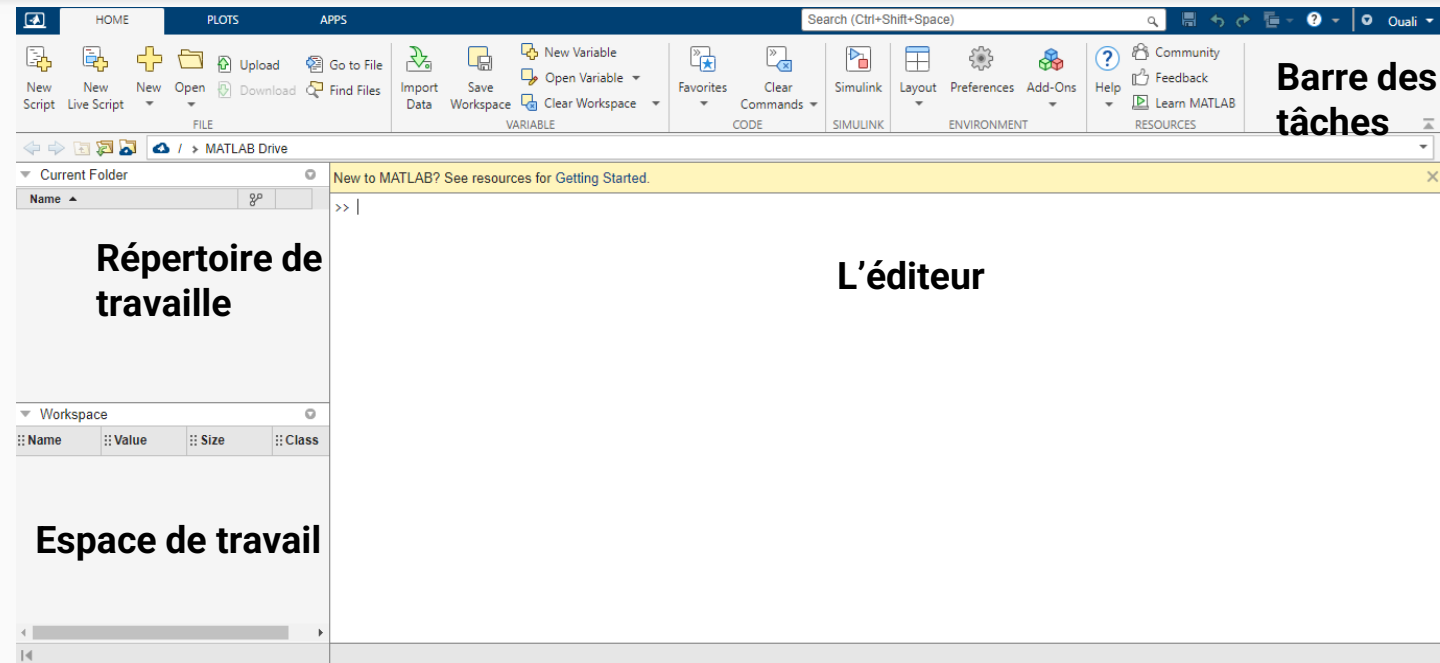
Maîtrise du logiciel matlab

matlab

- Généralités
- Introduction a Matlab
- Interface matlab
- Listes de quelques commandes d'environnement
- Exemple

chapitre 01: Maîtrise du logiciel matlab

Interface Matlab

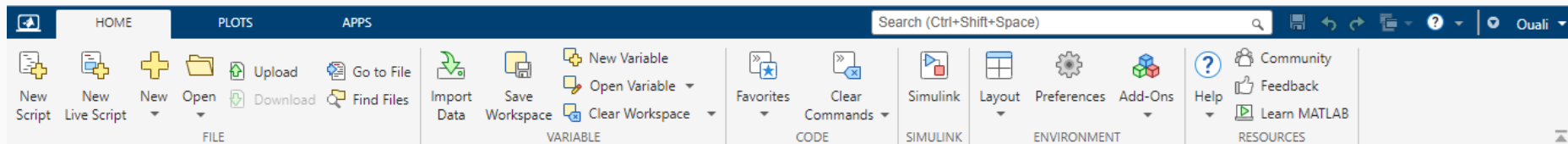


- Selon la version utilisée, l'interface peut changer légèrement mais les points centraux resteront identiques. Généralement l'interface de Matlab se compose de plusieurs zones.

chapitre 01: Maîtrise du logiciel matlab

Interface Matlab

Cette interface présente l'éditeur textuel fourni par l'environnement Matlab, il offre l'espace et les outils nécessaires pour l'écriture des scripts Matlab, citons l'ouverture, la sauvegarde, d'importation et l'exportation, d'un fichier Matlab ,etc.

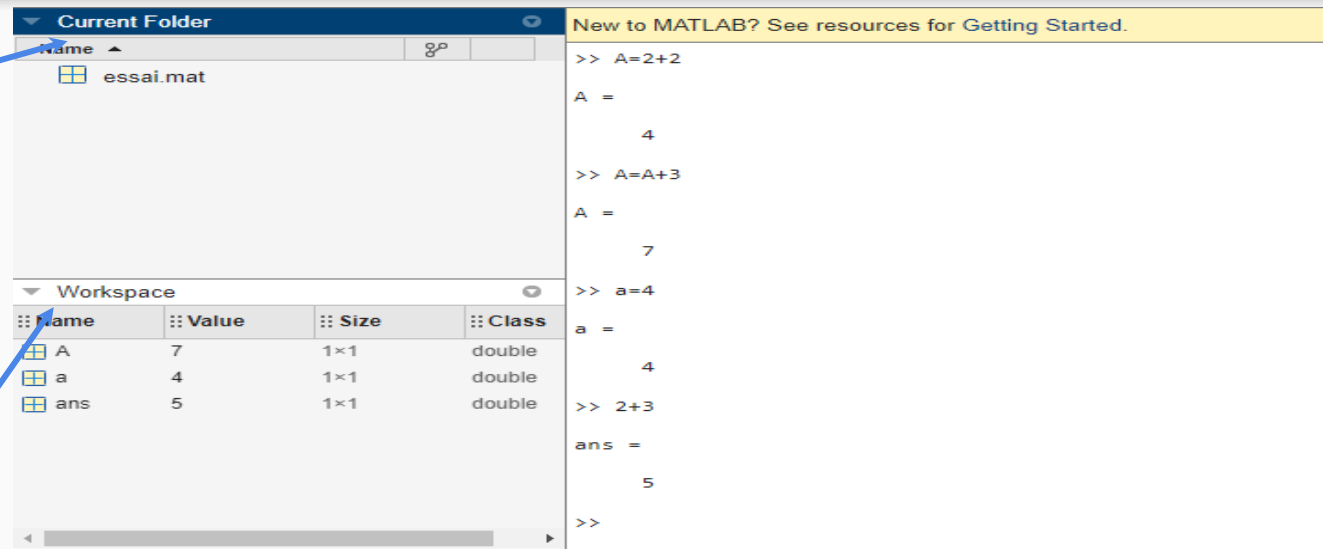


chapitre 01: Maîtrise du logiciel matlab

Interface Matlab

L'explorateur de fichiers (Current Folder)

permet de visualiser les fichiers scripts (.m) et de les ouvrir pour les éditer ou exécuter.

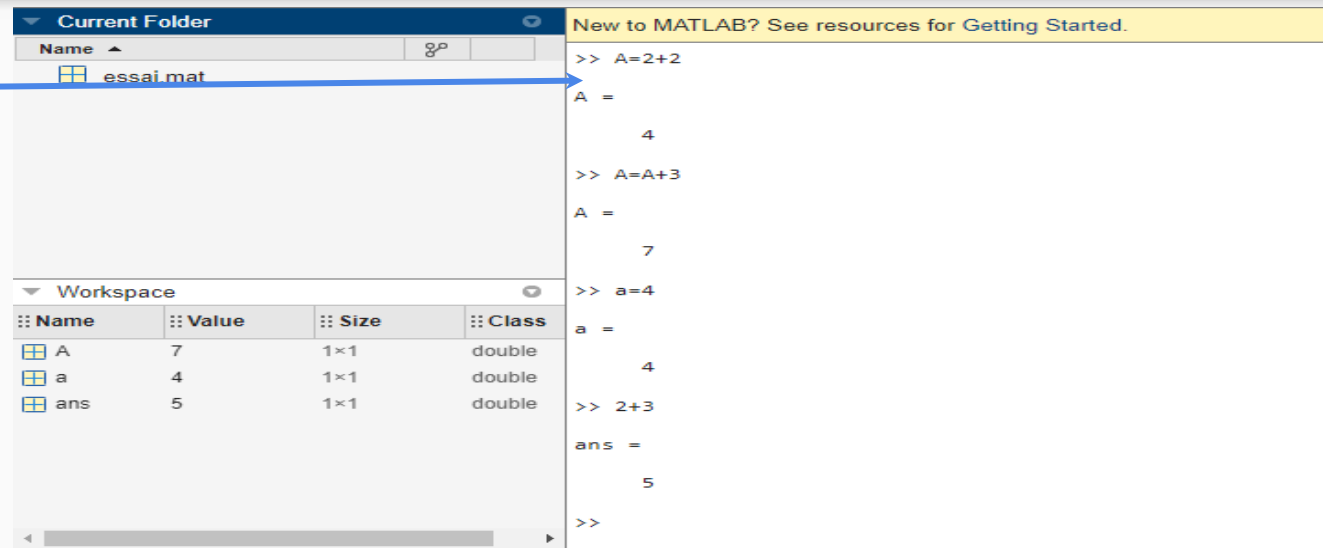


La zone des variables (Workspace) permet de visualiser toutes les variables en mémoire à l'instant présent (les noms ainsi que les contenus de ces variables). Un double-clic sur le nom de la variable permet d'afficher sa valeur tandis qu'un clic-droit sur une variable offre de nombreuses options (copier, coller, supprimer etc).

chapitre 01: Maîtrise du logiciel matlab

Interface Matlab

Zone de commande (Command Window) c'est le terminal qui permet d'écrire des commandes et de visualiser leur résultats. Une ligne commence toujours par ».



Maîtrise du logiciel matlab

matlab

- Généralités
- Introduction a Matlab
- Interface matlab
- Listes de quelques commandes d'environnement
- Exemple

chapitre 01: Maîtrise du logiciel matlab

Listes de quelques commandes d'environnement

Commande	Description
clc	Permet d'effacer l'ensemble des commandes MATLAB introduite dans la zone de commande
clear all	Permet de supprimer l'historique
quit ou exit	Permettent de quitter l'interface MATLAB
help	C'est la fonction la plus importante pour comprendre MATLAB. Afin d'avoir des informations sur l'utilisation d'une fonction matlab : »help Nomfonction.
lookfor	Cette fonction est utilisée lorsque on cherche une commande sans avoir une idée sur le nom de cette commande : »lookfor sinus.
doc	Permet d'avoir une description détaillée avec des exemples : » doc sin
who	Affiche la listes des variables de l'espace de travail
whos	Affiche la listes des variables avec description détaillée de l'espace de travail

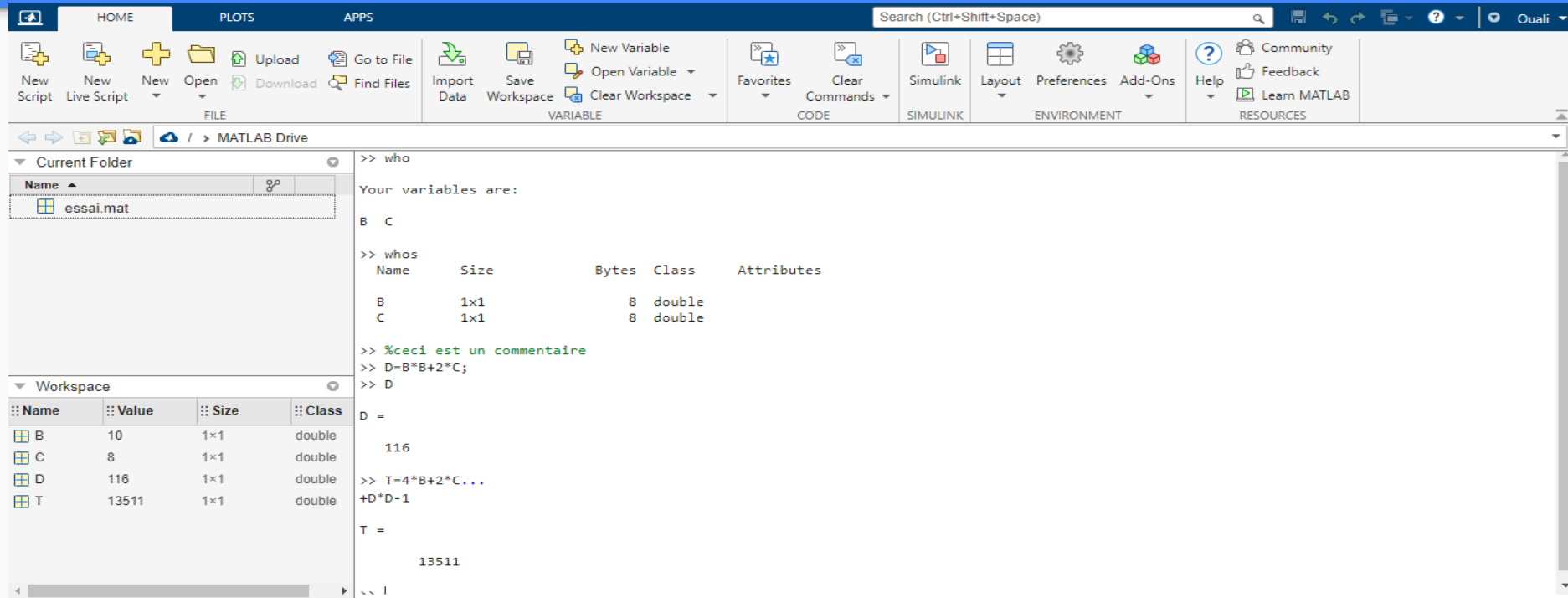
Maîtrise du logiciel matlab

matlab

- Généralités
- Introduction a Matlab
- Interface matlab
- Listes de quelques commandes d'environnement
- Exemple

chapitre 01: Maîtrise du logiciel matlab

Exemple



The image shows the MATLAB desktop environment. The top ribbon includes tabs for HOME, PLOTS, and APPS, along with a search bar. The HOME tab is active, displaying various toolbars for file operations, workspace management, and code execution. The left sidebar shows the 'Current Folder' and 'Workspace' panels. The 'Current Folder' panel shows a file named 'essai.mat'. The 'Workspace' panel displays a table of variables in the workspace.

Name	Value	Size	Class
B	10	1x1	double
C	8	1x1	double
D	116	1x1	double
T	13511	1x1	double

The main window shows the MATLAB Command Window with the following code and output:

```
>> who
Your variables are:
B C

>> whos
      Name      Size      Bytes  Class  Attributes
      B         1x1         8    double
      C         1x1         8    double

>> %ceci est un commentaire
>> D=B*2+C;
>> D
D =
    116

>> T=4*B+2*C...
+D*D-1
T =
    13511

>> |
```

Maîtrise du logiciel matlab

La programmation sous matlab

- Introduction & généralités
- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations
- Vecteurs
- Matrices

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Introduction & généralités

- Les programmes sous Matlab s'appellent scripts,
- Un script Matlab ne contient pas de partie déclarative et ne contient pas d'entête du programme,
- Un script Matlab se compose d'un ensemble d'instruction,
- Les variables sont déclarées automatiquement (les types de variables et dimensions de variables n'ont pas à être déclarés préalablement avant leur utilisation et manipulation, même pour les tableaux et les matrices. Il suffit simplement d'affecter (=) une valeur au nom de la variable depuis la fenêtre de commande.).

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- Dans le cas où la variable existe déjà, Matlab modifie son contenu et, si nécessaire, effectue une allocation d'espace mémoire selon le besoin (exemple : redimensionnement d'un tableau de variables).
- La création et la manipulation de variables est basée sur des expressions. Un nom (identifiant) de variable valide doit respecter les conditions suivantes :
 - Un Identifiant de variable ne peut pas être un mot clé propre à Matlab
 - Commencer par une lettre suivie de lettres, chiffres ou caractères souligné « _ »
 - Un Identifiant de variable ne peut pas contenir d'espace
 - Utiliser des lettres dans l'intervalle a-z et A-Z
 - Les caractères spéciaux et accentués ne sont pas autorisés (@, #, é, è...)
 - Contenir jusqu'à 63 caractères maximum.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemples

Quels sont les exemples valide et non valide parmi les variables ci-dessous:

- temp1 , temp-1, temp_1,tempé_1, X_min , mat_B1,01XY ,Temp1

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemples

- ❖ Exemples valides :
 - temp1 , Temp1, temp_1, X_min , mat_B1
- ❖ Exemples non valides :
 - 01XY (commence par un chiffre),
 - temp-1 (contient -),
 - tempé_1 (contient un caractère accentué)

Remarque : Matlab fait une distinction entre les minuscules et les majuscules. La variable MAT_A est différente de la variable mat_a

Maîtrise du logiciel matlab

La programmation sous matlab

- Introduction & generalites
- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations
- Vecteurs
- Matrices

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- Le typages de données standards:
 - Le type "double" représente les nombres réels, entier et même les nombres complexe avec deux valeurs (partie réelle et partie imaginaire 'codé sur 8 octets')

Exemple

```
>> a=2 %le pourcentage permet d'introduire des commentaires  
a = 2  
>> b=2.02 %nombre réel  
b = 2.0200  
>> f=0.000123 %notation décimale  
f = 1.2300e-04 %Avec résultat en notation scientifique (e)
```

```
>>c=2.34*10^(-6) %expression_1  
c = 2.3400e-06 %notation scientifique équivalente de  
l'expression_1
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- Le typages de données standards:
 - Le type complexe : dans Matlab, un nombre complexe est donné sous la forme $z = a + bi$. Stocké de façon interne sur 2×8 octets, respectivement pour la partie réelle et la partie imaginaire. Les fonctions usuelles de manipulation des nombres complexes sont prédéfinies dans Matlab (`real(z)`, `imag(z)`, `abs(z)`, `arg(z)` et `conj(z)`).

Exemple

```
>> z=1+2.3i
z = 1.0000 + 2.3000i
>> real(z) %Partie réelle de z
ans = 1
>> imag(z) %Partie imaginaire de z
ans = 2.3000
```

```
>> conj(z) %Calculer le conjugué de z
ans = 1.0000 - 2.3000i
>> abs(z) %Calculer le module de z
ans = 2.5080
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- Le typages de données standards:
 - le type "char" qui représente les caractères et les chaînes de caractères, ce type est codé sur 2 octets pour chaque caractère(en utilisant la codification Unicode), pour créer une chaîne de caractères il faut la déclarer avec des guillemets simples (') au début et à la fin.

Exemple

```
>> CH='Matlab' %Créer une chaîne de caractères CH = Matlab  
>> CH(3) %Afficher le 2ième élément de la chaîne CH  
ans = t
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- Le typages de données standards:
 - le type "logical" représente deux valeurs logiques "true" représenté par 1 et "false" représenté par la valeur 0. Ce type est codé sur 1 octet

Exemple

```
>> x= true
x = 1
>> y= false
y = 0
>>d=1==5
d=
logical
0
```

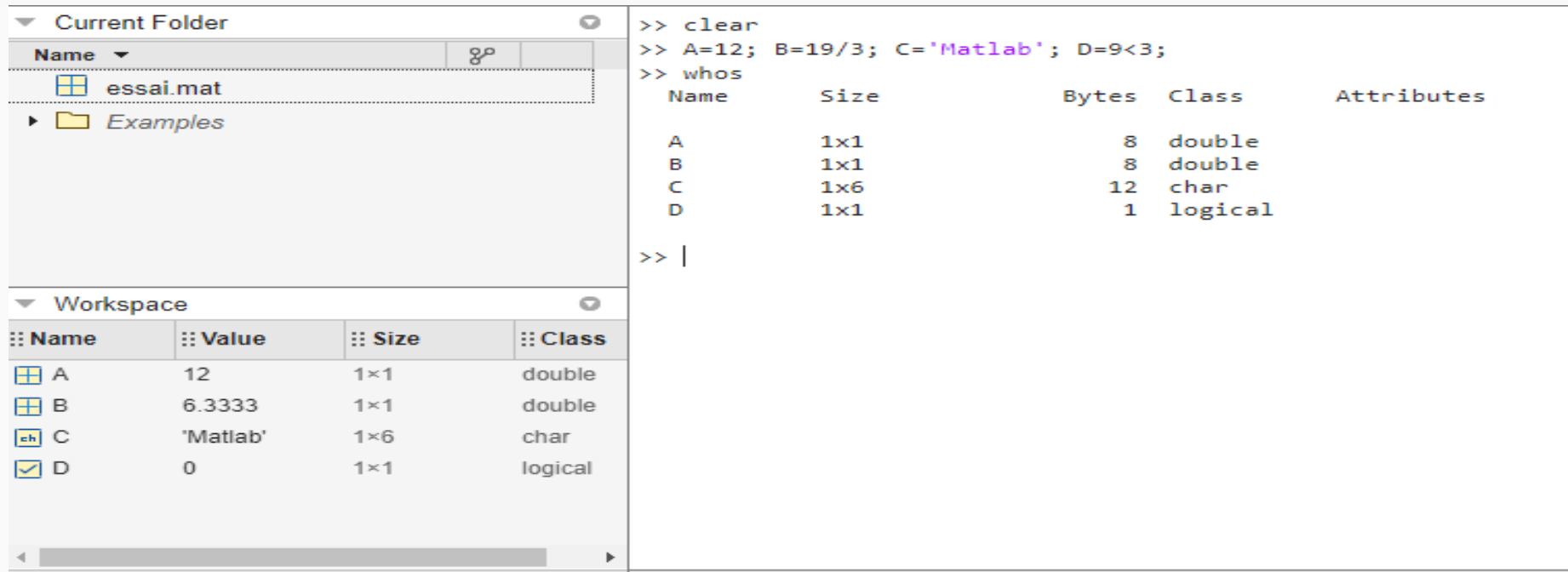
```
>>k=8==5+3

k =
logical

1
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab



The screenshot displays the MATLAB environment. On the left, the 'Current Folder' panel shows a file named 'essai.mat' and a subfolder 'Examples'. Below it, the 'Workspace' panel lists four variables: A (double, 1x1), B (double, 1x1), C (char, 1x6), and D (logical, 1x1). On the right, the Command Window shows the execution of the following code:

```
>> clear
>> A=12; B=19/3; C='Matlab'; D=9<3;
>> whos
```

Name	Size	Bytes	Class	Attributes
A	1x1	8	double	
B	1x1	8	double	
C	1x6	12	char	
D	1x1	1	logical	

The Command Window prompt is currently at '>> |'.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **Vecteurs :**

- ❖ un vecteur ligne se déclare entre crochets [..] en séparant les éléments avec des espaces ou des virgules(,).
- ❖ Pour un vecteur colonne, le séparateur est un point- virgule(;).

```
>> v = [1 2 3] % vecteur ligne avec des espaces au milieu  
v = 1 2 3  
>> v = [1, 2, 3] % vecteur ligne avec des virgules au milieu  
v = 1 2 3
```

```
>> z = [4;5;6] % vecteur colonne avec des points-virgules au milieu  
z = 4  
    5  
    6  
>> z = [4 5 6]' % Il est également possible d'utiliser  
l'opération de transposition  
z = 4  
    5  
    6
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **Matrice :**

- ❖ une matrice se déclare comme les vecteurs, en séparant les colonnes par des espaces et les lignes par des points- virgule.

```
>> A = [1 2; 3 4] %Créer une matrice 2x2
```

```
A =
```

```
1 2
```

```
3 4
```

```
>> A2=[1,2,3;4,5,6;7,8,9]%Créer une matrice 3x3
```

```
A2 =
```

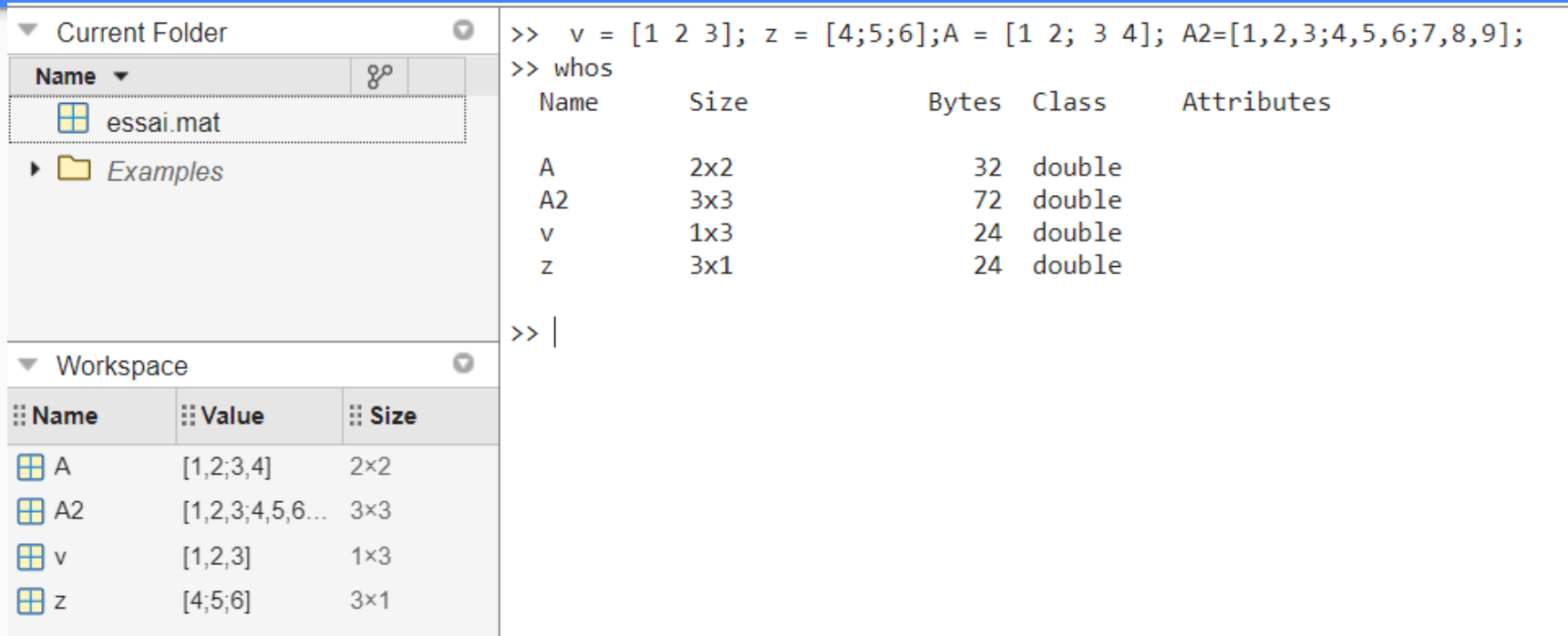
```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab



The screenshot displays the MATLAB environment. On the left, the 'Current Folder' pane shows a file named 'essai.mat' and a folder named 'Examples'. Below it, the 'Workspace' pane lists the variables currently in memory: A (2x2), A2 (3x3), v (1x3), and z (3x1). On the right, the Command Window shows the execution of MATLAB code and the output of the 'whos' command.

Current Folder

- essai.mat
- Examples

Workspace

Name	Value	Size
A	[1,2;3,4]	2x2
A2	[1,2,3;4,5,6...]	3x3
v	[1,2,3]	1x3
z	[4;5;6]	3x1

Command Window

```
>> v = [1 2 3]; z = [4;5;6]; A = [1 2; 3 4]; A2=[1,2,3;4,5,6;7,8,9];  
>> whos
```

Name	Size	Bytes	Class	Attributes
A	2x2	32	double	
A2	3x3	72	double	
v	1x3	24	double	
z	3x1	24	double	

```
>> |
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **Le typage de données (typage dynamique)**

Matlab permet d'adapter les types des opérateurs à fin de permettre aux opérations de s'effectuer,

1. Avantages:

- Le programmeur n'est pas obligé de définir à l'avance le type de chaque variable et être coincé tout au long de l'exécution par ce type,
- Matlab effectue des conversions implicitement sans le signaler; ce qui peut augmenter le temps d'exécution et parfois mener à de très bons résultats

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab




Exemple

```
>> A=5,B=2.5,C=A+B
```

▼ Current Folder

Name ▼		
 <code>essai.mat</code>		
▶  <code>Examples</code>		

▼ Workspace

Name	Value	Size	Class
 A	5	1×1	double
 B	2.5000	1×1	double
 C	7.5000	1×1	double

A =

5

B =

2.5000

C =

7.5000

>>

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

Le type d'une variable est déterminé de manière automatique à partir de l'expression mathématique ou de la valeur affectée à la variable. La commande suivante :

```
>> a=45; b='Hello'; c=a+5i; d=true;
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

Le type d'une variable est déterminé de manière automatique à partir de l'expression mathématique ou de la valeur affectée à la variable. La commande suivante :

```
>> a=45; b='Hello'; c=a+5i; d=true;
```

Name	Size	Bytes	Class
Attributes			
a	1x1	8	double
b	1x5	10	char
c	1x1	16	double complex
d	1x1	1	logical

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Remarque

NB: Il est impossible de déclarer le type de variable lorsque l'on crée une variable dans Matlab. Mais pour vérifier le type d'une variable on peut utiliser les fonctions **ischar**, **islogical**, **isreal**.

```
>>b='Bonjour';C=true;D=2;E=3.50;  
  ischar(b)  
  ans =  
  logical  
  1  
>>islogical(C)  
  ans =  
  logical  
  1  
>>isreal(D)  
  ans =  
  logical  
  1
```

```
>>isreal(E)  
  ans =  
  logical  
  1  
>>ischar(E)  
  ans =  
  logical  
  0  
>>islogical(D)  
  ans =  
  logical  
  0
```


Maîtrise du logiciel matlab

La programmation sous matlab

- Introduction & generalites
- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations
- Vecteurs
- Matrices

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **Les formats d'affichage**

1. Matlab dispose de plusieurs formats d'affichage des réels. Par défaut le format est le `format court` à 5 chiffres.
2. Matlab dispose également des formats `format short g` et `format long g` qui utilise la « meilleure » des deux écritures à virgule fixe ou à virgule flottante.
3. On obtient tous les formats d'affichage possibles en tapant `help format`.
4. On impose un format d'affichage en tapant l'instruction de format correspondante dans la fenêtre de contrôle.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **Les formats d'affichage**

- » `format rat` %forme rationnelle.
- » `format long` %forme décimale avec 15 chiffres avec notation en virgule flottante.
- » `format long e` %forme exponentielle longue.
- » `format hex` %forme hexadécimale telle que représentée en mémoire
- » `format short e` %forme exponentielle courte.
- » `format short` %forme par défaut à 5 chiffres avec notation en virgule flottante.

```
»pi;  
»format long  
pi  
ans =  
3.141592653589793  
»format short  
»pi  
ans =  
3.1416
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonction d'entrée input**

La lecture sous Matlab se fait en appelant la commande input. La syntaxe est comme suit:

» `var=input('phrase');`

- ❖ Une phrase sera affichée et MATLAB attend que l'utilisateur saisisse une donnée au clavier.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonctions de sortie disp**

- ❖ Permet d'afficher la valeur d'une variable ou une chaîne de caractères (messages).
- ❖ L'autre façon d'afficher la valeur d'une variable est de taper son nom.
- ❖ La commande disp se contente d'afficher la valeur de la variable sans afficher ce qui peut améliorer certaines présentations.
- ❖ On utilise également la commande disp pour afficher un message et un ou plusieurs résultats en même temps.
- ❖ Les trois syntaxes possibles pour la fonction disp sont les suivants :

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonctions de sortie disp**

- ❖ Les trois syntaxes possibles pour la fonction disp sont les suivants :

- *Affichage d'un message*

- »disp('this is a message');

this is a message

- Affichage de la valeur d'une variable var

- »A=15;

»disp(A);

15

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonctions de sortie disp**

- *Affichage d'un message et d'un résultat*

syntaxe: `disp(['message', num2str(var)])`

Exemple:

```
»A=15; B='Hello';
```

```
»disp(['La valeur de A est ', num2str(A), ' et la valeur de B est ', num2str(B)])
```

```
La valeur de A est :15 et la valeur de B est Hello
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonction de sortie sprintf**

- ❖ La commande sprintf permet l'impression de variables selon un modèle donné.
Un modèle d'édition se présente sous la forme du symbole pourcent (%) suivi d'indications permettant de composer le contenu du champ à imprimer, en particulier sa longueur en nombre de caractères.

Le modèle d'édition utilisé par MATLAB est le modèle d'édition du Langage C.

La syntaxe de la commande sprintf : `sprintf('format', variables)`

- Les formats d'impression en Matlab sont donnés dans le tableau suivant:

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

format	conversion en
%d	entier
%e	Pour une notation à virgule flottante où la partie exposant est délimitée par un e minuscule
%E	Même notation mais E remplace e
%f	Pour une notation à virgule fixe
%g	format libre
%s	Pour chaîne de caractères

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonction de sortie sprintf**

- ❖ En plus du caractère donnant le type des données, on peut éventuellement préciser certains paramètres du format d'impression, qui sont spécifiés entre le % et le caractère de conversion dans l'ordre suivant :

- La largeur du champ d'impression :

- ☐ **%7d** spécifie qu'au moins 7 caractères seront réservés pour imprimer l'entier.

- La précision :

- ☐ **%.12f** signifie qu'un flottant sera imprimé avec 12 chiffres après la virgule.
- ☐ **%10.2f** signifie que l'on réserve 12 caractères (incluant le point .) pour imprimer le flottant et que 2 d'entre eux sont destinés aux chiffres après la virgule.
- ☐ Quant à la précision si elle n'est pas spécifiée, elle correspond par défaut à 4 chiffres après la virgule.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

Nous voulons avoir l'interface suivante:

Donner la valeur de X? 12.345

Donner la valeur de Y? 6.789

Affichage

La valeur de X est 12.3 et la valeur de Y est 6.79

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

Nous voulons avoir l'interface suivante:

Donner la valeur de X? 12.345

Donner la valeur de Y? 6.789

Affichage

La valeur de X est 12.3 et la valeur de Y est 6.79

```
» X=input('Donner la valeur de X?');
```

Donner la valeur de X?

12.345

```
» Y=input('Donner la valeur de Y');
```

Donner la valeur de Y

6.789

```
» sprintf('La valeur de X est %.1f et la valeur de Y est  
%.2f ',X,Y)
```

ans =

'La valeur de X est 12.3 et la valeur de Y est 6.79'

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

- **La fonction de sortie fprintf**

- ☐ C'est une fonction inspirée du langage C
- ☐ Elle permet d'afficher plusieurs variables et messages en tenant compte du format défini dans l'appel de la fonction.
- ☐ Pour afficher le contenu des variables, il faut définir le masque d'affichage du contenu de la variable(et un masque pour chaque variable)

Syntaxe:

```
NomVariable;  
dis(NomVariable);  
disp('message');  
fprintf('message %masque_Affichage', NomVariable);
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

Nous voulons avoir l'interface suivante (en utilisant fprintf):

Donner la valeur de X? 12.345

Donner la valeur de Y? 6.789

Affichage

La valeur de X est 12.35 et la valeur de Y est 7

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

Nous voulons avoir l'interface suivante (en utilisant fprintf):

Donner la valeur de X? 12.345

Donner la valeur de Y? 6.789

Affichage

La valeur de X est 12.35 et la valeur de Y est 7

```
» X=input('Donner la valeur de X?');
```

Donner la valeur de X?

12.345

```
» Y=input('Donner la valeur de Y');
```

Donner la valeur de Y

6.789

```
» fprintf('La valeur de X est %.2f et la valeur de Y est  
%.0f ',X,Y);
```

La valeur de X est 12.35 et la valeur de Y est 7

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Remarques

- ❖ Si on a besoin d'afficher le caractère % on le doublera %% pour qu'il ne soit pas interprété comme le début d'un format.
- ❖ On peut avoir besoin d'inclure dans la chaîne de caractères format, un certain nombre de caractères spéciaux dont la liste est donnée dans le tableau ci-dessous.

Caractères en Matlab	signification
<code>\n</code>	fin de ligne
<code>\t</code>	tabulation horizontale
<code>\v</code>	tabulation verticale
<code>\b</code>	retour arrière
<code>\r</code>	saut horizontal

Maîtrise du logiciel matlab

La programmation sous matlab

- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations
- Vecteurs
- Matrices

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

- Les opérateurs de base
 - Matlab utilise la codification des opérateurs de base utilisée dans la plupart des langages de programmation évolués avec des petits changements.
 - Parmi les particularités de Matlab est d'avoir des opérateurs spécifiques aux calculs matriciels, de plus de la caractéristique du typage dynamique, Matlab exécute la bonne opération en fonction du type des termes de l'opérateur.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Les opérateurs logiques sous matlab

Opérateur	Description
<code>~ a</code>	NOT - retourne 1 si a est égal 0, et 0 si a est différent de 0
<code>a == b</code>	Retourne 1 si a est égal b, 0 autrement
<code>a < b</code>	Retourne 1 si a est inférieur à b, 0 autrement
<code>a > b</code>	Retourne 1 si a est supérieur à b, 0 autrement
<code>a <= b</code>	Retourne 1 si a est inférieur ou égal à b, 0 autrement
<code>a >= b</code>	Retourne 1 si a est supérieur ou égal à b, 0 autrement
<code>a & b</code> ou <code>a && b</code>	Retourne 1 si a et b sont vrais, 0 autrement
<code>a b</code> ou <code>a b</code>	Retourne 0 si a et b sont faux, 1 autrement
<code>xor(a,b)</code>	Retourne 1 si a et b sont faux ou vrais les deux, 0 autrement
<code>any(a)</code>	Retourne 1 si un des éléments de a est non nul
<code>all (a)</code>	Retourne 1 si tous les éléments de a sont non nuls

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Exemple:

Si on utilise l'opérateur + pour deux valeurs de type char Matlab remplace les caractères par leurs valeurs numériques(codage unicode) et effectue l'opération d'addition(l'opération de la somme est définie dans l'espace des nombres)

```
»A='Bonjour'
```

```
A =
```

```
    'Bonjour'
```

```
»B='Bonsoir'
```

```
B =
```

```
    'Bonsoir'
```

```
»C=A+B
```

```
C =
```

```
    132    222    220    221    222
```

```
    222    228
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Des caractères spéciaux

Opérateur	Description
'	Transposé d'une matrice
+	Addition
-	Soustraction
*	Multiplication
/	Division à droite
\	Division à gauche
^	Puissance

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Les instructions de contrôle:

- **Les instructions conditionnelles**
 - **L'instruction IF**

1. La forme simple est donnée comme suit:

```
if (condition)
    instruction
end
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Les instructions de contrôle:

- **Les instructions conditionnelles**
 - **L'instruction IF**

2.La forme générale est donnée sous la forme suivante:

```
if (condition)
    instruction ou ensemble d'instructions
else
    instruction ou ensemble d'instructions
end
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Les instructions de contrôle:

- **Les instructions conditionnelles**
 - **L'instruction IF**

2. Une autre forme se présente comme suit:

```
if (condition 1)
    instruction ou ensemble d'instructions
elseif(condition 2)
    instruction ou ensemble d'instructions
else
    instruction ou ensemble d'instructions
end
```


chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 01

Écrire un script Matlab qui permet de:

- 1.lire deux valeurs entières **A** et **b***
- 2.Trouver le maximum entre **A** et **b***
- 3. Sauvegarder maximum dans la variable **Max***



Stop

SECTION



📁 / > MATLAB Drive > lilia

ConditionIF.m  


ConditionIF.m

▼ Workspace

Name	Value	Size
------	-------	------

A	10	1x1
---	----	-----

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} a \quad 5 \quad 1 \times 1$$

 b 123 1x1

 Max 123 1x1

```
1 A=input('donner un valeur a A');
2 b=input('donner un valeur a b');
3 if(A>b)
4     Max=A;
5 else
6     Max=b;
7 end
8 fprintf('le Maximum est %d',Max);
```

Command Window

```
>> ConditionIF
donner un valeur a A
10
donner un valeur a b
123
le Maximum est 123
>>
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 02

Écrire un script Matlab qui permet de:

- 1.lire une valeur entière entre 1..12*
- 2.Afficher le mois équivalent à ce chiffre.*

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple

```
disp('entrer un chiffre entre 1 et 12');  
mois=input('');  
if(mois==1)  
    disp('vous êtes au mois de janvier');  
elseif(mois==2)  
    disp('vous êtes au mois de février');  
elseif(mois==3)  
    disp('vous êtes au mois de mars');  
elseif(mois==4)  
    disp('vous êtes au mois de Avril');  
elseif(mois==5)  
    disp('vous êtes au mois de mai');  
elseif(mois==6)  
    disp('vous êtes au mois de juin');
```

```
elseif(mois==7)  
    disp('vous êtes au mois de juillet');  
elseif(mois==8)  
    disp('vous êtes au mois de Août');  
elseif(mois==9)  
    disp('vous êtes au mois de septembre');  
elseif(mois==10)  
    disp('vous êtes au mois de octobre');  
elseif(mois==11)  
    disp('vous êtes au mois de novembre');  
elseif(mois==12)  
    disp('vous êtes au mois de décembre');  
end
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Les instructions de contrôle:

- *Les instructions conditionnelles du choix switch*

```
switch (choix )  
case val1  
    action 1  
case val2  
    action 2  
...  
case valn  
    action n  
otherwise  
    autre  
end
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 03

Écrire un script Matlab qui permet de:

- 1.Lire une chaine de caractère qui représente les mois de L'an ou les jours de la semaine*
- 2.Afficher si c'est un mois ou bien c'est un jour de semaine.*

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 03

```
chaine=input('veuillez saisir un jours de semaine ou un mois');
switch (chaine)
    case
        {'janvier','fevrier','mars','avril','mai','juin','juillet','aout','septembre','octobre','novembre','decembre'}
            disp('vous avez saisi un mois')
        case {'samedi','dimanche','lundi','mardi','mercredi','jeudi','vendredi'}
            disp('vous avez saisi un jour de semaine ')
        otherwise
            disp('erreur')
end
```

Current Folder

Name
conditionSWITCH.m
ConditionIF.m
ConditionIF.m

Workspace

Name	Value	Size
chaine	'dimanche'	1x8

ConditionIF.m x ConditionIF.m x conditionSWITCH.m x

```
1 chaine=input('veuillez saisir un jours de semaine ou un mois');
2 switch (chaine)
3     case {'janvier','fevrier','mars','avril','mai','juin','juillet','août','septembre','octobre','novemb
4         disp('vous avez saisie un mois')
5     case {'samedi','dimanche','lundi','mardi','mercredi','jeudi','vendredi'}
6         disp('vous avez saisie un jour de semaine ')
7     otherwise
8         disp('erreur')
9 end
```

Command Window

```
>> conditionSWITCH
veuillez saisir un jours de semaine ou un mois
'dimanche'
vous avez saisie un jour de semaine
>>
```


chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

- *L'instruction répétitive while*

```
while (condition )  
    Instruction 1;  
    ...  
    Instruction n;  
end
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 04

Écrire un programme qui calcule la puissance d'un nombre donné en suivant ces étapes:

1. Lire deux chiffres **X** et **n**
2. Calculer **Xⁿ** en utilisant l'instruction while.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 04

```
disp('calculer la puissance X^n');  
X=input('veuillez introduire un entier');  
n=input('veuillez saisir la puissance');  
i=1;puis=1;  
while(i<=n)  
    puis=puis*X  
    i=i+1  
end  
fprintf('%d puissance %d est de %d',X,n,puis);
```

Current Folder

Name ▲	
lilia	
ConditionIF.m	
ConditionIFelse.m	
conditionSWITCH.n	
linstruction WHILE.r	

Workspace

Name	Value	Size
i	4	1×1
n	3	1×1
puis	216	1×1
X	6	1×1

new.m x while.m x +

```

1      disp('calculer la puissance X^n');
2      X=input('veuillez introduire un entier');
3      n=input('veuillez saisir la puissance');
4      i=1;puis=1;
5      while(i<=n)
6          puis=puis*X;
7          i=i+1;
8      end
9      fprintf('%d puissance %d est de %d',X,n,puis);
  
```

Command Window

```

>> new
calculer la puissance X^n
veuillez introduire un entier
6
veuillez saisir la puissance
3
6 puissance 3 est de 216
  
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

- *L'instruction répétitive for*

```
for indice=valeur initiale:pas:valeur finale  
    Instruction 1;  
    ...  
    Instruction n;  
end
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les structures de programmations

Quelques exemples

<pre>for n=6:-2:0 disp(n); end</pre>	<pre>6 4 2 0</pre>
<pre>for T=[1 5 3 7] disp(T); end</pre>	<pre>1 5 3 7</pre>
<pre>K=[-2 8 -1 0]; for s=K disp(s) end</pre>	<pre>-2 8 -1 0</pre>

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Exemple 05

Ecrire un programme qui calcule la puissance d'un nombre donné en suivant ces étapes:

1. Lire deux chiffres **X** et **n**
2. Calculer **Xⁿ** en utilisant l'instruction for.

Current Folder

Name ▲	
conditionSWITCH.m	
for1.m	
linstruction_WHILE.m	
new.m	
while1.m	

Workspace

Name	Value	Size
i	3	1x1
n	3	1x1
puis	8	1x1
X	2	1x1

new.m x for1.m x +

```

1  disp('calculer la puissance X^n');
2  X=input('veuillez introduire un entier X');
3  n=input('veuillez saisir la puissance n');
4  puis=1;
5  for i=1:n
6      puis=puis*X;
7  end
8  fprintf('%d puissance %d est de %d',X,n,puis);

```

Command Window

```

>> for1
calculer la puissance X^n
veuillez introduire un entier X
2
veuillez saisir la puissance n
3
2 puissance 3 est de 8
>>

```


Maîtrise du logiciel matlab

La programmation sous matlab

- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations
- Vecteurs
- Matrices

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs

Un vecteur sous Matlab:

- C'est un ensemble d'éléments du même type
- Vecteur ligne >> $v = [1 \ 2 \ 3]$ ou >> $v = [1, 2, 3]$
- Vecteur colonne >> $z = [4;5;6]$ ou >> $z = [4 \ 5 \ 6]'$
- Concaténer deux vecteurs

```
»A=[1 2 3 4] ;  
»B=[5,6,7] ;  
»C=[A B]  
C=  
1 2 3 4 5 6 7
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs

Un vecteur sous Matlab:

- la taille des vecteurs est manipulée dynamiquement (ce fait automatique) par Matlab;
- fonction `length()` permet de retourner la taille d'un tableau.

```
»A=[1 2 3 4 5 6 7] ;  
»length(A)  
»A=  
    7
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs

Un vecteur sous Matlab:

- L'accès à un élément dans un vecteur se fait grâce à son indice dans le tableau.
- le premier élément dans un vecteur correspond à l'indice 1

```
»A=[1 -2 3 0 5 -6 7] ;  
»A(4)  
»A=  
0
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs

Un vecteur sous Matlab:

- Il est possible d'extraire un sous-vecteur par la commande Nom Vecteur (indice initiale : indice final)

```
»A=[1 -2 3 0 5 -6 7] ;  
»A(4:6)  
» 0 5 -6
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs

Un vecteur sous Matlab:

- Si les composants d'un vecteur X sont ordonnés avec des valeurs consécutives, nous pouvons le noter avec la notation suivante : **X = premier_élément : dernier_élément** (les crochets sont facultatifs)

```
>>X = 1:8 % [1:8]  
>>X =  
1 2 3 4 5 6 7 8
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs

Un vecteur sous Matlab:

- Il est possible de créer un vecteur espacé linéairement en utilisant **[a : h : b]** où les valeurs du vecteur sont entre **a** et **b** avec un espacement **h**.

```
»A=[1 :3 :15]
```

```
»A=
```

```
1 4 7 10 13
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs spéciaux

Il existe des vecteurs spéciaux prédéfinis dans Matlab :

- `ones(1, n)`

- vecteur ligne de longueur **n** dont tous les éléments valent 1

The screenshot displays the MATLAB environment. The 'Current Folder' pane on the left shows a file named 'vecteurs_speciaux.m'. The 'Workspace' pane below it lists two variables: 'ans' with a value of [1,1,1,1,1] and size 1x5, and 'n' with a value of 5 and size 1x1. The 'Command Window' on the right shows the execution of the command `n=5; ones(1, n)`, resulting in the output `ans = 1 1 1 1 1`.

```
1 n=5
2 ones(1, n)
3
```

Name	Value	Size
ans	[1,1,1,1,1]	1x5
n	5	1x1

```
ans =
    1    1    1    1    1
>>
```


chapitre 01: Maîtrise du logiciel matlab

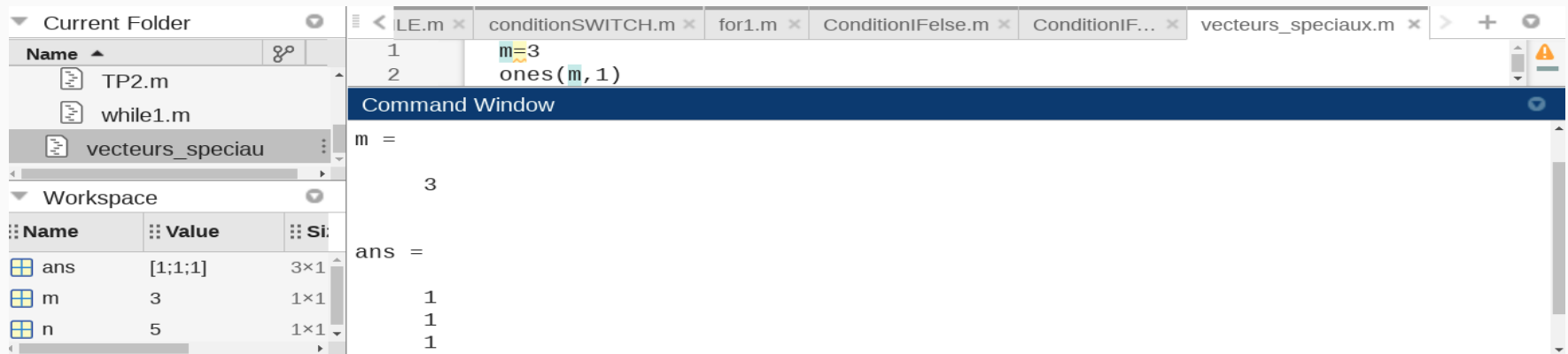
La programmation sous Matlab

Les vecteurs spéciaux

Il existe des vecteurs spéciaux prédéfinis dans Matlab :

- `ones(m, 1)`

- vecteur colonne de longueur m dont tous les éléments valent 1



The screenshot displays the MATLAB environment. The Command Window shows the execution of the command `ones(m, 1)` where `m` is assigned the value 3. The output is a 3x1 column vector of ones, displayed as `ans =` followed by the values 1, 1, and 1 on separate lines. The Workspace window on the left shows the current state of variables: `ans` is a 3x1 matrix, `m` is a scalar 3, and `n` is a scalar 5. The Current Folder window shows the file `vecteurs_speciaux.m` is open.

Name	Value	Size
ans	[1;1;1]	3x1
m	3	1x1
n	5	1x1

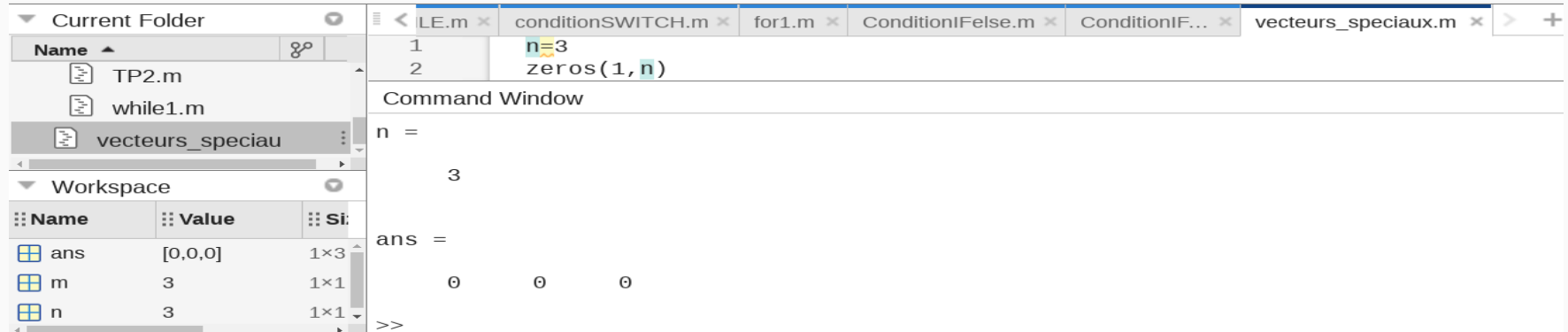
chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs spéciaux

Il existe des vecteurs spéciaux prédéfinis dans Matlab :

- `zeros(1,n)`
- vecteur ligne de longueur n dont tous les éléments valent 0



The screenshot shows the MATLAB environment. The Command Window displays the following code and output:

```
n =  
    3  
  
ans =  
    0    0    0  
  
>>
```

The Workspace window shows the following variables:

Name	Value	Size
ans	[0,0,0]	1×3
m	3	1×1
n	3	1×1

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs spéciaux

Il existe des vecteurs spéciaux prédéfinis dans Matlab :

- `zeros(m,1)`

- vecteur colonne de longueur m dont tous les éléments valent 0



The screenshot shows the MATLAB environment. The Command Window displays the following code and output:

```
1 m=2
2 zeros(m,1)

Command Window

m =

     2

ans =

     0
     0
```

The Workspace window shows the following variables:

Name	Value	Size
ans	[0;0]	2x1
m	2	1x1
n	3	1x1

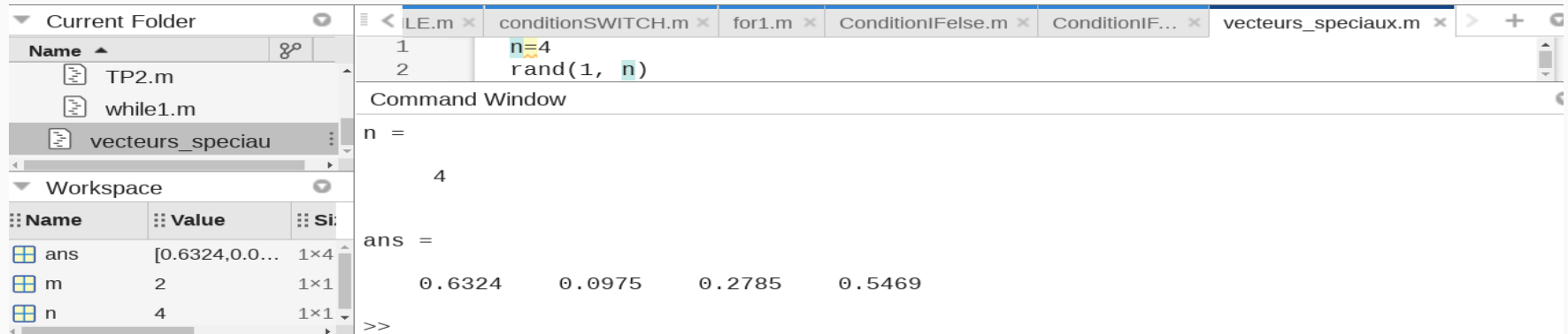
chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs spéciaux

Il existe des vecteurs spéciaux prédéfinis dans Matlab :

- `rand(1,n)`
- vecteur ligne de longueur n dont les éléments sont générés de manière aléatoire entre 0 et 1



The screenshot shows the MATLAB environment. The Command Window displays the following code and output:

```
1  
2 n=4  
   rand(1, n)  
  
Command Window  
  
n =  
  
     4  
  
ans =  
  
    0.6324    0.0975    0.2785    0.5469  
  
>>
```

The Workspace window shows the following variables:

Name	Value	Size
ans	[0.6324, 0.0975, 0.2785, 0.5469]	1x4
m	2	1x1
n	4	1x1

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les vecteurs spéciaux

Il existe des vecteurs spéciaux prédéfinis dans Matlab :

- `rand(m,1)`
- vecteur colonne de longueur m dont les éléments sont générés de manière aléatoire entre 0 et 1



The screenshot shows the MATLAB IDE interface. The top toolbar includes navigation icons and the 'MATLAB Drive' path. The 'Current Folder' pane on the left lists files: TP2.m, while1.m, and vecteurs_speciau. The 'Workspace' pane below it shows variables: ans (2x1 double), m (1x1 double), and n (1x1 double). The main editor window displays a script with the following code:

```
1  
2  
m=2  
rand(m, 1)
```

The 'Command Window' at the bottom shows the execution results:

```
m =  
    2  
  
ans =  
    0.1270  
    0.9134
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux vecteurs

● `V = [val var expr ...]`

- Créer un vecteur ligne V contenant des valeurs (val), des variables (var), ou expressions (expr) spécifiées.

```
K=-6,  
C=12,  
V=[0 2 K C+K C 10]
```

```
K =  
    -6  
C =  
    12  
V =  
     0     2    -6     6    12    10
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux vecteurs

- | | |
|--|---|
| <ul style="list-style-type: none">• <code>V(indices)=[]</code> | <ul style="list-style-type: none">• Suppression des éléments indicés du vecteur V.• Matlab effectue un redimensionnement dynamique après destruction des cases |
|--|---|

```
v=0:2:20  
v(3:6)  
v(3:6)=[  
]
```

```
v =  
0     2     4     6     8    10    12    14    16    18    20  
  
ans =  
4     6     8    10  
  
v =  
0     2    12    14    16    18    20
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux vecteurs

- `length(v)`

- Renvoie le nombre d'éléments (taille) du vecteur V (ligne ou colonne).

```
v=0:2:20  
length(v)  
v(3:6)  
v(3:6)=[ ]  
length(v)
```

```
v =  
    0     2     4     6     8    10    12    14    16    18    20  
ans =  
    11  
ans =  
     4     6     8    10  
v =  
     0     2    12    14    16    18    20  
ans =  
     7
```


chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux vecteurs

● `V= linspace(V0 ,V f ,n)`

- Créer un vecteur de suite arithmétique en précisant : le premier terme V_i , le dernier terme V_f et le nombre de termes n .
- Matlab calcule automatiquement le pas
(La raison= $(V_f - V_0)/(n-1)$)

`V=linspace(10,20,5)`

`V =`

10.0000 12.5000 15.0000 17.5000 20.0000

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux vecteurs

●diag(Vecteur)

A partir d'un Vecteur ligne ou colonne, la fonction retourne une matrice carrée dont la diagonale principale porte les éléments du vecteur et les autres éléments sont égaux à "0"

```
V=linspace(0,6,4)
```

```
diag(V)
```

```
V =
```

```
    0    2    4    6
```

```
ans =
```

```
    0    0    0    0
```

```
    0    2    0    0
```

```
    0    0    4    0
```

```
    0    0    0    6
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Opérations vectorielles

Les opérations arithmétique usuelles $+$, $-$, $*$, $/$ doivent être utilisés avec précautions pour les vecteurs. La somme et la différence sont des opérations termes à termes, et nécessitent donc des vecteurs de la même dimension.

```
A=[2 10 30];B=[3 6 20];
```

```
A+B
```

```
A-B
```

```
ans =
```

```
5      16      50
```

```
ans =
```

```
-1       4      10
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Opérations vectorielles

Le produit `*` est le produit matriciel. Pour utiliser la multiplication, la division termes à termes on doit remplacer `*` par `.*`, `/` par `./`

```
A=[2 10 30];B=[3 6 20];
```

```
A.*B
```

```
A./B
```

```
ans =
```

```
6      60     600
```

```
ans =
```

```
0.6667     1.6667     1.5000
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Opérations vectorielles

```
X=[1 5 3 9 7 0];  
sum(X)%la somme des éléments du vecteur X  
prod(X)%le produit des éléments du vecteur X  
max (X)%le plus grand élément du vecteur X  
min(X)%le plus petit élément du vecteur X  
mean(X)%la moyenne des éléments du vecteur X  
sort(X)%Ordonner les éléments du vecteur X  
par ordre croissant  
fliplr (X)%Renverser l'ordre des éléments du  
vecteur X
```

```
ans =  
    25  
ans =  
     0  
ans =  
     9  
ans =  
     0  
ans =  
    4.1667  
ans =  
     0     1     3     5     7     9  
ans =  
     0     7     9     3     5     1
```

Maîtrise du logiciel matlab

La programmation sous matlab

- Le typage de données
- Les entrées/sorties sous matlab
- Les structures de programmations
- Vecteurs
- Matrices

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Une matrice Matlab est un tableau à deux dimensions de (**$n \times m$**) éléments de types nombres réels ou complexes ou de caractère

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Une matrice carrée est caractérisée par le fait d'avoir le même le nombre de lignes et le nombre de colonnes ($n=m$)

Exemple: est une matrice carrée 4×4 ($n=m=4$)

	1	1	1	1
A	2	2	2	2
	3	3	3	3
	4	4	4	4

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Addition A+B de deux matrices de mêmes dimensions : le résultat est une matrice C de même dimension.

Exemple:

$$\begin{array}{c} \mathbf{A} \end{array} \begin{array}{ccc} -1 & 6 & 6 \\ 2 & 9 & 0 \end{array} + \begin{array}{c} \mathbf{B} \end{array} \begin{array}{ccc} 5 & 9 & 2 \\ 0 & 8 & 1 \end{array} = \begin{array}{ccc} 4 & 15 & 8 \\ 2 & 17 & 1 \end{array}$$

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Soustraction A-B de deux matrices de mêmes dimensions : le résultat de la soustraction donne une matrice C de même dimension

Exemple:

$$\begin{array}{r} \mathbf{A} \quad -5 \ 8 \ 6 \\ \quad \quad 3 \ 9 \ 0 \end{array} - \begin{array}{r} \mathbf{B} \quad 8 \ 3 \ 2 \\ \quad \quad 5 \ 4 \ 1 \end{array} = \begin{array}{r} -13 \ 5 \ 4 \\ \quad -2 \ 5 \ -1 \end{array}$$

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

La multiplication de deux Matrices $A(N_A \times M_A) * B(N_A \times M_A)$ est réalisable seulement si le nombre de colonnes de la matrice A est égale au nombre de lignes de la matrice B c'est-à-dire : $M_A = N_B$.
Le résultat de la multiplication est une matrice de dimension $N_A \times M_B$.

Exemple:

$$\begin{array}{ccc} \begin{array}{ccc} 4 & 0 & 1 \\ 2 & 5 & 3 \end{array} & * & \begin{array}{cccc} 4 & 2 & 9 & 2 \\ 7 & 8 & 0 & 1 \\ 3 & 0 & 1 & 1 \end{array} \\ & & = \end{array}$$

19	8	37	9	$4*4+0*7+1*3=16+3=$	19
52	44	21	12	$4*2+0*8+1*0=$	8
				$4*9+0*0+1*1=36+1=$	37
				$4*2+0*1+1*1=8+1=$	9

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Multiplication d'une matrice par un scalaire se fait en multipliant tous les éléments de la matrice par le scalaire en question

Exemple:

$$3 \quad * \quad \begin{matrix} 4 & 2 & 9 & 2 \\ 7 & 8 & 0 & 1 \\ 3 & 0 & 1 & 1 \end{matrix} \quad = \quad \begin{matrix} 12 & 6 & 27 & 6 \\ 21 & 24 & 0 & 3 \\ 9 & 0 & 3 & 3 \end{matrix}$$

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Transposition : il s'agit d'une matrice notée A^T de dimensions (MxN) obtenue en échangeant les lignes et les colonnes d'une matrice initiale A de dimensions (NxM)

Exemple:

	4	2	9	2		4	7	3
A	7	8	0	1	A^t	2	8	0
	3	0	1	1		9	0	1
						2	1	1

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

La Diagonale d'une matrice est l'ensemble des éléments d'une matrice carrée qui se situent du coins haut gauche vers le coin bas droit.

Exemple:

	4	2	9	2
A	7	8	0	1
	3	0	1	1
	1	1	1	1

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

Les Matrices :

Matrice identité : il s'agit d'une matrice diagonale égale à 1

$$A_{IJ} = \{A_{IJ} = 1 \text{ / } I=J, A_{IJ} = 0 \text{ / } I \neq J\}$$

Exemple:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux matrices

Mat=[v_{11} v_{12} ... v_{1m} ;
 v_{21} v_{22} ... v_{2m} ;
 ;
 v_{n1} v_{n2} ... v_{nm}]

Créer une Matrice Mat de dimension (n x m) avec les éléments v_{ij} .
Les éléments d'une ligne sont séparés par des espace ou virgules, les différentes lignes sont délimitées par des points-virgules ou par la touche enter.

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux matrices

Exemple

Cette matrice peut être écrite en MATLAB avec une des syntaxes suivantes:

```
>>A = [1,2,3,4 ; 5,6,7,8 ; 9,10,11,12] ;  
>>A = [1 2 3 4 ; 5 6 7 8 ; 9 10 11 12] ;  
>>A = [1,2,3,4  
        5,6,7,8  
        9,10,11,12] ;  
>> A=[[1;5;9] , [2;6;10] , [3;7;11] , [4;8;12]] ;
```

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux matrices

$\text{Mat}=[V_{\text{col1}} \ V_{\text{col2}} \ \dots]$

Il est aussi possible de définir une matrice Mat par concaténation de vecteurs colonnes V_{coli} ou de vecteurs ligne V_{li} .
Les séparateurs entre les vecteurs colonnes est l'espace,

chapitre 01: Maîtrise du logiciel matlab

La programmation sous Matlab

quelques fonctions liées aux matrices

$\text{Mat}=[V_{li1} ; V_{li2} ; \dots]$

Les séparateurs entre les vecteurs lignes est le (;)
points-virgules