



جامعة المص<u>ستقبل</u> AL MUSTAQBAL UNIVERSITY كليصصة العطصوم



1<sup>st</sup> class 2024- 2025

# Mathematics

Practical **MATLAB** 

# Lecture 1

# Asst. Lect. Mohammed Jabbar

Mohammed.Jabbar.Obaid@uomus.edu.iq

الرياضيات الاساسية: المرحلة الاولى مادة الرياضيات ماتلاب المحاضرة الاولى استاذ المادة: م.م محمد جبار Cybersecurity Department قسم الأمن السيبرانى

# Contents

Intro	oductio	n to MATLAB for Numerical Computations	1
1.1	Matlab	Programming	1
1.2	Unders	standing the MATLAB Environment	1
	1.2.1	MATLAB Desktop Components	2
Basi	c Synta	X	4
2.1	Hands	on Practice	4
2.2	Variab	les	6
	2.2.1	Creating Variables	7
	2.2.2	Variable Types	8
	2.2.3	Clearing Variables	8
	2.2.4	Checking Variable Information	8
	<ul> <li>Intro</li> <li>1.1</li> <li>1.2</li> <li>Basi</li> <li>2.1</li> <li>2.2</li> </ul>	Introduction         1.1       Matlab         1.2       Underse         1.2.1       1.2.1         Basic Synta       Synta         2.1       Hands         2.2       2.2.1         2.2.2       2.2.2         2.2.3       2.2.4	Introduction to MATLAB for Numerical Computations         1.1       Matlab Programming

# **1** Introduction to MATLAB for Numerical Computations

MATLAB (short for MATrix LABoratory) is a high-level programming environment widely used for numerical computations, data analysis, algorithm development, and visualization. It is especially powerful for working with matrices and arrays, making it ideal for scientific and engineering applications.

# 1.1 Matlab Programming

A computer program is a sequence of instructions in a given language that achieves a specific task.

MATLAB programming is centered around matrix and array manipulations, offering an extensive collection of built-in functions and toolboxes for numerical computation, algorithm development, data analysis, and visualization.

# **1.2 Understanding the MATLAB Environment**

MATLAB development IDE can be launched from the icon created on the desktop. Understanding the MATLAB environment is essential for efficiently using its features. Below are the key components of the MATLAB environment.

🔺 MATLAB R2023b — 🗗 🔿 🗙						
HOME PLOTS APPS		🔚 🍐 🖄 🛱 🔁 ൙ 🚍 🕐 👁 Search Documentation 🛛 🔎 🐥 Sign In				
Image: New New New New New Script Live Script         New Open Image: Compare New Determine New New New New New New New New New Ne	Wariable         Wariable           Clean         Save Workspace           Data         Clear Workspace           VARIALE         Cobe	Image: Simulink         Image: Preferences         Image: Simulink         Image:				
🗇 🕪 💽 🛜 📁 🕨 C: 🕨 Users 🕨		م •				
Current Folder 💿	Command Window	$\odot$				
Name -	>> x=1					
Public						
🖽 🚾 user	x =					
	1					
	>> y=2					
	у -					
	2					
	-					
	$f_{x} >> y=2$					
Dataile						
Workspace	1					
Name  Value						
x 1						
y z						
nn.l	]					

Figure 1: The Main Working Window in Matlab

# 1.2.1 MATLAB Desktop Components

📣 MATLAB R2	023b													-	0	×
HOME	PL	DTS	APPS								% <b>h h</b> 4	) ¢	🔄 🕐 💿 Search Doc	umentation	P 💄	Sign In
3. 📑			Car ter	1		🔜 Variable 🔻	>>	🚽 Analyze Code			OPreferences	6	PANELS			
New New	r u v New	Open	Find Files	Import	Clean	📲 Save Workspace	Favorites	😽 Run and Time	Simulink	Lavout	🚰 Set Path	Add-	Command Window	Ctrl+0		
Script Live So	ript 🔻	•	Compare	Data	Data	🍃 Clear Workspace 🔻	-	🍃 Clear Commands 🔻	Sintainte	-	Parallel 🔻	100	Command History	Ctrl+1		
4	FIL					VARIABLE		CODE	SIMULINK		ENVIRONMENT		Current Folder	Ctrl+2		<b>A</b>
Current Folder	► C:	<ul> <li>Users</li> </ul>	•		Com	mand Window							Workspace	Ctrl+3		• •
Name 4				0	>	> x=1							Next Visible Panel	Ctrl+Tab		
🗄 📜 Public													Previous Visible Panel	Ctrl+Shift+Tab		
🗄 🦲 user					x	=							Next Tabbed Panel	Ctrl+Page Down		
						1							Previous Tabbed Panel	Ctrl+Page Up		
						> v=2										
						2										
					У	-										
						2										
					fr	×										
						> y-2										
Details				^												
Workspace				۲	1											
Name ^	Value															
<sup>x</sup> y	2															

Figure 2: MATLAB Desktop Components: Switch Window

### 1. Command Window:

- The Command Window is where you can type and execute MATLAB commands interactively. For example, entering a calculation like 3+4 will immediately show the result.
- You can also run scripts and functions from the Command Window by typing their names.

С	iommand Window	$\odot$
	>> x=1	
	x =	
	1	
	>> y=2	
	Y =	
	2	
f>	<i>x</i> >>	
_		

Figure 3: Command Window

## 2. Workspace:

- The Workspace shows all the variables currently in memory. You can view their values and details like size and type.
- You can double-click a variable in the Workspace to open the Variable Editor and inspect or modify its contents.

Workspace		
Name 🔺	Value	
🛨 x	1	
🛨 y	2	

Figure 4: Workspace

3. **Command History:** This window stores all the commands you have previously entered in the Command Window. You can easily re-execute commands from here by double-clicking them.

Command History	⊙×
3x prime_number	
Prime = primes(512*512)	
- prime_number	
sqrt(23000)	
Prime = primes(300*300)	
sqrt(8713)	
Prime = primes(100001)	
sqrt(9592)	
Prime = primes(200000)	
sqrt(17984)	
Prime = primes(300000)	
sqrt(25997)	
Prime = primes(400000)	
sqrt(33860)	
Prime = primes(500000)	
sqrt(14538)	

Figure 5: Command History

4. **Current Folder:** The Current Folder window displays the files and folders in your working directory. This is the default location where MATLAB will save or look for scripts, functions, and data files.

Current Folder	6
Name 🔺	
🗄 📒 Public	
🗄 📒 user	

Figure 6: Current Folder

# 2 Basic Syntax

MATLAB has a simple and intuitive syntax designed for numerical computing and matrix manipulations. Here's an overview of the basic syntax rules and commands to get started with MATLAB.

# 2.1 Hands on Practice

Type a valid expression, for example,

5 + 5

%And press ENTER. When you click the Execute button, or type Ctrl+E, MATLAB ... executes it immediately and the result returned is:

ans =

10

#### Let us take up few more examples:

 $3^2 \%3$  raised to the power of 2

%Press ENTER.

ans =

9

### Another example,

 $\sin(pi/2)$  % sine of angle 90 ( $\pi/2$ )

%Press ENTER.

ans =

1

#### Another example,

7/0 %Divide by zero

%Press ENTER.

ans =

Inf %Warning: division by zero

## Another example,

732\*20.3

%Press ENTER.

ans =

1.4860e + 04

MATLAB provides some special expressions for some mathematical symbols, like  $\pi, \infty, \sqrt{a}$  etc. Nan stands for 'not a number'.

Function	Description	Syntax
sin(x)	Sine of x	$\sin(x)$
$\cos(x)$	Cosine of x	$\cos(x)$
tan(x)	Tangent of x	$\tan(x)$
asin(x)	Inverse sine (arcsin) of x	$\sin^{-1}(x)$
acos(x)	Inverse cosine (arccos) of x	$\cos^{-1}(x)$
atan(x)	Inverse tangent (arctan) of x	$\tan^{-1}(x)$
exp(x)	Exponential function of x	$e^x$
log(x)	Natural logarithm of x	$\log(x)$
log10(x)	Base-10 logarithm of x	$\log_{10}(x)$
sqrt(x)	Square root of x	$\sqrt{x}$
abs(x)	Absolute value of x	x
floor(x)	Round towards negative infinity	floor(x)
ceil(x)	Round towards positive infinity	ceil(x)
round(x)	Round to nearest integer	round(x)
sinh(x)	Hyperbolic sine of x	$\sinh(x)$
$\cosh(x)$	Hyperbolic cosine of x	$\cosh(x)$
tanh(x)	Hyperbolic tangent of x	tanh(x)
asinh(x)	Inverse hyperbolic sine of x	$sinh^{-1}(x)$
acosh(x)	Inverse hyperbolic cosine of x	$\cosh^{-1}(x)$
atanh(x)	Inverse hyperbolic tangent of x	$\tanh^{-1}(x)$

Table 1: Table of Special Functions in MATLAB

Operator	Description	Example
+	Addition	a+b
-	Subtraction	a-b
*	Multiplication	a * b
/	Right Division	$\frac{a}{b}$
.+	Element-wise Addition	A. + B
	Element-wise Subtraction	A B
.*	Element-wise Multiplication	A. * B
./	Element-wise Right Division	A./B

Table 2: Table of Arithmetic Operators in MATLAB

# 2.2 Variables

In MATLAB environment, every variable is an array or matrix. You can assign variables in a simple way.

Variables are used to store data for manipulation and analysis. Here's a brief overview of how to create and use variables in MATLAB:

#### 2.2.1 Creating Variables

You can create variables by simply assigning a value to a name. Variable names must start with a letter, followed by letters, digits, or underscores.

```
x = 5 %Numeric variable
%Press ENTER
x =
       5
y = sin(67) %Numeric variable
%Press ENTER
y =
       -0.8555
name = 'MATLAB' \% String variable
%Press ENTER
name =
       'MATLAB'
A = [1, 2, 3; 4, 5, 6] % Matrix variable
%Press ENTER
A =
       1 \ 2 \ 3
       4 5 6
% Accessing Variables: You can access the value of a variable by simply typing its ...
name.
у
%Press ENTER
y =
```

-0.8555

### 2.2.2 Variable Types

MATLAB supports various data types, including:

- Numeric: Scalars, vectors, and matrices.
- Strings: Character arrays and string arrays.
- Logical: Boolean values (true/false).
- Cells: Cell arrays for heterogeneous data.
- Structures: Data structures for complex data types.

## 2.2.3 Clearing Variables

You can remove variables from the workspace using the clear command.

clear x %Press ENTER %Removes variable x clear all %Press ENTER %Clears all variables

**Remark.** The (clc) command in MATLAB is used to clear the Command Window. It does not delete variables from memory, but it removes all the text and results from the Command Window, giving you a fresh, clean workspace for the next set of outputs.

### 2.2.4 Checking Variable Information

Use the (whos) command to see the variables in the workspace along with their sizes and types.

whos %See all variables

Remark. In MATLAB, the semicolon (;) has two main purposes:

 Suppress Output: When you use a semicolon at the end of a statement or command, MATLAB will execute the command but suppress the output in the Command Window. For example:

x = 5; %The value is assigned, but no output is displayed. %Without the semicolon:

x =

5

2. Separate Commands on the Same Line: You can use a semicolon to separate multiple statements or commands on the same line. For example:

a = 3; b = 7; c = a + b;

 $\%\,\mathrm{This}$  executes all three commands in sequence, a, b, and c.