Lecture1

Introduction to MATLAB

MATLAB is a powerful language for technical computing. The name MATLAB stands for MATrix LABoratory, because its basic data element is a matrix (array). MATLAB can be used for math computations, modeling and simulations, data analysis and processing, visualization and graphics, and algorithm development. This chapter had been introduced some principles in the MATLAB.

1.1 *Introduction to MATLAB*

It is assumed that the software is installed on the computer, and that the user can start the program. Once the program starts, the MATLAB desktop window opens (Figure 1-1). The window contains four smaller windows: the Command Window, the Current Folder Window, the Workspace Window, and the Command History Window. This is the default view that shows four of the various windows of MATLAB. A list of several windows and their purpose is given in Table 1-1. The Start button on the lower left side can be used to access MATLAB tools and features.

1. Command Window: The Command Window is MATLAB's main window and opens when MATLAB is started. It is convenient to have the Command Window as the only visible window, and this can be done by either closing all the other windows (click on the x at the top right-hand side of the window you want to close) or by first selecting the Desktop Layout in the Desktop menu, and then selecting Command Window Only from the submenu that opens.

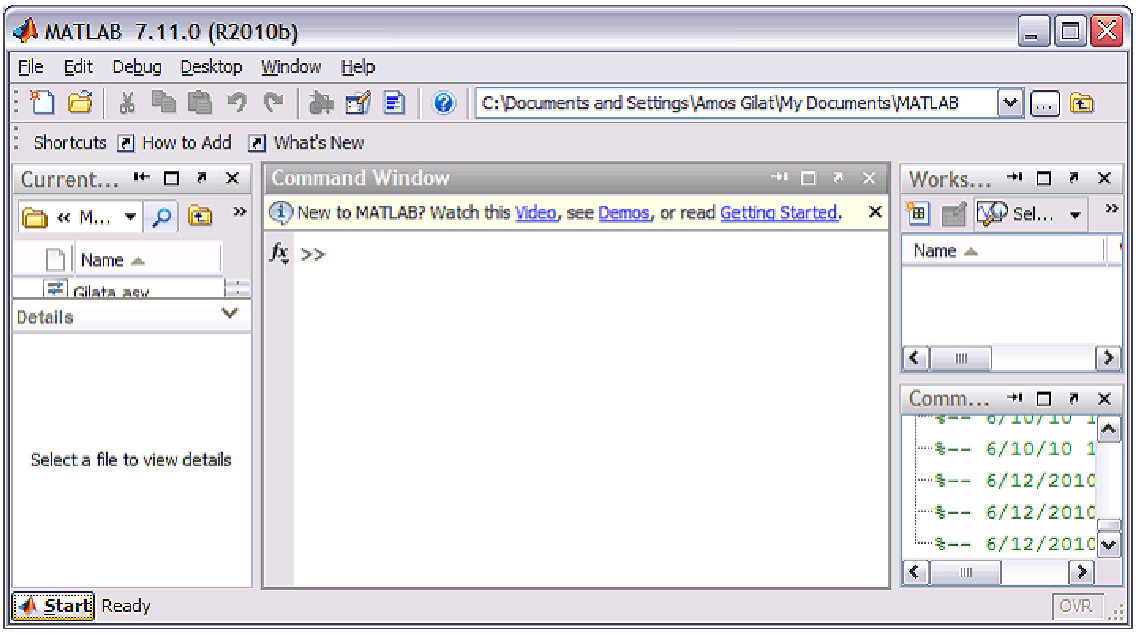


Figure 1-1: The default view of MATLAB desktop.

Table 1-1: MATLAB windows

Window	Purpose
Command Window	Main window, enters variables, runs programs.
Figure Window	Contains output from graphic commands.
Editor Window	Creates and debugs script and function files.
Help Window	Provides help information.
Command History Window	Logs commands entered in the Command Window.
Workspace Window	Provides information about the variables that are used.
Current Folder Window	Shows the files in the current folder.

2. Editor Window: The Editor Window is used for writing and editing programs. This window is opened from the File menu. An example of an Editor Window is shown in Figure 1-2.

3. Help Window: The Help Window contains help information. This window can be opened from the Help menu in the toolbar of any MATLAB window. The Help Window is interactive and can be used to obtain information on any feature of MATLAB. Figure 1-3 shows an open Help Window.

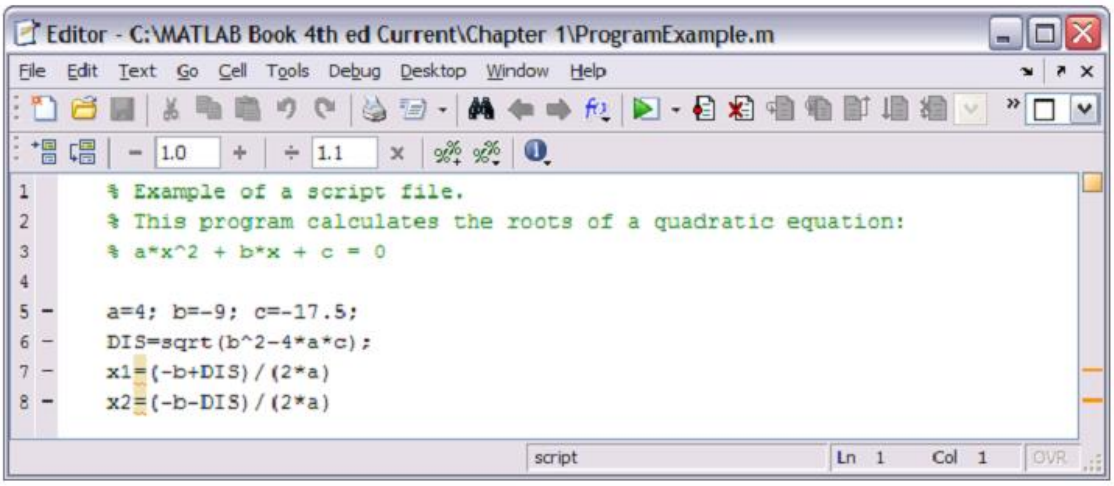


Figure 1-2: Example of an Editor Window.

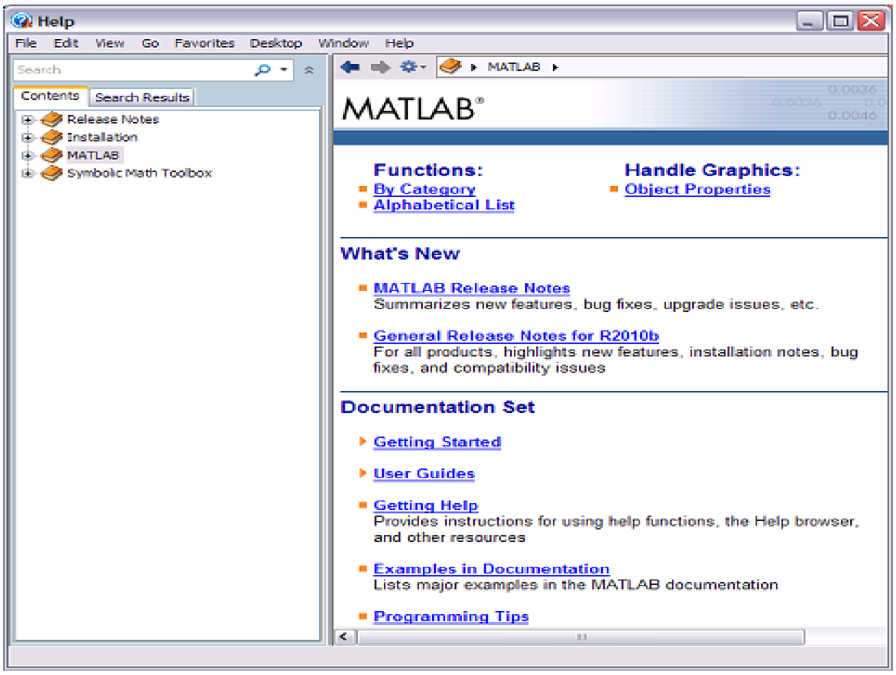


Figure 1-3: The Help Window.

1.2. *Arithmetic Operations with Scalars*

Numbers can be used in arithmetic calculations directly (as with a calculator) or they can be assigned to variables, which can subsequently be used in calculations. The symbols of arithmetic operations are:

Operation	Symbol	Example
Addition	+	5 + 3
Subtraction	–	5 – 3
Multiplication	*	5 * 3
Right division	/	5 / 3
Left division	\	5 \ 3 = 3 / 5
Exponentiation	^	5 ^ 3 (means 5^3 = 125)

MATLAB executes the calculations according to the order of precedence displayed below. This order is the same as used in most calculators.

Precedence	Mathematical Operation
First	Parentheses. For nested parentheses, the innermost are executed first.
Second	Exponentiation.
Third	Multiplication, division (equal precedence).
Fourth	Addition and subtraction.

The simplest way to use MATLAB is as a calculator. This is done in the Command Window by typing a mathematical expression and pressing the Enter key. MATLAB calculates the expression and responds by displaying *ans* = and the numerical result of the expression in the next line this is demonstrated in Tutorial 1-1.

Tutorial 1-1: Using MATLAB as a calculator.

```
>> 7+8/2
ans =
    11
>> (7+8)/2
ans =
    7.5000
>> 4+5/3+2
ans =
    7.6667
>> 5^3/2
ans =
   62.5000
>> 27^(1/3)+32^0.2
ans =
     5
>> 27^1/3+32^0.2
ans =
    11

>> 0.7854-(0.7854)^3/(1*2*3)+0.785^5/(1*2*3*4*5)...
-(0.785)^7/(1*2*3*4*5*6*7)

ans =
    0.7071
>>
```

- >> 7+8/2 ← Type and press **Enter**. 8/2 is executed first.
- >> (7+8)/2 ← Type and press **Enter**. 7+8 is executed first.
- 4+5/3+2: 5/3 is executed first.
- 5^3/2: 5^3 is executed first, /2 is executed next.
- 27^(1/3)+32^0.2: 1/3 is executed first, 27^(1/3) and 32^0.2 are executed next, and + is executed last.
- 27^1/3+32^0.2: 27^1 and 32^0.2 are executed first, /3 is executed next, and + is executed last.
- Type three periods ... (and press **Enter**) to continue the expression on the next line.
- The last expression is the first four terms of the Taylor series for sin(π/4).

1.3. Display Formats

The user can control the format in which MATLAB displays output on the screen. In Tutorial 1-1, the output format is fixed-point with four decimal digits (called short), which is the default format for numerical values. The format can be changed with the format command. Once the format command is entered, all the output that follows is displayed in the specified format. Several of the available formats are listed and described in Table 1-2. MATLAB has several other formats for displaying numbers. Details of these formats can be obtained by typing help format in the Command Window. The format in which numbers are displayed does not affect how MATLAB computes and saves numbers.

Table 1-2: Display formats

Command	Description	Example

`format short`	Fixed-point with 4 decimal digits for: $0.001 \le number \le 1000$ Otherwise display format `short e`.	`>> 290/7`
		`ans =`
		`41.4286`

`format long`	Fixed-point with 15 decimal digits for: $0.001 \le number \le 100$ Otherwise display format `long e`.	`>> 290/7`
		`ans =`
		`41.428571428571431`

`format short e`	Scientific notation with 4 decimal digits.	`>> 290/7`
		`ans =`
		`4.1429e+001`

`format long e`	Scientific notation with 15 decimal digits.	`>> 290/7`
		`ans =`
		`4.142857142857143e+001`

`format short g`	Best of 5-digit fixed or floating point.	`>> 290/7`
		`ans =`
		`41.429`

`format long g`	Best of 15-digit fixed or floating point.	`>> 290/7`
		`ans =`
		`41.4285714285714`

`format bank`	Two decimal digits.	`>> 290/7`
		`ans =`
		`41.43`

1.4. Elementary Math Built-in Functions

In addition to basic arithmetic operations, expressions in MATLAB can include functions. MATLAB has a very large library of built-in functions. A function has a name and an argument in parentheses. Some commonly used elementary MATLAB mathematical built-in functions are given in Tables 1-3,1.4 and 1-5. A complete list of functions organized by category can be found in the Help Window.

Table 1-3: Elementary math functions

Function	Description	Example

`sqrt(x)`	Square root.	`>> sqrt(81)`
		`ans =`
		`9`

`nthroot(x,n)`	Real *n*th root of a real number *x*. (If *x* is negative *n* must be an odd integer.)	`>> nthroot(80,5)`
		`ans =`
		`2.4022`

`exp(x)`	Exponential (e^x).	`>> exp(5)`
		`ans =`
		`148.4132`

`abs(x)`	Absolute value.	`>> abs(-24)`
		`ans =`
		`24`

`log(x)`	Natural logarithm. Base *e* logarithm (ln).	`>> log(1000)`
		`ans =`
		`6.9078`

`log10(x)`	Base 10 logarithm.	`>> log10(1000)`
		`ans =`
		`3.0000`

`factorial(x)`	The factorial function *x*! (*x* must be a positive integer.)	`>> factorial(5)`
		`ans =`
		`120`

The inverse trigonometric functions are asin(x), acos(x), atan(x), acot(x) for the angle in radians; and asind(x), acosd(x), atand(x), acotd(x) for the angle in degrees. The hyperbolic trigonometric functions are sinh(x), cosh(x), tanh(x), and coth(x). Table 1-4 uses pi, which is equal to π.

Table 1-4: Trigonometric math functions

Function	Description	Example

`sin(x)`
`sind(x)`
	Sine of angle *x* (*x* in radians).
	Sine of angle *x* (*x* in degrees).
		`>> sin(pi/6)`
		`ans =`
		`0.5000`

`cos(x)`
`cosd(x)`
	Cosine of angle *x* (*x* in radians).
	Cosine of angle *x* (*x* in degrees).
		`>> cosd(30)`
		`ans =`
		`0.8660`

`tan(x)`
`tand(x)`
	Tangent of angle *x* (*x* in radians).
	Tangent of angle *x* (*x* in degrees).
		`>> tan(pi/6)`
		`ans =`
		`0.5774`

`cot(x)`
`cotd(x)`
	Cotangent of angle *x* (*x* in radians).
	Cotangent of angle *x* (*x* in degrees).
		`>> cotd(30)`
		`ans =`
		`1.7321`

Table 1-5: Rounding functions

Function	Description	Example

`round(x)`	Round to the nearest integer.	`>> round(17/5)`
		`ans =`
		`3`

`fix(x)`	Round toward zero.	`>> fix(13/5)`
		`ans =`
		`2`

`ceil(x)`	Round toward infinity.	`>> ceil(11/5)`
		`ans =`
		`3`

`floor(x)`	Round toward minus infinity.	`>> floor(-9/4)`
		`ans =`
		`-3`

`rem(x,y)`	Returns the remainder after *x* is divided by *y*.	`>> rem(13,5)`
		`ans =`
		`3`

`sign(x)`	Signum function. Returns 1 if $x > 0$, -1 if $x < 0$, and 0 if $x = 0$.	`>> sign(5)`
		`ans =`
		`1`

Lecture 2

1.5 Defining Scalar Variables

In order to store a value in a MATLAB session, or in a program, a variable is used. The Workspace Window shows variables that have been created. On easy way to create a variable is to use an assignment statement. The format of an assignment statement is:

Variable_name = A numerical value, or a computable expression

The left-hand side of the assignment operator can include only one variable name. The right-hand side can be a number, or a computable expression that can include numbers and/or variables that were previously assigned numerical values. When the Enter key is pressed the numerical value of the right-hand side is assigned to the variable, and MATLAB displays the variable and its assigned value in the next two lines. The following shows how the assignment operator works:

```
>> x=15
x =
    15

>> x=3*x-12
x =
    33
>>
```

The number 15 is assigned to the variable x.

MATLAB displays the variable and its assigned value.

A new value is assigned to x. The new value is 3 times the previous value of x minus 12.

A variable can be named according to the following rules:

•Must begin with a letter .
•Can be up to 63 characters long.
•Can contain letters, digits, and the underscore character.
•Cannot contain punctuation characters (e.g., Period , comma, semicolon).
•MATLAB is case sensitive: it distinguishes between uppercase and lowercase letters. For example, AA, Aa, aA, and aa are the names of four different variables.
•No spaces are allowed between characters (use the underscore where a space is Desired).
•Avoid using the name of a built-in function for a variable (i.e., avoid using cos, sin, exp, sqrt, etc.). Once a function name is used to define a variable, the function cannot be used.

Examples

1. A trigonometric identity is given by:

$$\cos^2\frac{x}{2} = \frac{\tan x + \sin x}{2\tan x}$$

Verify that the identity is correct by calculating each side of the equation, substituting $x = \frac{\pi}{5}$.

Solution:

```
%Example 1
x=pi/5;
LHS=cos(x/2)^2
RHS=(tan(x)+sin(x))/(2*tan(x))
```

Answer:

LHS =

0.9045

RHS =

0.9045

2. Define the variable x=π/8 , find the *value of polynomial f(x)=sin(x)=x-x3/3!+x5/5!* , and compare with the exact value by calculate the error.

Solution:

```
%Example 2
%apfun is the approximated vale
%exc if the exact value
format long
x=pi/8;
apfun=x-x^3/factorial(3)+x^5/factorial(3)
exc=sin(x)
error=abs(apfun-exc)
```

Answer:

apfun =

0.384162389281559

exc =

0.382683432365090

error = 0.001478956916470

3. An electron, which has a mass of 9.1×10^{31} kg, moves with a speed of 0.750c. Find its relative momentum and compare this value with the momentum calculated from the classical expression.

Hint: relativistic momentum is given by $p = \dfrac{m_e u}{\sqrt{1 - \dfrac{u^2}{c^2}}}$. Where *u* and *c* , are the

Velocity of the object and the light respectively, and *me* is the mass of electron.

Answer:

```
%Example 3
%c is speed of light
%u is speed of object
%p relative momentum  ,pclass is the clasical momentum
% me is the mass
clear all
clc
c=3e08; ,u=0.75*c;, me=9.1e-31;
p=(me*u)/sqrt(1-u^2/c^2)
pclass=me*u
```

Answer:

p =

3.0955e-022

pclass =

2.0475e-022

PROBLEMS:

The following problems can be solved by writing commands in the Command Window, or by writing a program in a script file and then executing the file.

4. Given: $\int \cos^2(ax)dx = \frac{1}{2}x - \frac{\sin 2ax}{4a}$. Use MATLAB to calculate the following definite integral: $\int_{\frac{\pi}{9}}^{\frac{3\pi}{5}} \cos^2(0.5x)dx$.

Solution 4:

4. The code is:

```
%problem 8
%maxv is the maximum value of the integration at  x1=3*pi/5
%minv is the minimum value of the integration at  x2=pi/9
%intg is the integration value or area under the curve
a=0.5;
x1=3*pi/5;, x2=pi/9;
maxv=0.5*x1-sin(2*a*x1)/(4*a);
minv=0.5*x2-sin(2*a*x2)/(4*a);
intg=maxv-minv
```

Answer:

intg =

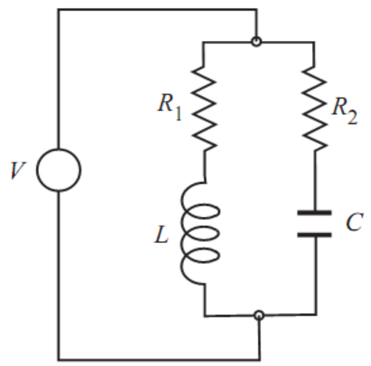
0.4634

5.

The resonant frequency f (in Hz) for the circuit shown is given by:

$$f = \frac{1}{2\pi}\sqrt{LC\frac{R_1^2C - L}{R_2^2C - L}}$$

Calculate the resonant frequency when $L = 0.2$ henrys, $R_1 = 1500$ ohms, $R_2 = 1500$ ohms, and $C = 2 \times 10^{-6}$ farads.



Solution 5:

5. **The code is:**

```
%problem 10
C=2e-06;,R1=1500;, L=0.2;,R2=1500;
f=(1/(2*pi))*sqrt(L*C*(R1^2*C-L)/(R2^2*C-L))
```

Answer:

f =

1.0066e-004