Numerical Analysis in Mechanical Engineering Course MEC 155

Introduction to MATLAB

Dr./ Ahmed Mohamed Nagib Elmekawy

Associate Professor, Mechanical Engineering department, Alexandria university, Egypt

Spring 2023

Chapter 1

An Overview of MATLAB

MATLAB stands for MATrix LABoratory

- MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming environment. Furthermore, MATLAB is a modern programming language environment: it has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. These factors make MATLAB an excellent tool for teaching and research.
- Matlab can build exe files that can be run independently
- MATLAB has many advantages compared to conventional computer languages (e.g.,C, FORTRAN) for solving technical problems. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning.
- The software package has been commercially available since 1984 and is now considered as a standard tool at most universities and industries worldwide.

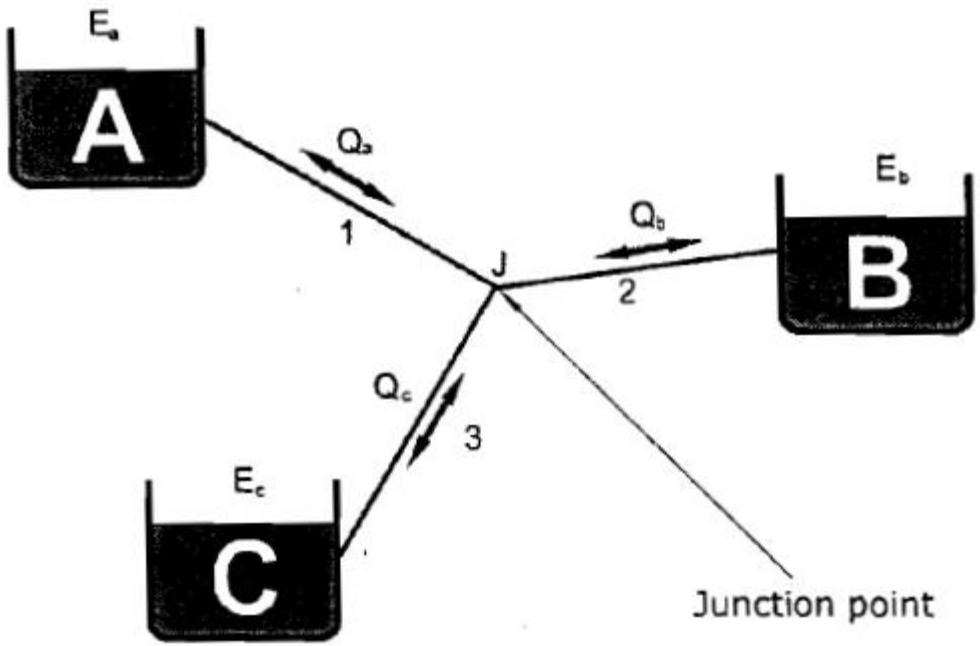
MATLAB stands for MATrix LABoratory

- It has powerful built-in routines that enable a very wide variety of computations. It also has easy to use graphics commands that make the visualization of results immediately available. Specific applications are collected in packages referred to as toolbox. There are toolboxes for signal processing, symbolic computation, control theory, simulation, optimization, and several other fields of applied science and engineering.
- In addition to the MATLAB documentation which is mostly available on-line
- Can be used to simulate the performance of mechanical systems such as Finite Element Methods and Computational Fluid Dynamics (CFD)

When will I use Matlab again during university?

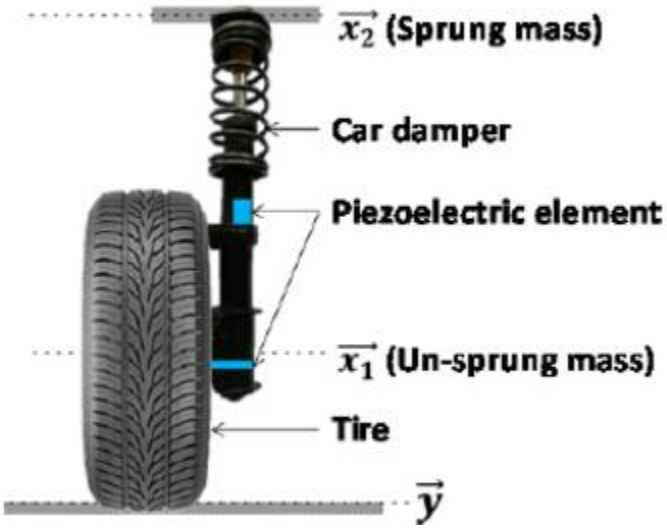
- Solve algebraic equations
- Solve try and error problems
- Integration and differentiation
- Optimization Course

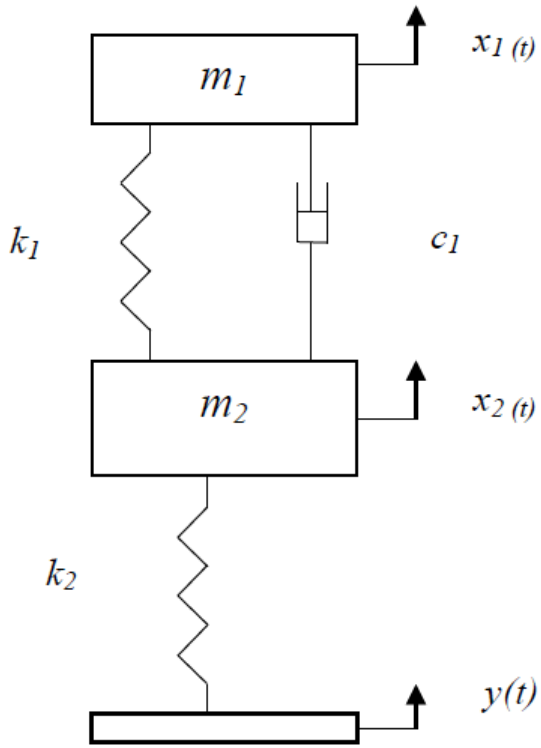
- Three Tank Problem



When will I use Matlab again during university?

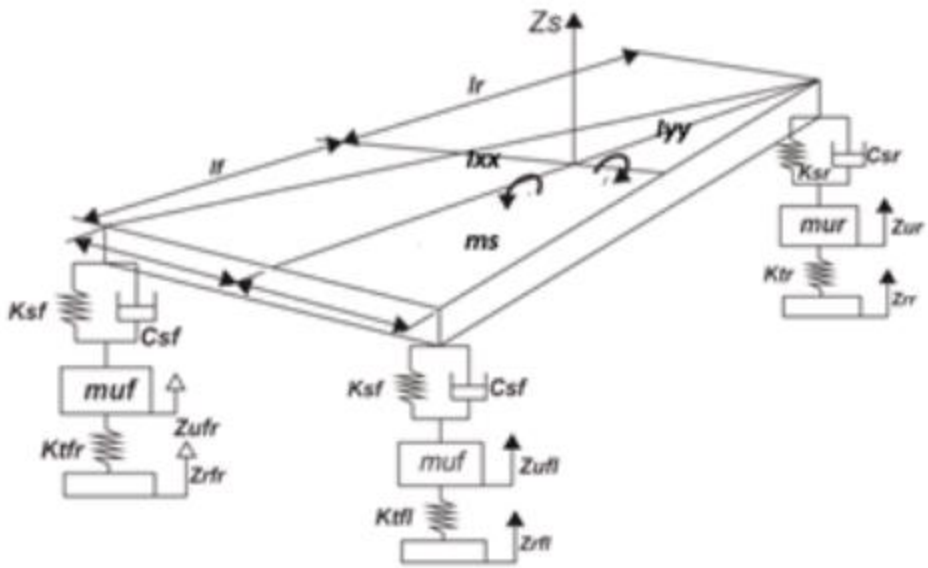
- Plot system performance in vibration course

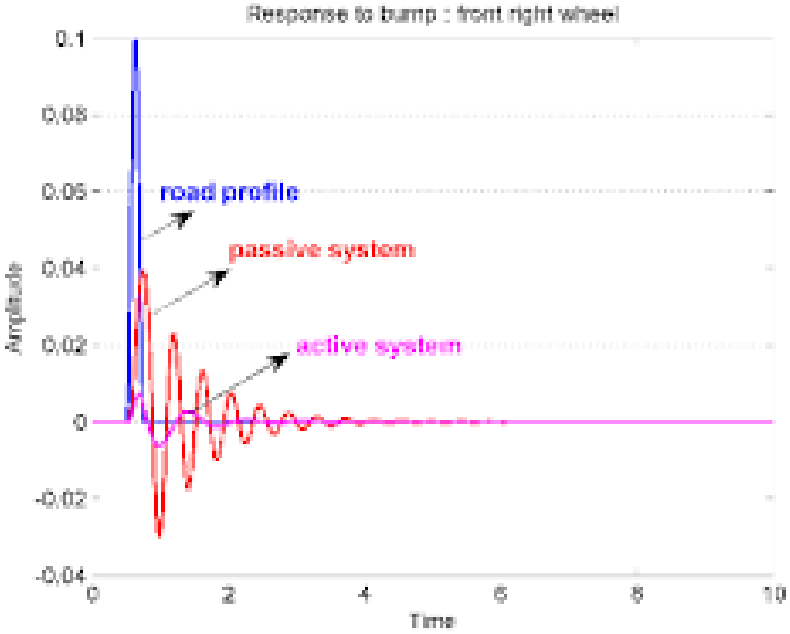




When will I use Matlab again during university?

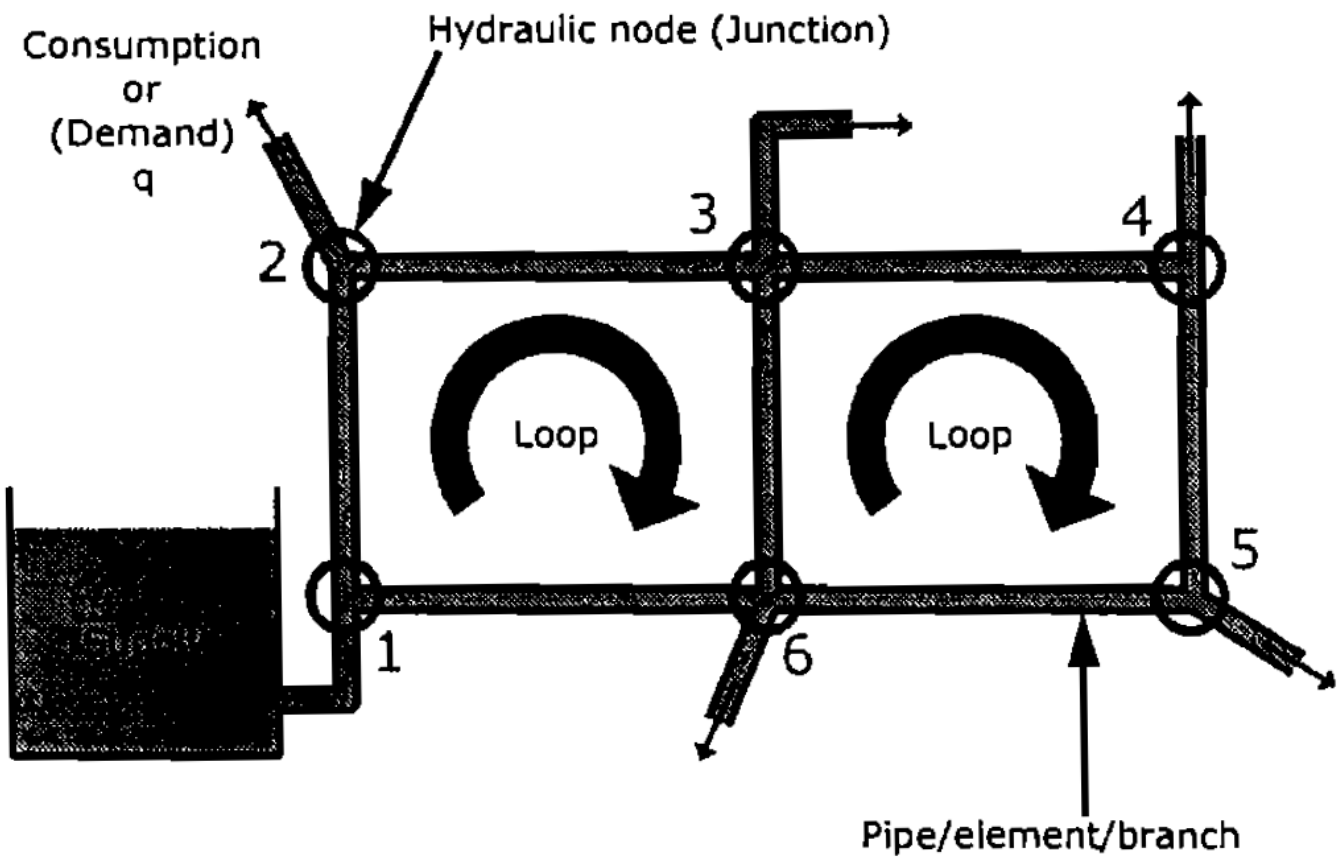
- Plot system performance in vibration course





When will I use Matlab again during university?

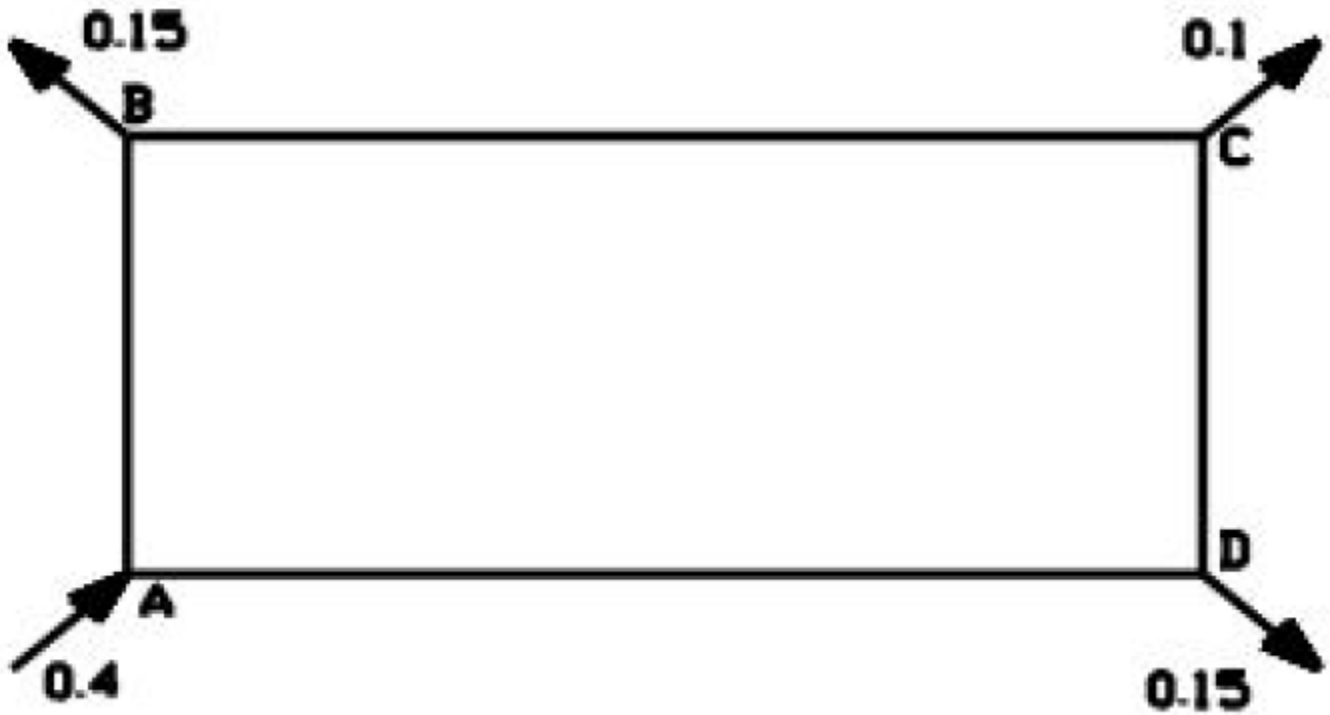
- Hydraulic Networks



When will I use Matlab again during university?

4. Water enters the four-sided ring main shown in Fig.3 at A at a rate of 0.4 m^3/s and is delivered at B, C, D at the rate of 0.15, 0.10, and 0.15 m^3/s respectively. All pipes are 0.6 m in diameter with a friction coefficient ' f ' of 0.0078 and 78their lengths are: AB = CD = 150 m, BC = 300 m, and DA = 240 m.

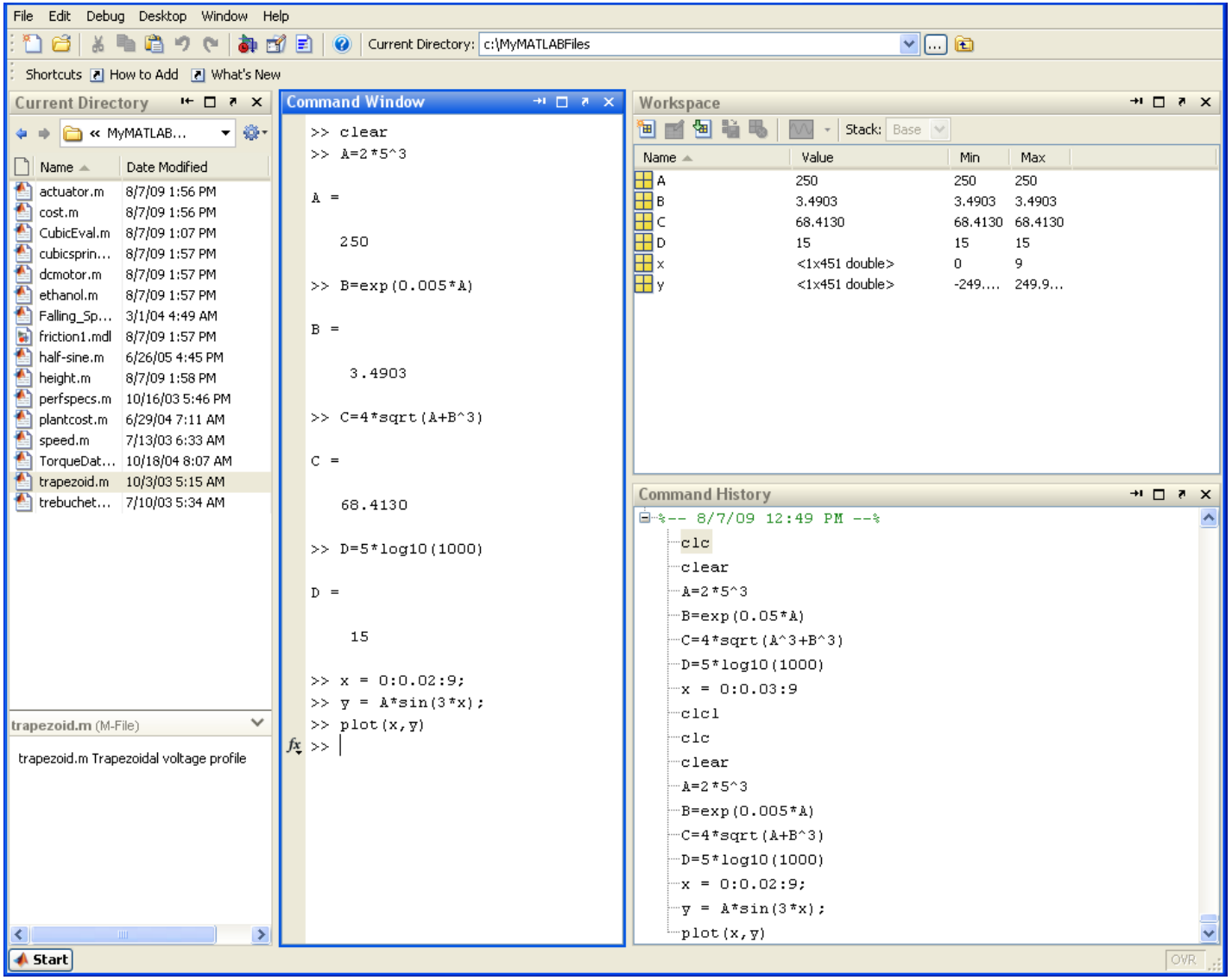
Determine the flow through each pipe and the pressures at B, C, and D if that at A is 105 kN/m^2.



When will I use Matlab again after graduation?

- Design of mechanical equipments such as heat exchanger
- Postgraduate course

The Default MATLAB Desktop:



Scalar arithmetic operations

Symbol	Operation	MATLAB form
^	exponentiation: a^b	a^b
*	multiplication: ab	a*b
/	right division: $a/b = \frac{a}{b}$	a/b
\	left division: $a\backslash b = \frac{b}{a}$	a\ b
+	addition: $a + b$	a+b
-	subtraction: $a - b$	a-b

Example 1.1

Type the following in command windows:

```
>> 8/10
ans =
     0.8000
>> 5*ans
ans =
     4
>> r=8/10
r =
   0.8000
>> r
r =
   0.8000
>> s=20*r
s =
   16
```

Variables

- You do not need to declare the variable names and sizes at the beginning of your program as Matlab declare them automatically.
- You should avoid using variable names that Matlab uses as a function or command such as:

plot	end
if	for
while	pi
sin	cos

- You can check to see if a command, function or file name already exists by using the `exist('name')` command.
- Lower and upper case variables are different.

Example 1.2

Type the following in command windows:

```
>> Y=5
Y =
      5
>> y=4
y =
      4
>> x=y+Y
x =
    9
```

Order of precedence

Precedence	Operation
First	Parentheses, evaluated starting with the innermost pair.
Second	Exponentiation, evaluated from left to right.
Third	Multiplication and division with equal precedence, evaluated from left to right.
Fourth	Addition and subtraction with equal precedence, evaluated from left to right.

Example 1.3 - Precedence
Type the following in command windows:

```
>> 8 + 3*5
ans =
     23
>> 8 + (3*5)
ans =
     23
>>(8 + 3)*5
ans =
     55
>>4^2 - 12 - 8/4*2
ans =
     0
>>4^2 - 12 - 8/(4*2)
ans =
     3
```

Commands for managing the work session

Command	Description
`clc`	Clears the Command window.
`clear`	Removes all variables from memory.
`clear var1 var2`	Removes the variables `var1` and `var2` from memory.
`exist('name')`	Determines if a file or variable exists having the name 'name'.
`quit`	Stops MATLAB.
`who`	Lists the variables currently in memory.
`whos`	Lists the current variables and sizes, and indicates if they have imaginary parts.
`:`	Colon; generates an array having regularly spaced elements.
`,`	Comma; separates elements of an array.
`;`	Semicolon; suppresses screen printing; also denotes a new row in an array.
`...`	Ellipsis; continues a line.

Special variables and constants

Command	Description
`ans`	Temporary variable containing the most recent answer.
`eps`	Specifies the accuracy of floating point precision.
`i,j`	The imaginary unit $\sqrt{-1}$.
`Inf`	Infinity.
`NaN`	Indicates an undefined numerical result.
`pi`	The number π.

Complex Number Operations

- The number $c_1 = 1 - 2i$ is entered as follows:
  `c1 = 1-2i`.
- An asterisk is not needed between i or j and a number, although it is required with a variable, such as `c2 = 5 - i*c1`.
- Be careful. The expressions
  `y = 7/2*i`

and

  `x = 7/2i`

give two different results:

$y = (7/2)i = 3.5i$

and

$x = 7/(2i) = -3.5i$.

Example 1.4
Type the following in command windows:

```
>> i= 5
i =
     5
>>s=3+7i
w =
     3.0000 + 7.0000i
>>w=3+7*i
w =
     38
```

Numeric display formats:

The `format` command controls how numbers appear on the screen. The table able gives the variants of this command. MATLAB uses many significant figures in its calculations, but we rarely need to see all of them. The default MATLAB display format is the short format, which uses four decimal digits. You can display more by typing format long, which gives 16 digits. To return to the default mode, type format short.

Command	Description and example
`format short`	Four decimal digits (the default); 13.6745.
`format long`	16 digits; 17.27484029463547.
`format short e`	Five digits (four decimals) plus exponent; 6.3792e+03.
`format long e`	16 digits (15 decimals) plus exponent; 6.379243784781294e−04.

Example 1.5

Type the following in command windows:

```
>> pi
ans =    3.1416
>> format long
>> pi
ans =   3.141592653589793
>> format short
>> pi
ans =    3.1416
```

Arrays

- The numbers 0, 1, 2, …, 10 can be assigned to the variable u by typing `u = 0:10`.
- To compute *w* = 5 sin *u* for *u* = 0, 1, 2, …, 10, the session is;

```
>>u = 0:10;
>>w = 5*sin(u);
>>plot(u,w)
```

- The single line, `w = 5*sin(u)`, computed the formula *w* = 5 sin *u* 11 times.

Arrays

- The numbers 0, 0.1, 0.2, …, 10 can be assigned to the variable u by typing `u = 0:0.1:10`.
- To compute *w* = 5 sin *u* for *u* = 0, 0.1, 0.2, …, 10, the session is;

```
>>q = 0:0.1:10;
>>r = 5*sin(u);
>>plot(u,w)
```

- The single line, `w = 5*sin(u)`, computed the formula *w* = 5 sin *u* 101 times.

Array Index

```
>>u(7)
ans =
      0.6000
```

← Gets the 7th element of u

```
>>w(7)
ans =
      2.8232
```

← Gets the 7th element of w

- Use the `length` function to determine how many values are in an array.

```
>>m = length(w)
m =
   101
```

Polynomial Roots

To find the roots of $x^2 - 5x + 6 = 0$, the session is

```
>>a = [1,-5,6];
>>roots(a)
ans =
     3.0000
     2.0000
```

The roots are $x = 3$ and $x = 2$.

Polynomial Roots

To find the roots of $x^3 - 7x^2 + 40x - 34 = 0$, the session is

```
>>a = [1,-7,40,-34];
>>roots(a)
ans =
     3.0000 + 5.000i
     3.0000 - 5.000i
     1.0000
```

The roots are $x = 1$ and $x = 3 \pm 5i$.

Some commonly used mathematical functions

Function	MATLAB syntax[1]
e^x	`exp(x)`
$\sqrt{x}$	`sqrt(x)`
$\ln x$	`log(x)`
$\log_{10} x$	`log10(x)`
$\cos x$	`cos(x)`
$\sin x$	`sin(x)`
$\tan x$	`tan(x)`
$\cos^{-1} x$	`acos(x)`
$\sin^{-1} x$	`asin(x)`
$\tan^{-1} x$	`atan(x)`

[1] The MATLAB trigonometric functions use radian measure.

Example 1.6
Type the following in command windows:

```
>> x=sin(0)
x =     0
>> x=sin(90)
x =    0.8940
>> x=sind(0)
x =     0
>> x=sind(90)
x =     1
```

M-files

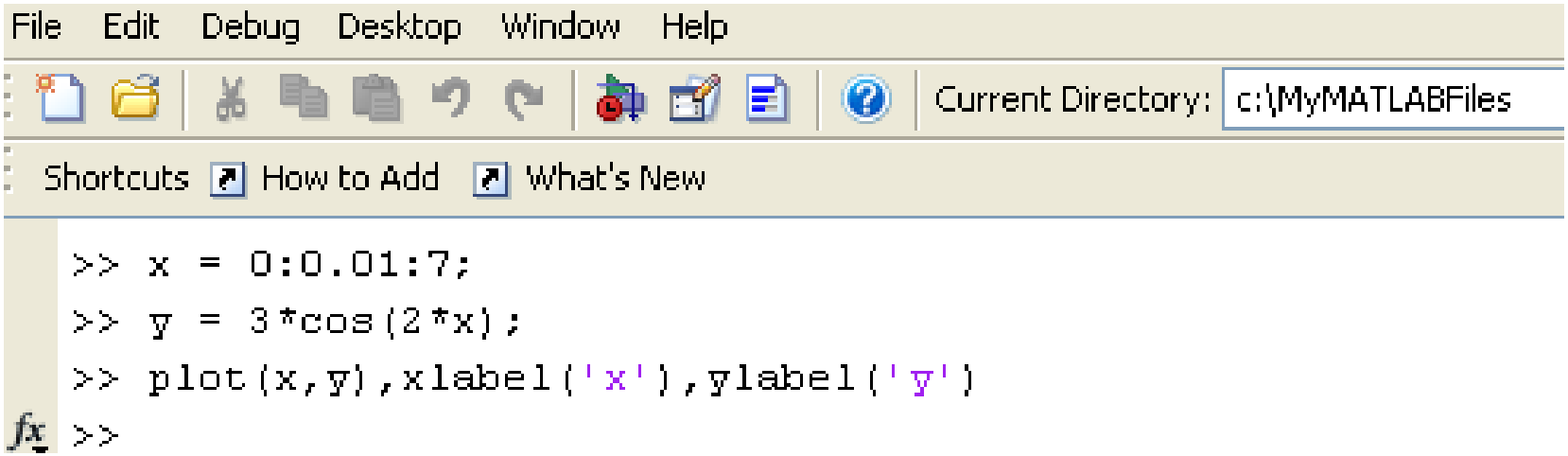
- Once you close Matlab program, All the commands you enter in the command windows is not saved and can not be used later.
- M-files can be saved and used later.
- From now all the programs will be written in M-files.
- The name of a script file must begin with a letter, and may include digits and the underscore character, up to 63 characters.
- Do not give a M-file the same name as a variable.
- Do not give a M-file the same name as a MATLAB command or function. You can check to see if a command, function or file name already exists by using the `exist` command.

M-files

- To create an M-file:

  Go to File ⟹ New

An example of an m-file is shown below:



Comments in Matlab

The comment symbol may be put anywhere in the line. MATLAB ignores everything to the right of the % symbol. For example,

```
>>% This is a comment.
>>x = 2+3 % So is this.
x =
    5
```

Note that the portion of the line before the % sign is executed to compute x.

Comments in Matlab

Commenting is used to organize the code and making it easy to understand the code later.
The comment symbol may be put anywhere in the line.
MATLAB ignores everything to the right of the % symbol.
For example,

```
% Program to solve Example 3-1

%Inputs
m=5; % mass
c=0; %damping
k=16; %stiffness
```

Note that the portion of the line before the % sign is executed.

The beginning of the program

We will start our program by clearing the variables from the memory, clearing any previous graph and clearing the command windows (If an error occurs, it will appear right away in the command window)

```
% Example 1
clear all; clf; clc

```

How does Matlab interact when we write a text in the command window?

When you type `problem1` in command window,

1. MATLAB first checks to see if problem1 is a variable and if so, displays its value.
2. If not, MATLAB then checks to see if problem1 is one of its own commands, and executes it if it is.
3. If not, MATLAB then looks in the current directory for a file named `problem1.m` and executes `problem1` if it finds it.
4. If not, MATLAB then searches the directories in its search path, in order, for `problem1.m` and then executes it if found.

System, directory, and file commands

Table 1.3–2,Page 23

Command	Description
`addpath dirname`	Adds the directory `dirname` to the search path.
`cd dirname`	Changes the current directory to `dirname`.
`dir`	Lists all files in the current directory.
`dir dirname`	Lists all the files in the directory `dirname`.
`path`	Displays the MATLAB search path.
`pathtool`	Starts the Set Path tool.
`pwd`	Displays the current directory.
`rmpath dirname`	Removes the directory `dirname` from the search path.
`what`	Lists the MATLAB-specific files found in the current working directory. Most data files and other non-MATLAB files are not listed. Use `dir` to get a list of all files.
`what dirname`	Lists the MATLAB-specific files in directory `dirname`.