

# MATLAB - Lecture # 4

## Script Files / Chapter 4

### Topics Covered:

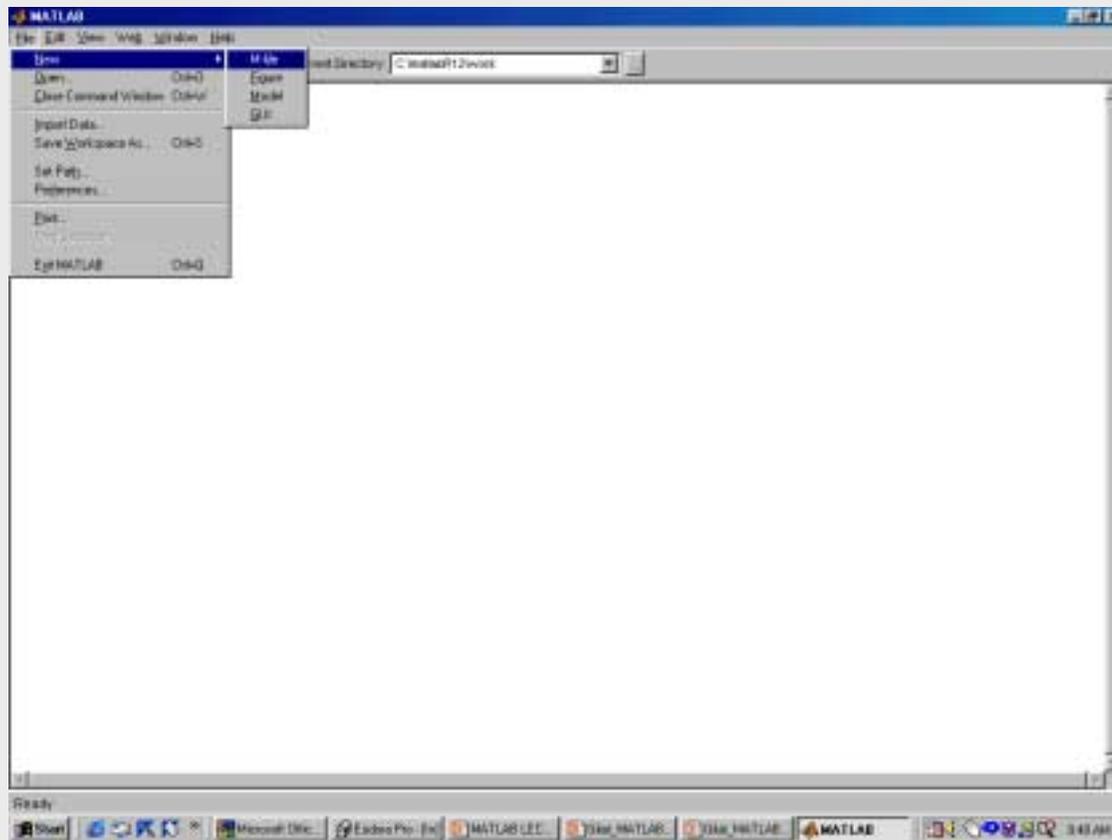
1. Script files.

## SCRIPT FILE

- A script file is a sequence of MATLAB commands, called a program.
- When a file runs, MATLAB executes the commands in the order they are written just as if they were typed in the Command Window.
- When a script file generates output, the output is displayed in the Command Window.
- Using a script file is convenient because it can be edited (corrected and/or changed) and executed many times.
- Script files are also called M-files because the extension .m is used when they are saved.

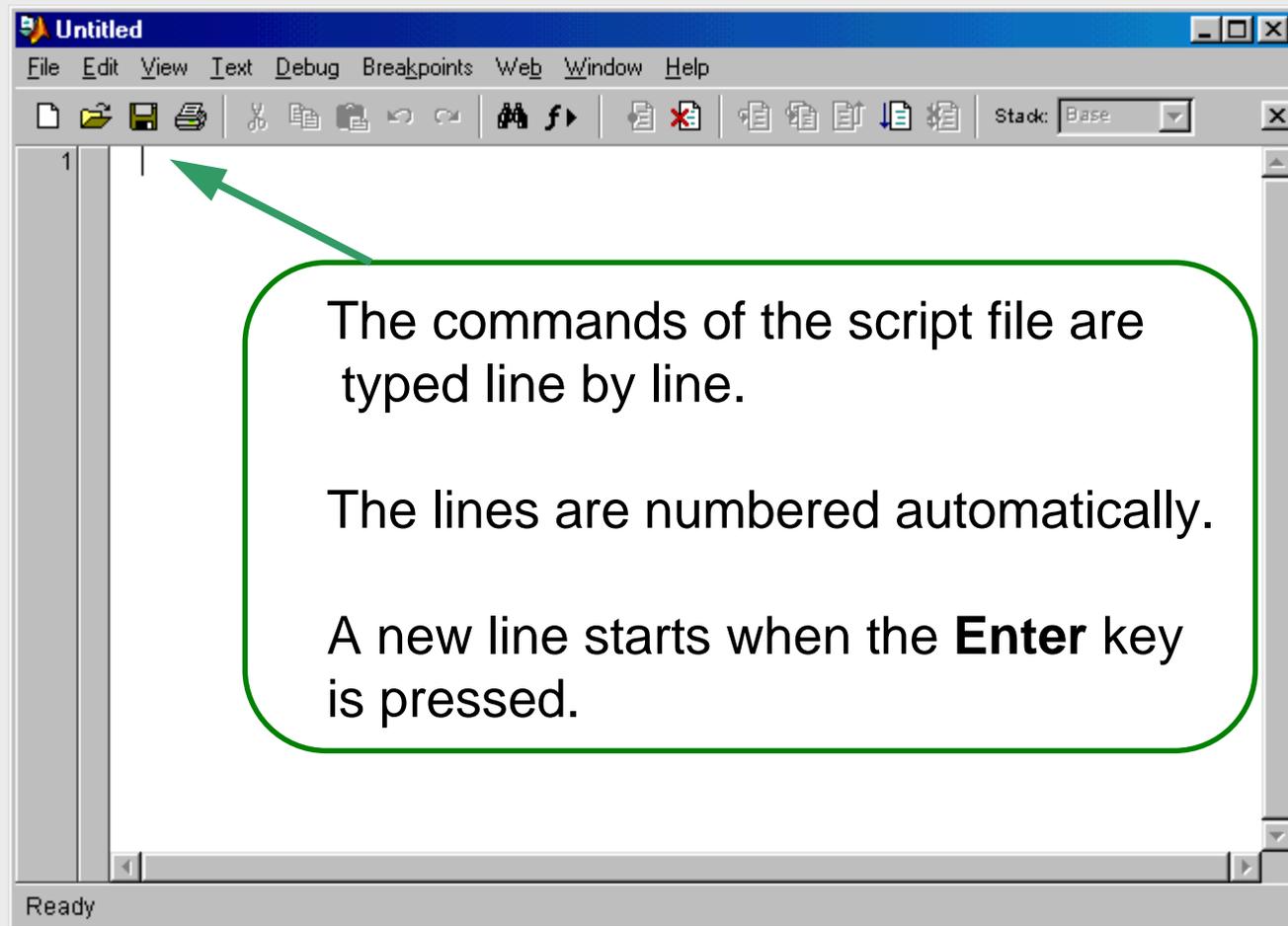
## CREATING A SCRIPT FILE

In the Command Window click on the **File** menu, select **New**, and then select **M-file**.



Once **M-file** is selected, the M-file Editor/Debugger Window opens.

## The M-file Editor/Debugger Window



# EXAMPLE OF A SCRIPT FILE

The screenshot shows a MATLAB script editor window titled "A:\Lecture3Example1.m". The window contains the following code:

```

1  % Example of a script file.
2  % This program calculates the square root of the numbers 1 through 10.
3  % It displays a vector x with the numbers 1 through 10
4  % and the vector y with the square root.
5  x = [1:10]
6  y = sqrt(x)

```

Annotations in the image include:

- A red arrow points from a green-bordered box labeled "The Run icon" to the Run icon in the toolbar.
- A yellow arrow points from a yellow-bordered box labeled "Define a vector  $x$ ." to the line `x = [1:10]`.
- A yellow arrow points from a yellow-bordered box labeled "Vector  $y$ , elements are square root of  $x$  elements." to the line `y = sqrt(x)`.
- A green-bordered box contains the text: "The text can be typed in this window, or it can be typed (and edited) in any text editor (e.g Microsoft Word) and then pasted here."
- A green-bordered box contains the text: "Recall that typing % at the beginning of a line designates the line as a comment, which is not executed."

The status bar at the bottom left of the window shows "Ready".

## SAVING A SCRIPT FILE

- ❖ Once the script file is completed, it must be saved. In our class use **Save As** and save in the floppy A drive, or the zip drive.
- ❖ The name of the script file follows the rules for names of variables in MATLAB. (Must begin with a letter, can include digits and underscore, up to 53 characters long, don't give the file a name of a variable that is used, or a predefined variable, don't use a name of a MATLAB command or a function.)

## RUNNING A SCRIPT FILE

- ❖ A script file can be executed from the Command Window by typing its name and pressing the **Enter** key.
- ❖ A script file can be executed from the Editor Window by clicking on the **Run** icon.
- ❖ To run a script file that is saved in drive A, the MATLAB search path has to be modified to include drive A, or the Current Directory has to be changed to drive A.
- ❖ To change the Current Directory to drive A type (in the command window):

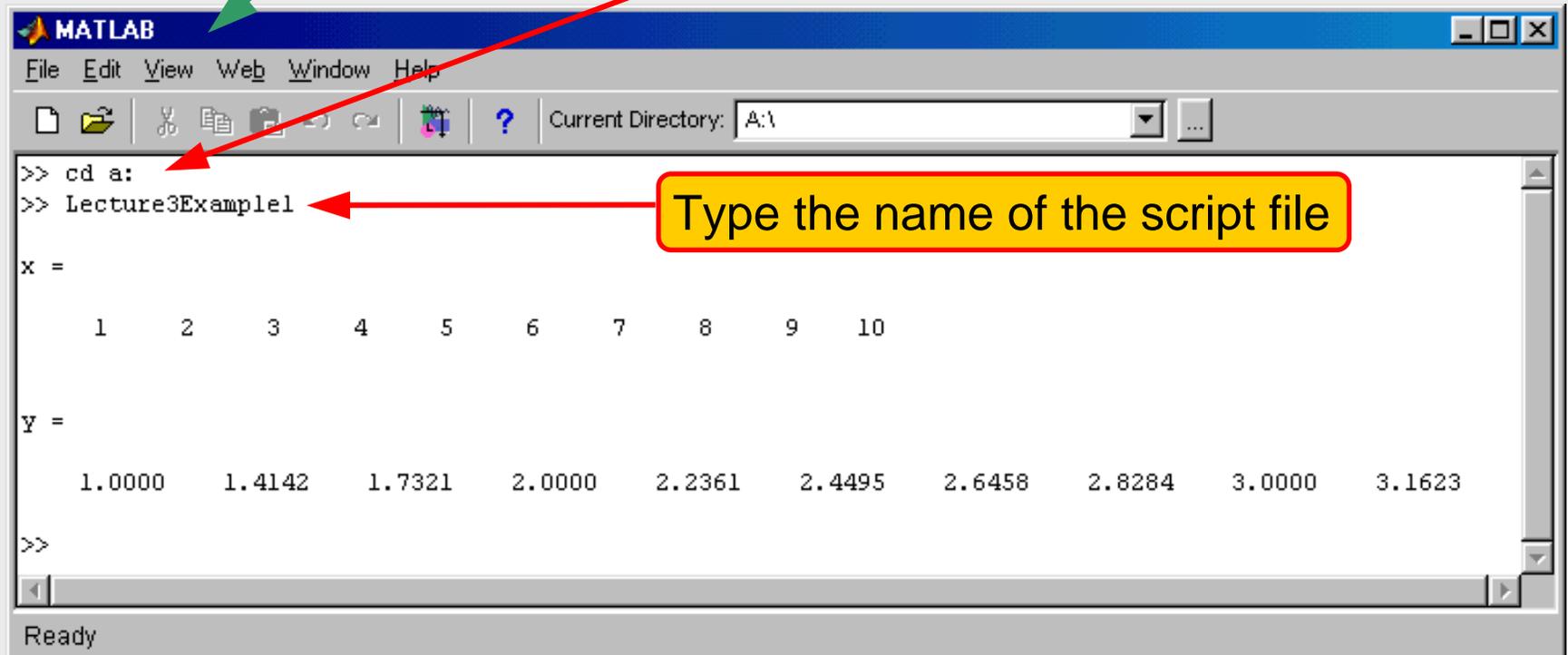
**cd a:**

Read pp. 79-81 in the book about other ways to change the Current Directory or the Search Path.

## EXAMPLE OF RUNNING A SCRIPT FILE

Command Window

Setting the Current directory to drive A



The screenshot shows the MATLAB Command Window interface. The title bar reads 'MATLAB'. The menu bar includes 'File', 'Edit', 'View', 'Web', 'Window', and 'Help'. The toolbar contains icons for file operations and a 'Current Directory' dropdown menu set to 'A:\'. The command prompt shows the following sequence of commands and output:

```
>> cd a:
>> Lecture3Example1
```

The output of the script is displayed as follows:

```
x =
     1     2     3     4     5     6     7     8     9    10

Y =
  1.0000  1.4142  1.7321  2.0000  2.2361  2.4495  2.6458  2.8284  3.0000  3.1623

>>
```

The status bar at the bottom left indicates 'Ready'. A yellow callout box points to the 'cd a:' command with the text 'Setting the Current directory to drive A'. Another yellow callout box points to the 'Lecture3Example1' command with the text 'Type the name of the script file'. A green callout box points to the Command Window title bar with the text 'Command Window'.

The output that is generated when the script file runs is printed in the Command Window.

## GLOBAL VARIABLES

- ❖ Global variables are variables that, once created in one part of MATLAB, are recognized in other parts of MATLAB.
- ❖ Variables that are created in the Command Window are recognized and can be used in a script file.
- ❖ Variables that are created in a script file are recognized and valid in the Command Window.

## INPUT TO A SCRIPT FILE

A script file is a program that can be executed with different values of its variables. This can be done in three different ways depending on where and how the variables are defined:

1. The variable is defined in the script file. To run the script file with different variable value, the file is edited and the value of the variable is changed. Then the file is saved, and executed.
2. The variable is defined in the command window. To run the script file with a different value, a new value is given to the variable in the Command Window. Then the script file is executed.

## INPUT TO A SCRIPT FILE

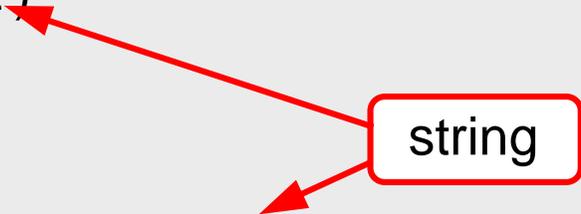
3. The variable is defined in the script file but a specific value is entered in the Command Window. When the script file runs the user is prompted to enter a value from the Command Window.

This is done by using the **input** command:

```
x = input('text')
```

For example:

```
x = input('Please enter a value for x')
```



string

Once a number (or a vector, or a matrix) is entered, x has this value.

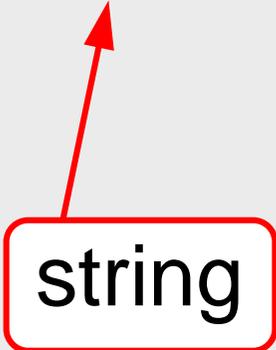
## OUTPUT FROM A SCRIPT FILE

- ❖ When a script file runs, output that is generated is displayed in the Command Window.
- ❖ Output is displayed automatically if a statement does not end with a semicolon.
- ❖ Output can also be displayed intentionally by using the **disp** command.

## The `disp` COMMAND

**`disp(A)`** Displays the content, but not the name, of the variable A.

**`disp('text')`** Displays the text (string) that is enclosed within the single quotes.



string

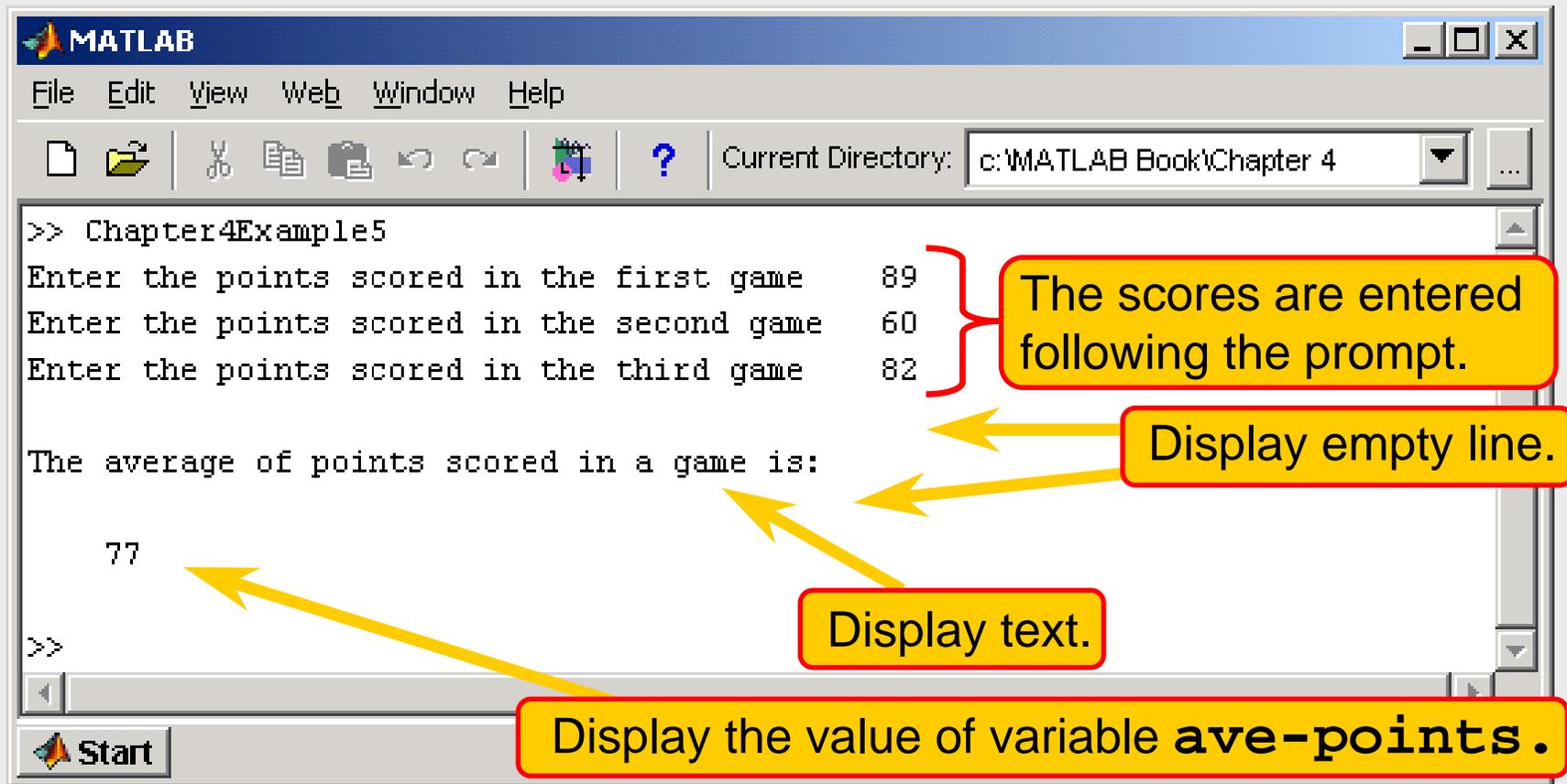
Every time a **`disp`** command is executed, the display it generates appears in a new line.

## EXAMPLE OF A SCRIPT FILE THAT USES THE `input` AND `disp` COMMANDS

```
C:\MATLAB BOOK\Chapter 4\Chapter4Example5.m
File Edit View Text Debug Breakpoints Web Window Help
Stack: Base
1 % This script file calculates the average points scored in three games.
2 % The point frm each game are assigned to the variables by
3 % using the input command.
4 % The disp command is used to display the output.
5 - game1 = input('Enter the points scored in the first game ');
6 - game2 = input('Enter the points scored in the second game ');
7 - game3 = input('Enter the points scored in the third game ');
8 - ave_points = (game1+game2+game3)/3;
9 - disp(' ')
10 - disp('The average of points scored in a game is:')
11 - disp(' ')
12 - disp(ave_points)
```

Display the value of variable **ave-points**.

# RUNNING THE SCRIPT FILE WITH THE `input` AND `disp` COMMANDS IN THE COMMAND WINDOW



The screenshot shows the MATLAB Command Window with the following content:

```
>> Chapter4Example5
Enter the points scored in the first game      89
Enter the points scored in the second game    60
Enter the points scored in the third game     82

The average of points scored in a game is:

    77
>>
```

Annotations in yellow boxes with arrows pointing to the corresponding output:

- The scores are entered following the prompt.
- Display empty line.
- Display text.
- Display the value of variable `ave-points`.

## CREATE AND DISPLAY A TABLE

This script file shows how to create and display a table. The table includes the population data from Lecture 2.

```
yr = [1984 1986 1988 1990 1992 1994 1996];
```

```
pop = [127 130 136 145 158 178 211];
```

The population data is entered in two row vectors.

```
tableYP(:,1)=yr';
```

yr is entered as the first column in the array `tableYP`.

```
tableYP(:,2)=pop';
```

pop is entered as the second column in array `tableYP`.

```
disp('    YEAR    POPULATION')
```

Display heading (first line).

```
disp('        (MILLIONS)')
```

Display heading (second line).

```
disp('    ')
```

Display an empty line.

```
disp(tableYP)
```

Display the array `tableYP`

## CREATE AND DISPLAY A TABLE

Executing the script file from the previous slide in the command window gives:

```
>> Lecture4Example3
YEAR      POPULATION
(MILLIONS)

1984      127
1986      130
1988      136
1990      145
1992      158
1994      178
1996      211
```

Display heading (first line).

Display heading (second line).

Display an empty line.

Display the array `tableYP`

## The `fprintf` COMMAND

- Can be used to display output on the screen or save it to a file.
- Text and numerical values can be intermixed and displayed in the same line.
- The format of the numbers can be controlled.

The **`fprintf`** command will not be covered in the class, but students who wish can learn the command from the book (pp. 87-94) and use it in the homework assignments

## MATLAB ASSIGNMENT 4:

1. MATLAB book, Chapter 4, Problem 1.
2. MATLAB book, Chapter 4, Problem 5.
3. MATLAB book, Chapter 4, Problem 7.

For each problem write a script file and execute it in the Command Window.

For each problem, the first two lines of the script file are:

```
% Assignment 4, Problem (write the number of the problem)
```

```
% Name: (first name, last name)
```

Submit a printout of the script file, and a printout of the command window.