

© Chen-Shao Hsu (許呈韶), 2010

## Introduction-1

- MATLAB is a powerful and popular software designed to solve a variety of math problems, such as:
- Find the numerical solution to differential equations: e.g.
   ODEs with time-varying coefficients, wave equations
- Perform Fourier/Lapace transforms
- Data processing: e.g. data statistics, sort, curve plots
- Hardware control: e.g. move a motorized stage
- Digital signal processing: e.g. enhance the image contrast
- Optimization (find a value/function corresponding to a extreme of a function/functional): e.g. genetic algorithm



## Introduction-2

- MATrix LABoratory is best known for its matrix calculation ability, e.g.
  - $$\begin{split} &a_{11}x + a_{12}y + a_{13}z = d_1 \ , \quad A = [a_{11} \ a_{12} \ a_{13} \ ; a_{21} \ \ldots ] \\ &a_{21}x + a_{22}y + a_{23}z = d_2 \ , \quad D = [d_1 \ ; d_2 \ ; d_3 \ ] \\ &a_{31}x + a_{32}y + a_{33}z = d_3 \ , \quad [x; y; z] = inv(A)*D \end{split}$$
- It is much easier to use than C/C++, because:
- No need to "declare" every variable in the beginning
- Flexible matrix manipulation
- No need to compile before execution



## Outline

- How to start: Open an M-file for programming.
- How to use vectors, matrixes, arrays, and the corresponding operators (+, -, \*, /, .\*, ./, ...)?
- How to use the built-in Math functions?
- How to generate a figure to show the result?
- How to save your data & figure?
- How to use For-loop to do the repeated task?



## M-file (script)

- M-file is used as a script(劇本), where you can edit (and debug) a sequence of instructions (like C) to complete a task. It can be saved for future use.
- Demo:



#### Vectors, Matrixes, Arrays

- Vectors: 1D sequence of numbers, categorized as column vectors & row vectors.
- Matrix: 2D sequence of numbers.
- Arrays: *n* dimensional. (Mathworks)





## Operators

- Example: A = [1 2 3]; B = [4 5 6]; (row vectors)
- A + B = [5 7 9]
- A B = [-3 3 3]
- A \* B = ??? Error using ==> mtimes
   Inner matrix dimensions must agree
- A .\* B = [4 10 8] (Element-by-element multiplication)
- A  $/B = [0.25 \ 0.4 \ 0.5]$  (Element-by-element division)
- A.^2 = [1 4 9] (Element-by-element squaring)



### **Matrix Manipulations**

Example: A = [1 2 3]; B = [4 5 6]; (row vectors)

- $A(2) = []; \Rightarrow A = [1 3] (Void)$
- [A B] = [1 2 3 4 5 6] (Concatenation)

• 
$$[A; B] = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$
 (Concatenation)



### **Built-in Math Functions**

#### Example: $x = [1:1:100] = [1 \ 2 \ 3 \ 4 \ ... \ 100]$ $y = x.^2; = [1 \ 4 \ 9 \ 16....10000]$ y = abs(x), max(x), sin(x), exp(x), fft(x), ...

Note: You can find the complete list of built-in functions by Help>Product Help>Matlab>Functions – (By Category or Alphabetical List)



## Plot Figures (plot)

 x = linspace(0, 2\*pi, 100); % equally spaced element values y = sin(x);

figure; plot(x,y);





Ultrafast Photonics Laboratory, National Tsing Hua University

## Save/Load Data

#### save FileName Variables

Example:

$$\mathbf{x} = [1:1:100] = [1 \ 2 \ 3 \ 4 \dots \ 100]$$

 $y = x^2; = [1 \ 4 \ 9 \ 16...10000]$ 

save data1 ("data1.mat" containing x, y is created) save data2 y ("data2.mat" containing y is created) save data3.txt -ascii ("data3.txt" containing x, y is created)

#### Ioad FileName Variables

load data1 (x, y of "data1.mat" appear in the working space) load data1 y (y of "data1.mat" appears in the working space)



## Save Figures

saveas(H, 'FileName', 'FileType')

Example:

x = linspace(0, 2\*pi, 100);

y = sin(x);

figure; plot(x,y);

saveas(gcf, 'Figure') ("Figure.fig" is created, can be edited)
saveas(gcf, 'Figure.tif') ("Figure.tif" image is created)



# For-loop

- Repeat a set of instructions for a specific number of times.
- Example:
  - $A = zeros(9,9); \ \% \text{ prepare a } 9*9 \text{ null matrix}$ for n = 1:9,  $A(n,m) = n^*m; \qquad A = \begin{bmatrix} 1 & 2 & \dots & 9 \\ 2 & 4 & \dots & 18 \\ \vdots & \vdots & \vdots & \vdots \\ 9 & 18 & \dots & 81 \end{bmatrix}$

end



## Alternative of For-loop

 Matlab is extremely inefficient in running for-loops, try to use matrix operation whenever possible.

• Example:

A = [1:1:9].'; % 9\*1 column vector

- B = [1:1:9]; % 1\*9 row vector
- $C = A^*B;$  % 9\*9 matrix  $\begin{bmatrix} 1 & 2 & \dots & 9 \\ 2 & 4 & \dots & 18 \end{bmatrix}$

$$C = \begin{vmatrix} 2 & 4 & \cdots & 18 \\ \vdots & \vdots & \vdots & \vdots \\ 9 & 18 & \cdots & 81 \end{vmatrix}$$



## **Further References**

- "Help" in MATLAB program
- MathWorks: <u>http://www.mathworks.com/</u>
- PTT: MATLAB 程設 ◎ MATLAB 程式討論區
- 張智星, MATLAB 程式設計 [入門篇], [進階篇]
- Just do it!

