

Programming in MATLAB

Dr. Risanuri Hidayat

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

Introduction

- So far in these lab sessions, all the commands were executed in the Command Window.
- The problem is that the commands entered in the Command Window cannot be saved and executed again for several times.
- Therefore, we need programming that do the following tasks:
 1. to *create* a file with a list of commands,
 2. *save* the file,
 3. *run* the file.

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

M-File Scripts

- A *script file* is an external file that contains a sequence of MATLAB statements.
- Script files have a filename extension `.m` and are often called M-files.
- M-files can be *scripts* that simply execute a series of MATLAB statements,
- or they can be *functions* that can accept arguments and can produce one or more outputs.

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

M-File Scripts

$$\begin{array}{rcl} x + 2y + 3z & = & 1 \\ 3x + 3y + 4z & = & 1 \\ 2x + 3y + 3z & = & 2 \end{array}$$

- Use the MATLAB *editor* File
-> New -> M-file.
- Enter the following
statements in the M-file:
`A = [1 2 3; 3 3 4; 2 3 3];`
`b = [1; 1; 2];`
`x = A\b`

- Save the file, for example,
example1.m.
- Run, in the command line, by
typing:

```
>> example1  
x =  
-0.5000  
1.5000  
-0.5000
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

M-File Scripts

- Create a file, say example2.m, which contains the following commands:

```
x = 0:pi/100:2*pi;  
y1 = 2*cos(x);  
y2 = cos(x);  
y3 = 0.5*cos(x);  
plot(x,y1,'--',x,y2,'-',x,y3,':')  
xlabel('0 ≤ x ≤ 2π')  
ylabel('Cosine functions')  
legend('2*cos(x)', 'cos(x)', '0.5*cos(x)')  
title('Typical example of multiple plots')  
axis([0 2*pi -3 3])
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

M-File functions

- This simple function shows the basic parts of an M-file.

```
function f = factorial(n) (1)  
% FACTORIAL(N) returns the factorial of N. (2)  
% Compute a factorial value. (3)  
f = prod(1:n); (4)
```

```
>> f = factorial(5)  
f =  
120
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

Difference between scripts and functions

SCRIPTS	FUNCTIONS
- Do not accept input arguments or return output arguments. - Store variables in a workspace that is shared with other scripts - Are useful for automating a series of commands	- Can accept input arguments and return output arguments. - Store variables in a workspace internal to the function. - Are useful for extending the MATLAB language for your application

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

arguments

- As mentioned above, the input arguments are listed inside parentheses following the function name.
- The output arguments are listed inside the brackets on the left side. They are used to transfer the output from the function file. The general form looks like this

`function [outputs] = function_name(inputs)`

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

arguments

<code>function C=FtoC(F)</code>	One input argument and one output argument
<code>function area=TrapArea(a,b,h)</code>	Three inputs and one output
<code>function [h,d]=motion(v,angle)</code>	Two inputs and two outputs

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

Input

- When the file is executed, the user is *prompted* to assign a value to the variable in the command prompt. This is done by using the input command. Here is an example.

```
game1 = input('Enter the points scored in the first game ');
game2 = input('Enter the points scored in the second game ');
game3 = input('Enter the points scored in the third game ');
average = (game1+game2+game3)/3
>> example3
>> Enter the points scored in the first game 15
>> Enter the points scored in the second game 23
>> Enter the points scored in the third game 10
average =
16
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

Output commands

- `disp` . Simple to use.
 - . Provide limited control over the appearance of output
- `fprintf` . Slightly more complicated than `disp`.
 - . Provide total control over the appearance of output

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

PR

- STL. Liz buys three apples, a dozen bananas, and one cantaloupe for \$2.36. Bob buys a dozen apples and two cantaloupe for \$5.26. Carol buys two bananas and three cantaloupe for \$2.77. How much do single pieces of each fruit cost?
- SIE. Write a function file that converts temperature in degrees Fahrenheit ($\pm F$) to degrees Centigrade ($\pm C$). Use `input` and `fprintf` commands to display a mix of text and numbers. Recall the conversion formulation, $C = 5/9 * (F - 32)$.
- SKI. Write a user-defined MATLAB function, with two input and two output arguments that determines the height in centimeters (cm) and mass in kilograms (kg) of a person from his height in inches (in.) and weight in pounds (lb).
 - (a) Determine in SI units the height and mass of a 5 ft.15 in. person who weight 180 lb.
 - (b) Determine your own height and weight in SI units.

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

Control Flow

- MATLAB has four control flow structures:
 - if statement,
 - for loop,
 - while loop,
 - switch statement

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

``if...end'' structure

- MATLAB supports the variants of “if” construct.
 - `if ... end`
 - `if ... else ... end`
 - `if ... elseif ... else ... end`
- The simplest form of the if statement is

```
if expression
    statements
end
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

``if...end'' structure

```
discr = b*b - 4*a*c;  
if discr < 0  
    disp('Warning: discriminant is  
        negative, roots are  
        imaginary');  
end
```

```
discr = b*b - 4*a*c;  
if discr < 0  
    disp('Warning: discriminant is  
        negative, roots are  
        imaginary');  
else  
    disp('Roots are real, but may be  
        repeated')  
end
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

```
discr = b*b - 4*a*c;  
if discr < 0  
    disp('Warning: discriminant is  
        negative, roots are  
        imaginary');  
elseif discr == 0  
    disp('Discriminant is zero, roots  
        are repeated')  
else  
    disp('Roots are real')  
end
```

``if...end'' structure

- elseif has no space between else and if (one word)
- no semicolon (;) is needed at the end of lines containing if, else, end
- indentation of if block is not required, but facilitate the reading.
- the end statement is required

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

Relational

OPERATOR	DESCRIPTION
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
==	Equal to
~=	Not equal to
&	AND operator
	OR operator
~	NOT operator

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

``for...end" loop

for variable = expression

statements

end

for ii=1:5

x=ii*ii

end

n = 5; A = eye(n);

for j=2:n

for i=1:j-1

A(i,j)=i/j;

A(j,i)=i/j;

end

end

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY

``while...end'' loop

```
while expression  
    statements  
end
```

- The statements are executed as long as expression is true.

```
x = 1  
while x <= 10  
    x = 3*x  
end
```

9/8/2009

Electrical Engineering Department,
GADJAH MADA UNIVERSITY