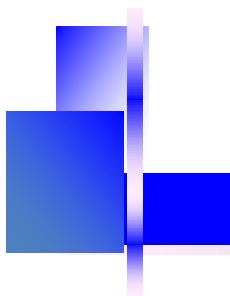




## Elective Course - 4<sup>th</sup> year



### New Programming Language (MATLAB)



# Control Statements

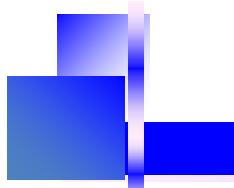
*prepared by*

***Eng./ Essam Nabil***

*Dept. of Industrial Electronics and Control Eng.*

*Faculty of Electronic Engineering*

*Minoufiya University*



# Lecture Outlines

## Control Statements:

### 1. Loop Statements

FOR Loop Statement

WHILE Loop Statement

### 2. Conditional Statements

IF –ELSE Statement

SWITCH Statement

## Input/output Commands:

Pause

Keyboard

input

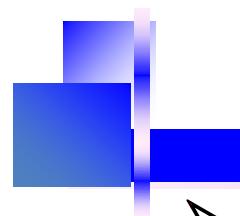
disp

fprintf

break

error

## Matlab Function design and Exercise.



# MATLAB Operators

## ➤ Relational Operators:

Operator	Description
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equal to
~=	Not equal to

## ➤ Logical Operators:

Logical Operation	Equivalent Function
A & B	and (A, B)
A   B	or (A, B)
~A	not (A)

# Loop Statements

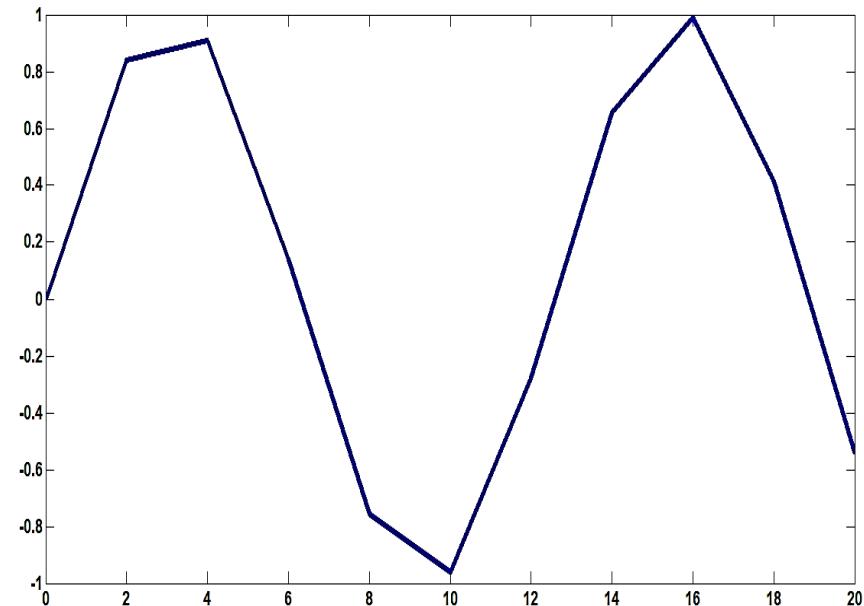
## □ FOR-Loop Statement:

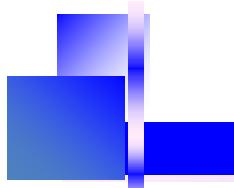
### Syntax:

```
for index=start : increment : end  
    statements;  
end
```

### Example:

```
for i = 0 : 10  
    x(i+1) = 2*i;  
    y(i+1) = sin(i);  
end  
plot(x,y)
```





# Loop Statements

## □ WHILE-Loop Statement:

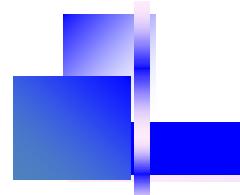
### Syntax:

```
while expression  
    statements;  
end
```

### Example:

```
n=0;  
while 2^n < 10  
    n=n+1;  
end
```





# Conditional Statements

## IF-ELSE Statement:

### Syntax1:

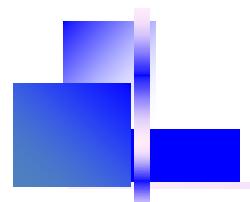
```
if expression  
    statements;  
end
```

### Syntax3:

```
if expression 1  
    statements1;  
elseif expression2  
    statements2;  
elseif expression3  
    statements3;  
...  
else  
    statementsn;  
end
```

### Syntax2:

```
if expression  
    statements1;  
else  
    statements2;  
end
```



# Conditional Statements

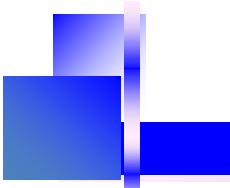
## Example:

3-bit A/D converter with analog input  $X$  and digital output  $Y$  represented as:

$Y = 0$	$X < -2.5$
$= 1$	$-2.5 \leq X < -1.5$
$= 2$	$-1.5 \leq X < -0.5$
$= 3$	$-0.5 \leq X < 0.5$
$= 4$	$0.5 \leq X < 1.5$
$= 5$	$1.5 \leq X < 2.5$
$= 6$	$2.5 \leq X < 3.5$
$= 7$	$X \geq 3.5$

## Solution:

```
function Y_dig=A2D(X)
%A2D performs 3-bit ADC with
%analog I/p X and digital O/p Y
if X<-2.5
    Y_dig=0
elseif (X>=-2.5)&(X<-1.5)
    Y_dig=1
elseif (X>=-1.5)&(X<-0.5)
    Y_dig=2
...
else
    Y_dig=7
end
```

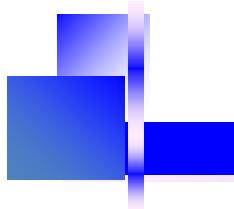


# Conditional Statements

## SWITCH-CASE Statement:

### Syntax:

```
switch expression (scalar or string)
    case value1
        statement1;
    case value2
        statement2;
    ...
otherwise
    statementn,
end
```

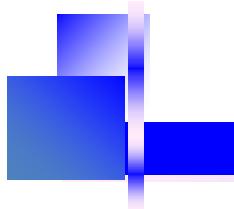


# Conditional Statements

X	Y
0.1	0.5
0.2	0.88
0.5	1.1
0.75	3.12
others	0

Solution:

```
switch X  
case 0.1  
    Y=0.5;  
case 0.2  
    Y=0.88;  
case 0.5  
    Y=1.1;  
case 0.75  
    Y=3.12;  
otherwise  
    Y=0;  
end
```



# Input/output Commands

---

## pause

:halt execution temporarily i.e. causes the currently executing M-file to stop and wait for you to press any key before continuing.

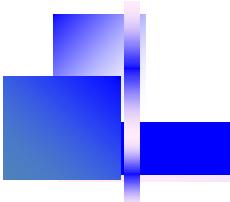
## keyboard

:halt execution and terminate the control to input from keyboard to end keyboard mode type "**return**" command.

## input

: To request user input as:

**user\_name = input(' Enter your name: ')**



# Input/output Commands

---

**disp (x)** :Display the variable x value without printing the variable name.

**disp ('text')** :Display 'text' on the screen.

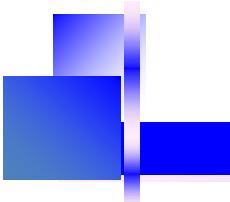
**fprintf (format,data)** :used to print both text (e.g. variable name) and variable value.

`>> fprintf( 'Result is %d' , 3 )`

Result is 3

`>> fprintf( 'Area of a circle with radius %d is %f' , 3, pi*3^2 )`

Area of a circle with radius 3 is 28.274334



# Input/output Commands

---

**break**

: Terminates execution of **for** or **while** loop.

**error**

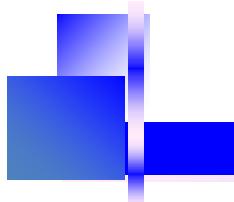
: Display user error defined message and abort function.

```
x = input(' Enter age of student : ')
```

```
if x<0
```

```
    error( 'Wrong age is entered, try again' )
```

```
end
```



# MATLAB Function Design

## ■ User-defined functions design steps:

1. From Menu bar select File->New->Function or open a new m-file editor.
2. Define Function m-file declaration line as:

```
function [x, y, z] = funName(in1, in2)
```

Must have the reserved word: **function**

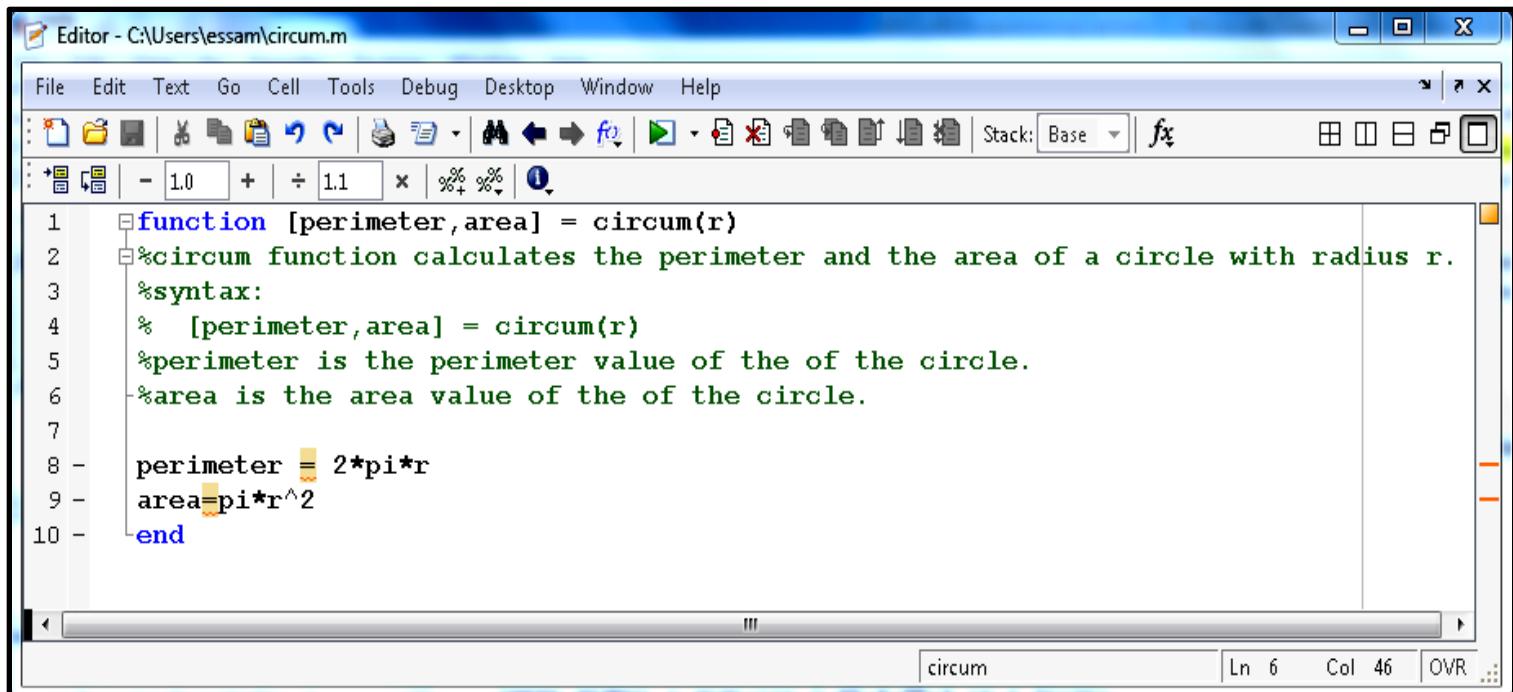
If more than one output, must be in brackets

Function name should match m-file name

Inputs must be specified

# MATLAB Function Design

3. Write Comment lines that describe detailed help information about your function.
4. Write all equations that required in your function.
5. Save the Functions m-file with the same name of the function.  
**(Don't change default name)**
6. Now, your function is ready for using.



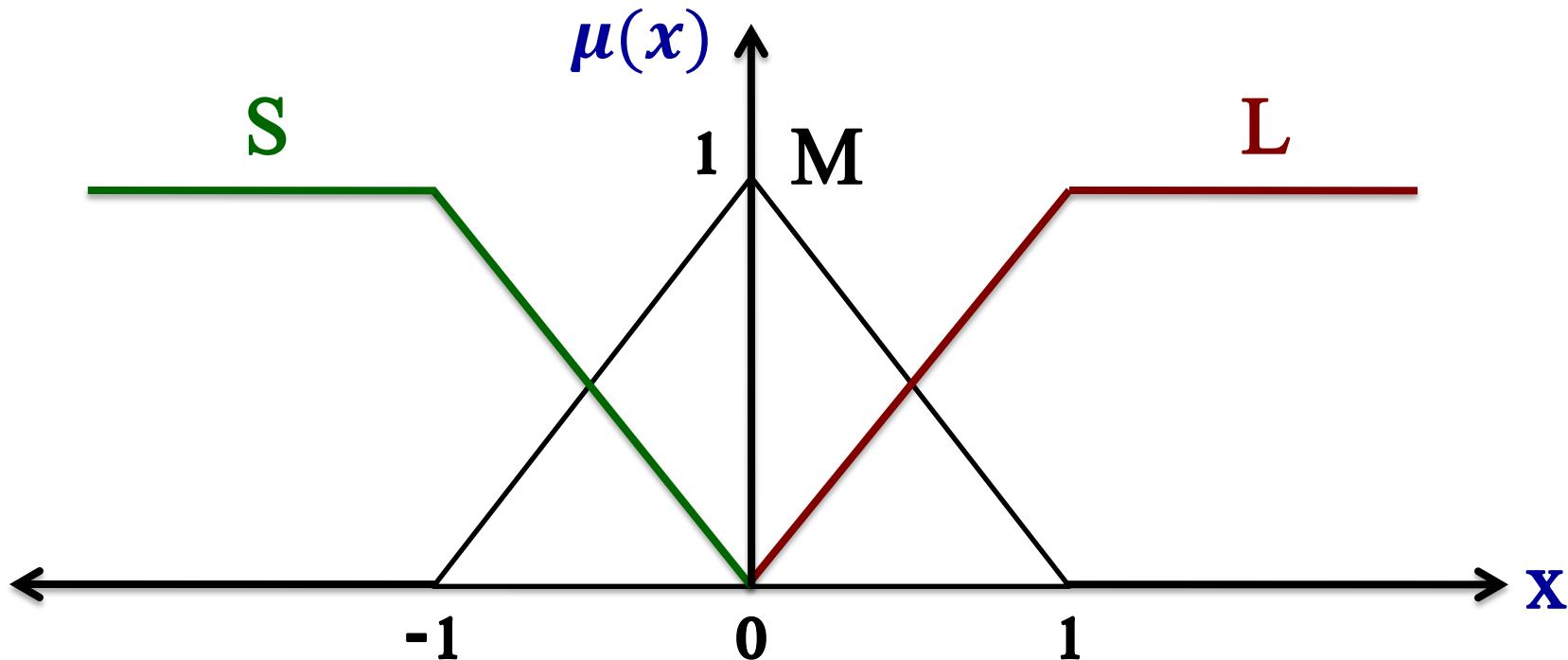
The screenshot shows the MATLAB Editor window with the file `C:\Users\essam\circum.m` open. The code defines a function `circum` that calculates the perimeter and area of a circle given its radius `r`. The function includes comments explaining its purpose, syntax, and the meaning of the output variables. The MATLAB interface includes a toolbar, a menu bar, and a command window at the bottom.

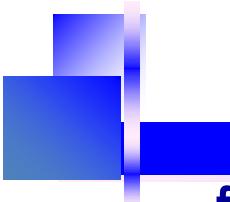
```
function [perimeter,area] = circum(r)
%circum function calculates the perimeter and the area of a circle with radius r.
%syntax:
% [perimeter,area] = circum(r)
%perimeter is the perimeter value of the of the circle.
%area is the area value of the of the circle.

perimeter = 2*pi*r
area=pi*r^2
end
```

# Exercise

► Design a MATLAB function to calculate the membership values  $S$ ,  $M$  and  $L$  for a given input  $x$  for the following represented membership function:





# Solution

---

```
function [S, M, L] = fuzz(x)
%fuzz function used to evaluate the inference of the membership function.
%[small,medium,large]=fuzz(x)
if(x<-1)
    S=1;
    M=0;
    L=0;
elseif(x>=-1)&(x<0)
    S=-x;
    M=x+1;
    L=0;
elseif(x>=0)&(x<1)
    S=0;
    M=-x+1;
    L=x;
else
    S=0;
    M=0;
    L=1;
end
```

# Questions and Discussion

