

البرمجة الإجرائية

Lecture No. 7

Flow control

ميكاترونك-سنة أولى-فصل أول

Dr. Eng. Essa Alghannam

Ph.D. Degree in Mechatronics Engineering

2024

أوجد خرج كل عبارة مما يلي في MATLAB (اكتب رقم العبارة ثم الإجابة)

```
x=[1 2 0;-1 0 1;1 2 3];  
y=[0 -6 0];
```

- a. `size_x=size(x,1)`
- b. `diag(x)`
- c. `inv(x) == eye(size(x))/x`
- d. `m=min(min([x;y]))`
- e. `t=[y;x]`
- f. `t(end,:)=zeros`
- g. `f=find(y==0)`
- h. `y(f)=1`
- i. `g=length(linspace(-10,-2,5))`

a) **size_x =**
3

ans =
1
b) 0
3

c) **ans =**
3×3 logical array
1 1 1
1 1 1
1 1 1

d) **m =**
-6

e) **t =**
0 -6 0
1 2 0
-1 0 1
1 2 3

f) **t =**
0 -6 0
1 2 0
-1 0 1
0 0 0

g) **f =**
1 3

h) **y =**
1 -6 1

i) **g =**
5

اكتب مجموعة تعليمات في MATLAB لإدخال ثلاث معادلات من الدرجة الثانية ومن ثم إيجاد الحلول لها دفعة واحدة وطباعة كل معادلة وحلها على نفس السطرو فوق الشكل التالي.

The solutions of $a_1x^2 + b_1x + c_1 = 0$ are x_1 x_2

```
a=input('input a as a 3*1 vector:');
b=input('input b as a 3*1 vector:');
c=input('input c as a 3*1 vector:');
delta=b.^2 - 4*a.*c;
x1=(-b+sqrt(delta))./(2*a);
x2=(-b-sqrt(delta))./(2*a);
disp(['The solutions of:' num2str(a(1)) 'x2+' num2str(b(1)) 'x+' num2str(c(1)) '=0
are:' num2str([x1(1) x2(1)]) ])
disp(['The solutions of:' num2str(a(2)) 'x2+' num2str(b(2)) 'x+' num2str(c(2)) '=0
are:' num2str([x1(2) x2(2)]) ])
disp(['The solutions of:' num2str(a(3)) 'x2+' num2str(b(3)) 'x+' num2str(c(3)) '=0
are:' num2str([x1(3) x2(3)]) ])
```

Flow Control

- if
- for
- while
- break

Flow Control Structures If Statement Syntax

```
if (Condition_1)  
    Matlab Commands  
elseif (Condition_2)  
    Matlab Commands  
elseif (Condition_3)  
    Matlab Commands  
else  
    Matlab Commands  
end
```

& is not so clear and it is used as logical and between two values

Flow Control Structures If Statement Syntax

```
if ((x>6) && (y==0))  
    Matlab Commands;  
End
```

```
x=input('enter a number')  
If(x==0)  
    disp('equal to zero')  
End
```

```
if (x<4)  
    Matlab Commands;  
else  
    Matlab Commands;  
end
```

```
x=input('enter a number')  
If(imag(x)~=0)  
    disp('complex')  
else  
    disp('real')  
end
```

```
if (x<4)  
    Matlab Commands;  
elseif (y~=2)  
    Matlab Commands;  
end
```

```
x=input('enter a number')  
If (abs(x)==0)  
    disp('zero')  
elseif (imag(x)==0)  
    disp('real')  
else  
    disp('complex')  
end
```


Flow Control Structures

if-else statement

We wish to use “if” and “else” statement to decide upon if the input integer is an odd or an even number. The Matlab code for the purpose is shown below:

```
x=input('enter an integer number:');  
if(rem(x,2))  
    disp('odd')  
else  
    disp('even')  
end
```

```
enter an integer number:-5  
odd
```

```
rem(-5,2)  
ans =  
    -1  
>> rem(-6,2)  
ans =  
    0
```

Flow Control Structures

if-else statement

Check the number :

```
clear
n=-6;
if(n<0)
    disp('input must be positive')
elseif rem(n,2)==0
    disp('even')
else
    disp('odd')
end
```



rem and mod

```
>> rem(-5,-2)
```

```
ans =  
-1
```

```
>> mod(-5,-2)
```

```
ans =  
-1
```

```
>> rem(-5,2)
```

```
ans =  
-1
```

```
>> mod(-5,2)
```

```
ans =  
1
```

x	y	x/y	fix(x/y)	floor(x/y)	rem(x/y)	mod(x/y)
-5	2	-2.5	-2	-3	-1	1
-5	-2	2.5	2	2	-1	-1

- $\text{rem}(x,y)$ returns $x - \text{fix}(x./y).*y$ if $y \neq 0$
- $\text{mod}(x,y)$ returns $x - \text{floor}(x./y).*y$ if $y \neq 0$

remainder after division
Modulus after division

$\text{rem}(x,y)$ and $\text{mod}(x,y)$ are equal if x and y have the same sign,
but differ by y if x and y have different signs.

Note: $\text{REM}(x,y)$, for $x \sim y$ and $y \neq 0$, has the same sign as x .

Note: $\text{MOD}(x,y)$, for $x \sim y$ and $y \neq 0$, has the same sign as y .

Flow Control Structures

switch case Statement Syntax

Switch expression

Case exp1

 commands1

Case {exp1, exp2, exp3}

 commands2

otherwise

 commands3

end

```
n = input('Enter a number: ');
```

```
switch n
```

```
    case -1
```

```
        disp('negative one')
```

```
    case 0
```

```
        disp('zero')
```

```
    case 1
```

```
        disp('positive one')
```

```
    otherwise
```

```
        disp('other value')
```

```
end
```

Enter a number: 5

other value

Enter a number: -1

negative one

```
grade = 'B';  
switch (grade)  
    case 'A'  
        fprintf('Excellent!\n' );  
    case 'B'  
        fprintf('Well done\n' );  
    case 'C'  
        fprintf('Well done\n' );  
% case {'B' , 'C'}  
%     fprintf('Well done\n' );  
    case 'D'  
        fprintf('You passed\n' );  
    case 'F'  
        fprintf('Better try again\n' );  
    otherwise  
        fprintf('Invalid grade\n' );  
end
```

When you run the file, it displays –
Well done



جامعة
المنارة
MANARA UNIVERSITY

Flow Control Structures

loops

- Two different types of loops exist in MATLAB.
 - *for*
 - *while*
- Their structure is as shown here

increment Initial and final value of control index

```
for index=init:inc:last
    Matlab Commands
end
```

```
for i=Index_Array
```

```
while condition_that_control_the_loop
    Matlab Commands
end
```

Flow Control Structures for loop Statement Syntax

```
for i=1:100  
    Some Matlab Commands;  
end
```

```
for j=1:3:200  
    Some Matlab Commands;  
end
```

command block

The loop variable is defined as a vector

It is scalar within the command block which is between for and end

تقوم for بتنفيذ كتلة من التعليمات عدد ثابت من المرات
عدد مرات التكرار iteration في بنية for معلوم

```
for m=13:-0.2:-21  
    Some Matlab Commands;  
end
```

ليس ضروريا أن يكون متحول الحلقة ذو قيمة تعاقبية consecutive

```
for k=[0.1 0.3 -13 12 7 -9.3]  
    Some Matlab Commands;  
end
```

Flow Control Structures for loop Statement Syntax

```
X=input('enter a number: ')\nfor i=1:5:X\n    disp([num2str(i) 'time(s)']);\nend
```

enter a number: 1

```
X =\n    1\n1 time(s)
```

enter a number: 2

```
X =\n    2\n1 time(s)\n1.5 time(s)\n2 time(s)
```

enter a number: 0

```
X =\n    0
```

enter a number: -1

```
X =\n   -1
```




جامعة
المنارة
MANARA UNIVERSITY

Flow Control Structures for loop Statement Syntax

- A simple for loop to calculate and display squares of numbers from 1 to 5.

```
for i=1:5  
    i^2  
end
```

ans =

1

ans =

4

ans =

9

ans =

16

ans =

25

Flow Control Structures for loop Statement Syntax

- Note that the squared values will be saved in the 'result' vector.
- The vector must be initialized before assigning any value to it.
- Note that this is the case for vectors and matrices only.
- We don't need to initialize scalars in MATLAB.

```
result(5)=0  
for i=1:5  
    result(i)=i^2  
end
```

result =

0 0 0 0 0

result =

1 0 0 0 0

result =

1 4 0 0 0

result =

1 4 9 0 0

result =

1 4 9 16 0

result =

1 4 9 16 25

Flow Control Structures for loop Statement Syntax

- In the following example odd numbers from 1 to 9 are printed.

```
for i=1:2:10  
    i  
end
```

```
i=  
1  
i=  
3  
i=  
5  
i=  
7  
i=  
9
```



جامعة
المنارة
MANARA UNIVERSITY

Flow Control Structures for loop Statement Syntax

```
num=0  
for x=9:-2:2  
    disp(x)  
    num=num+1  
end  
disp(num)
```

```
num =  
    0  
    9  
num =  
    1  
    7  
num =  
    2  
    5  
num =  
    3  
    3  
num =  
    4  
    4
```

Flow Control Structures for loop Statement Syntax

```
num=0  
for x=(1:4)  
    disp(x)  
    num=num+1  
end  
disp(num)
```

num =

0

1

2

3

4

num =

1

1

مرور واحد لان x شعاع عمودي

Flow Control Structures

While loop Statement Syntax

- While loop continue to execute “Matlab Commands” if the given “condition” is true.
- Beware of infinite loops!
- A condition is true if it is not equal to ‘0’.
- *Don't need to know the number of iterations*

Example

```
while (condition)
    Matlab Commands
end
```

```
while ((a>3) && (b==5))
    Some Matlab Commands;
end
```

Flow Control Structures

While loop Statement Syntax

- A while loop to display numbers from 1 to 10 is shown here. simple

In this while loop at each iteration the value of 'i' is incremented after the value is printed.

```
clc  
clear  
i=0;  
while (i<5)  
    i=i+1  
end
```

```
i =  
    1  
i =  
    2  
i =  
    3  
i =  
    4  
i =  
    5
```

Flow Control Structures

While loop Statement Syntax

- Now we can show the usage of these logical operations in the while loop as condition.
- Say $a=1$, then the following code results in the execution of loop until the condition $a < b$ gets FALSE.

```
clc  
clear  
a=1;b=5;  
while (a<b)  
    b=b-1  
end
```

```
b =  
    4  
  
b =  
    3  
  
b =  
    2  
  
b =  
    1
```


Flow Control Structures

While loop Statement Syntax

Calculate the value of eps

```
clc  
clear  
num=0;EPS=1;  
while (1+EPS) > 1  
    EPS = EPS / 2;  
    num=num+1;  
end  
EPS, EPS=EPS*2, eps, num
```

EPS =
1.1102e-16

EPS =
2.2204e-16

ans =
2.2204e-16

num =
53

Flow Control Structures

While loop Statement Syntax

```
clc  
clear  
x=[2 3 1+i];  
while (x)  
    % while abs(x)  
    beep  
    pause(1)  
end
```

Complex values cannot be converted to logical.
Error



جامعة
المنارة
MANARA UNIVERSITY

Flow Control Structures

While loop Statement Syntax

```
clc
clear
x=[2 3 1+i];
% while (x)
while abs(x)
    beep
    pause(1)
end
```

```
>> abs(x)
ans =
    2.0000    3.0000    1.4142
```

- تكرار الحلقة عدد لانتهائي من المرات لان طوييلة
- كل عنصر غير معدومة وبالتالي العبارة محققة
- إذا كان احد عناصر x معدوما لن يتم الدخول الى الحلقة لان العبارة غير محققة. مثلا

```
x=[2 3 0+0i]
```

Flow Control Structures

While loop Statement Syntax

```
clc  
clear  
x=[2 3 0+0i];  
% while (x)  
for i= abs(x)  
% while abs(x)  
    beep  
    pause(1)  
end
```

ثلاث تكرارات للحلقة

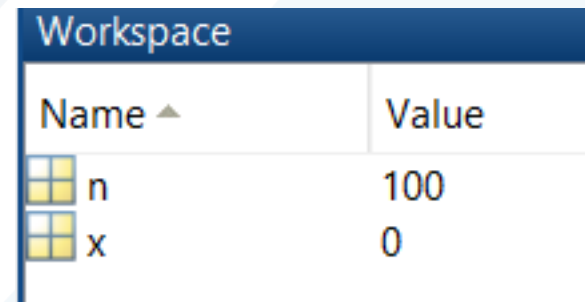
Flow Control Structures continue - break

```
clc  
clear  
x=100; n=0;  
while 1  
    x=x-1;n=n+1;  
    if x==0  
        break;  
    end  
end  
disp(n)
```

النتيجة
100

تسبب تعليمة BREAK الخروج من الحلقة التي تحويها

نهاية التنفيذ



Name ^	Value
n	100
x	0

```
clc  
clear  
n=0;  
for x=-10:10  
    n=n+1;  
    if x==0  
        break;  
    end  
end  
disp(n)
```

النتيجة

11

Flow Control Structures continue - break

```
clc  
clear  
n=0;  
for x=1:55  
    n=n+1;  
    continue  
    break;  
end  
disp(n)
```

تسبب تعليمة continue الانتقال إلى التكرار التالي للحلقات

⚠ This statement (and possibly following ones) cannot be reached. [Details](#)

النتيجة
55



جامعة
المنارة
MANARA UNIVERSITY

Flow Control Structures continue - break

```
clc  
clear  
n=100;  
while n>3  
    n=n-1;  
    continue  
    n=100;  
end  
disp(n)
```

تسبب تعليمة continue الانتقال إلى التكرار التالي للحلقات

النتيجة

3

Flow Control Structures

Example:

Matrix its elements are the row number[^] the column number :

```
clc
clear
n=5;m=6;
for i=1:n
    for j=1:m
        C(i,j)=i^j;
    end
end
C
```

C =	1	1	1	1	1	1
	2	4	8	16	32	64
	3	9	27	81	243	729
	4	16	64	256	1024	4096
	5	25	125	625	3125	15625

Thanks .

