



ME 444

MATLAB® FOR ENGINEERS

Lecturer:

Dr. Nurettin Furkan DOĞAN

Mechanical Engineering Department

4 th Floor Office No: 303

nfdogan@gantep.edu.tr

0342 317 1200- 2569



Office Hours:

Tuesday: 13.30- 15:00

Friday : 10.00- 11.30

CHAPTER 6

CONTROL STRUCTURES-2

switch Conditional Statement

- The switch statement can often be used in place of an else-if or an if statement with many elseif statements.
- Switch statements are used when you want to see if an expression is equal to one of several possible values.
- The general writing format is:

```
switch switch_statement
    case probability_1
        operations1
    case probability_2
        operations2
    case probability_3
        operations3
    (can be continued any number of times.)
    otherwise operations
end
```

✓ Here, the **switch_statement** written next to the **switch** is the name of the data that is checked (compared) in our program. We check this data by using the probabilities next to the **cases** we have written below.

✓ If **switch_statement** is matched with one of the possibilities given in the **cases**, the action in that part will be taken.

✓ Otherwise, the operation in the **otherwise** part will be executed. That is, if none of the probabilities are met, the transactions here will take place.

switch Conditional Statement

- **Switch** statements are used when you want to see if an expression is **equal to one of several possible values**. In other words, in the probabilities written next to the **case**, only the **switch_expression** is checked whether it is equal to that probability or not, and while doing this, a comparison expression such as **< (small) / > (large)** cannot be written.

```
switch a
    case a<10
        operations1
    case a>10
        operations2
otherwise operations
end
```

WRONG → SMALL OR LARGE COMPARISON
CANNOT BE DONE IN switch/case
STRUCTURE

If Conditional Statement

Example: Let's rewrite the function we wrote in the previous lesson with the switch structure.

```
%This function returns the letter equivalent of the entered grade (1-10)
function let=grade(gpa)
%First check the gpa is valid or not
    if gpa<0 || gpa>10
        let='X';
    else %If the gpa is valid, check for the letter
        switch gpa
            case 9 || 10
                let='A';
            case 8
                let='B';
            case 7
                let='C';
            case 6
                let='D';
            otherwise
                let='F';
        end
    end
end
end
```

Here, if the grade value is not less than 0 or greater than 10, the switch structure will work.

We performed greater-than-less than control with if else structure!

If Conditional Statement

Example: Let's rewrite the function we wrote in the previous lesson with the switch structure.

```
%This function returns the letter equivalent of the entered grade (1-10)
function let=grade(gpa)
%First check the gpa is valid or not
    if gpa<0 || gpa>10
        let='X';
    else %If the gpa is valid, check for the letter
        switch let
            case 9 || 10
                let='A';
            case 8
                let='B';
            case 7
                let='C';
            case 6
                let='D';
            otherwise
                let='F';
        end
    end
end
```

When we send an input(grade) value with the name of the function in the command window...

Command Window

```
>> grade2(5)
```

```
ans =
```

```
F
```

```
>> grade2(8)
```

```
ans =
```

```
B
```

```
>> grade2(15)
```

```
ans =
```

```
X
```

When an invalid grade value is entered...

If Conditional Statement

Example: Let's rewrite the function we wrote in the previous lesson with the switch structure.

```
%This function returns the letter equivalent of the entered grade (1-10)
function let=grade(gpa)
%First check the gpa is valid or not
    if gpa<0 || gpa>10
        let='X';
    else %If the gpa is valid, check for the letter
        switch let
            case {9,10}
                let='A';
            case 8
                let='B';
            case 7
                let='C';
            case 6
                let='D';
            otherwise
                let='F';
        end
    end
end
end
```



Instead of OR || expression, we can write as many equality probabilities as we want inside curly braces { }.

In this way, we can more easily make the operation work if it fulfills any condition among more than one option.

- There are many functions in MATLAB that test whether something is true; these functions have names starting with "is".
- As an example, let's see the **ismember** function :

```
>>a=1:2:10  
a =  
     1     3     5     7     9  
  
>> b=2;  
>> c=5;  
>> ismember(b,a)  
ans =  
     0  
  
>> ismember(c,a)  
ans =  
     1
```

«IS» Functions in MATLAB

- Let's use the `isletter` function :

```
Command Window

>> isletter(a)

ans =

    logical

     0

>> isletter('a')

ans =

    logical

     1
```

```
a=input('Please enter a letter:','s');
if isletter(a)
    disp('It is a letter. ');
else
    disp('It is not a letter!');
end
```



```
Command Window

>> Untitled5
Please enter a letter:k
It is a letter.
>> Untitled5
Please enter a letter:5
It is not a letter!
```

- Let's use the `isnumeric` function :

```
Command Window

>> isnumeric(pi)

ans =

    logical

     1

>> isnumeric(3+4i)

ans =

    logical

     1
```

- Let's use the **isempty** function :

```
Command Window
>> vector=[ ];
>> isempty(vector)

ans =

    logical

     1

>> vector=[vector 5];
>> isempty(vector)

ans =

    logical

     0
```

«IS» Functions in MATLAB

- **isbetween** function : Determines elements within date and time interval.

```
tf = isbetween(t,tlower,tupper)
```

```
tlower <= t <= tupper
```

Command Window

```
>> tlower=datetime(2014,9,13)

tlower =

    datetime

    13-Sep-2014

>> tupper='20-Sep-2014'

tupper =

    '20-Sep-2014'

>> t=tlower+caldays(2:2:10)

t =

    1x5 datetime array

    15-Sep-2014    17-Sep-2014    19-Sep-2014    21-Sep-2014    23-Sep-2014

>> tf=isbetween(t,tlower,tupper)

tf =

    1x5 logical array

    1    1    1    0    0
```

- **iskeyword** function determines whether input is MATLAB keyword

```
tf = iskeyword(txt)
iskeyword
```

```
Command Window

>> iskeyword('while')

ans =

    logical

     1

>> iskeyword('card')

ans =

    logical

     0
```

«IS» Functions in MATLAB

- **isspace** function determines which characters are space characters

TF = isspace(A)

Command Window

```
>> c='Gaziantep Uni. Mechanical Eng. Dept. ';  
>> isspace(c)
```

ans =

1×37 logical array

Columns 1 through 31

0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1

Columns 32 through 37

0 0 0 0 0 1

«IS» Functions in MATLAB

- **issymmetric** function determines if the matrix is symmetric or skew-symmetric

```
tf = issymmetric(A)
```

Command Window

```
>> A=[1 0 1i;0 1 0;-1i 0 1]
```

```
A =
```

```
1.0000 + 0.0000i    0.0000 + 0.0000i    0.0000 + 1.0000i  
0.0000 + 0.0000i    1.0000 + 0.0000i    0.0000 + 0.0000i  
0.0000 - 1.0000i    0.0000 + 0.0000i    1.0000 + 0.0000i
```

```
>> tf=issymmetric(A)
```

```
tf =
```

```
logical
```

```
0
```

```
>> A(3,1)=1i;
```

```
>> tf=issymmetric(A)
```

```
tf =
```

```
logical
```

```
1
```

Next week

Chapter 7

LOOPS 1