



Control Statements

Conditional statements enable MATLAB to make decisions. The process is similar to the way we (humans) make decisions.

A condition stated. If the condition is met, one set of actions is taken. If the condition is not met, either nothing is done, or a second set of actions is taken.

Example:

If I win the Lottery,
I will quit college, buy a new car, and go fishing.
If I do not win the Lottery,
I will study harder so that I can get a better job.

Selection statements that test the results of relational or logical functions or operators are the decision-making structures that allow the flow of command execution to be controlled.

Operator	Relationship or Logical Operation	Example
<code>==</code>	Equal to	<code>x == b</code>
<code>~=</code>	Not equal to	<code>k ~= 0</code>
<code><</code>	Less than	<code>t < 12</code>
<code>></code>	Greater than	<code>a > -5</code>
<code><=</code>	Less than or equal to	<code>7 <= f</code>
<code>>=</code>	Greater than or equal to	<code>(4+r/6) >= 2</code>
<code>~</code>	NOT– negates the logical value of an expression	<code>~(b < 4*g)</code>
<code>&</code>	AND – both expressions must evaluate to true for result to be true	<code>t > 0)&&(c == 5)</code>
<code> </code>	OR evaluate to true for result to be true – either expression must	<code>(p > 1) (m > 3)</code>



If Statement

The if statement is used to determine whether or not a statement or group of statements is to be executed

General form: in figure (1)

If conditional statement
 commands
end

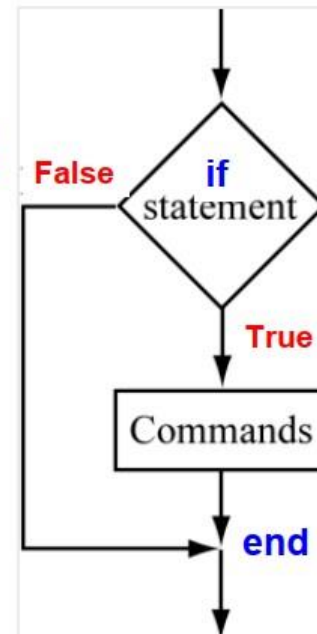


Figure 1

- the condition is any boolean expression
- the action is any number of valid statements (including, possibly, just one)
- if the condition is true, the action is executed – otherwise, it is skipped entirely

Example 1

A script file that demonstrates the use of the if-end statement. The user is asked to enter three grades. The program calculates the average of the grades. If the average is less than 60, a message: The student did not pass the course. is printed.

Solution

```
score = input('Enter (as a vector) the scores of the three tests: ');  
ave_grade = (score(1) + score(2) + score(3))/3;  
disp('The average grade is: ')
```



```
disp(ave_grade)
    if ave_grade < 60
        disp('The student did not pass the course.')
    end
```

If-else Statement

The if-else statement chooses between two actions

General form: in figure (2)

```
if conditional statement
```

```
    command group 1
```

```
else
```

```
    command group 2
```

```
end
```

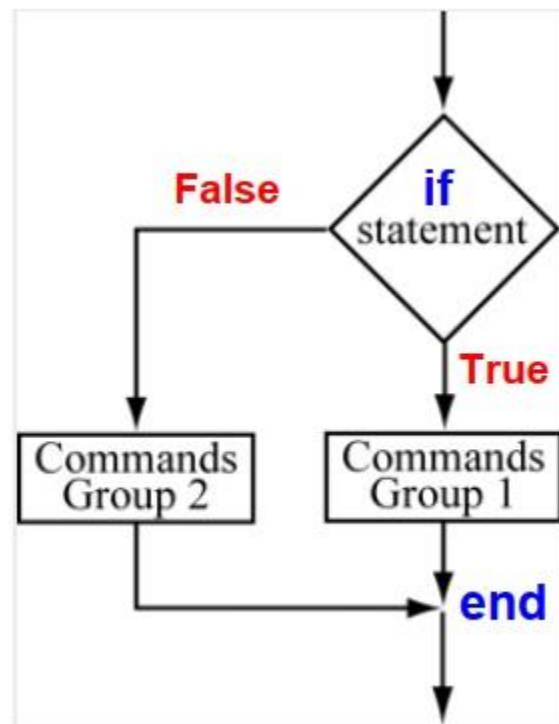


Figure 2

One and only one action is executed; which one depends on the value of the condition (action1 if it is logical true or action2 if it is false)

Example 2

A script file that demonstrates the use of the **if-else-end** statement. The user is asked to enter three grades. The program calculates the average of the grades. If the average is less than 60, a message: The student did not pass the course. is printed. Otherwise, a message: The student passed the course. is printed.



Solution

```
score = input('Enter (as a vector) the scores of the three tests ');  
ave_grade = (score(1) + score(2) + score(3))/3;  
disp('The average grade is: ')  
disp(ave_grade)  
if ave_grade < 60  
    disp('The student did not pass the course.')  
else  
    disp('The student passed the course.')  
end
```

If-elseif Statement

MATLAB has an elseif clause which shortens nested if-else

General form: in figure (3)

```
if conditional statement 1  
    command group 1  
elseif conditional statement 2  
    command group 2  
else  
    command group 3  
end
```

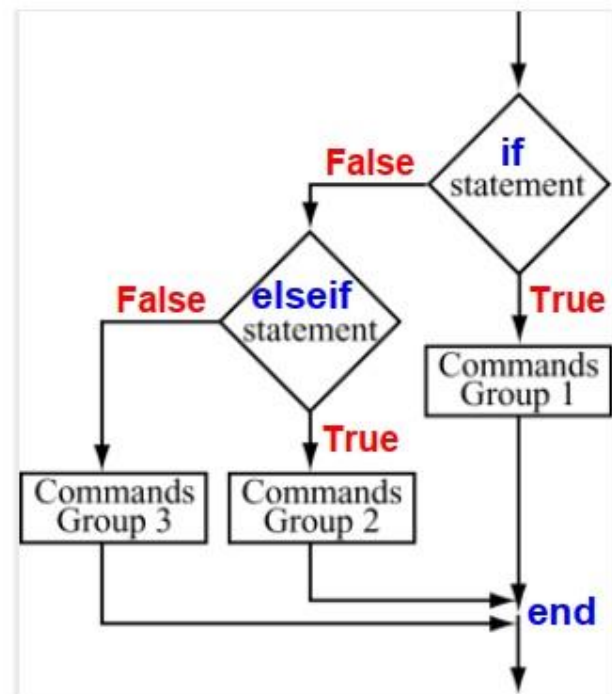


Figure 3

Example 3



A script file that demonstrates the use of the **if-elseif-else-end** statement. The program calculates the tip in a restaurant according to the amount of the bill. If the bill is less than 10\$ the tip is \$1.80. Between \$10 and \$60 the tip is 18% of the bill. Above \$60 the tip is 20% of the bill

Solution

```
bill = input('Enter the amount of the bill (in dollars): ');  
if bill <= 10  
    tip = 1.8;  
elseif (bill > 10) & (bill <= 60)  
    tip = bill*0.18;  
else  
    tip = bill*0.2;  
end  
disp('The tip is (in dollars):')  
disp(tip)
```

Switch Selection Structure

The switch selection structure provides an alternative to using the **if**, **elseif**, and **else** commands. Anything programmed using if structures can also be programmed using switch structures. The advantage of the switch structure is that in some situations, it yields code that is more readable.

A **switch** block conditionally executes one set of statements from several choices. Each choice is covered by a **case** statement. The switch block tests each case until one of the cases is *true*.

When a **case** is *true*, MATLAB executes the corresponding statements and then *exits* the switch block. The **otherwise** block is *optional* and executes only when *no case* is *true*. The *syntax* of **switch** statement in MATLAB is:

```
switch <switch_expression>
```

```
case <case_expression>
```



```
<statement(s)>  
case <case_expression>  
    <statement(s)>  
otherwise  
    <statement(s)>  
end
```

Example 4

```
clc  
n = -1  
switch n  
    case -1  
        disp('negative one')  
    case 0  
        disp('zero')  
    case 1  
        disp('one')  
    otherwise  
        disp('something else')  
end
```

More examples

EX5: If Statement

Once you've checked that it is working correctly, change the values of x and y to zero, in figure (4):

```
x=5;  
y=10;  
if x>0
```



```
if y>0
    z=x/y;
end
disp(z)
```

EX6: If-else Statement

Once you've checked that it is working correctly,
change the value of x to 50, in figure (5)

```
x=50;
y=10;
if x<y
    z=x*y;
else
    z=x/y;
end
disp(z)
```

EX7: If-elseif Statement

Once you've checked that it is working correctly, change the value of x to 5 and then to 7, in figure (6):

```
x=-1;
y=2;
if x<0
    z=x*y;
elseif x>0 & x<6
    z=x/y;
else
    z=x^y;
end
disp(z)
```

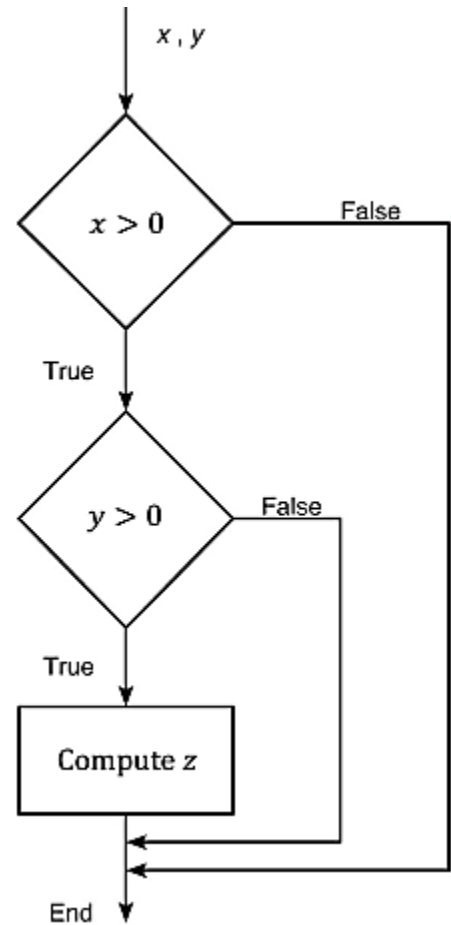


Figure 4

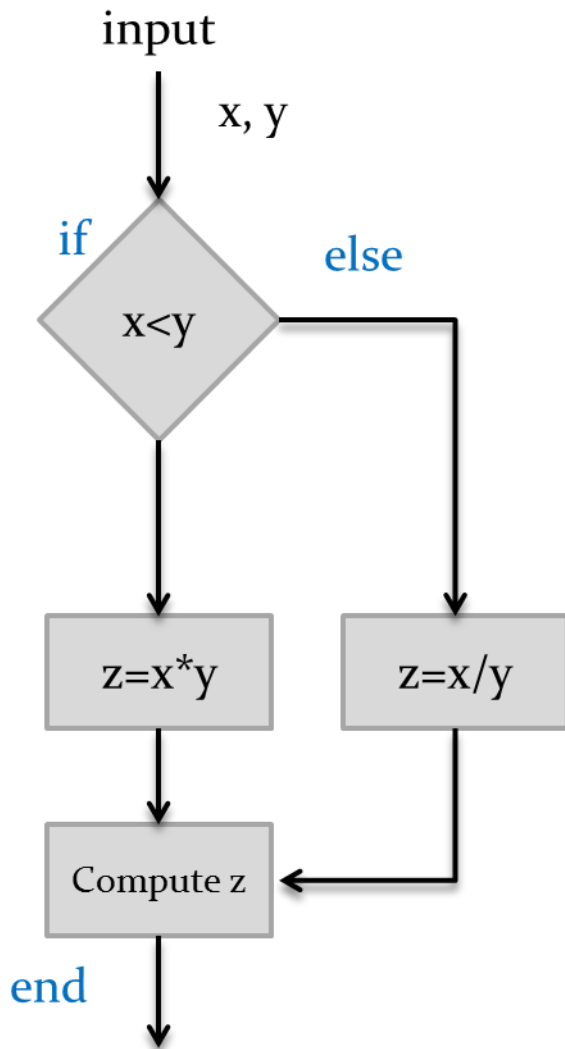


Figure 5

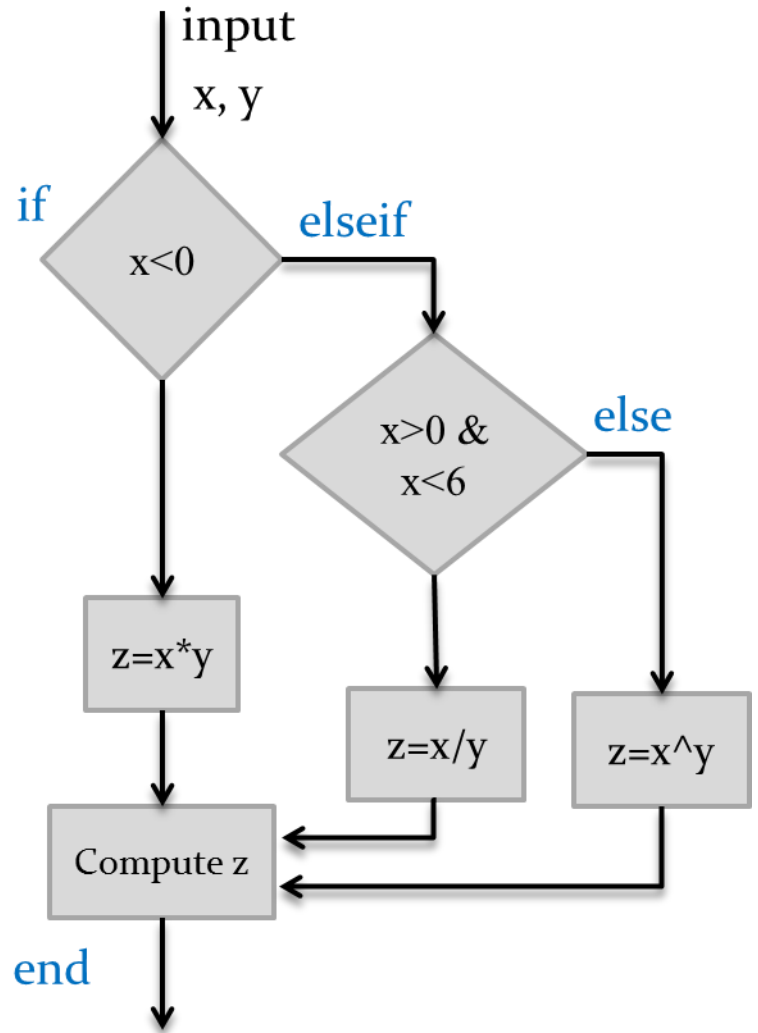


Figure 6



EX8: Switch Statement

A=2;

Switch A

case A==1

B=2

case A==2

B=8

case A==3

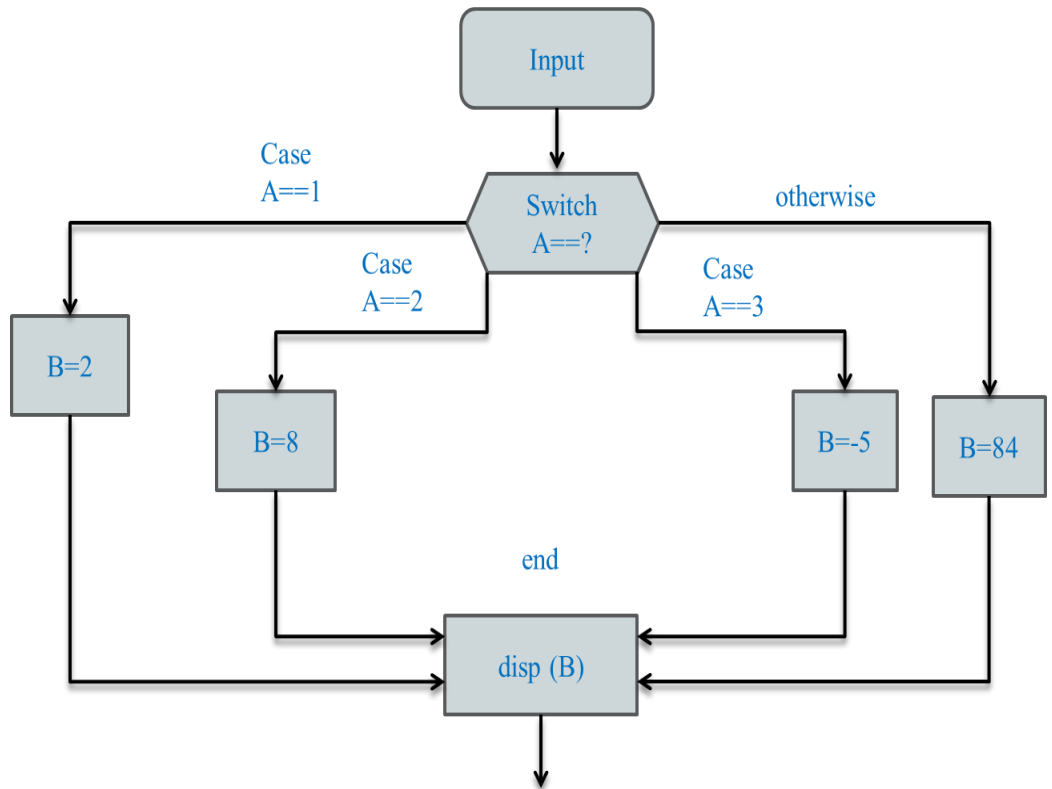
B= -5

otherwise

B= 84

end

disp (B)



H.W 5

Write a script file using Conditional **If-Elseif-Else** statements to evaluate the following function, assuming that $x = -2, 0,$ and 6 . The function is:

$$y = \begin{cases} e^{x+1} & \text{for } x < -1 \\ 2 + \cos(\pi x) & \text{for } -1 \leq x \leq 5 \\ 10(x - 5) + 1 & \text{for } x > 5 \end{cases}$$