



Lecture 5: Conditional Statements & Loops in Matlab

1- Conditional Statements (if, elseif and else)

Syntax

```
if expression
statements
elseif expression
statements
else
statements
end
```

Exp1: Let's consider that FIRAS is a student at the Biomedical Engineering department. He had an exam in mathematics, and we would like to evaluate his mark.

Means: ≥ 90 : Excellent

≥ 80 : Very good

≥ 70 : Good

≥ 50 : Passed

< 50 : Failed

Using: **if, elseif and else**

```
FIRASMARK = randi(100,1);
if FIRASMARK >= 90
disp ('EXCELLENT')
elseif FIRASMARK >= 80
disp ('VERY GOOD')
elseif FIRASMARK >= 70
disp ('GOOD')
elseif FIRASMARK >= 50
disp ('PASSED')
else
disp ('FAILED')
end
```

Exp2: In hospitals, the heart rate monitors used to measure/display the heart rate in real time. Program it to give an alarm in low and/or high heart rate.

Means: ≥ 100 : Dangerous very high
 ≥ 90 : High heart rate
 ≥ 40 : Well
 ≥ 35 : Low heart rate
 < 35 : Dangerous very low

```
heartrate = randi(200,1);  
if heartrate >= 100  
    disp ('PEEP DANGEROUS VERY HIGHT')  
elseif heartrate >= 90  
    disp ('HIGH HEART RATE')  
elseif heartrate >= 40  
    disp ('WELL')  
elseif heartrate >= 35  
    disp ('LOW HEART RATE')  
else  
    disp ('PEEP DANGEROUS VERY LOW')  
end
```

2-Loops in Matlab (while loop, for loop, nested loops)

While loop:

The **while loop** repeatedly executes statements while a specified condition is true.

Syntax

```
while <expression>  
    <statements>  
end
```

Exp.1:

```
n = 1;  
while n > 0  
    n = 5  
end
```

Exp.2:

```
n=10;  
while n > 0  
    n=n-1  
end
```

Exp.3: Program to find the number 8 from a series of random numbers using **while** loop

```
n = 1;
while n
if randi(50,1) == 8
disp(['The random number equivalent to 8 found at ', num2str(n), '
step'])
break
end
n = n + 1;
end
```

Output:

The random number equivalent to 8 found at step ??

for loop:

A **for loop** is used to repeat a statement or a group of statements for a fixed number of times.

Syntax

```
for index = values
<program statements>
...
end
```

Exp1:

```
for i = (1:5);
q = i * 2
end
```

Exp2:

```
for i = (1:5);
disp (i)
end
```

Nested loop: MATLAB also allows using one loop inside another loops.

Exp1:

```
r = zeros(10,10)
for i = 1:10
r(i,1) = i
end
```

Exp2:

```
r = zeros(10,10)
for i = 1:10
for j = 1:10
r (i,j) = i
end
end
```