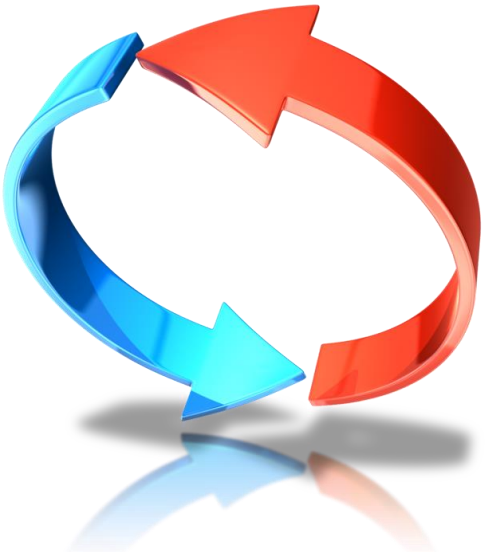
Module 04: Loop Statement



Chul Min Yeum
Assistant Professor
Civil and Environmental Engineering
University of Waterloo, Canada

UNIVERSITY OF WATERLOO
FACULTY OF ENGINEERING

```
mat1 = [1 2 3; 4 5 6; 7 8 9];

val = 0;
for ii=1:3
    val = val + mat1(ii,ii);
end
```

M04-Q1: Which of the best describe the following script?

```
mat1 = [1 2 3; 4 5 6; 7 8 9];

val = 0;
for ii=1:3
    val = val + mat1(ii,ii);
end
```

1) Adding all numbers in *mat1*
2) Adding all numbers at a diagonal locations in *mat1*
3) Adding all numbers at the first column of *mat1*
4) Adding all numbers at the first row of *mat1*

```
mat1 = [1 2 3 4; 5 6 7 8; 9 10 11 12];

[nr, nc] = size(mat1);

val = 0;
for ii=1:2:nc
    for jj=1:nr
        val = val + mat1(jj, ii);
    end
end
```

M04-Q2: Which of the best describe the following script?

```
mat1 = [1 2 3 4; 5 6 7 8; 9 10 11 12];

[nr, nc] = size(mat1);

val = 0;
for ii=1:2:nc
    for jj=1:nr
        val = val + mat1(jj, ii);
    end
end
```

1) Adding all number in *mat1*
2) Adding all odd numbers in *mat1*
3) Adding all numbers at even column locations in *mat1*
4) Adding all numbers at odd column locations in *mat1*

```
vec = 1:25
mat1 = zeros(5, 5);

for ii=1:numel(vec)
    mat1(ii) = vec(end-ii+1)
end
```

25	24	23	22	21
20	19	18	17	16
15	14	13	12	11
10	9	8	7	6
5	4	3	2	1

(A)

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

(B)

25	20	15	10	5
24	19	14	9	4
23	18	13	8	3
22	17	12	7	2
21	16	11	6	1

(C)

M04-Q3: What is the array finally assigned to *mat1*?

```
vec = 1:25
mat1 = zeros(5, 5);

for ii=1:numel(vec)
    mat1(ii) = vec(end-ii+1)
end
```

★★

1) A
2) B
3) C
4) No answer

25	24	23	22	21
20	19	18	17	16
15	14	13	12	11
10	9	8	7	6
5	4	3	2	1

(A)

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

(B)

25	20	15	10	5
24	19	14	9	4
23	18	13	8	3
22	17	12	7	2
21	16	11	6	1

(C)

```
vec = 1:25
mat1 = zeros(5, 5);

count = 1;
for ii=1:5
    for jj=1:5
        mat1(ii, jj) = vec(count);
        count = count + 1;
    end
end
```

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

(A)

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

(B)

25	24	23	22	21
20	19	18	17	16
15	14	13	12	11
10	9	8	7	6
5	4	3	2	1

(C)

M04-Q4: What is the array finally assigned to *mat1*?

```
vec = 1:25
mat1 = zeros(5, 5);

count = 1;
for ii=1:5
    for jj=1:5
        mat1(ii, jj) = vec(count);
        count = count + 1;
    end
end
```

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

(B)

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

(A)

25	24	23	22	21
20	19	18	17	16
15	14	13	12	11
10	9	8	7	6
5	4	3	2	1

(C)

1) A
2) B
3) C
4) No answer

```
vec = [10 11 55 33 22 11 20 10]

nvec = numel(vec);
val = 0;

for ii=1:nvec

    if rem(vec(ii), 2) == 1
        val = val+vec(ii);
    end
end
```

M04-Q5: Which of the best describe the following script?

```
vec = [10 11 55 33 22 11 20 10]

nvec = numel(vec);
val = 0;

for ii=1:nvec

    if rem(vec(ii), 2) == 1
        val = val+vec(ii);
    end
end
```

1) Adding all numbers in *vec*
2) Adding all odd numbers in *vec*
3) Adding all even numbers in *vec*
4) Adding all numbers at odd locations in *vec*

```
mat1 = [1 10 11 14; 2 3 42 3; 0 1 4 5]
[nr, nc] = size(mat1);

mat_out = [];

for ii=1:nr
    for jj=1:nc
        if rem(mat1(ii,jj), 2) == 1
            mat_out = [mat_out; [ii jj]];
        end

    end
end
```

★★

M04-Q6: Which of the best describe the values in *mat_out*?

```
mat1 = [1 10 11 14; 2 3 42 3; 0 1 4 5]
[nr, nc] = size(mat1);

mat_out = [];

for ii=1:nr
    for jj=1:nc
        if rem(mat1(ii,jj), 2) == 1
            mat_out = [mat_out; [ii jj]];
        end

    end
end
```

★★

1) even values in *mat1*
2) odd values in *mat1*
3) subscript indexes of odd values in *mat1*
4) linear indexex of odd values in *mat1*

```
isTask = true;

for ii=1:numel(vec)-1
    if vec(ii) > vec(ii+1)
        isTask = false;
        break
    end
end
```

★★★

M04-Q7: Which of the best describe the following script?

```
isTask = true;

for ii=1:numel(vec)-1
    if vec(ii) > vec(ii+1)
        isTask = false;
        break
    end
end
```

1) Check if the values in *vec* are in an ascending order.
2) Check if the values in *vec* are in a descending order.
3) Check if the maximum value in *vec* is not a single.
4) Check if there are the same values in *vec*.

```
vec = [1 3 5 3 2 5 3 5 6];

num_3 = 0;
loc = 1;

while num_3 ~= 3
    if vec(loc) == 3
        num_3 = num_3 + 1;
    end
   loc = loc + 1;
end
```

★★★

M04-Q8: What is a value finally assigned to *loc* ?

```
vec = [1 3 5 3 2 5 3 5 6];

num_3 = 0;
loc = 1;

while num_3 ~= 3
    if vec(loc) == 3
        num_3 = num_3 + 1;
    end
   loc = loc + 1;
end
```

1) 2
2) 4
3) 7
4) 8

```
vec = [1 3 5 3 2 5 3 5 6];

num_3 = 0;

for ii = 1:numel(vec)
    if vec(ii) == 3
        num_3 = num_3 + 1;
        loc = ii;
    end
    if num_3 == 2
        break
    end
end
```

★★★

M04-Q9: What value is assigned to ‘loc’?

```
vec = [1 3 5 3 2 5 3 5 6];

num_3 = 0;

for ii = 1:numel(vec)
    if vec(ii) == 3
        num_3 = num_3 + 1;
        loc = ii;
    end
    if num_3 == 2
        break
    end
end
```

★★★

1) 2
2) 4
3) 7
4) 8

```
vec = 1:25;
mat = reshape(vec,5,5);

count = 1;
for ii = 1:5
    for jj = 1:5
        mat(jj,ii) = count^2
        count = count + 1;
    end
end
```

M04-Q10: What is the best description of the resulting '`mat`'?

```
vec = 1:25;
mat = reshape(vec,5,5);

count = 1;
for ii = 1:5
    for jj = 1:5
        mat(jj,ii) = count^2
        count = count + 1;
    end
end
```

1) Assign the squared linear index value to each corresponding location in 'mat'
2) Assign the square of (column + row) index values to respective locations
3) Square every value in 'mat'
4) None of the other answers are a good description

```
char_vec = 'aaabbbcccdddeeeaaabbb';

nchar = numel(char_vec);

val = 0;
for ii=1:nchar

    isa = char_vec(ii) == 'a';
    isb = char_vec(ii) == 'b';

    if or(isa, isb)
        val = val + 1;
    end
end
```

★★

M04-11: What value is assigned to ‘`val`’?

```
char_vec = 'aaabbbcccdddeeeaaabbb';

nchar = numel(char_vec);

val = 0;
for ii=1:nchar

    isa = char_vec(ii) == 'a';
    isb = char_vec(ii) == 'b';

    if or(isa, isb)
        val = val + 1;
    end
end
```

★★

1) 6
2) 10
3) 12
4) 16

Given

```
m = randi(10, 5, 5)
```

```
for ii=1:5
    for jj=1:5
        if m(ii,jj) == 10
            loc = 10;
        else
            loc = 0;
        end
    end
end
```

(A)

```
loc = 0;
for ii=1:5
    for jj=1:5
        if m(ii,jj) == 10
            loc = 10;
            break;
        end
    end
end
```

(C)

```
loc = 0;
isrun = false;
for ii=1:5
    for jj=1:5
        if m(ii,jj) == 10
            loc = 10;
            isrun = true;
            break;
        end
    end
    if isrun
       break;
    end
end
```

(B)

```
loc = 0;
for ii=1:25
    if m(ii) == 10
        loc = 10;
        break;
    end
end
```

(D)

M04-12: Which of the scripts produce a different 'loc' value?

Given

```
m = randi(10, 5, 5)
```

```
for ii=1:5
    for jj=1:5
        if m(ii,jj) == 10
            loc = 10;
        else
            loc = 0;
        end
    end
end
```

(A)

```
loc = 0;
for ii=1:5
    for jj=1:5
        if m(ii,jj) == 10
            loc = 10;
            break;
        end
    end
end
```

(C)

```
loc = 0;
isrun = false;
for ii=1:5
    for jj=1:5
        if m(ii,jj) == 10
            loc = 10;
            isrun = true;
            break;
        end
    end
    if isrun
       break;
    end
end
```

(B)

```
loc = 0;
for ii=1:25
    if m(ii) == 10
        loc = 10;
        break;
    end
end
```

(D)

1) A
2) B
3) C
4) D