

```
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
```

- **Answer:** Adding all numbers at diagonal locations in mat1
- **Comment:** Review “For-Loop Statement” (p.110) and an example of “Summation of Values in a Matrix” (p.115).

```
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
```

- **Answer:** Adding all numbers at odd column locations in mat1
- **Comment:** Review “For-Loop Statement” (p.110) and an example of “Summation of Values in a Matrix” (p.115).

```
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
```



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)

- **Answer: C**
- **Comment:** Review “For-Loop Statement” (p.110) and an example of “Summation of Values using Linear Indexing” (p.119).

```

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)



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

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)



- **Answer: B**
- **Comment:** Review “Nested For-Loop Statement” (p.114) and an example of “Summation of Values in a Matrix” (p.115-118).


```
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
```

- **Answer:** Adding all odd numbers in vec
- **Comment:** Review “Combining for-loop and if-else Statement” (p.123) and an example of “Example: Combining for-loop and If-elseif Statement” (p.124).

```
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
```



- **Answer:** subscript indexes of odd values in mat1
- **Comment:** Review an example of “Example: Find a Value” (p.126-7).

```
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
```

- **Answer:** Check if the values in vec are in an ascending order.
- **Comment:** Review “Break” (p. 131) and an example of “Example: Use Break” (p.132).

```
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
```

- **Answer:** 8
- **Comment:** Review “While Loop” (p.133) and an example of “Example: Summation” (p.134).


```
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
```



- **Answer: 4**
- **Comment:** Review “Break” (p.131) and example “Use Break” (p.132).

```
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
```

- **Answer:** Assign the squared linear index value to each corresponding location in 'mat'
- **Comment:** Review “Summation of Values using Linear Indexing” (p.119).

```
char_vec = 'aaabbbbccccdddeeeaaabbbb';  
  
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 = 'aaabbbcccdddeeeaaabb';  
  
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
```



- **Answer:** 12
- **Comment:** Review example “Use Break” (p.132).

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
```



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