

```
% Hassan El-seoudy
% ID : 3780 % GP4 : S1 "Odd"
https://github.com/Hassan-Elseoudy/DSP\_MATLAB/blob/master/Assignment1.m
% Generate the following sequence: V=1 4 9 16 25 ..... 16 9 4
```

```
v = zeros(1,49);
for i = 1:1:25
    v(i) = (i*i);
    if (i >= 25)
        for j = 24 : -1 : 1
            i++;
            v(i) = (j*j);
        end
    end
end
disp(v)
```

Columns 1 through 13:

```
1    4    9   16   25   36   49   64   81  100  121  144  169
```

Columns 14 through 26:

```
196  225  256  289  324  361  400  441  484  529  576  625  576
```

Columns 27 through 39:

```
529  484  441  400  361  324  289  256  225  196  169  144  121
```

Columns 40 through 49:

```
100   81   64   49   36   25   16    9    4    1
```

```
% Add 2 to the last 3 elements. %
```

```
v ([end-2 end-1 end]) = [v(end-2)+2 v(end-1)+2 v(end)+2];
```

Columns 1 through 13:

```
1    4    9   16   25   36   49   64   81  100  121  144  169
```

Columns 14 through 26:

```
196  225  256  289  324  361  400  441  484  529  576  625  576
```

Columns 27 through 39:

```
529  484  441  400  361  324  289  256  225  196  169  144  121
```

Columns 40 through 49:

```
100   81   64   49   36   25   16   11    6    3
```

```
% Reverse the order of the last 10 elements. %
```

```
count = 0;
```

```
for i = 39:1:44
```

```
    temp = v(i);
```

```
    v(i) = v(end - count);
```

```
    v(end - count) = temp;
```

```
    count+= 1;
```

```
end
```

Columns 1 through 13:

1	4	9	16	25	36	49	64	81	100	121	144	169
---	---	---	----	----	----	----	----	----	-----	-----	-----	-----

Columns 14 through 26:

196	225	256	289	324	361	400	441	484	529	576	625	576
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Columns 27 through 39:

529	484	441	400	361	324	289	256	225	196	169	144	3
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	---

Columns 40 through 49:

6	11	16	25	36	49	64	81	100	121
---	----	----	----	----	----	----	----	-----	-----

% For the first 48 elements, add the elements in the even places to that in
% the odd places and store the output in the odd places.

```
for i = 1:2:48  
v(i) += v(i+1);  
End
```

Columns 1 through 11:

5	4	25	16	61	36	113	64	181	100	265
---	---	----	----	----	----	-----	----	-----	-----	-----

Columns 12 through 22:

144	365	196	481	256	613	324	761	400	925	484
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Columns 23 through 33:

1105	576	1201	576	1013	484	841	400	685	324	545
------	-----	------	-----	------	-----	-----	-----	-----	-----	-----

Columns 34 through 44:

256	421	196	313	144	9	6	27	16	61	36
-----	-----	-----	-----	-----	---	---	----	----	----	----

Columns 45 through 49:

113	64	181	100	121
-----	----	-----	-----	-----

```
m = [1 2 3 4;-1 -2 -3 -4;1 2 3 4;-1 -2 -3 -4];  
% Reflect array (M) left-side right %  
m = fliplr(m);  
disp (m)
```

4	3	2	1
-4	-3	-2	-1
4	3	2	1
-4	-3	-2	-1

```
m = [1 2 3 4;-1 -2 -3 -4;1 2 3 4;-1 -2 -3 -4];  
% Reflect array (M) upside down %  
m = flipud(m);  
disp (m)
```

```
-1 -2 -3 -4
1 2 3 4
-1 -2 -3 -4
1 2 3 4
```

```
m = [1 2 3 4;-1 -2 -3 -4;1 2 3 4;-1 -2 -3 -4]; % Swap columns 2 and 3 of array (M) %
x = zeros(4,1);
x = m ([1 2 3 4],2);
m ([1 2 3 4],2) = m ([1 2 3 4],3);
m ([1 2 3 4],3) = x;
disp(m)
```

```
1 3 2 4
-1 -3 -2 -4
1 3 2 4
-1 -3 -2 -4
```

```
m = [1 2 3 4;-1 -2 -3 -4;1 2 3 4;-1 -2 -3 -4]; % Swap rows 1 and 4 of array (M) %
x = zeros(1,4);
x = m (1,[1 2 3 4]);
m (1,[1 2 3 4]) = m (4,[1 2 3 4]);
m (4,[1 2 3 4]) = x;
disp(m)
```

```
-1 -2 -3 -4
-1 -2 -3 -4
1 2 3 4
1 2 3 4
```

```
m = [1 2 3 4;-1 -2 -3 -4;1 2 3 4;-1 -2 -3 -4]; % Shuffle (M) from [1 2 3 4] to [1 3 4 2] %
m_new ([1 2 3 4],:) = m([1 3 4 2],:);
m = m_new;
disp(m)
```

```
1 2 3 4
1 2 3 4
-1 -2 -3 -4
-1 -2 -3 -4
```

```
m = [1 2 3 4;-1 -2 -3 -4;1 2 3 4;-1 -2 -3 -4];% shuffle (M) from [1 2 3 4] to [3 2 4 1]. %
m_new (:,[1 2 3 4]) = m(:, [3 2 4 1]);
m = m_new;
disp(m)
```

```
3 2 4 1
-3 -2 -4 -1
3 2 4 1
-3 -2 -4 -1
```