Exercise 1

A discrete-time signal: $x(n) = \begin{cases} 1 + n/3, & -3 \le n \le -1 \\ 1, & 0 \le n \le 3 \\ 0, & elsewhere \end{cases}$

1. Determine its values and sketch the signal $x(n)$

Case 1.

x(n) = 1 + n/3

n(-3) = 1 + -3/3 = 0

n(-2) = 1 + -2/3 = 1/3

n(-1) = 1 + -1/3 = 2/3

Case 2.

n(0) = 1

n(1) = 1

n(2) = 1

n(3) = 1

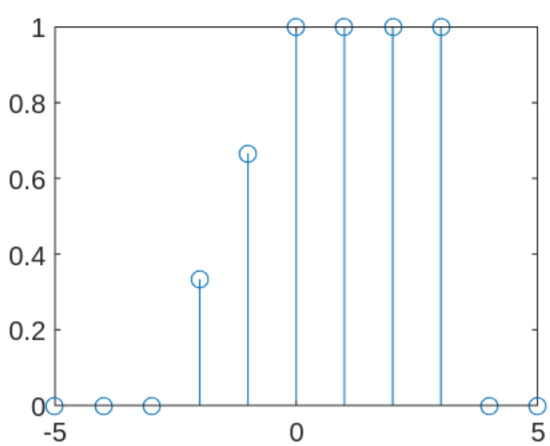
```
x = [-5:5]
```

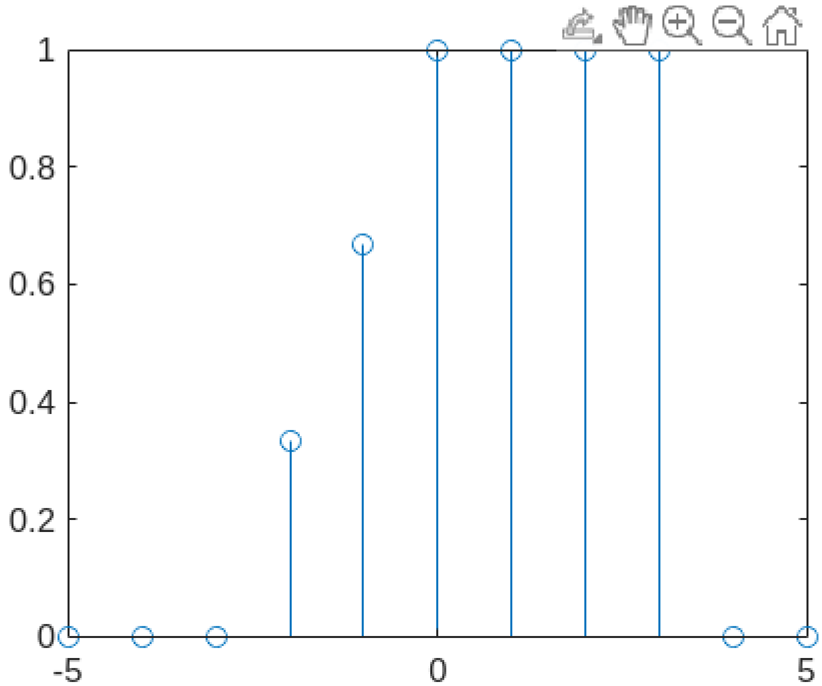
```
x = 1×11
    -5    -4    -3    -2    -1     0     1     2     3     4     5
```

```
y = [0, 0, 0, 1/3, 2/3, 1, 1, 1, 1, 0, 0]
```

```
y = 1×11
         0         0         0    0.3333    0.6667    1.0000    1.0000    1.0000 ···
```

```
stem(x, y)
```





2. Sketch the signals that result if we:

- First fold $x(n)$ and then delay the resulting signal by four samples

Fold signal

y = [0, 0, 0, 1/3, 2/3, 1, 1, 1, 1, 0, 0, 0, 0]

x = [-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7]

y = [0, 0, 0, 0, 1, 1, 1, 1, 2/3, 1/3, 0, 0, 0]

x = [-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7]

Delay four samples

y = [1, 1, 1, 1, 2/3, 1/3, 0, 0, 0, 0, 0, 0, 0]

x = [-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6]

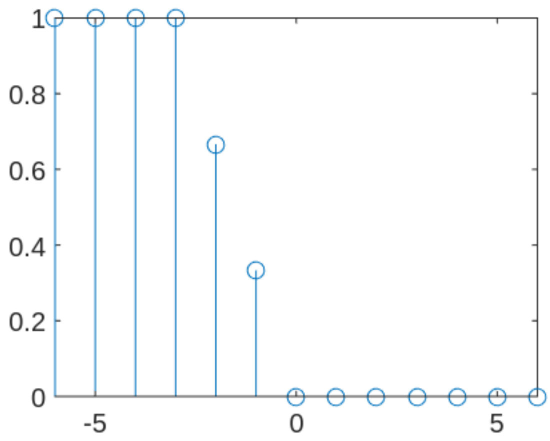
```
x = [-6:6]
```

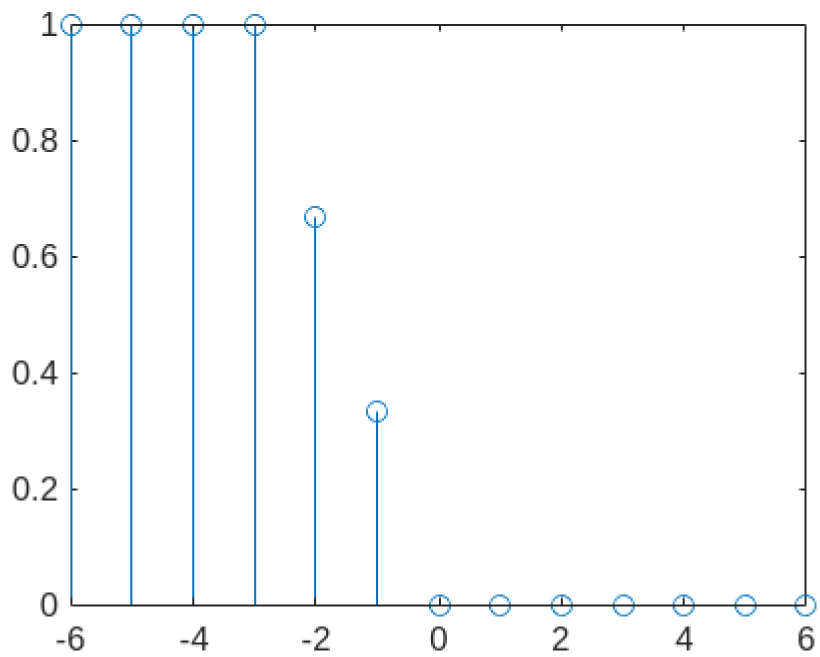
```
x = 1×13
    -6    -5    -4    -3    -2    -1     0     1     2     3     4     5     6
```

```
y = [1, 1, 1, 1, 2/3, 1/3, 0, 0, 0, 0, 0, 0, 0]
```

```
y = 1×13
    1.0000    1.0000    1.0000    1.0000    0.6667    0.3333         0         0···
```

```
stem(x, y)
```





- First delay $x(n)$ four samples and then fold.

Delay four samples

y = [0, 0, 0, 0, 1/3, 2/3, 1, 1, 1, 1, 0, 0, 0]

x = [-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6]

y = [1/3, 2/3, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0]

x = [-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6]

Fold the signal

y = [1/3, 2/3, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0]

x = [-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6]

y = [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 2/3, 1/3]

x = [-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6]

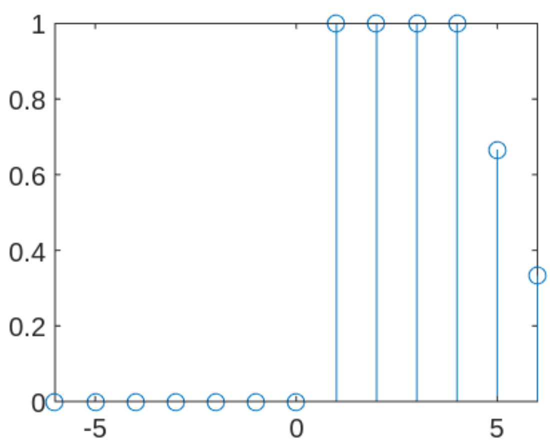
```
x = [-6:6]
```

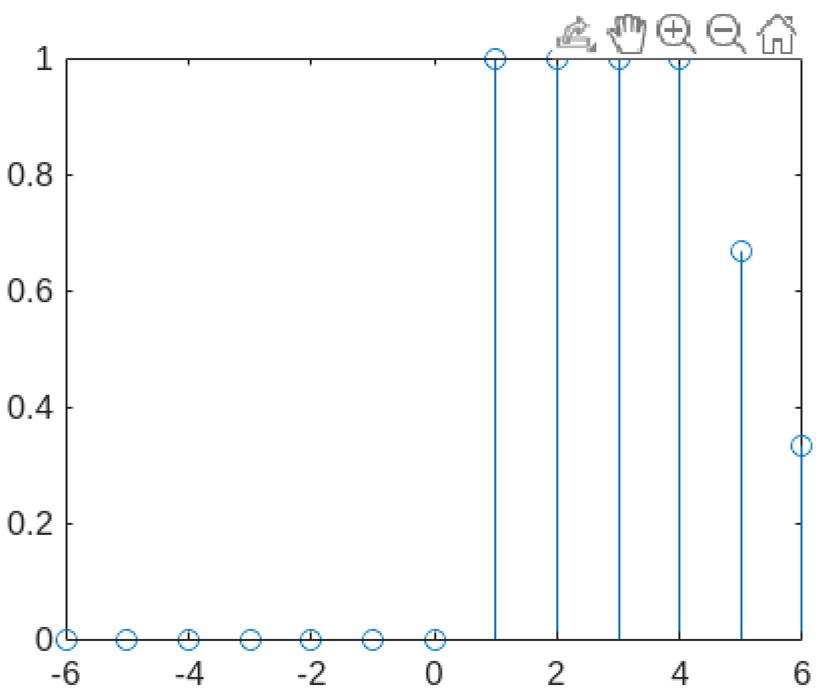
```
x = 1×13
    -6    -5    -4    -3    -2    -1     0     1     2     3     4     5     6
```

```
y = [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 2/3, 1/3]
```

```
y = 1×13
         0         0         0         0         0         0         0    1.0000 ···
```

```
stem(x, y)
```





3. Sketch the signal $x(-n+4)$

n=0; x(4); y = 0

n=1; x(3); y = 1

n=2; x(2); y = 1

n=3; x(1); y = 1

n=4; x(0); y = 1

n=5; x(-1); y = 2/3

n=6; x(-2); y = 1/3

n=7; x(-3); y = 0

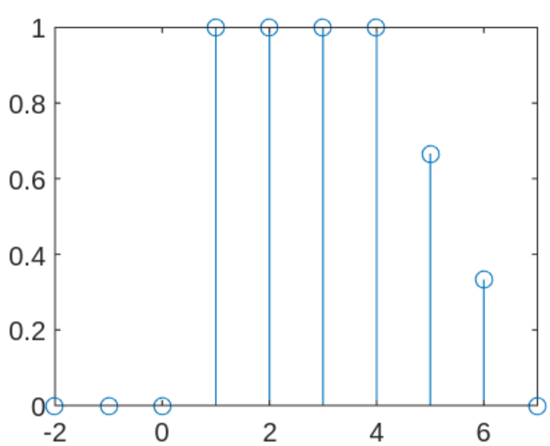
```
x = [-2:7]
```

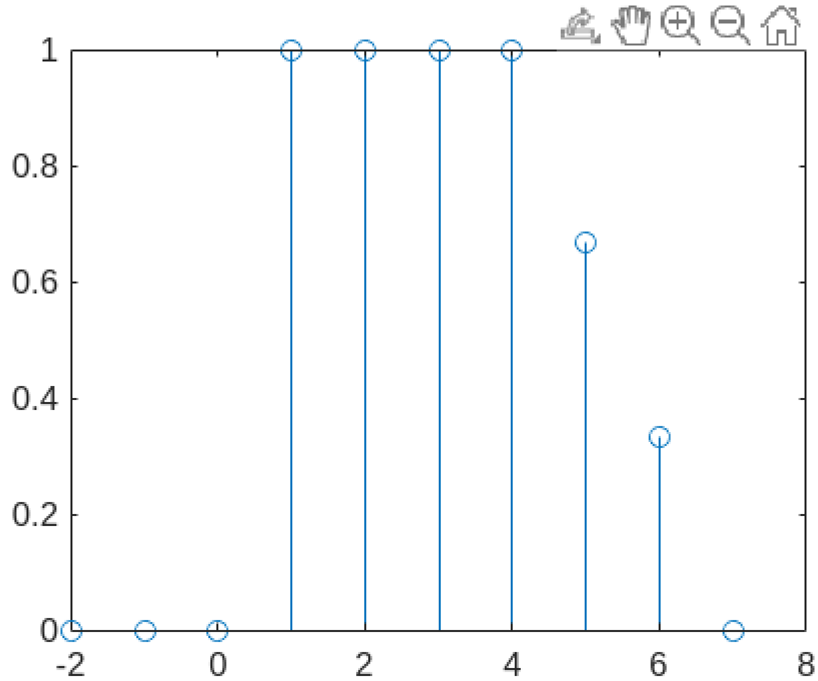
```
x = 1×10
    -2    -1     0     1     2     3     4     5     6     7
```

```
y = [0, 0, 0, 1, 1, 1, 1, 2/3, 1/3, 0]
```

```
y = 1×10
         0         0         0    1.0000    1.0000    1.0000    1.0000    0.6667 ···
```

```
stem(x, y)
```





4. Compare the results in parts b) and c) and derive a rule for obtaining the signal $x(-n+k)$ from $x(n)$

The rule would be :

$$x(-n+k) = \begin{cases} 1 + (-n+k)/3, & -3 \le -n+k \le -1 \\ 1, & 0 \le -n+k \le 3 \\ 0, & elsewhere \end{cases}$$

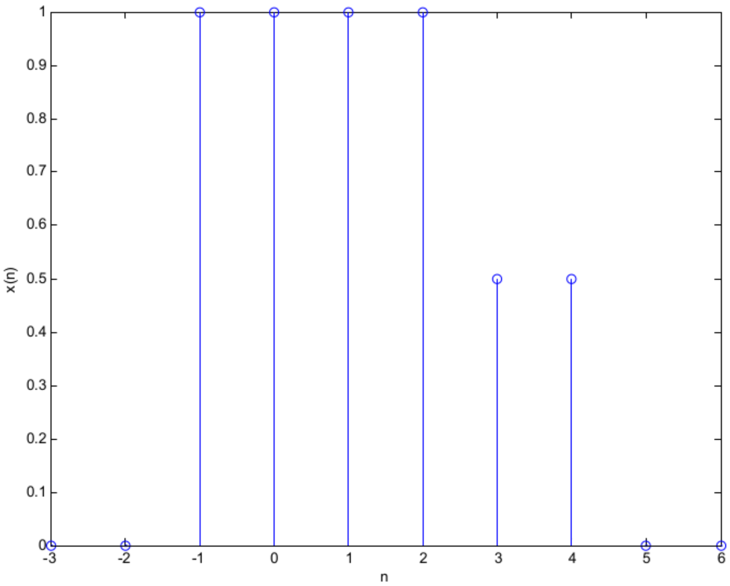
5. Can you express the signal in terms of $x(n)$ in terms of signals $\delta(n)$ and $u(n)$?

No. For $\delta(n)$ or even $\delta(n-k)$, the signal will result in a shifted impulse, instead of the signal described by $x(n)$

$\mu(n)$ and $\mu(n-k)$ also cannot describe $x(n)$ as will just shift a train of impulses.

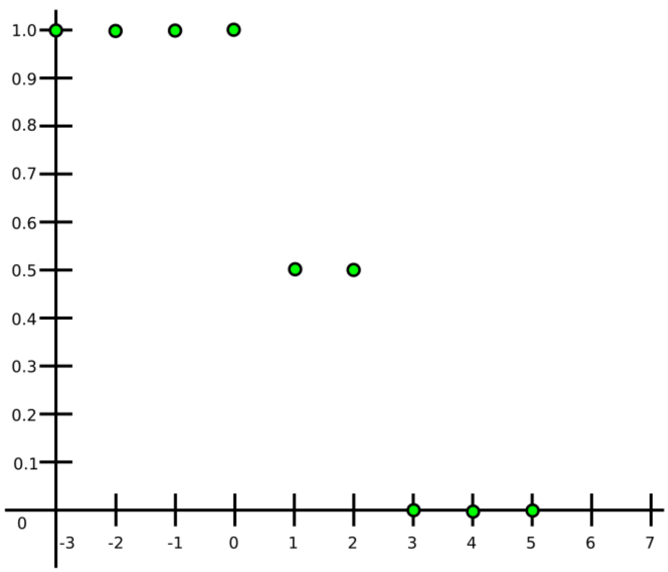
Exercise 2

A discrete-time signal $x(n)$ is shown in the next figure.

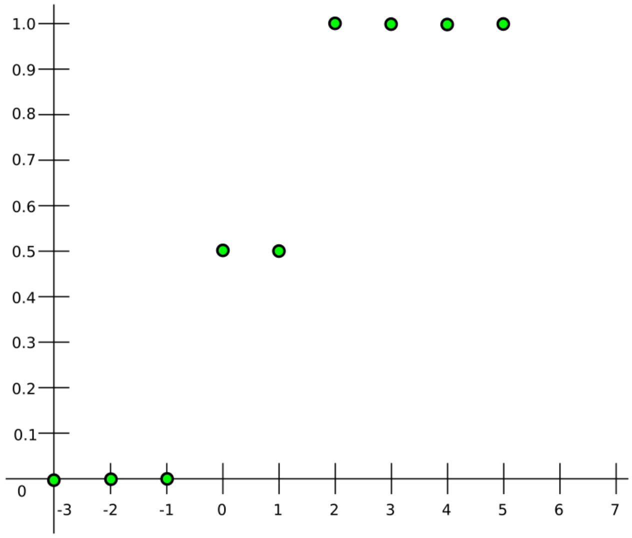


Sketch and label carefully each of the following signals:

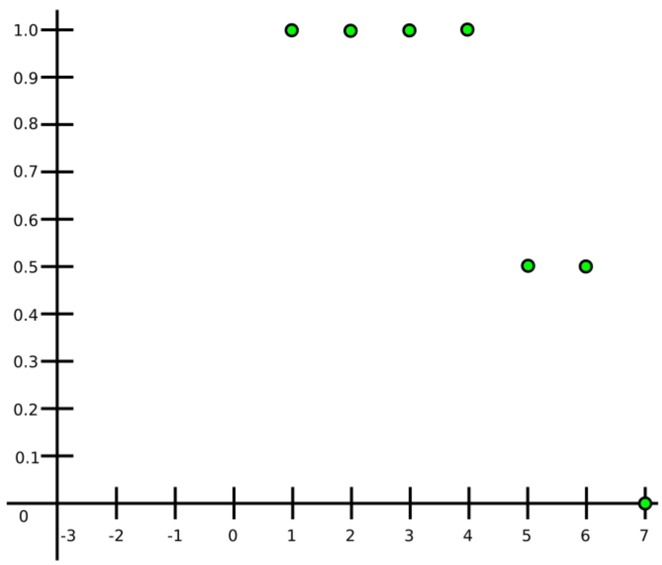
a. $x(n-2)$



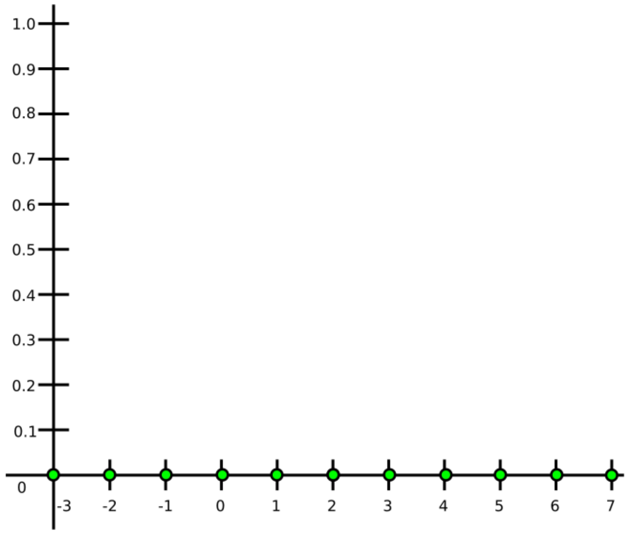
b. $x(4-n)$



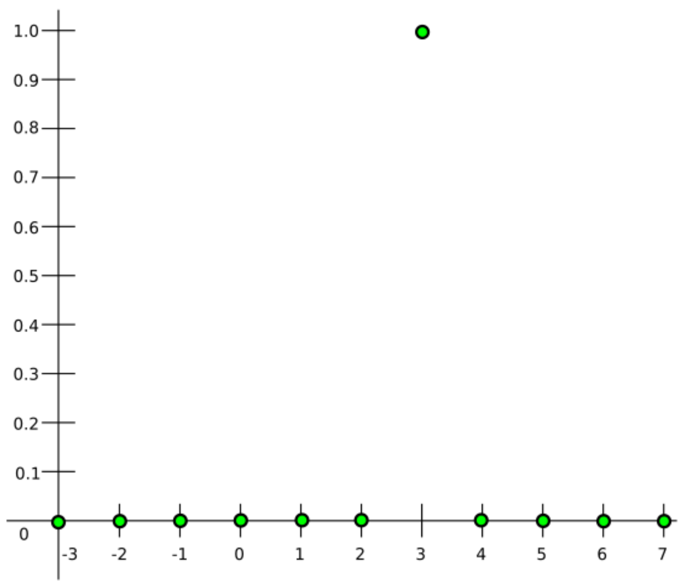
c. $x(n+2)$



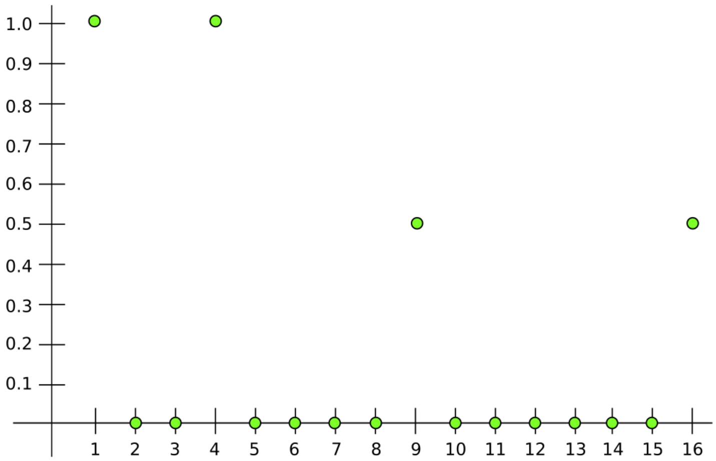
d. $x(n)\mu(2-n)$



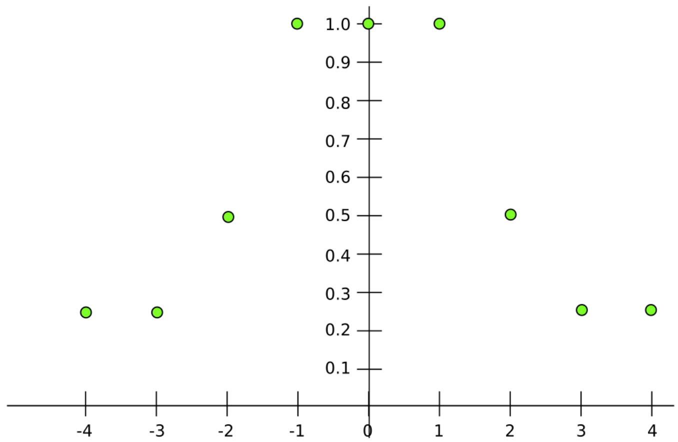
e. $x(n+1)\delta(n-3)$



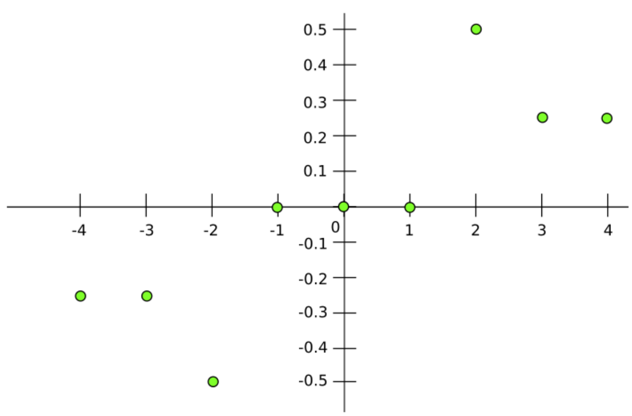
f. $x(n^2)$



g. Even part of $x(n)$



h. Odd part of $x(n)$



Exercise 3

Plot in matlab the Exercise 2

$x(n)$

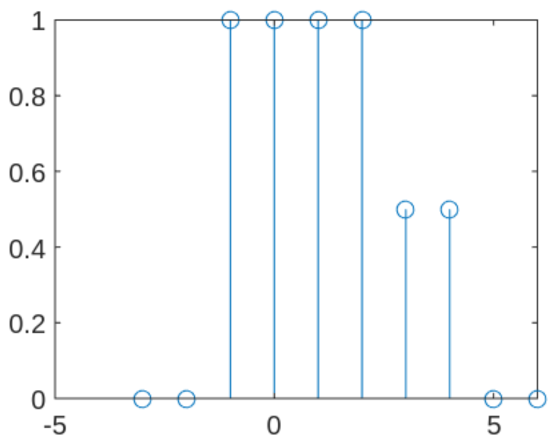
```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ···
```

```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

```
stem(n, x)
```



a. $x(n-2)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ···
```

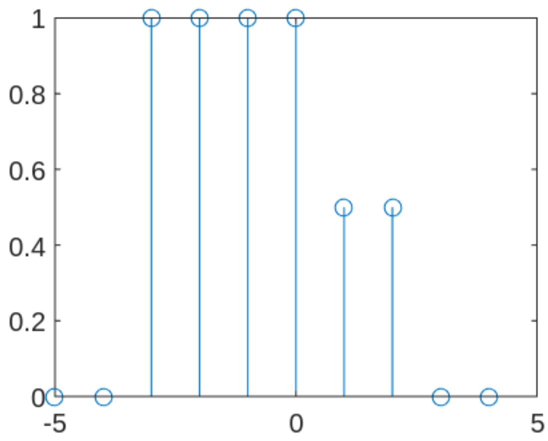
```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

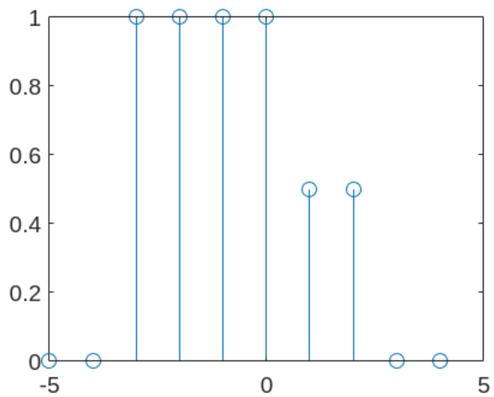
```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

```
[xf, nf] = sigshift(x, n, -2)
```

```
xf = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ···
nf = 1×10
    -5    -4    -3    -2    -1     0     1     2     3     4
```

```
stem(nf, xf)
```





b. $x(4-n)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ···
```

```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

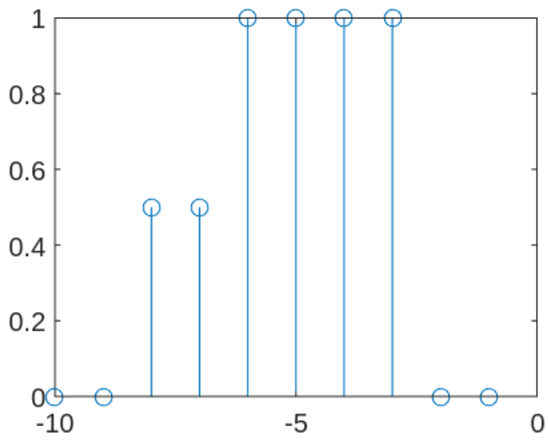
```
[xF, nF] = sigfold(x, n)
```

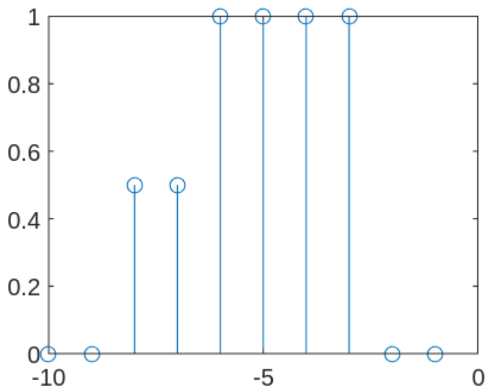
```
xF = 1×10
         0         0    0.5000    0.5000    1.0000    1.0000    1.0000    1.0000 ···
nF = 1×10
    -6    -5    -4    -3    -2    -1     0     1     2     3
```

```
[xf, nf] = sigshift(xF, nF, -4)
```

```
xf = 1×10
         0         0    0.5000    0.5000    1.0000    1.0000    1.0000    1.0000 ···
nf = 1×10
   -10    -9    -8    -7    -6    -5    -4    -3    -2    -1
```

```
stem(nf, xf)
```





c. $x(n+2)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ⋯
```

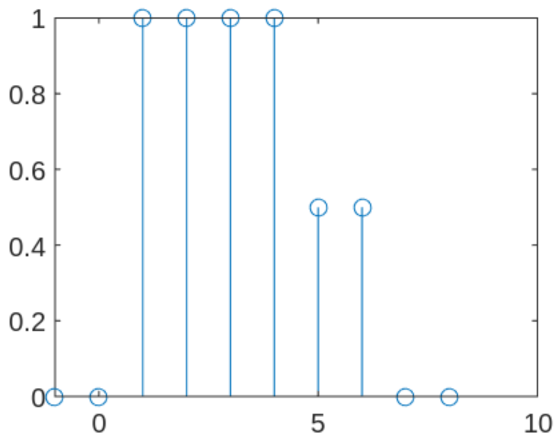
```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

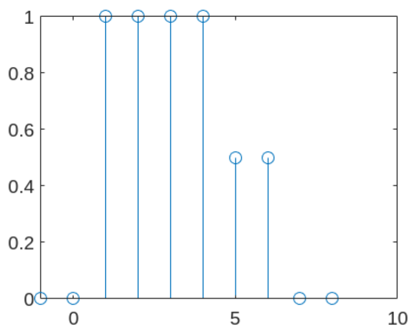
```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

```
[xf, nf] = sigshift(x, n, 2)
```

```
xf = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ⋯
nf = 1×10
    -1     0     1     2     3     4     5     6     7     8
```

```
stem(nf, xf)
```





d. $x(n)\mu(2-n)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ···
```

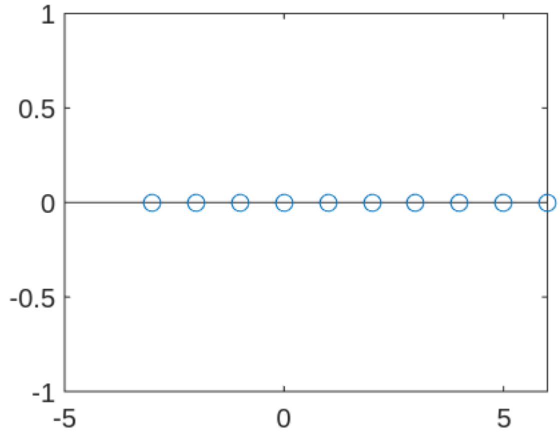
```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

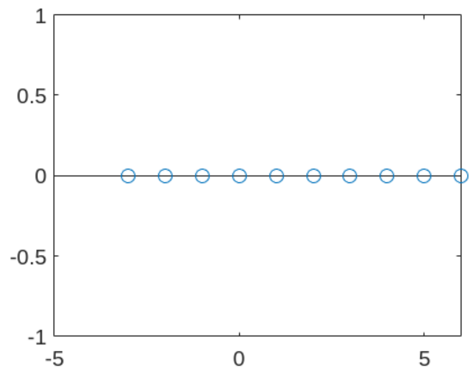
```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

```
u = [1, 1, 0, 0, 0, 0, 0, 0, 0, 0]
```

```
u = 1×10
     1     1     0     0     0     0     0     0     0     0
```

```
stem(n, x.*u)
```





e. $x(n+1)\delta(n-3)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ⋯
```

```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

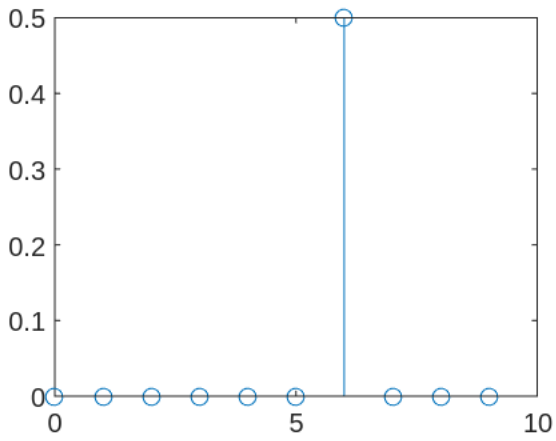
```
s = [0, 0, 0, 0, 0, 0, 1, 0, 0, 0]
```

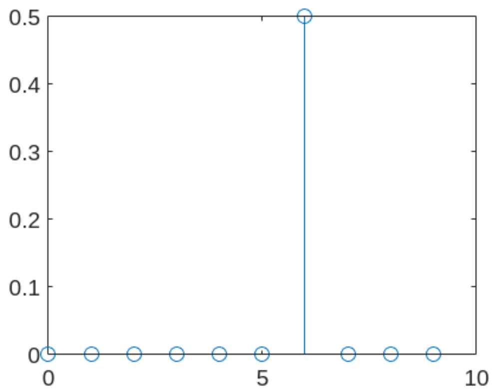
```
s = 1×10
     0     0     0     0     0     0     1     0     0     0
```

```
[xf, nf] = sigshift(x, n, 3)
```

```
xf = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ⋯
nf = 1×10
     0     1     2     3     4     5     6     7     8     9
```

```
stem(nf, xf.*s)
```





f. $x(n^2)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ⋯
```

```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

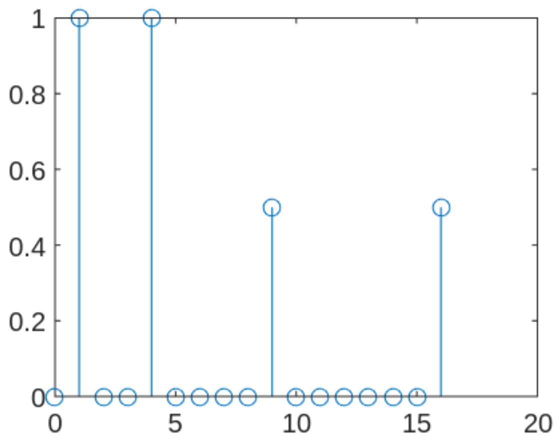
```
xexp = [0, 1.0, 0, 0, 1.0, 0, 0, 0, 0, 0.5, 0, 0, 0, 0, 0, 0, 0.5]
```

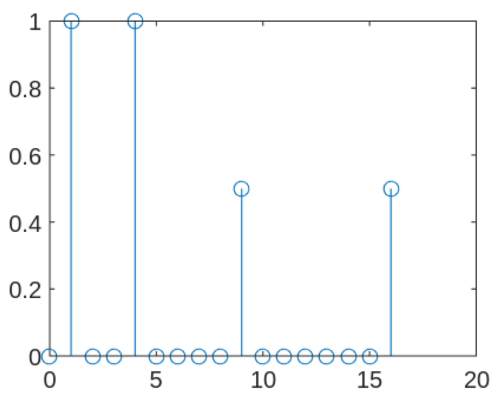
```
xexp = 1×17
         0    1.0000         0         0    1.0000         0         0         0 ⋯
```

```
nexp = [0:16]
```

```
nexp = 1×17
     0     1     2     3     4     5     6     7     8     9    10    11    12 ⋯
```

```
stem(nexp, xexp)
```





g. Even part of $x(n)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ⋯
```

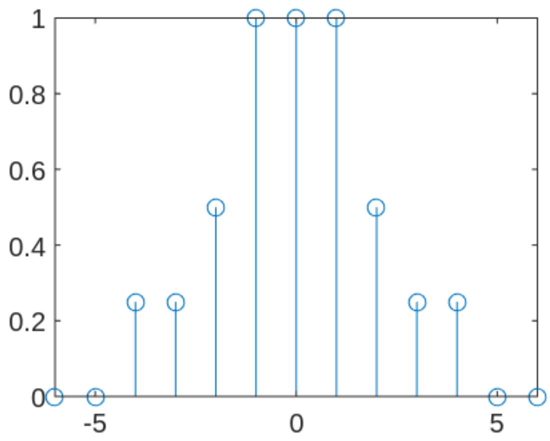
```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

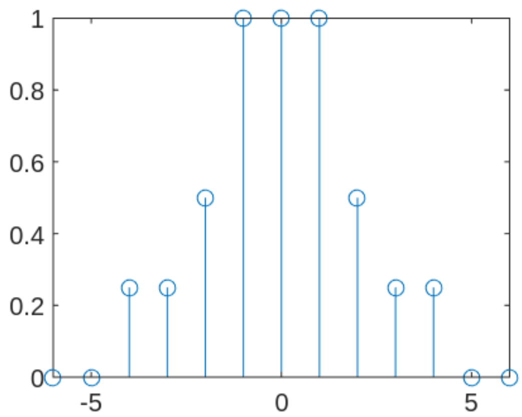
```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

```
[e, o, m] = evenodd(x, n)
```

```
e = 1×13
         0         0    0.2500    0.2500    0.5000    1.0000    1.0000    1.0000 ⋯
o = 1×13
         0         0   -0.2500   -0.2500   -0.5000         0         0         0 ⋯
m = 1×13
    -6    -5    -4    -3    -2    -1     0     1     2     3     4     5     6
```

```
stem(m, e)
```





h. Odd part of $x(n)$

```
x = [0, 0, 1.0, 1.0, 1.0, 1.0, 0.5, 0.5, 0, 0]
```

```
x = 1×10
         0         0    1.0000    1.0000    1.0000    1.0000    0.5000    0.5000 ···
```

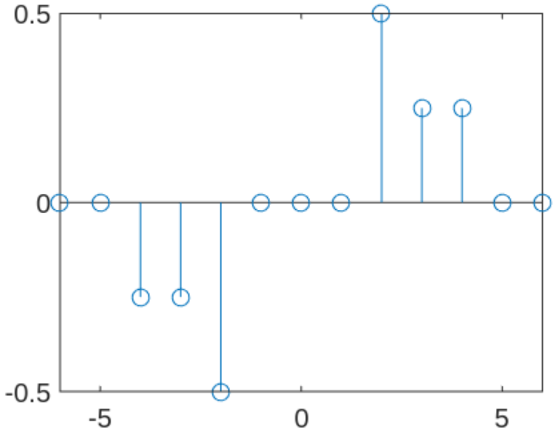
```
n = [-3, -2, -1, 0, 1, 2, 3, 4, 5, 6]
```

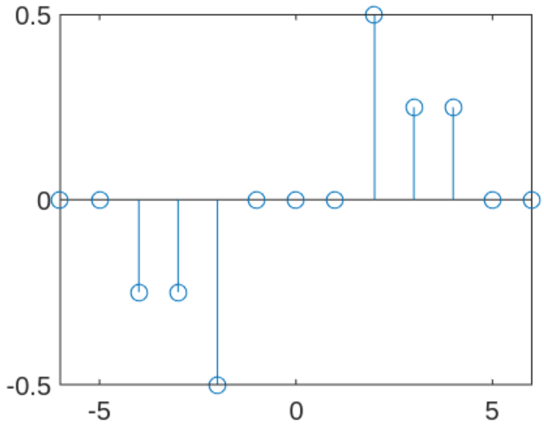
```
n = 1×10
    -3    -2    -1     0     1     2     3     4     5     6
```

```
[e, o, m] = evenodd(x, n)
```

```
e = 1×13
         0         0    0.2500    0.2500    0.5000    1.0000    1.0000    1.0000 ···
o = 1×13
         0         0   -0.2500   -0.2500   -0.5000         0         0         0 ···
m = 1×13
    -6    -5    -4    -3    -2    -1     0     1     2     3     4     5     6
```

```
stem(m, o)
```





Exercise 4

Mitra exercises 2.1, 2.3, 2.4, 2.5, 2.8 and 2.23

Exercise 2.1

Find the norm L_1, L_2 and L_inf of the next finite length sequences.

```
x1 = [4.5, -2.68, -0.14, 3.91, 2.62, -0.43, -4.81, 3.21, -0.55]
```

```
x1 = 1×9
    4.5000   -2.6800   -0.1400    3.9100    2.6200   -0.4300   -4.8100    3.2100 ⋯
```

```
% L_1 norm
L_1 = sum(abs(x1))
```

```
L_1 = 22.8500
```

```
% L_2 norm
L_2 = sqrt(sum(x1.^2))
```

```
L_2 = 9.1396
```

```
% L_inf norm
L_inf = max(abs(x1))
```

```
L_inf = 4.8100
```

```
% Matlab verification.
L1m = norm(x1, 1)
```

```
L1m = 22.8500
```

```
L2m = norm(x1, 2)
```

```
L2m = 9.1396
```

```
Linf = norm(x1, "inf")
```

```
Linf = 4.8100
```

```
x2 = [0.92, 2.34, 3.37, 1.9, -2.59, -0.75, 3.48, 3.33]
```

```
x2 = 1×8
    0.9200    2.3400    3.3700    1.9000   -2.5900   -0.7500    3.4800    3.3300
```

```
% L_1 norm
L_1 = sum(abs(x2))
```

```
L_1 = 18.6800
```

```
% L_2 norm
L_2 = sqrt(sum(x2.^2))
```

```
L_2 = 7.1944
```

```
% L_inf norm
L_inf = max(abs(x2))
```

```
L_inf = 3.4800
```

```
% Matlab verification.
L1m = norm(x2, 1)
```

```
L1m = 18.6800
```

```
L2m = norm(x2, 2)
```

```
L2m = 7.1944
```

```
Linf = norm(x2, "inf")
```

```
Linf = 3.4800
```

Exercise 2.4

Write the sequence $x[n] = [1, 3, -2, -4]$ in terms of the step unit sequence $\mu[n]$

Assuming the values in x[n] are n, then

$\mu[n] = [1, 1, 0, 0]$

But, if the signal is correct and the n values are not specified, then $\mu[n]x[n] = [1, 3, -2, -4]$, if the signal starts at n=0

Exercise 2.5

Consider the next sequences (values not specified are zero), get the sequences:

$$x(n) = \{-4, 5, 1, -2, -3, 0, 2\} \quad -3 \leq n \leq 3$$
$$y(n) = \{6, -3, -, 1, 0, 8, 7, -2\} \quad -1 \leq n \leq 5$$
$$w(n) = \{3, 2, 2, -1, 0, -2, 5\} \quad 2 \leq n \leq 8$$

a. $c(n) = x(-n + 2)$

```
x = [-4, 5, 1, -2, -3, 0, 2]
```

```
x = 1×7
    -4     5     1    -2    -3     0     2
```

```
n = [-3, -2, -1, 0, 1, 2, 3]
```

```
n = 1×7
    -3    -2    -1     0     1     2     3
```

```
[xf, xn] = sigfold(x, n)
```

```
xf = 1×7
     2     0    -3    -2     1     5    -4
xn = 1×7
    -3    -2    -1     0     1     2     3
```

```
[cf, cn] = sigshift(xf, xn, -2)
```

```
cf = 1×7
     2     0    -3    -2     1     5    -4
cn = 1×7
    -5    -4    -3    -2    -1     0     1
```

```
cf
```

```
cf = 1×7
     2     0    -3    -2     1     5    -4
```

b. $d(n) = y(-n - 3)$

```
y = [6, -3, -1, 0, 8, 7, -2]
```

```
y = 1×7
     6    -3    -1     0     8     7    -2
```

```
n = [-1, 0, 1, 2, 3, 4, 5]
```

```
n = 1×7
    -1     0     1     2     3     4     5
```

```
[yf, nf] = sigfold(y, n)
```

```
yf = 1×7
    -2     7     8     0    -1    -3     6
nf = 1×7
    -5    -4    -3    -2    -1     0     1
```

```
[df, dn] = sigshift(yf, nf, 3)
```

```
df = 1×7
    -2     7     8     0    -1    -3     6
dn = 1×7
    -2    -1     0     1     2     3     4
```

```
df
```

```
df = 1×7
    -2     7     8     0    -1    -3     6
```

c. $e(n) = w(-n)$

```
w = [3, 2, 2, -1, 0, -2, 5]
```

```
w = 1×7
     3     2     2    -1     0    -2     5
```

```
n = [2, 3, 4, 5, 6, 7, 8]
```

```
n = 1×7
     2     3     4     5     6     7     8
```

```
[wf, nf] = sigfold(w, n)
```

```
wf = 1×7
     5    -2     0    -1     2     2     3
nf = 1×7
    -8    -7    -6    -5    -4    -3    -2
```

```
wf
```

```
wf = 1×7
     5    -2     0    -1     2     2     3
```

d. $u(n) = x(n) + y(n-2)$

```
x = [-4, 5, 1, -2, -3, 0, 2]
```

```
x = 1×7
    -4     5     1    -2    -3     0     2
```

```
nx = [-3, -2, -1, 0, 1, 2, 3]
```

```
nx = 1×7
    -3    -2    -1     0     1     2     3
```

```
y = [6, -3, -1, 0, 8, 7, -2]
```

```
y = 1×7
     6    -3    -1     0     8     7    -2
```

```
ny = [-1, 0, 1, 2, 3, 4, 5]
```

```
ny = 1×7
    -1     0     1     2     3     4     5
```

```
[yf, nyf] = sigfold(y, ny)
```

```
yf = 1×7
    -2     7     8     0    -1    -3     6
nyf = 1×7
    -5    -4    -3    -2    -1     0     1
```

```
[sa, san] = sigadd(x, nx, yf, nyf)
```

```
sa = 1×9
    -2     7     4     5     0    -5     3     0     2
san = 1×9
    -5    -4    -3    -2    -1     0     1     2     3
```

```
sa
```

```
sa = 1×9
    -2     7     4     5     0    -5     3     0     2
```

e. $v(n) = x(n)w(n+4)$

```
x = [-4, 5, 1, -2, -3, 0, 2]
```

```
x = 1×7
    -4     5     1    -2    -3     0     2
```

```
nx = [-3, -2, -1, 0, 1, 2, 3]
```

```
nx = 1×7
    -3    -2    -1     0     1     2     3
```

```
w = [3, 2, 2, -1, 0, -2, 5]
```

```
w = 1×7
     3     2     2    -1     0    -2     5
```

```
nw = [2, 3, 4, 5, 6, 7, 8]
```

```
nw = 1×7
     2     3     4     5     6     7     8
```

```
[wf, nf] = sigshift(w, nw, -4)
```

```
wf = 1×7
     3     2     2    -1     0    -2     5
nf = 1×7
    -2    -1     0     1     2     3     4
```

```
[sm, snm] = sigmult(x, nx, wf, nf)
```

```
sm = 1×8
     0    15     2    -4     3     0    -4     0
snm = 1×8
```

```
    -3    -2    -1     0     1     2     3     4
```

```
sm
```

```
sm = 1×8
     0    15     2    -4     3     0    -4     0
```

f. $s(n) = y(n) - w(n-4)$

```
y = [6, -3, -1, 0, 8, 7, -2]
```

```
y = 1×7
     6    -3    -1     0     8     7    -2
```

```
ny = [-1, 0, 1, 2, 3, 4, 5]
```

```
ny = 1×7
    -1     0     1     2     3     4     5
```

```
w = [3, 2, 2, -1, 0, -2, 5]
```

```
w = 1×7
     3     2     2    -1     0    -2     5
```

```
nw = [2, 3, 4, 5, 6, 7, 8]
```

```
nw = 1×7
     2     3     4     5     6     7     8
```

```
[wf, nf] = sigshift(w, nw, 4)
```

```
wf = 1×7
     3     2     2    -1     0    -2     5
nf = 1×7
     6     7     8     9    10    11    12
```

```
[sa, san] = sigadd(y, ny, wf.*-1, nf)
```

```
sa = 1×14
     6    -3    -1     0     8     7    -2    -3    -2    -2     1     0     2···
san = 1×14
    -1     0     1     2     3     4     5     6     7     8     9    10    11···
```

```
sa
```

```
sa = 1×14
     6    -3    -1     0     8     7    -2    -3    -2    -2     1     0     2···
```

g. $r(n) = 3.5y(n)$

```
y = [6, -3, -1, 0, 8, 7, -2]
```

```
y = 1×7
     6    -3    -1     0     8     7    -2
```

```
ny = [-1, 0, 1, 2, 3, 4, 5]
```

```
ny = 1×7
    -1     0     1     2     3     4     5
```

```
r = y.^3.5
```

```
r = 1×7
1.0e+03 *
   0.5291 + 0.0000i   0.0000 - 0.0468i   0.0000 - 0.0010i   0.0
```

Exercise 2.8

Find the symmetric and asymmetric conjugate of the next sequences.

Use the following formulas:

$$x_{1\,cs}(n) = \frac{1}{2}\left[x(n) + x_1^*(-n)\right] \qquad x_{1\,ca}(n) = \frac{1}{2}\left[x(n) - x_1^*(-n)\right]$$

a. $x_1(n) = \{1 + j4, -2 + j5, 3 - j2, -7 + j3, -1 + j1\}, \quad -2 \le n \le 2$

$$\{x_1^*[n]\} = \{1 - j4, -2 - j5, 3 + j2, -7 - j3, -1 - j1\}$$

$$\{x_1^*[-n]\} = \{-1 - j1, -7 - j3, 3 + j2, -2 - j5, 1 - j4\}$$

$$x_{1cs}(n) = \{j1.5, -4.5 + j1, 3, -4.5 - j1, -j1.5\}$$

$$x_{1ca}(n) = \{1 + j2.5, 2.5 + j4, -j2, -2.5 + j4, -1 + j2.5\}$$

b. $x_2(n) = e^{j\pi\frac{n}{3}}$

c. $x_3(n) = je^{-j\pi\frac{n}{3}}$