# EE301 Homework-1

İsmail Enes Bülbül, Eren Meydanlı, Ahmet Caner Akar

October 30, 2022

## Question 1

### System 1

**1 -** The system is memoryless since the value of the output $y(t)$ depends on only the value of the input $x(t)$ at present time instant.
**2 -** To determine the linearity of the system, apply the input $\alpha x(t)$ and let the output $\hat{y}(t) = \alpha x(t) cos(2\pi f_0 t) = \alpha y(t) \longrightarrow$ the system is linear.
**3 -** The system is causal since the output $y(t)$ does not depend on the future value of the input $x(t)$.
**4 -** The system is time-variant. To see this, apply the input $x(t-t_0)$ and let the output $\hat{y}(t) = x(t-t_0) cos(2\pi f_0 t) \neq y(t-t_0) = x(t-t_0) cos(2\pi f_0 (t-t_0))$.
**5 -** A system is said to be stable if any bounded input creates a bounded output. Let $x(t)$ be a bounded. That is, there exist $M > 0$ such that $|x(t)| \leq M$. Also, $-1 \leq cos(2\pi f_0 t) \leq 1$. Thus, $|y(t)| = |x(t) cos(2\pi f_0 t)| \leq |x(t)| \, |cos(2\pi f_0 t)| \leq M \longrightarrow$ so the system is stable.

### System 2

**1 -** The system is memoryless.
**2 -** The system is non-linear since if we apply the input $\alpha x(t)$ then the output becomes $\hat{y}(t) = c_1 \alpha x(t) + c_2 (\alpha x(t))^2 = \alpha c_1 x(t) + \alpha^2 c_2 x^2(t) \neq \alpha y(t) = \alpha c_1 x(t) + \alpha c_2 x^2(t)$
**3 -** The system is causal.
**4 -** The system is time-invariant consider that if we apply the input $x(t-t_0)$ and let the output $\hat{y}(t) = c_1 x(t-t_0) + c_2 x^2(t-t_0) = y(t-t_0)$
**5 -** Apply a bounded input $x(t)$ such that $|x(t)| \leq M$ then $|y(t)| = |c_1 x(t) + c_2 x^2(t)| \leq$

$|c_1|\,|x(t)| + |c_2|\,|x^2(t)| \leq |c_1|\,M + |c_2|\,M^2 < \infty \longrightarrow$ the system is stable.

## System 3

**1 -** The system is memoryless.

**2 -** The system is non-linear. Apply the input $\alpha x(t)$ and let the output $\hat{y}(t) = \alpha x(t) + 4 \neq \alpha y(t) = \alpha x(t) + \alpha 4$

**3 -** The system is causal.

**4 -** Apply the input $x(t - t_0)$:

$\hat{y}(t) = x(t - t_0) + 4 = y(t - t_0) \longrightarrow$ the system is time-invariant.

**5 -** Let $x(t)$ be bounded input such that $|x(t)| \leq M$. Then, $|y(t)| = |x(t) + 4| \leq |x(t)| + 4 \leq M + 4$ (which is a finite number). So, the system is stable.

## System 4

**1 -** The system is not memoryless as it is not causal.

**2 -** The system is linear.

**3 -** For $t < 0$, $y(t)$ is calculated by future values of $x(t)$, therefore, the system is not causal.

**4 -** Apply the input $x(t - t_0)$:

$\hat{y}(t) = x((t - t_0)/3) = y(t - t_0) \longrightarrow$ the system is time-invariant.

**5 -** Let $x(t)$ be bounded input such that $|x(t)| \leq M$. Then, $|y(t)| = |x(t/3)|$ (the expanded version of $x(t)$) $\leq M \longrightarrow$ the system is stable.

## System 5

**1 -** The system has a memory since the value of the output $y(t)$ depends on the future values of the input $x(t)$.

**2 -** The system is linear.

**3 -** The system is not causal because the output $y(t)$ does not depend only on present and past values of the input.

**4 -** Apply $x(t - t_0)$:

$\hat{y}(t) = tx(t - t_0 + 5) \neq y(t - t_0) = (t - t_0)x(t - t_0 + 5) \longrightarrow$ the system is time-variant.

**5 -** The system is not stable. To see this, consider the following example:

$x(t) = u(t) \Rightarrow x(t + 5) = u(t + 5)$ also $|x(t + 5)| = |u(t + 5)| \leq 1$ which means that the input is bounded. However, the output $y(t) = tu(t+5)$ is not bounded. Therefore, the system is not stable.

### System 6

$$y(t) = u(x(t)) = \begin{cases} 1, & x(t) \geq 0 \\ 0, & x(t) < 0 \end{cases}$$

**1 -** The system is memoryless.
**2 -** The system is non-linear, apply $\alpha x(t)$:
$\hat{y}(t) = u(\alpha x(t))$ may not be equal to $\alpha y(t) = \alpha u(x(t))$. For example, let the input $x(t) = u(t)$ then for $t \geq 0$, the $x(t) = 1$. Also, choose $\alpha = -1$. As a result, for $t \geq 0$: $u(\alpha x(t)) = 0 \neq \alpha y(t) = \alpha u(x(t)) = -1$.
**3 -** The system is causal.
**4 -** Apply $x(t - t_0)$:
$\hat{y}(t) = u(x(t - t_0)) = y(t - t_0) \longrightarrow$ the system is time-invariant.
**5 -** Remember that $|u(t)| \leq 1$ so it is bounded. In this system, whatever the input $x(t)$ is, the output $y(t) = u(x(t)) = \begin{cases} 1, & x(t) \geq 0 \\ 0, & x(t) < 0 \end{cases}$. Hence, the system is stable.

# Question 2

## a)

$y_1(t) = x_1(t) * h(t) = \int_{-\infty}^{\infty} \alpha^{t-\tau} u(t-\tau)\, d\tau = \int_{-\infty}^{t} \alpha^{t-\tau}\, d\tau$
$y_1(t) = \alpha^t \int_{-\infty}^{t} \alpha^{-\tau}\, d\tau = \alpha^t \int_{-\infty}^{t} e^{ln(\alpha^{-\tau})}\, d\tau = \alpha^t \int_{-\infty}^{t} e^{-\tau ln\alpha}\, d\tau$
Let $u = -\tau ln\alpha$ then $\frac{du}{d\tau} = -ln\alpha \Rightarrow \frac{du}{ln\alpha} = -d\tau$
Then, the integral becomes: $y_1(t) = \alpha^t \int_{\tau=-\infty}^{\tau=t} e^u \, \frac{-du}{ln\alpha}$
Thus, $y_1(t) = \frac{-\alpha^t}{ln\alpha} \left( e^u \Big|_{\tau=-\infty}^{\tau=t} \right) \Rightarrow y_1(t) = \frac{-1}{ln\alpha}$

$y_2(t) = x_2(t) * h(t) = \int_{-\infty}^{\infty} x_2(\tau) h(t-\tau)\, d\tau = \int_{-\infty}^{\infty} u(\tau) \alpha^{t-\tau} u(t-\tau)\, d\tau$
For $t < 0$, $y_2(t) = 0$
For $t \geq 0$, $y_2(t) = \int_0^t \alpha^{t-\tau}\, d\tau$
As we did in the calculation of $y_1(t)$, use the method of integration by substution.
Then, the integral becomes: $y_2(t) = \frac{-\alpha^t}{ln\alpha} \left( e^u \Big|_{\tau=0}^{\tau=t} \right) \Rightarrow y_2(t) = \frac{\alpha^t - 1}{ln\alpha}$, for $t \geq 0$

Therefore, $y_2(t) = \frac{\alpha^t - 1}{ln\alpha} u(t)$

Similarly, $y_3(t) = x_3(t) * h(t) = \int_{-\infty}^{\infty} x_3(\tau)h(t-\tau)\,d\tau = \int_{-\infty}^{\infty} u(-\tau-1)\alpha^{t-\tau}u(t-\tau)\,d\tau$
For $t < -1$, $y_3(t) = \int_{-\infty}^{t} \alpha^{t-\tau}\,d\tau = \frac{-1}{ln\alpha}$
For $t \geq -1$, $y_3(t) = \int_{-\infty}^{-1} \alpha^{t-\tau}\,d\tau = \frac{-\alpha^{t+1}}{ln\alpha}$
Therefore, $y_3(t) = \frac{-1}{ln\alpha} + (\frac{-\alpha^{t+1}+1}{ln\alpha})u(t+1)$

As it can be seen in above calculations, to evaluate the output signal y(t) for each input signal x(t), we applied the convolution operation to each of them with the impulse response h(t). Although the integral expressions would be the same, the range of integration is different for each input. Also, the output $y_3(t)$ is a time-reversed and shifted version of $y_2(t)$.

### b)

Let $x_5(t) = u(t+1)$. Then $x_4(t) = x_5(t) - x_2(t)$.
$y_5(t) = x_5(t) * h(t) = y_2(t+1) = \frac{\alpha^{t+1}-1}{ln\alpha}u(t+1)$
$y_4(t) = x_4(t) * h(t) = (x_5(t) - x_2(t)) * h(t)$
$= (x_5(t) * h(t)) - (x_5(t) * h(t))$
$= y_5(t) - y_2(t)$
$y_4(t) = \frac{\alpha^{t+1}-1}{ln\alpha}u(t+1) - \frac{\alpha^t-1}{ln\alpha}u(t)$

## Question 3

### a)

Recall that in continuous-time systems one can apply the derivative operation to any arbitrary input signal. For example,
$\delta(t) = \frac{d}{dt}u(t) \longrightarrow \int_{-\infty}^{\infty} \delta(t)\,dt$
Also, it can be applied by using the formal definition of the derivative:
$\lim_{h\to 0} \frac{x(t+h)-x(t)}{h}$
However, in discrete-time systems h cannot go to zero and the minimum value for h can be one. So, in discrete time the derivative expression becomes
$\lim_{h\to 1} \frac{x(n+h)-x(n)}{h} = x[n+1] - x[n]$
Thus, the derivative operation in continuous-time systems is analogous of the difference operation in discrete-time. Hence, we can obtain the impulse response of the difference operation as: $h[n] = \delta[n] - \delta[n-1]$

## b)

By convolution, $y[n] = x[n] * h[n] = x[n] * (\delta[n] - \delta[n-1])$
By distributive property of the convolution operation,
$x[n] * (\delta[n] – \delta[n-1]) = (x[n] * \delta[n]) – (x[n] * \delta[n-1])$
$y[n] = x[n] – x[n-1]$

## c)

$e^{j\Omega_0 n}(1 - e^{-j\Omega_0}) = (cos(\Omega_0 n) + jsin(\Omega_0 n))(1 - e^{-j\Omega_0})$
$= cos(\Omega_0 n) + jsin(\Omega_0 n) - (cos(\Omega_0 n) + jsin(\Omega_0 n))(cos(\Omega_0 n) - jsin(\Omega_0 n))$
$= cos(\Omega_0 n) + jsin(\Omega_0 n) - cos(\Omega_0 n)cos(\Omega_0) + jcos(\Omega_0 n)sin(\Omega_0) - jsin(\Omega_0 n)cos(\Omega_0) - sin(\Omega_0 n)sin(\Omega_0)$
$= cos(\Omega_0 n) + jsin(\Omega_0 n) - (cos(\Omega_0 n)cos(\Omega_0) + sin(\Omega_0 n)sin(\Omega_0)) - j(sin(\Omega_0 n)cos(\Omega_0) - cos(\Omega_0 n)sin(\Omega_0))$
$= cos(\Omega_0 n) + jsin(\Omega_0 n) - cos(\Omega_0 n - \Omega_0) - jsin(\Omega_0 n - \Omega_0)$
$= cos(\Omega_0 n) - cos(\Omega_0(n-1)) + j(sin(\Omega_0 n) - sin(\Omega_0(n-1)))$

$|y[n]| = [(cos(\Omega_0 n) - cos(\Omega_0(n-1)))^2 + (sin(\Omega_0 n) - sin(\Omega_0(n-1)))^2]^{1/2}$
$= [cos^2(\Omega_0 n) - 2cos(\Omega_0 n)cos(\Omega_o(n-1)) + cos^2(\Omega_0(n-1)) + sin^2(\Omega_0 n) - 2sin(\Omega_0 n)sin(\Omega_o(n-1)) + sin^2(\Omega_0(n-1))]^{1/2}$
$= \sqrt{2 - 2[cos(\Omega_0 n)cos(\Omega_0(n-1)) + sin(\Omega_0 n)sin(\Omega_0(n-1))]}$
$= \sqrt{2 - 2cos(\Omega_0 n - \Omega_0(n-1))}$
$= \sqrt{2 - 2cos(\Omega_0)}$

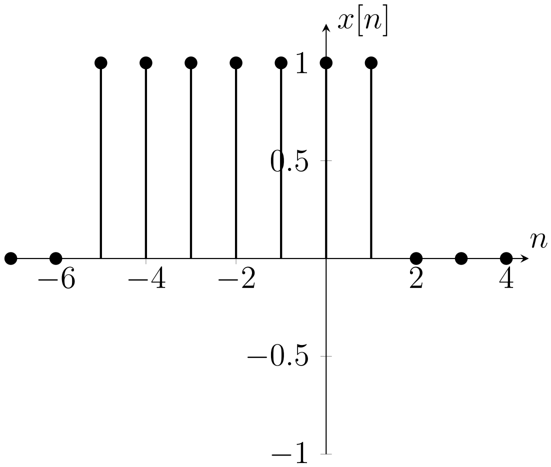
## d)

$\Omega_0 = 0 \longrightarrow |y[n]| = \sqrt{2 - 2cos(0)} = 0$
$\Omega_0 = \frac{2\pi}{8} \longrightarrow |y[n]| = \sqrt{2 - 2cos(\frac{2\pi}{8})} = \sqrt{2 - \sqrt{2}}$
$\Omega_0 = \frac{2\pi}{4} \longrightarrow |y[n]| = \sqrt{2 - 2cos(\frac{2\pi}{4})} = \sqrt{2}$
$\Omega_0 = \frac{2\pi}{2} \longrightarrow |y[n]| = \sqrt{2 - 2cos(\frac{2\pi}{2})} = 2$

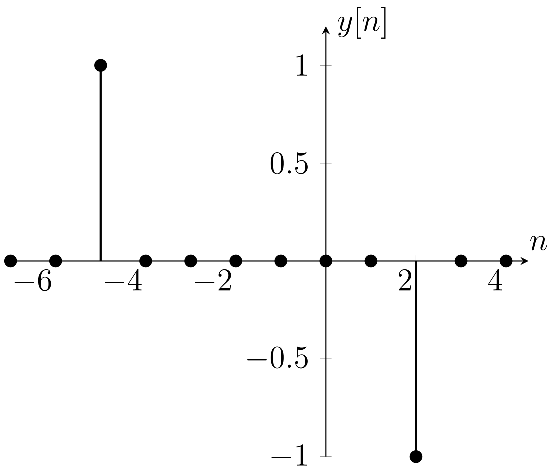
As we have shown in part a), the given discrete-time LTI system is analagous to the derivative operation in continuous-time systems. Also, please remember that the derivative is defined as the rate of change of a function with respect to a variable. Here, as the frequency of the input x[n] is increased, the period becomes smaller and the values of the input function are changed more rapidly. Therefore, the rate of change of a input function is increased as the frequency is increased, and the modulus of the output $|y[n]|$ is getting larger values.

## e)

$x[n] = u[n+5] - u[n-2]$ can be plotted as:




We can calculate $y[n]$ for $x[n] = u[n+5] - u[n-2]$ by using the answer in (b) as:
$y[n] = u[n+5] - u[n+2] - (u[n+4] + u[n-3])$
$y[n]$ can be plotted as:




From the graph of $y[n]$, we can say that $x[n]$ has edges at $n = -5$ and $n = 2$.

## f)

From (d), we can say that magnitude of the output signal of the system increases as the frequency increases. In (f), $x[n]$ has rapid changes at $n = -5$ and $n = 2$, and as a result,

$y[n]$ has magnitude 1 at $n = -5$ and $n = 2$, and 0 elsewhere.

# Question 4

## a)

$$h_1[n] = \sum_{k=-\infty}^{n+2} \delta[k] = \begin{cases} 1, & n \geq -2 \\ 0, & n < -2 \end{cases} = u[n+2]$$

$$h_2[n] = \sum_{k=-\infty}^{n-2} \delta[k] = \begin{cases} 1, & n \geq 2 \\ 0, & n < 2 \end{cases} = u[n-2]$$

## b)

**i:** Both systems are not memoryless as the output signals depend on past values of input signals.
**ii:** System 1 is not causal as the output signal depends on the values of the input signal at future time $(n+1, n+2)$. System 2, however, is causal since the output singal only depends the past values of the input signal.
**iii:** Both systems have unbounded output signals for the input signal chosen as unit step function. Therefore, the systems are not stable.

## c)

### i.

The systems are connected in parallel, so we can calculate the impulse response of the overall system $h[n]$ as:
$h[n] = h_1[n] - h_2[n]$
$h[n] = u[n+2] - u[n-2]$

### ii.

We can calculate the output signal of the overall system $y[n]$ for an arbitrary input signal $x[n]$ as:
$y[n] = x[n-1] + x[n] + x[n+1] + x[n+2]$
The system is neither memoryless nor causal since $y[n]$ depends future values of $x[n]$.

For a bounded input signal, we can say that there exist $M > 0$ such that $|x[n]| \leq M$. Then $|y[n]| \leq 4M$ and $y[n]$ is bounded, therefore the system is stable.

# Question 5

## a)

In calculating part of the question, we added zero vectors to end and beginning of the main vector to obtain correct results in range of $0 \leq n \leq N-1$, which can be seen in Appendix. Since we manually did that zero adding part, we deviated the result for $n > N$ range. Because of this situation, our calculated result has only N value in range. On the other hand, MATLAB's built-in function, *conv()*, evaluate the convolution so that the size of the output is equal to $N + L - 1$. That's why the output graphs of convolution operation are different by using our function and MATLAB's built-in function as shown in Appendix. In fact, to get the same result that we got by using our function, one can specify an input argument, *same*, to MATLAB's function so that the result gives the central part of the convolution operation. Otherwise, the built-in function gives the full convolution whose length is equal to $N + L - 1$.

## b)

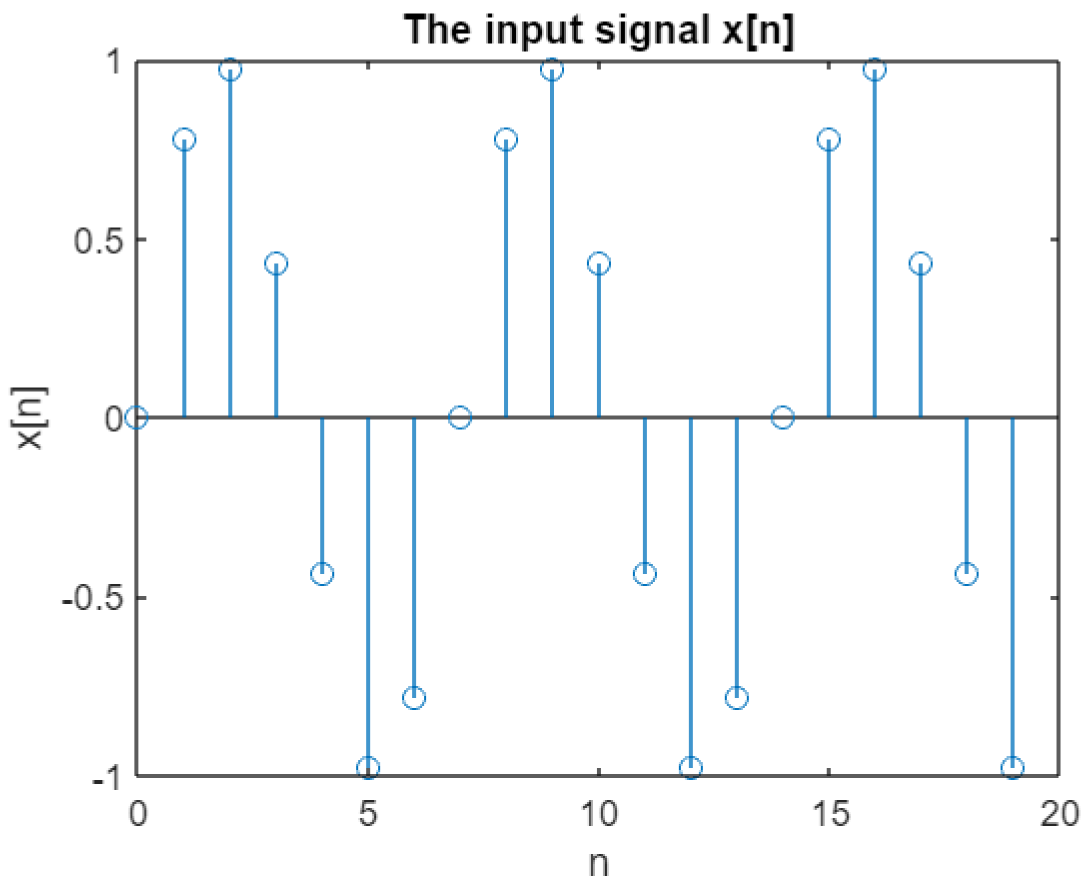
The output signal looks more smoother in larger values of L. As we can observe that h[n] impulse response created accordingly to L which directly affects the convolution range. By saying range, we mean that how many $x*h$ multiplication we will use to calculate one element of the output. So, this is why that the output signal gain more elements and more precise values when L gets larger. And also, we can observe that the L value expands the output range by expanding h[n] size. The MATLAB code and the resulting graphs for different values of L can be seen in Appendix.

# Appendix

## Part a)

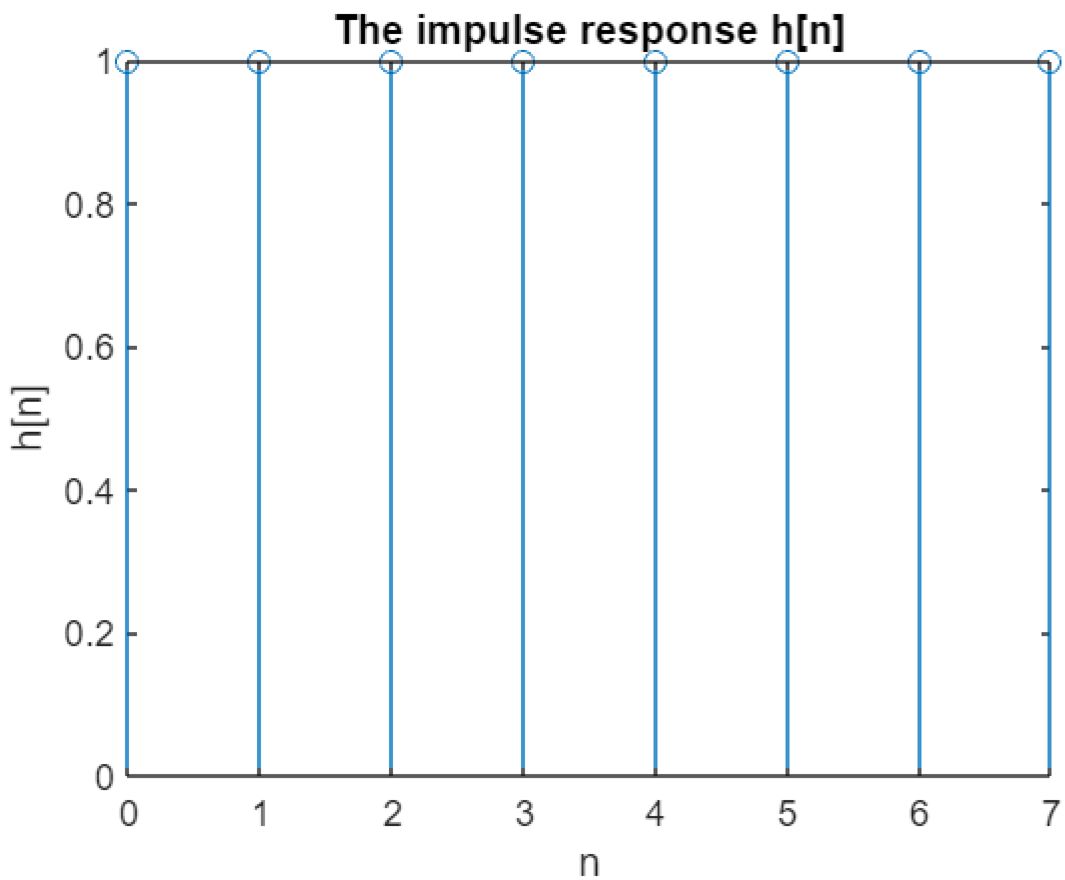
**Step 1:** Generate the input signal x[n]

```
N = 20;
n = 0:1:N-1;
x = sin(2*pi*n/7);
figure;
stem(n,x);
xlabel('n');
ylabel('x[n]');
title('The input signal x[n]');
```




**Step 2:** Generate and plot the impulse response h[n]

```
L = N-12;
h = ones(1,L);
figure;
stem(0:L-1,h);
xlabel('n');
ylabel('h[n]');
title('The impulse response h[n]');
```




**Step 3:** Time-reversed impulse response h[-n]

```
h_reversed = flip(h);
```

**Step 4:** Zero-padding

```
x_padded = [zeros(1,L-1) x  zeros(1,L-1)];
```

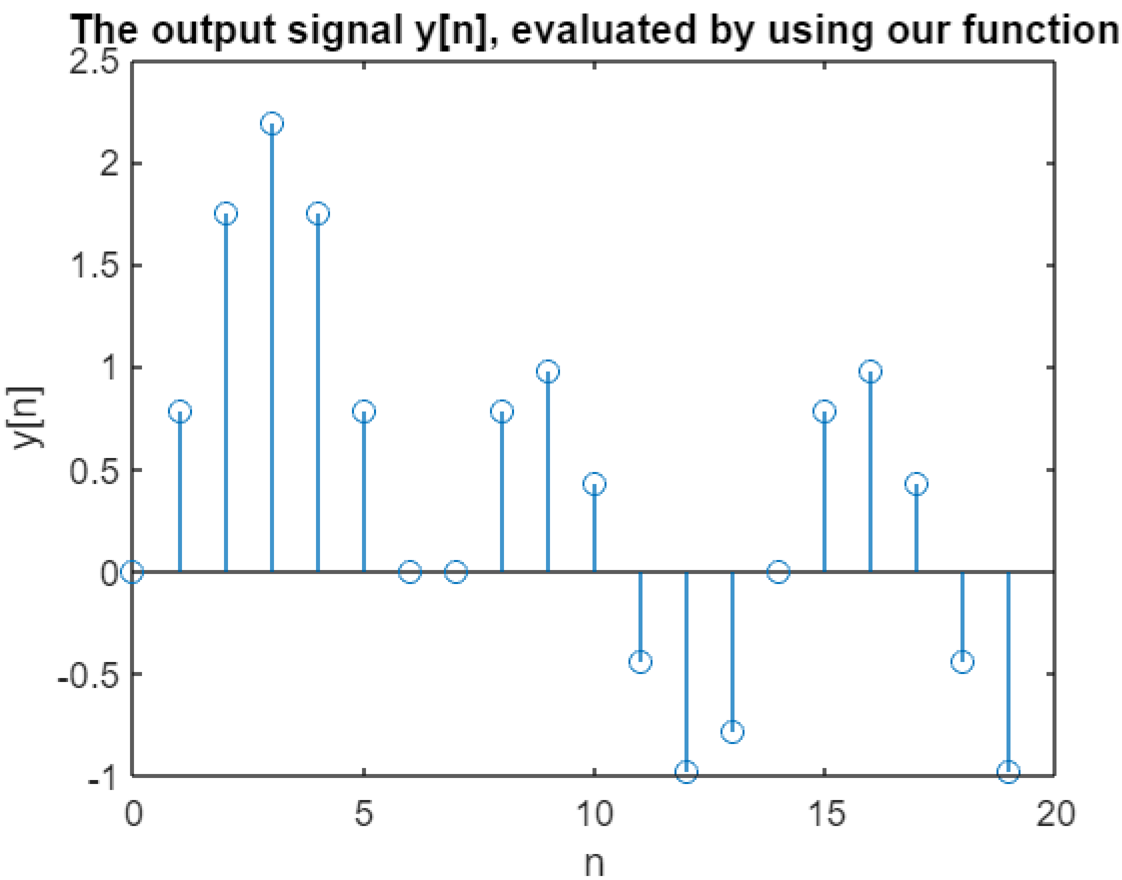
**Step 5:** Evaluate the convolution sum

```
y = zeros(1,N+L-1);
for i=1:N+L-1
    y(i) = x_padded(i:i+L-1)*h_reversed';
end
```

**Step 6:** Plot the output signal y[n]

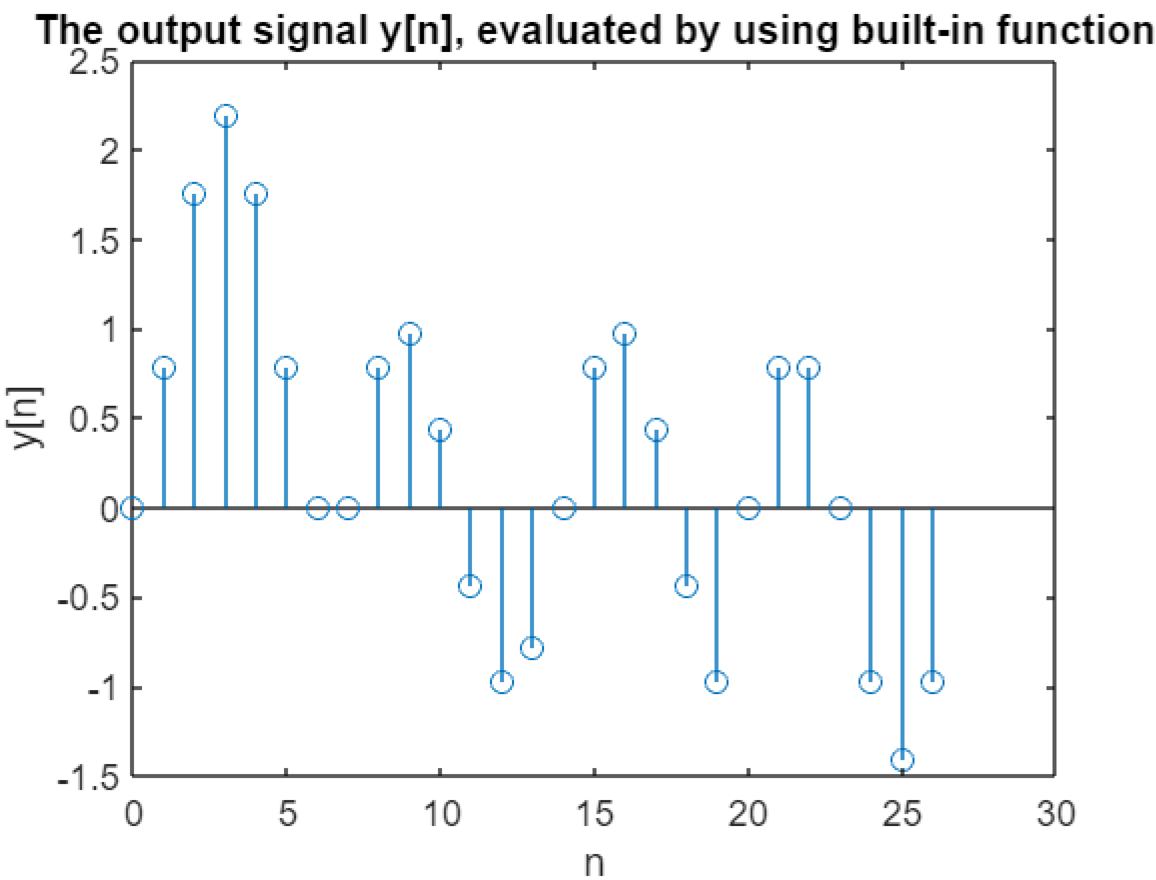
```
y = y(1:N);
figure;
stem(n,y);
xlabel('n');
ylabel('y[n]');
```

```
title('The output signal y[n], evaluated by using our function');
```




**Step 7:** MATLAB built-in convolution function

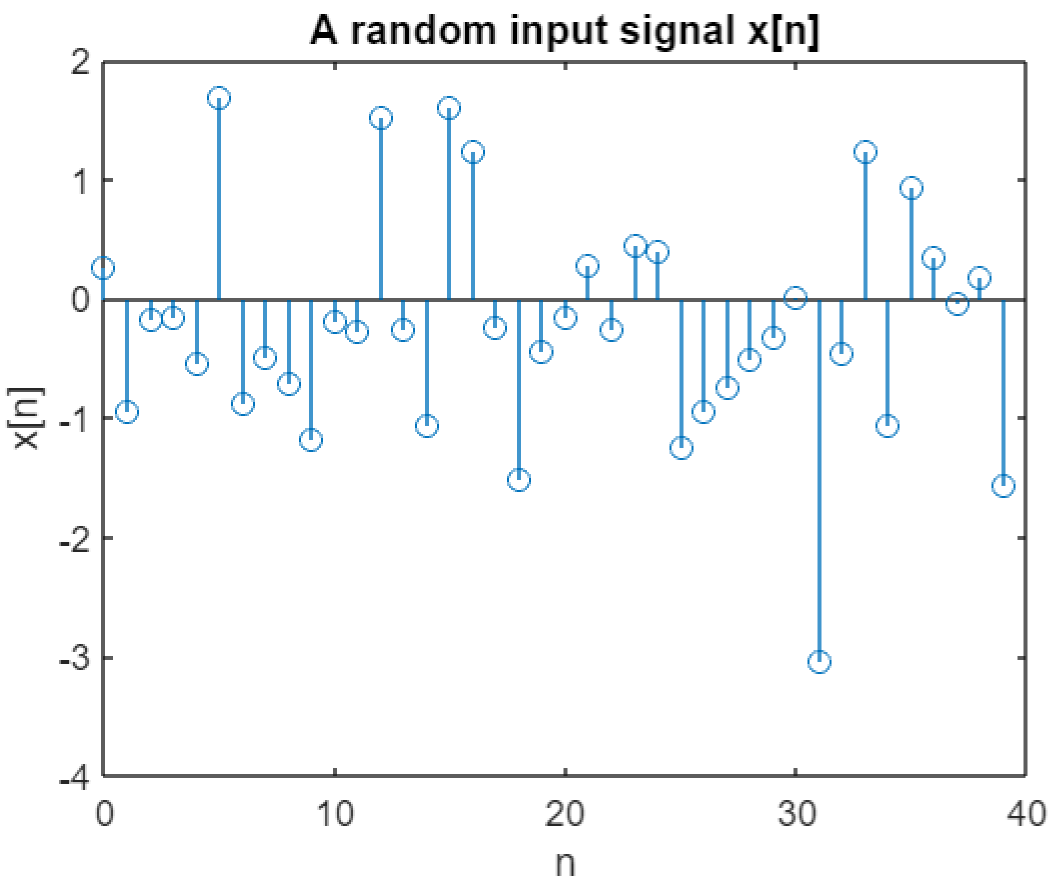
```
y = conv(x,h);
figure;
stem(0:N+L-2,y);
xlabel('n');
ylabel('y[n]');
ylabel('y[n]');
title('The output signal y[n], evaluated by using built-in function');
```




## Part b)

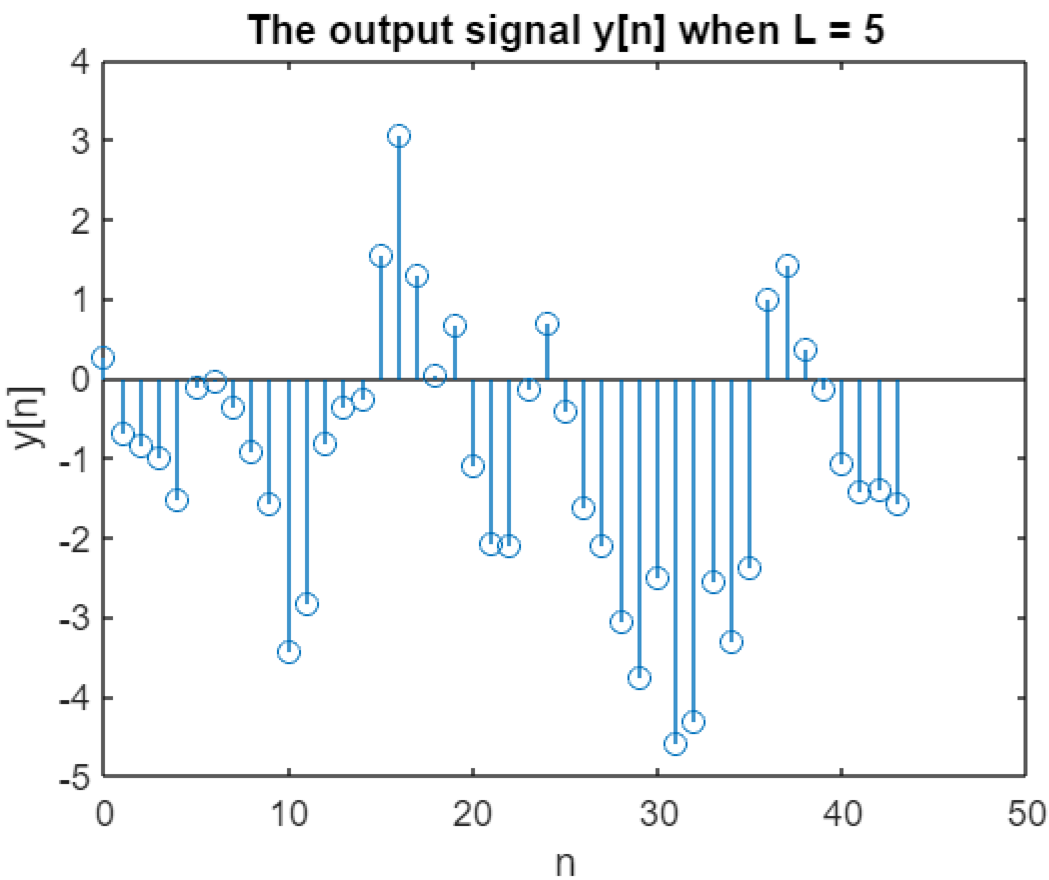
**Step 1:** Generate random input signal x[n]

```
N = 40;
n = 0:1:N-1;
x = randn(1,N);
figure;
stem(n,x);
title('A random input signal x[n]');
xlabel('n');
ylabel('x[n]');
```

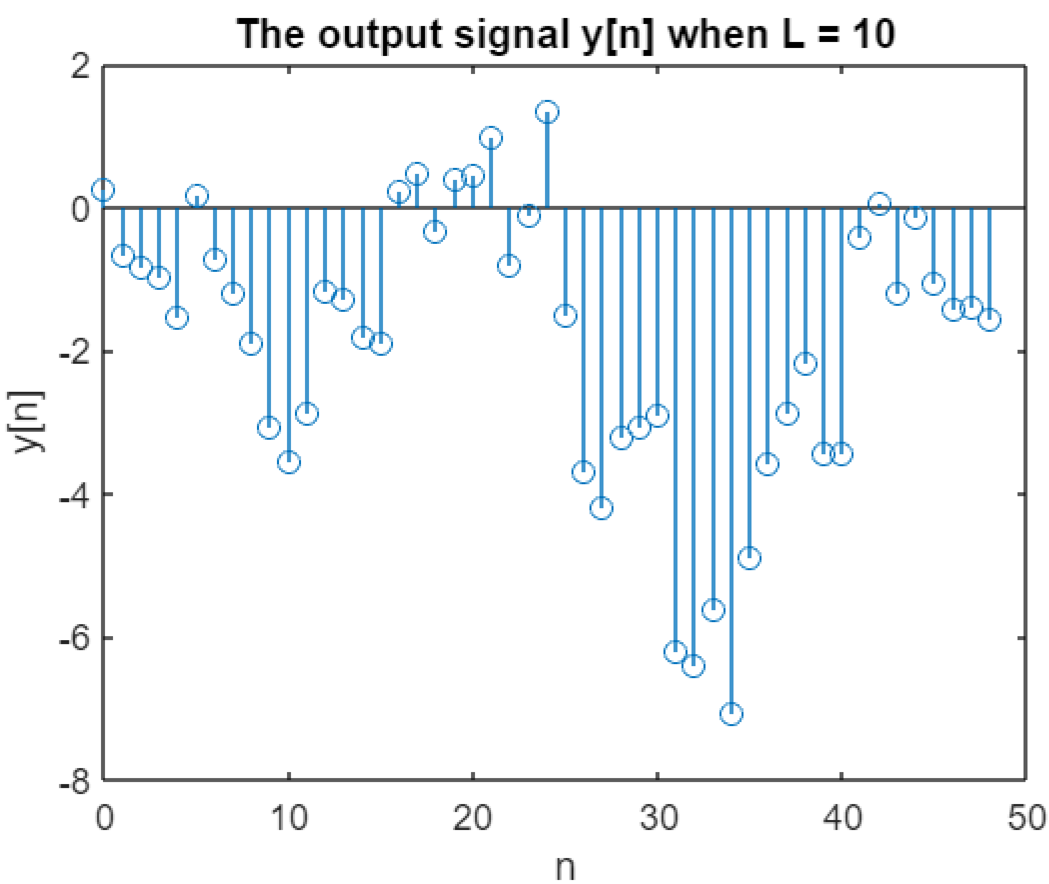



**Step 2:** Plot the output signal y[n] for different values of L

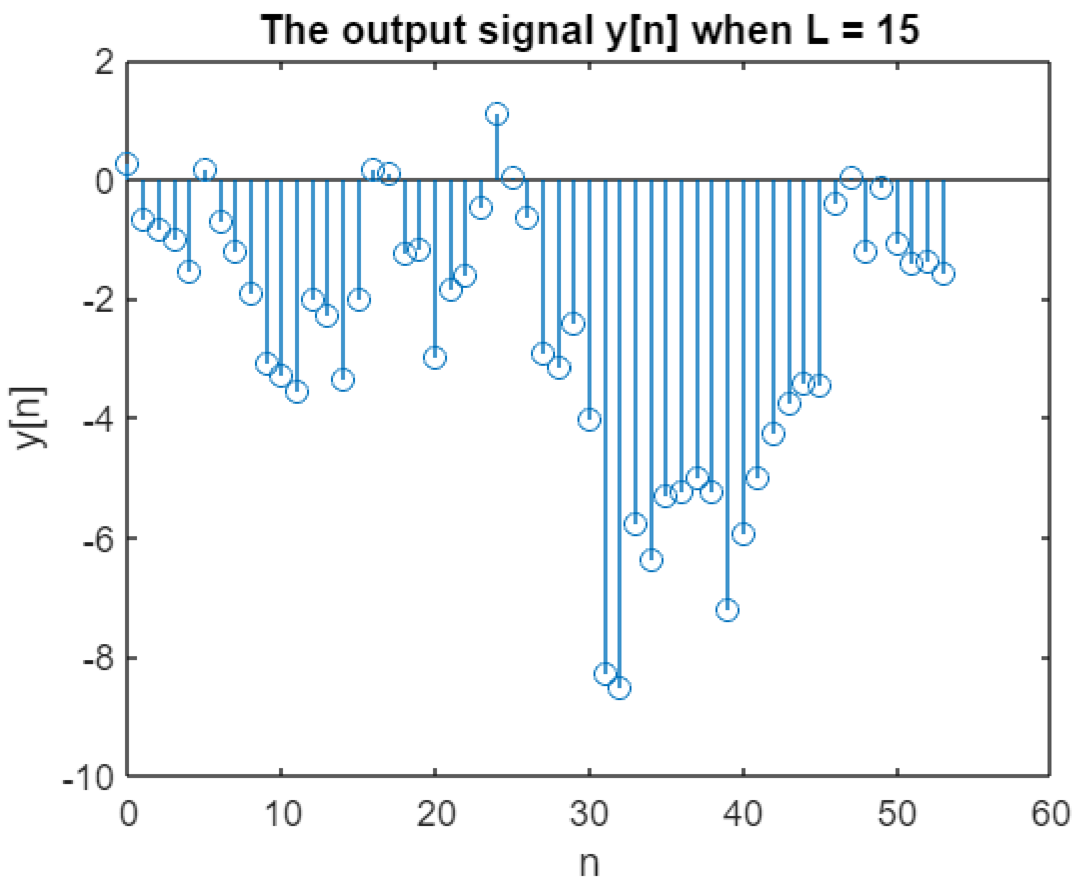
```
for l=5:5:30
    L = l;
    h = ones(1,L);
    y = conv(x,h);
    figure;
    stem(0:N+L-2,y);
    title(['The output signal y[n] when L = ',num2str(L)]);
    xlabel('n');
    ylabel('y[n]');
end
```

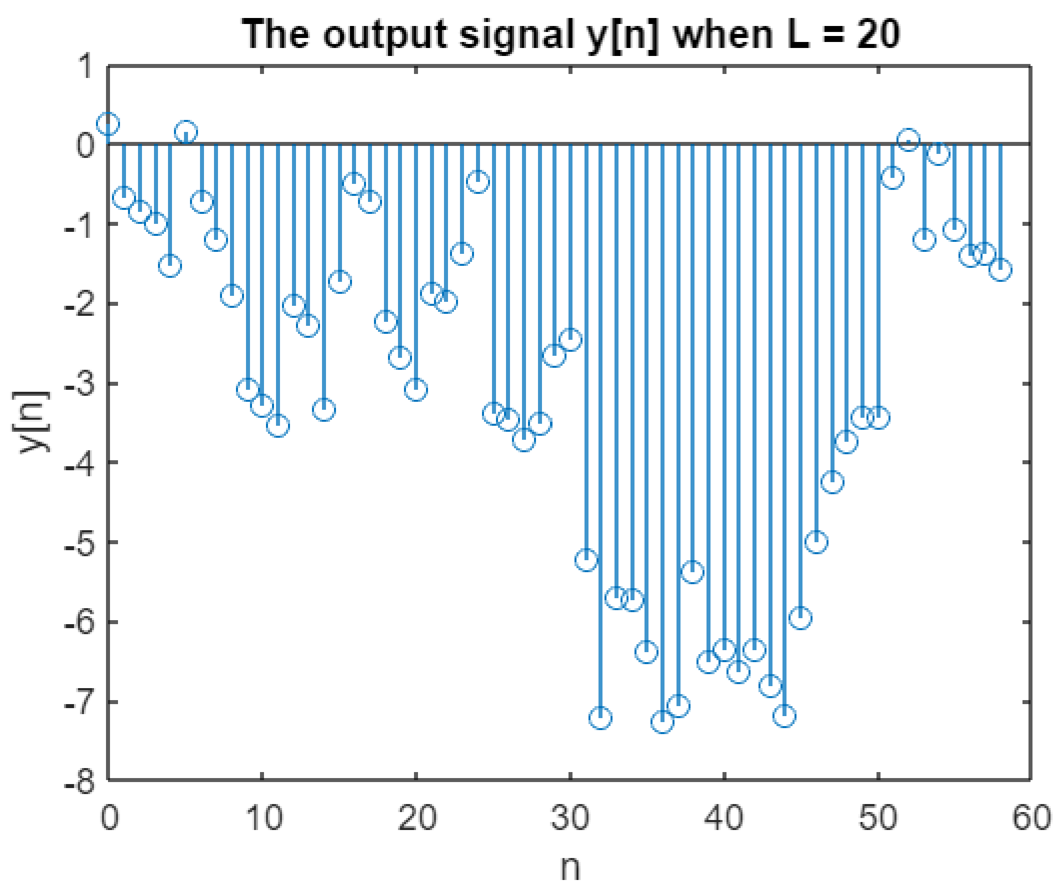


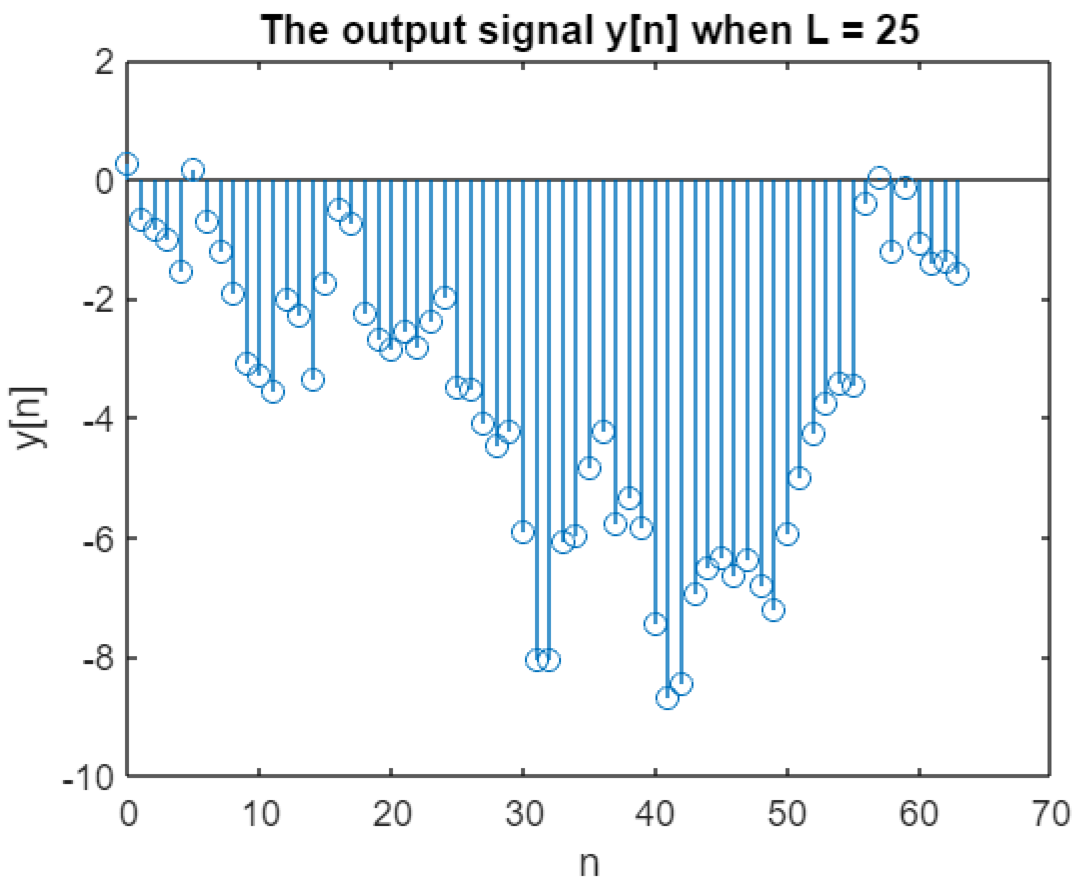
The output signal y[n] when L = 5
4
3
2
1
0
-1
-2
-3
-4
-5
y[n]
0
10
20
30
40
50
n

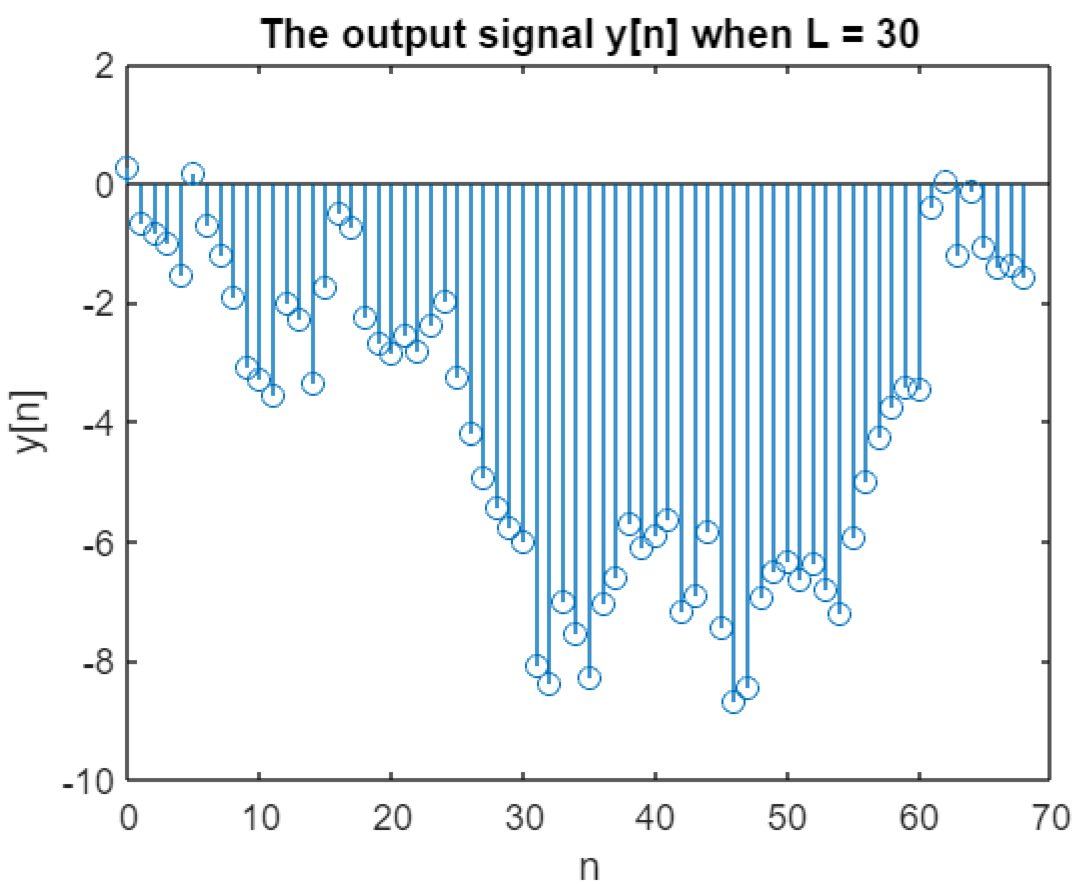



The output signal y[n] when L = 10
2
0
-2
-4
-6
-8
y[n]
0
10
20
30
40
50
n


The output signal y[n] when L = 15
y[n]
n
2
0
-2
-4
-6
-8
-10
0
10
20
30
40
50
60


The output signal y[n] when L = 20
y[n]
n
1
0
-1
-2
-3
-4
-5
-6
-7
-8
0
10
20
30
40
50
60


The output signal y[n] when L = 25
y[n]
n
2
0
-2
-4
-6
-8
-10
0
10
20
30
40
50
60
70


The output signal y[n] when L = 30
y[n]
n
2
0
-2
-4
-6
-8
-10
0
10
20
30
40
50
60
70