CONVOLUTION AND PROPERTIES OF CONVOLUTION

LAB # 08

CSE301L Signals & Systems Lab

Submitted by: **Shah Raza**

Registration No: **18PWCSE1658**

Class Section: **B**

"On my honor, as a student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: ______________

Submitted to: **Engr. Durr-e-Nayab**

Friday, July 24th, 2020

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Lab Objectives:

Objectives of this lab are as follows:

- Making Signals Causal and Non-Causal
- Convolution
- Properties of Convolution

Task # 1:

Sample the signal given in the above example to get its discrete-time counterpart. Make the resultant signal causal. Display the lollipop plot of each signal.

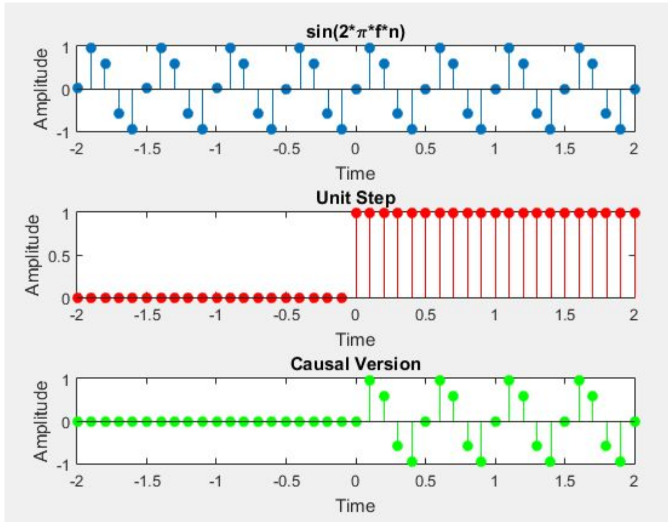
Problem Analysis:

To make a signal causal we multiply the given signal with a unit step signal where u[n] is one only for n>=0.

Code:

```
n= -2:1/10:2;
x1 = sin(2*pi*2*n);
subplot(3,1,1);
stem(n,x1,'filled');
xlabel('Time');
ylabel('Amplitude');
title('sin(2*\pi*f*n)');
u = (n>=0);
x2 = x1.*u;
subplot(3,1,2);
stem(n,u, 'r','filled');
xlabel('Time');
ylabel('Amplitude');
title('Unit Step');
subplot(3,1,3);
stem(n,x2, 'g','filled');
xlabel('Time');
ylabel('Amplitude');
title('Causal Version');
```

Output:



Task # 2:

A signal is said to be anti-causal if it exists for values of n<0. Make the signal given in above example anti-causal.

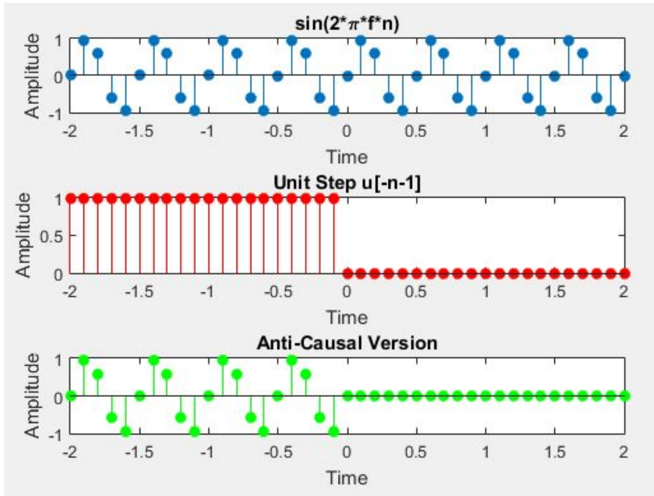
Problem Analysis:

To make a signal causal we multiply the given signal with a unit step signal where u[n] is one only for n<0.

Code:

```
n= -2:1/10:2;
x1 = sin(2*pi*2*n);
subplot(3,1,1);
stem(n,x1,'filled');
xlabel('Time');
ylabel('Amplitude');
title('sin(2*\pi*f*n)');
u = (n<0);
x2 = x1.*u;
subplot(3,1,2);
stem(n,u, 'r','filled');
xlabel('Time');
ylabel('Amplitude');
title('Unit Step u[-n-1]');
subplot(3,1,3);
stem(n,x2, 'g','filled');
xlabel('Time');
ylabel('Amplitude');
title('Anti-Causal Version')
```

Output:



Task # 3:

Create a function by name of sig_causal in matlab that has two input arguments: (i) a discrete-time signal, and (ii) a position vector. The function should make the given signal causal and return the resultant signal to the calling program.

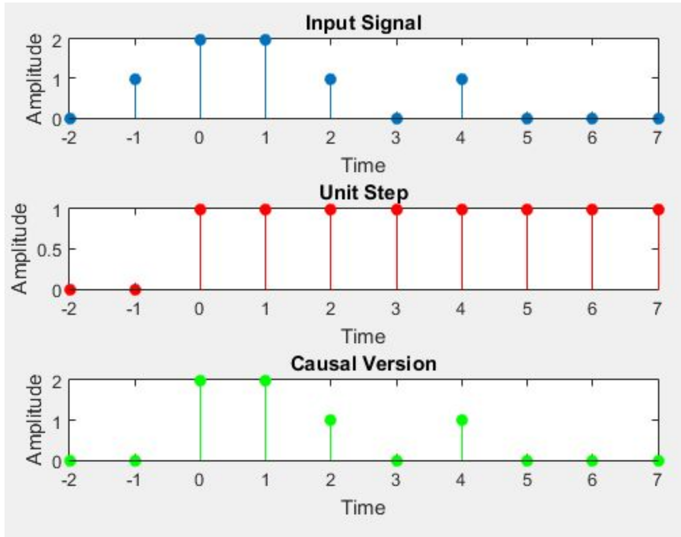
Problem Analysis:

To make a function that converts a given signal into causal we multiply the given signal with a unit step signal where u[n] is one only for n>=0.

Code:

```
function Sig_causal(n,x)
subplot(3,1,1);
stem(n,x,'filled');
xlabel('Time');
ylabel('Amplitude');
title('Input Signal');
u = (n>=0);
x1 = x.*u;
subplot(3,1,2);
stem(n,u, 'r','filled');
xlabel('Time');
ylabel('Amplitude');
title('Unit Step');
subplot(3,1,3);
stem(n,x1, 'g','filled');
xlabel('Time');
ylabel('Amplitude');
title('Causal Version');
end
```

Output:



Task # 4:

Convolve the following signals:

x =[2 4 6 4 2]; h =[3 -1 2 1];

Plot the input signal as well as the output signal.

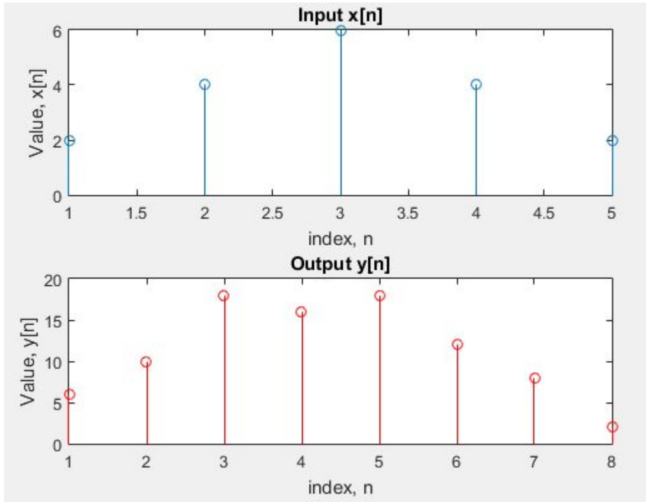
Problem Analysis:

To convolve the two signals, we use the function conv() and then plot the results.

Code:

```
h = [3 -1 2 1];
x = [2 4 6 4 2];
y = conv(h,x);
subplot(2,1,1);
stem(x);
title('Input x[n]');
xlabel('index, n');
ylabel('Value, x[n]');
subplot(2,1,2);
stem(y,'r');
title('Output y[n]');
xlabel('index, n');
ylabel('Value, y[n]');
```

Output:



Task # 5:

Convolution is associative. Given the three signal x1[n], x2[n], and x3[n] as:
x1[n]= [3 1 1]; x2[n]= [4 2 1]; x3[n]= [3 2 1 2 3];
Show that (x1[n] * x2[n]) * x3[n] = x1[n] * (x2[n] * x3[n]).

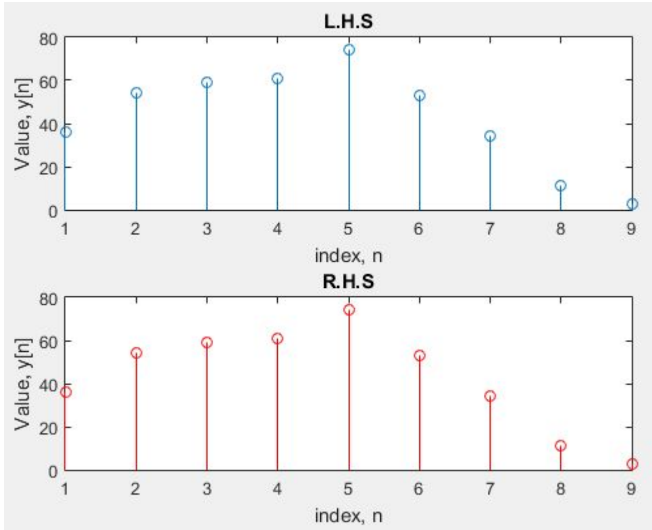
Problem Analysis:

To verify the associative property of convolution, first convolve x1 and x2 and then convolve the result with x3 on LHS. On the other side first convolve x2 and x3 and then convolve the result with x1.

Code:

```
x1 = [3 1 1];
x2 = [4 2 1];
x3 = [3 2 1 2 3];
y1 = conv(x1,x2);
y2 = conv(y1,x3);
subplot(2,1,1);
stem(y2);
title('L.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
y3=conv(x2,x3);
y4=conv(y3,x1);
subplot(2,1,2);
stem(y4,'r');
title('R.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
```

Output:



Task # 6:

Convolution is commutative. Given x[n] and h[n] as:

X[n]=[1 3 2 1]; H[n]=[1 1 2];

Show that x[n] * h[n] = h[n] * x[n].

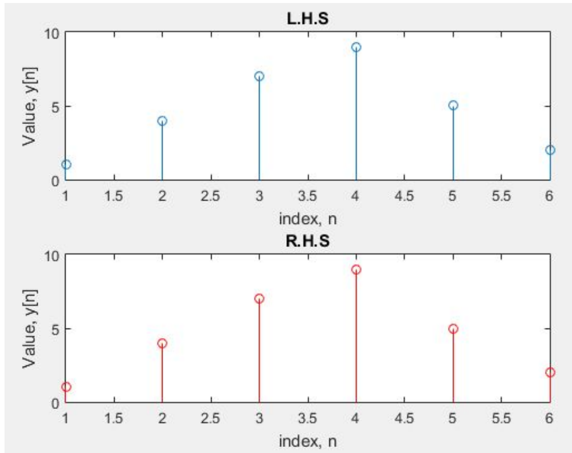
Problem Analysis:

To verify the commutative property of convolution, first convolve x with h on LHS. On the other side convolve h with x.

Code:

```
x = [1 3 2 1];
H = [1 1 2];
y = conv(x,H);
subplot(2,1,1);
stem(y);
title('L.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
y1=conv(H,x);
subplot(2,1,2);
stem(y1,'r');
title('R.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
```

Output:



Task # 7:

Given the impulse response of the systems as:

$$h[n]= 2\delta[n] + \delta[n-1]+2\delta[n-2]+ 4\delta[n-3]+ 3\delta[n-4]$$

If the input $x[n] = \delta[n]+ 4\delta[n-1] +3\delta[n-2] + 2\delta[n-3]$ is applied to the system, determine the output of the system.

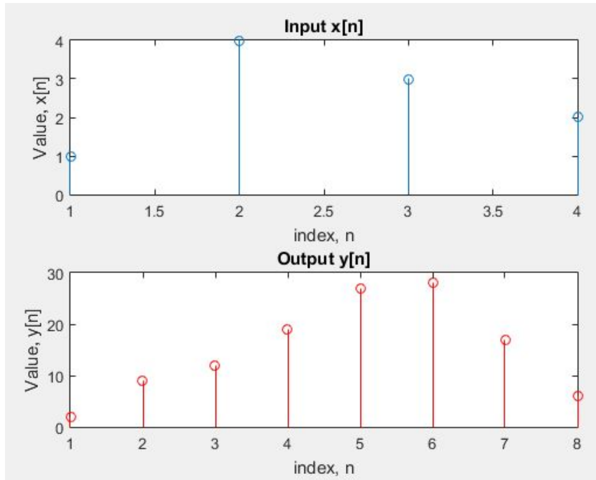
Problem Analysis:

To find the output of the system convolve x with h, using the function conv().

Code:

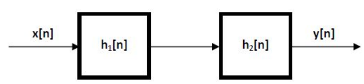
```
h = [2 1 2 4 3];
x = [1 4 3 2];
y = conv(x,h);
subplot(2,1,1);
stem(x);
title('Input x[n]');
xlabel('index, n');
ylabel('Value, x[n]');
subplot(2,1,2);
stem(y,'r');
title('Output y[n]');
xlabel('index, n');
ylabel('Value, y[n]');
```

Output:



Task # 8:

Two systems are connected in cascade.



h1[n]=[1 3 2 1]; h2[n]=[1 1 2]

If the input x[n] = δ[n]+ 4δ[n-1] +3δ[n-2] +2δ[n-3] is applied, determine the output.

Problem Analysis:

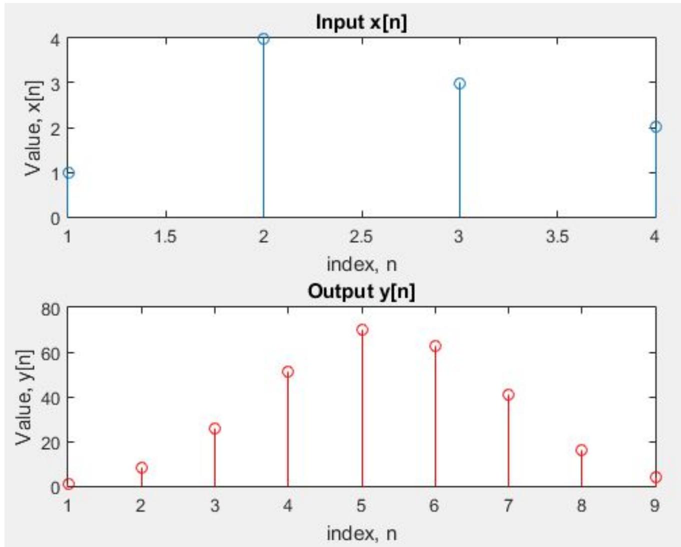
As the two systems are connected in series, we first convolve x and h1 and then convolve the answer with h2.

Code:

```
h1 = [1 3 2 1];
h2 = [1 1 2];
x = [1 4 3 2];
y = conv(x,h1);
y1= conv(y,h2);
subplot(2,1,1);
stem(x);
title('Input x[n]');
xlabel('index, n');
ylabel('Value, x[n]');
subplot(2,1,2);
stem(y1,'r');
title('Output y[n]');
```

```
xlabel('index, n');
ylabel('Value, y[n]');
```

Output:



Task # 9:

Given the signals:

$$x1[n]= 2\delta[n] -3\delta[n-1]+ 3\delta[n-2] +4\delta[n-3] -2\delta[n-4]$$
$$x2[n]= 4\delta[n]+ 2\delta[n-1]+ 3\delta[n-2] - \delta[n-3] -2\delta[n-4]$$
$$x3[n]= 3\delta[n]+ 5\delta[n-1]-3\delta[n-2] +4 \delta[n-3]$$

Verify that

$$x1[n] * (x2[n] * x3[n]) = (x1[n] * x2[n]) * x3[n]$$
$$x1[n] * x2[n]= x2[n] * x1[n]$$

Problem Analysis:

To verify the associative property of convolution, first convolve x2 and x3 and then convolve the result with x1 on LHS. On the other side first convolve x1 and x2 and then convolve the result with x3.

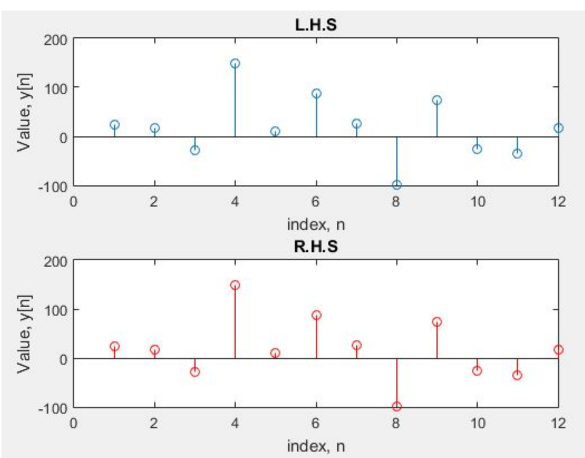
To verify the commutative property of convolution, first convolve x1 with x2 on LHS. On the other side convolve x2 with x1.

Code:

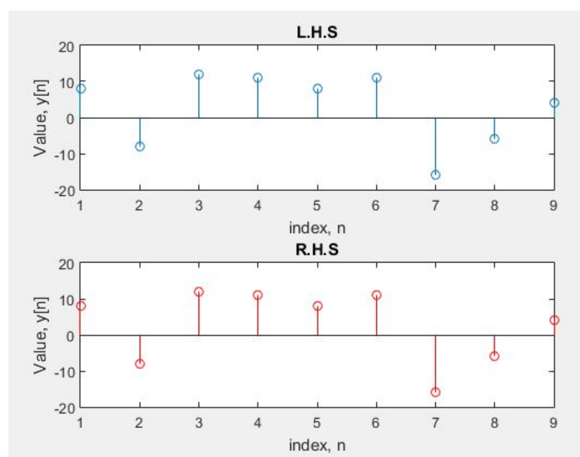
```
x1 = [2 -3 3 4 -2];
x2 = [4 2 3 -1 -2];
x3 = [3 5 -3 4];
figure(1); % for association law
```

```
y = conv(x2,x3);
y1= conv(y,x1);
subplot(2,1,1);
stem(y1);
title('L.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
y2 = conv(x1,x2);
y3= conv(y2,x3);
subplot(2,1,2);
stem(y3,'r');
title('R.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
figure(2); %for commutative law
y4 = conv(x1,x2);
subplot(2,1,1);
stem(y4);
title('L.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
y5 = conv(x2,x1);
subplot(2,1,2);
stem(y5,'r');
title('R.H.S');
xlabel('index, n');
ylabel('Value, y[n]');
```

Output:



Fig(1)



Fig(2)