Lab Assignment – 03 – Spring 2020

Signal & Systems

Atharva Deshpande || UG-1 || CSE || S20190010043

Indian Institute of Information Technology, Sri City.

Sri City, Andhra Pradesh.

January 30, 2020.

Summary:

- In this assignment we learnt how to find the convolution of two signals.
- Two methods were mainly used. One of them being using 'for loop'.
- The other method was a straight forward way to find convolution using the in-built 'conv' function and we basically used this method to verify the previous one.

---

1) $y_1[n] = (x_1[n]+x_2[n]) * h[n]$

   $y_2[n] = x_1[n]*h[n] + x_2[n]*h[n]$

   x_1, x_2 and h were given.

   We had to plot x_1, x_2 y_1 y_2 and h.

   Here's the code for 1) a):

   ➔ First, we find y_1 & y_2 by using 'for loop'.

   ➔ Then we run the code and get its output.

atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\lab3one.m

lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
%Second part x1*h + x2*h

o=5;p=0:o-1; %Since there are 5 num in x1,h & they work as indices of values of x1,h: (0,1,2,3,4)
x1=zeros(size(p));x1(p==0)=-2;x1(p==1)=2;x1(p==2)=3;x1(p==3)=1;x1(p==4)=12;
subplot(3,3,1);stem(p,x1);title('x1[n]');
h=zeros(size(p));h(p==0)=3;h(p==1)=-2;h(p==2)=-5;h(p==3)=1;h(p==4)=-4;
subplot(3,3,5);stem(p,h);title('h[n]');
m=length(x1);n=length(h);l=m+n-1;
x1e=zeros(1,l);
h1e=zeros(1,l);
y1=h1e;
x1e(1:m)=x1;
x1e(m+1:l)=0;
h1e(1:n)=h;
h1e(n+1:l)=0;
for i=1:l
    y1(i)=0;
    for k=1:i
        y1(i)=y1(i)+h1e(k)*x1e(i-k+1);
    end
end
b=4;j=0:b-1; %Since there are 4 numbers in x2 & they work as indices of values of x2 : (0,1,2,3)
x2=zeros(size(j));
x2(j==0)=1;x2(j==1)=-1;x2(j==2)=4;x2(j==3)=-2;
```

atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\lab3one.m

lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
subplot(3,3,3);stem(j,x2);title('x2[n]');
f=length(x2);d=n;l=f+d-1;
x2e=zeros(1,l);
h2e=zeros(1,l);
y2=h2e;
x2e(1:f)=x2;
x2e(f+1:l)=0;
h2e(1:d)=h;
h2e(d+1:l)=0;
for g=1:l
    y2(g)=0;
    for k=1:g
        y2(g)=y2(g)+h2e(k)*x2e(g-k+1);
    end
end
%Since y1 has 9 elements and y2 has 8, for addition purpose we have to pad y2(9)=0
y2(9)=0;
%We can also make an algorithm that if y1 not equal to y2
%The lower one will be padded with zeroes until the end i.e number of values in y1

%As we know x1:(0->4) x2:(0->3) h(0->4) | max(x1,x2)=4 | y1+y2 E (0->8) = 9 nos (5+5-1)
v=0:8;
subplot(3,3,9);stem(v,y1+y2);title('y2[n]');
```

atharva deshpande ▸ Documents ▸ MATLAB

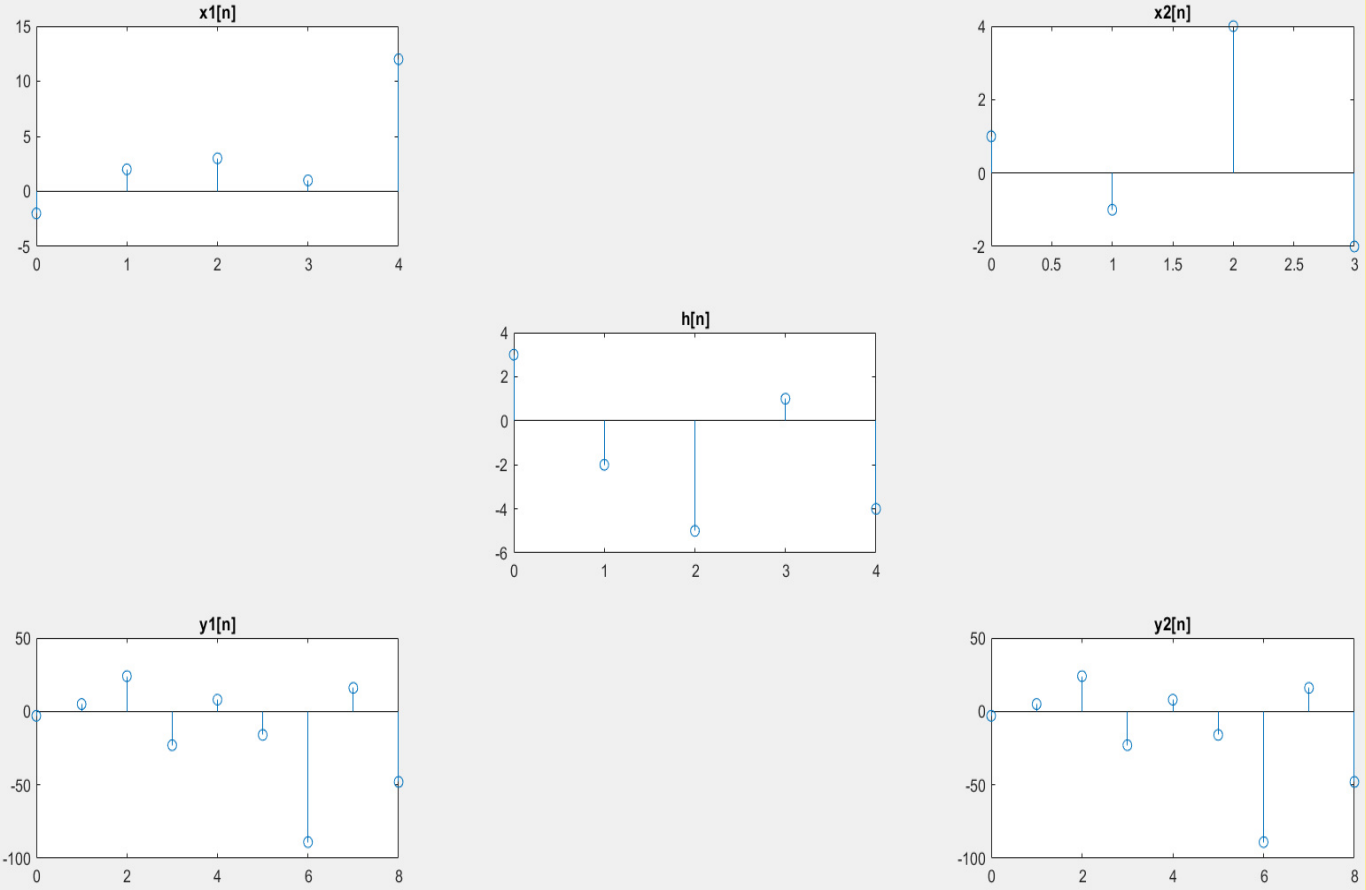
Editor - C:\Users\atharva deshpande\Documents\MATLAB\lab3one.m

lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
%First Part (x1+x2)*h

%For adding x1,x2 both need to be of the same length
for q=b+1:o
    x2(q)=0;
end
z=x1+x2;
mm=length(z);nn=n;l=mm+nn-1;
z1e=zeros(1,l);
hz1e=zeros(1,l);
y3=hz1e;
z1e(1:mm)=z;
z1e(mm+1:l)=0;
hz1e(1:nn)=h;
hz1e(nn+1:l)=0;
for i=1:l
    y3(i)=0;
    for k=1:i
        y3(i)=y3(i)+hz1e(k)*z1e(i-k+1);
    end
end
subplot(3,3,7);stem(v,y3);title('y1[n]');
```

Here's the output:



Here's the code for 1) b):

➔ First, we find y_1 & y_2 by using 'conv' function.

➔ Then we run the code and get its output.

atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\verify3one.m

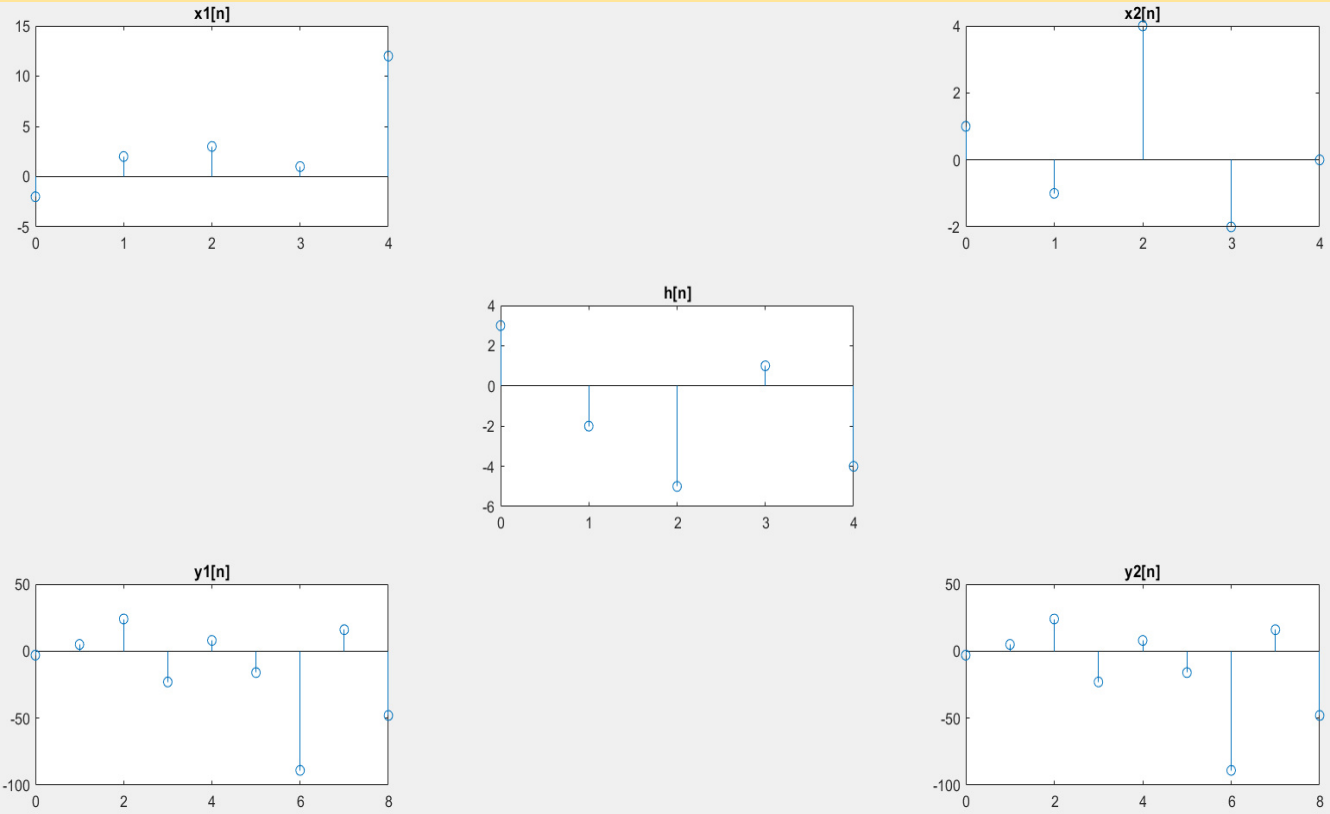
lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
clc
clear
close all

%

m=5;n=0:m-1;%m=max(x1,x2)=max(5,4)=5
x1=zeros(size(n));
x1(n==0)=-2;x1(n==1)=2;x1(n==2)=3;x1(n==3)=1;x1(n==4)=12;
subplot(3,3,1);stem(n,x1);title('x1[n]');
x2=zeros(size(n));
x2(n==0)=1;x2(n==1)=-1;x2(n==2)=4;x2(n==3)=-2;
subplot(3,3,3);stem(n,x2);title('x2[n]');
h=zeros(size(n));
h(n==0)=3;h(n==1)=-2;h(n==2)=-5;h(n==3)=1;h(n==4)=-4;
subplot(3,3,5);stem(n,h);title('h[n]');
z1=x1+x2;%First Part
y1=conv(z1,h);
p=0:2*m-2;%p=0:8->9 values or from 0:(m+n-1)-1 as we are starting from 0 & (m,n)=(5,4)
subplot(3,3,7);stem(p,y1);title('y1[n]');
z2=conv(x1,h);z3=conv(x2,h);y2=z2+z3;%Second Part
subplot(3,3,9);stem(p,y2);title('y2[n]');
```

Here's the output:



As we can see, the output signals y_1 & y_2 are coming out to be one and the same in both the ways (conv and for loop). So, the given output is verified successfully.

2) Now x and h were already given, we had to find the output signal y.

Here's the code for 2) a):

➔ First, we find y by using 'for loop'.

➔ Then we run the code and get its output.

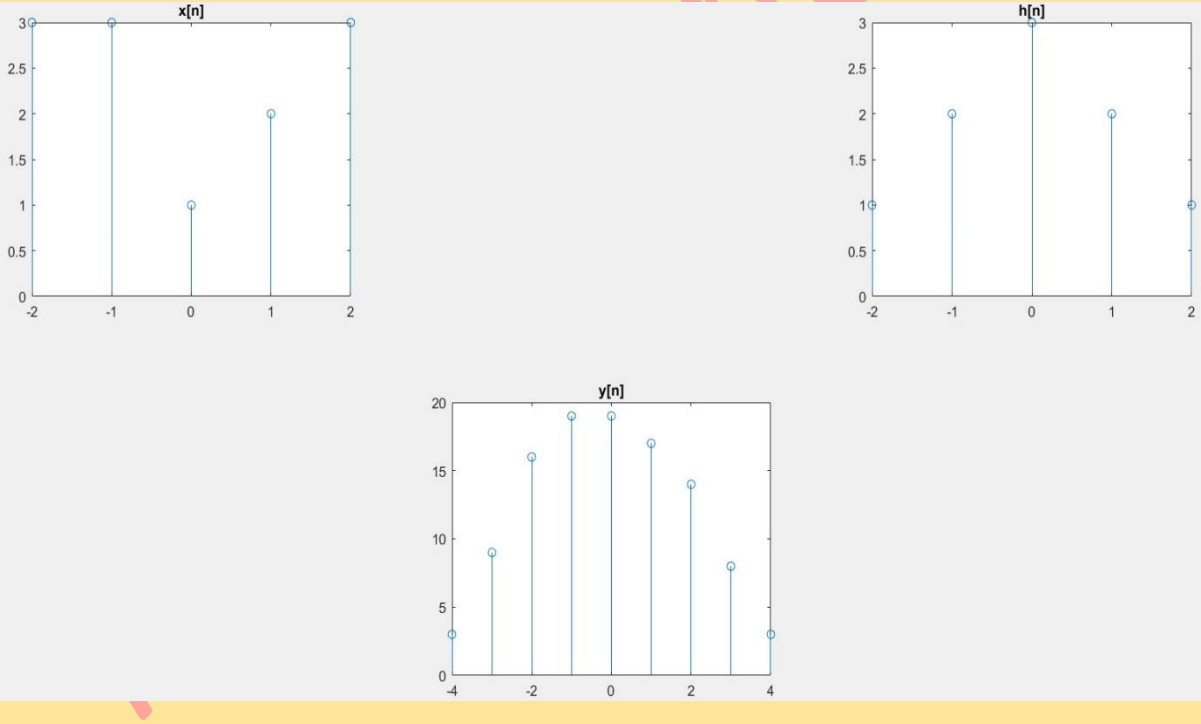
atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\lab3two.m

lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
o=2;p=-o:1:o;
x=zeros(size(p));
x(p==-2)=3;x(p==-1)=3;x(p==0)=1;x(p==1)=2;x(p==2)=3;
subplot(2,3,1);stem(p,x);title('x[n]');
h=zeros(size(p));
h(p==-2)=1;h(p==-1)=2;h(p==0)=3;h(p==1)=2;h(p==2)=1;
subplot(2,3,3);stem(p,h);title('h[n]');
m=length(x);n=length(h);l=m+n-1;
xe=zeros(1,l);
he=zeros(1,l);
y=he;
xe(1:m)=x;
xe(m+1:l)=0;
he(1:n)=h;
he(n+1:l)=0;
for i=1:l
    y(i)=0;
    for k=1:i
        y(i)=y(i)+he(k)*xe(i-k+1);
    end
end
v=-2*o:1:2*o;%Needless to say it should be double the range (here=2) from both the sides.
subplot(2,3,5);stem(v,y);title('y[n]');
```

Here's the output:



Here's the code for 2) b):

➔ First, we find y by using 'conv' function.

➔ Then we run the code and get its output.

atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\verify3two.m

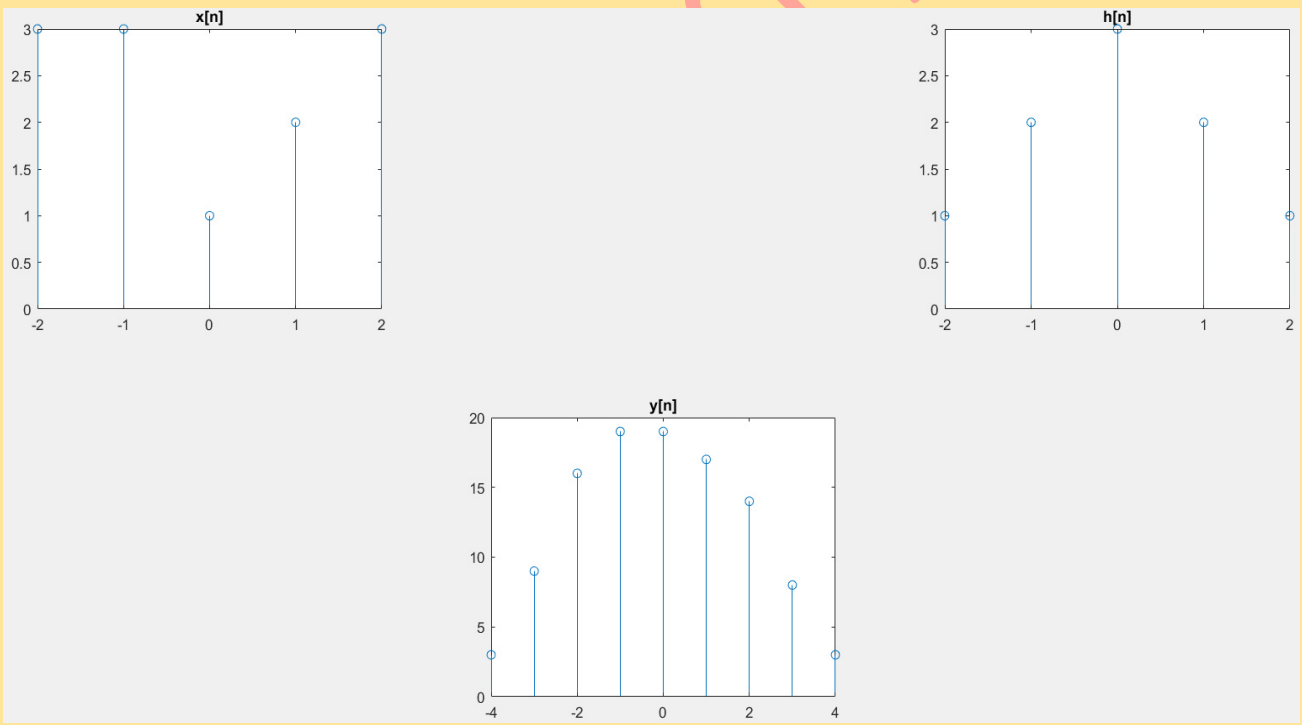
lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
clc
clear
close all

%

m=2;
n=-m:1:m;
x=zeros(size(n));
x(n==-2)=3;x(n==-1)=3;x(n==0)=1;x(n==1)=2;x(n==2)=3;
subplot(2,3,1);stem(n,x);title('x[n]');
h=zeros(size(n));
h(n==-2)=1;h(n==-1)=2;h(n==0)=3;h(n==1)=2;h(n==2)=1;
subplot(2,3,3);stem(n,h);title('h[n]');
y=conv(x,h);
p=-2*m:1:2*m;%Needless to say it should be double the range (here=2) from both the sides.
subplot(2,3,5);stem(p,y);title('y[n]');
```

Here's the output:

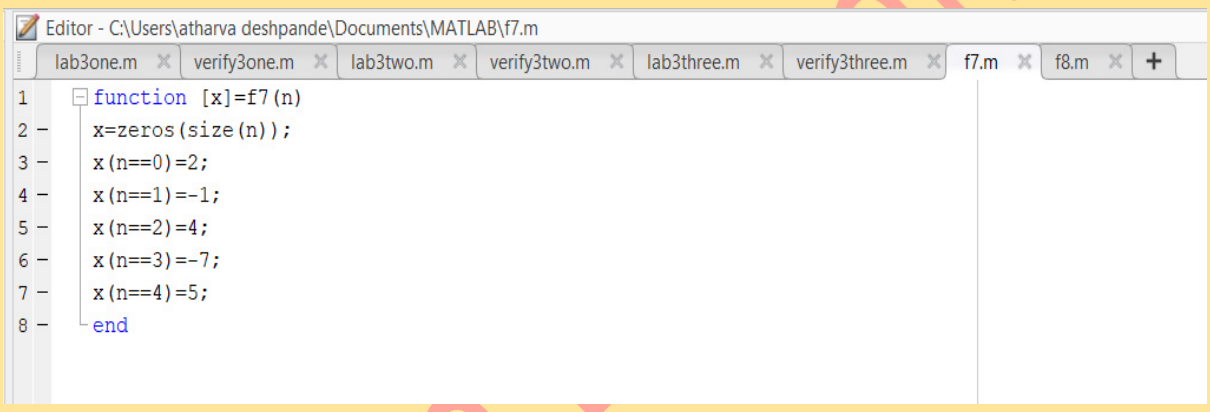


As we can see, the output signal y is coming out to be one and the same in both the ways (conv and for loop). So, the given output is verified successfully.

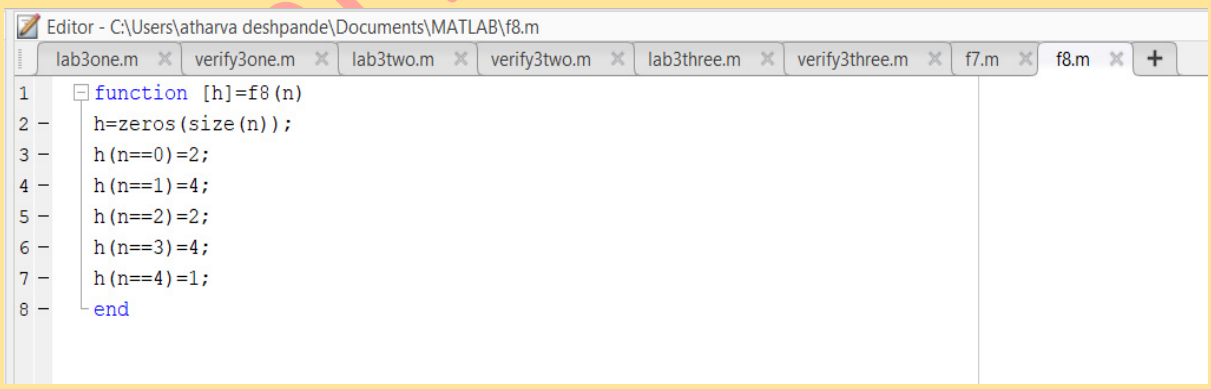
3) x and h are given. We have to find the output for shifted and scaled version of x and h. Basically, there are two parts in one script and we have to verify the output using the in-built function.

Here's the code for 3) a):

- ➔ First, we find y by using 'for loop'.
- ➔ Then we run the code and get its output.
- ➔ Here we have used two functions f7 & f8.



f7 function



f8 function

atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\lab3three.m

lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
o=5;p=-(o-1):1:o-1;
x1=f7(p);subplot(2,2,1);stem(p,x1);title('x[n]');
h1=f8(p);subplot(2,2,2);stem(p,h1);title('h[n]');
h2=f8(-p);
m=5;n=5;l=m+n-1;
x1e=zeros(1,l);
h2e=zeros(1,l);
y1=h2e;
x1e(1:m)=x1(o:2*o-1);
x1e(m+1:l)=0;
h2e(1:n)=h2(1:o);
h2e(n+1:l)=0;
for i=1:l
    y1(i)=0;
    for k=1:i
        y1(i)=y1(i)+h2e(k)*x1e(i-k+1);
    end
end
v=-o+1:1:o-1;
```

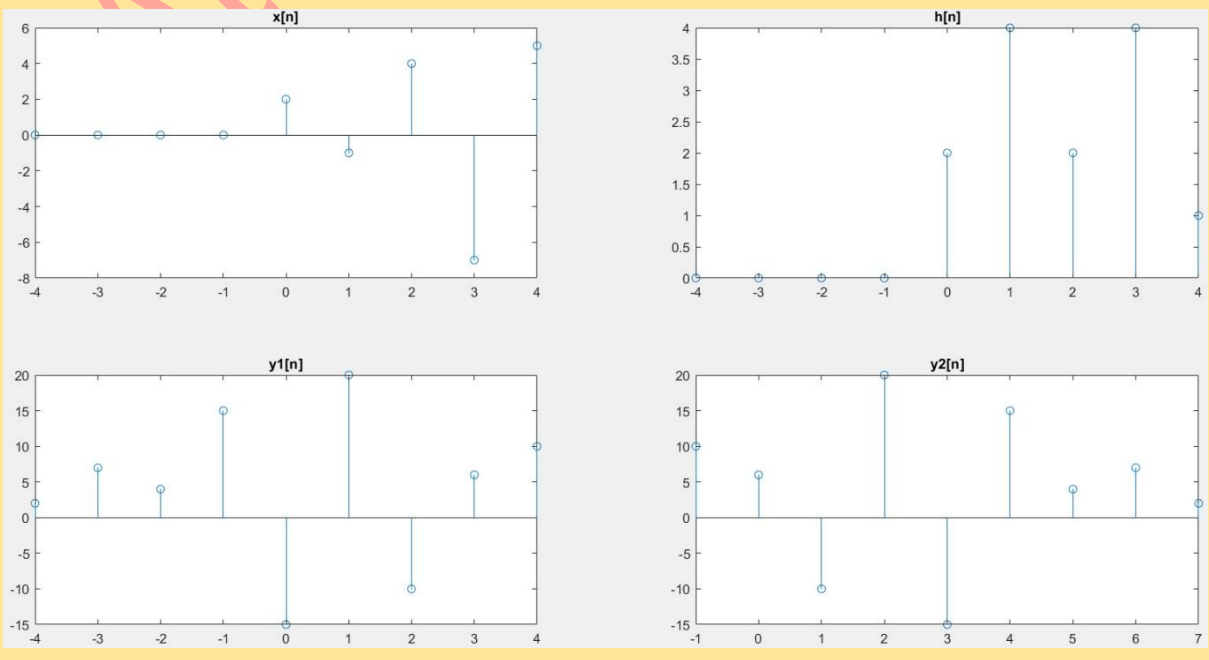
atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\lab3three.m

lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
v=-o+1:1:o-1;
subplot(2,2,3);stem(v,y1);title('y1[n]');
x2=f7(3-p);
m=5;n=5;l=m+n-1;
x2e=zeros(1,l);
h1e=zeros(1,l);
y2=h1e;
x2e(1:m)=x2(4:8);
x2e(m+1:l)=0;
h1e(1:n)=h1(o:2*o-1);
h1e(n+1:l)=0;
for i=1:l
    y2(i)=0;
    for k=1:i
        y2(i)=y2(i)+h1e(k)*x2e(i-k+1);
    end
end
u=3-(m-1):1:((m-1)-1+(m-1));%x:(-1:3) h:(0:4) hence y:(-1+0:4+3)
subplot(2,2,4);stem(u,y2);title('y2[n]');
```

Here's the output:



Here's the code for 3) b):

➔ First, we find y by using 'conv' function.

➔ Then we run the code and get its output.

atharva deshpande ▸ Documents ▸ MATLAB

Editor - C:\Users\atharva deshpande\Documents\MATLAB\verify3three.m

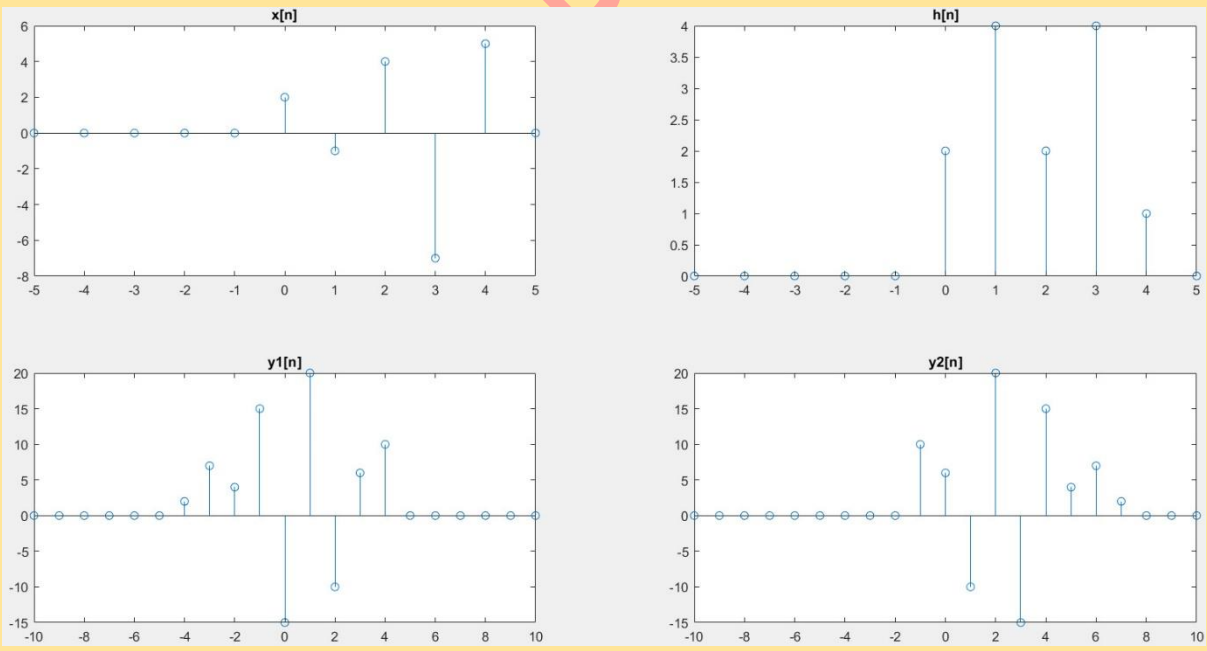
lab3one.m | verify3one.m | lab3two.m | verify3two.m | lab3three.m | verify3three.m

```
clc
clear
close all

%

m=5;
n=-m:1:m;
x=zeros(size(n));
x1=f7(n);subplot(2,2,1);stem(n,x1);title('x[n]');%real input signal
h1=f8(n);subplot(2,2,2);stem(n,h1);title('h[n]');%real system equation h
h2=f8(-n);
y1=conv(x1,h2);
p=-2*m:1:2*m;
subplot(2,2,3);stem(p,y1);title('y1[n]');
x2=f7(3-n);
y2=conv(x2,h1);
subplot(2,2,4);stem(p,y2);title('y2[n]');
```

Here's the output:



As we can see, the output signals y_1 & y_2 are coming out to be one and the same in both the ways (conv and for loop). So, the given output is verified successfully.