**Program #1**

EE5350

Burhan Ayub Wani

1001050444

CONV1.m

function [y Ny]=CONV1(x,Nx,h,Nh)

Ny=Nx+Nh;

for n=0:Ny

y(n+1)=0;

for k=0:Nh

if(n-k+1>=1 && n-k+1<=Nx+1)

y(n+1) = y(n+1) + h(k+1)\*x(n-k+1); % Formula for convolution

end

end

end

end

Main.m

clc;

clear all;

close all;

%defining h(n) signal with 25 samples which is defined as:

for n=0:8

h(n+1)=n;

end

for n=9:16

h(n+1)=16-n;

end

for n=17:24

h(n+1)=0;

end

Nh=24;

%plot figure

figure;

stem (0:Nh,h);

title ('signal h');

xlabel ('n');

ylabel ('h(n)');

%defining x(n) signal with 90 samples which is defined as::-

for n=0:4

x(n+1)=0;

end

for n=5:15

x(n+1)=1;

end

for n=16:19

x(n+1)=0;

end

for n=20:35

x(n+1)=-1.5;

end

for n=36:44

x(n+1)=0;

end

for n=45:65

x(n+1)=2;

end

for n=66:89

x(n+1)=0;

end

Nx=89;

% plot figure

figure;

stem (0:Nx,x);

title ('signal x');

xlabel ('n');

ylabel ('x(n)');

%calling convolution function

[y Ny] = CONV1(x,Nx,h,Nh);

%plot figure

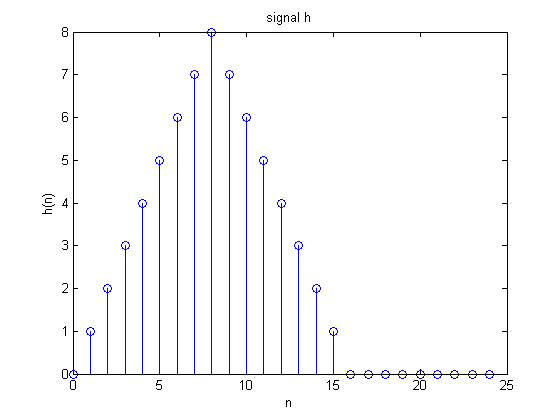
figure;

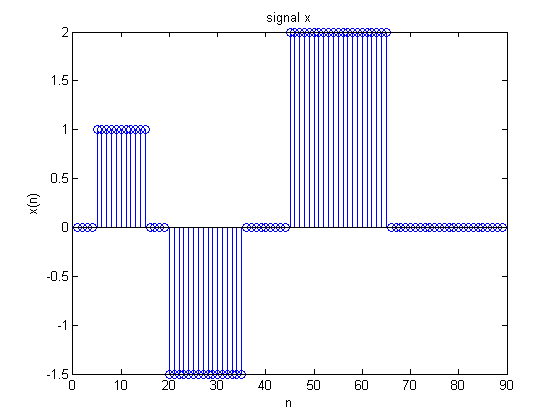
stem (0:Ny,y);

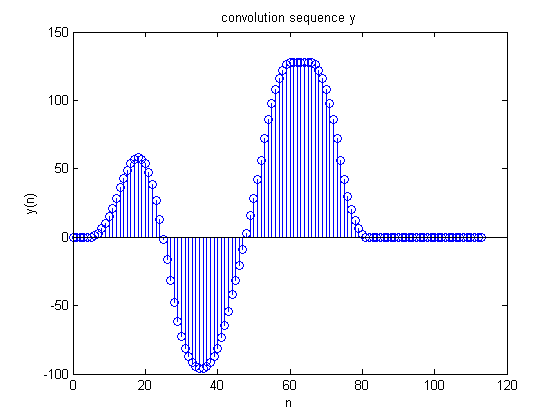
title ('convolution sequence y');

xlabel ('n');

ylabel('y(n)');







**Take home message:** We can calculate the convolution of real time signals in matlab which can be effective in designing filters. Although Matlab has its own Convolution function, we can write algorithms on our own to specify the same.

As can be seen from the plot of the Convolution result, it is simply the product of the samples of two signals at the same intervals of time. H(n) may be the impulse response of a filter and x(n) an input signal. Y(n) gives the output of the filter when the input signal is fed into it.