



UNDER THE SUPERVISION OF
DR. SYED SIBTAIN KHALID
(PROF.)

DIGITAL SIGNAL PROCESSING USING MATLAB PROGRAMMING

A MINOR-PROJECT BY
MUHAMMAD FEROZ KHAN
2018-333-041
V SEMESTER / III YEAR
BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

Program-1

Aim

Write MATLAB code for the following discrete-time signals:

- i. Impulse Sequence
- ii. Step Sequence
- iii. Ramp Sequence
- iv. Real Exponential Sequence
- v. Sinusoidal Sequence

Software used

MATLAB/GNU Octave/SciLab

Code

```
%IMPULSE SEQUENCE  
clc;  
n=-3:3;  
subplot(3,3,1);  
impulse= (n==0);  
stem(n,impulse);  
ylabel('amplitude');  
xlabel('n');  
title('Impulse Sequence')
```

```

%STEP SEQUENCE
n=-5:5;
subplot(3,3,2);
unit_step = (n>=0);
stem(n,unit_step);
ylabel('amplitude');
xlabel('n');
title('Step Sequence')

%RAMP SEQUENCE
n=-4:4;
subplot(3,3,3);
ramp_function = n.* (n>=0);
stem(ramp_function);
ylabel('amplitude');
xlabel('n');
title('Ramp Sequence')

%REAL EXPONENTIAL SEQUENCE
n=-20:20; %increasing exponential
subplot(3,3,4);
x_n=2.* (1.2.^n);
stem(n,x_n);
ylabel('amplitude');
xlabel('n');
title('Exponential Sequence-Increasing')

```

```

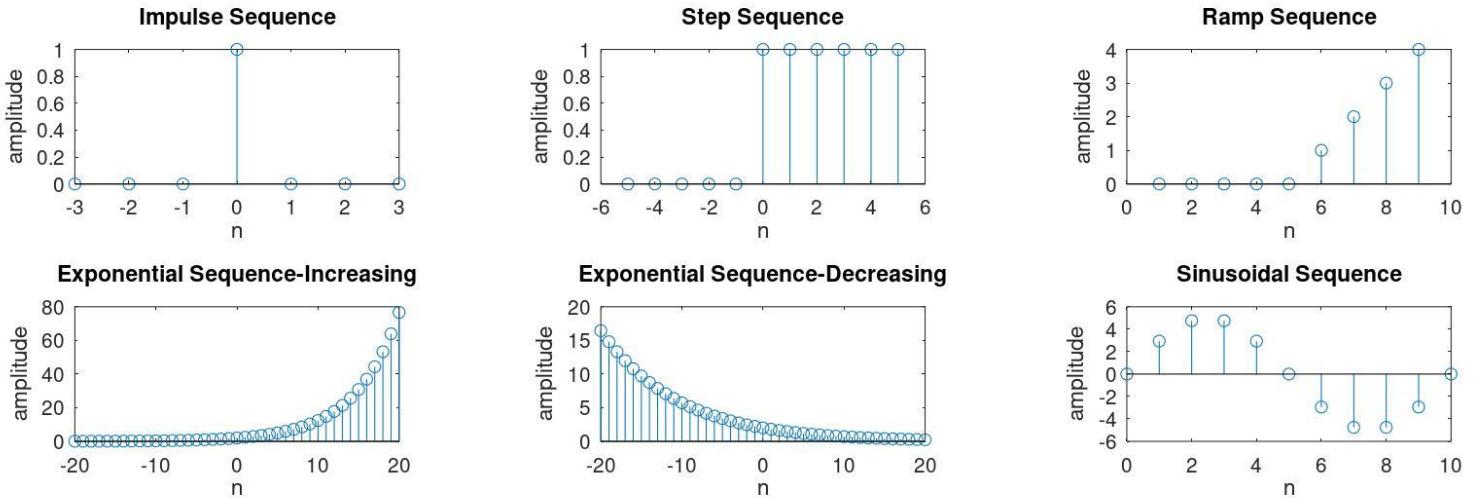
n=-20:20; %decaying exponential
subplot(3,3,5);
x_n=2.* (0.9.^n);
stem(n,x_n);
ylabel('amplitude');
xlabel('n');
title('Exponential Sequence-Decreasing')

%SINUSOIDAL SEQUENCE

A=5;
f=0.1;
n=0:1:10;
subplot(3,3,6);
xt=A*sin(2*pi*f*n);
stem(n,xt);
ylabel('amplitude');
xlabel('n');
title('Sinusoidal Sequence')

```

Output



Program-2

Aim

Write a MATLAB code for the addition of discrete time signal.

Software used

MATLAB/GNU Octave/SciLab

Code

```
x1 = [1,2,3,4,5];
n1 = [1:5];
x2 = [5,6,3,2];
n2 = [1:4];
subplot(3,1,1);
stem(n1,x1);
title("Signal 1");
xlabel("n");
ylabel("x(n)");
subplot(3,1,2);
stem(n2,x2);
title("Signal 2");
xlabel("n");
ylabel("x(n)");
subplot(3,1,3);
```

```
[y,n] = sigadd(x1,n1,x2,n2);

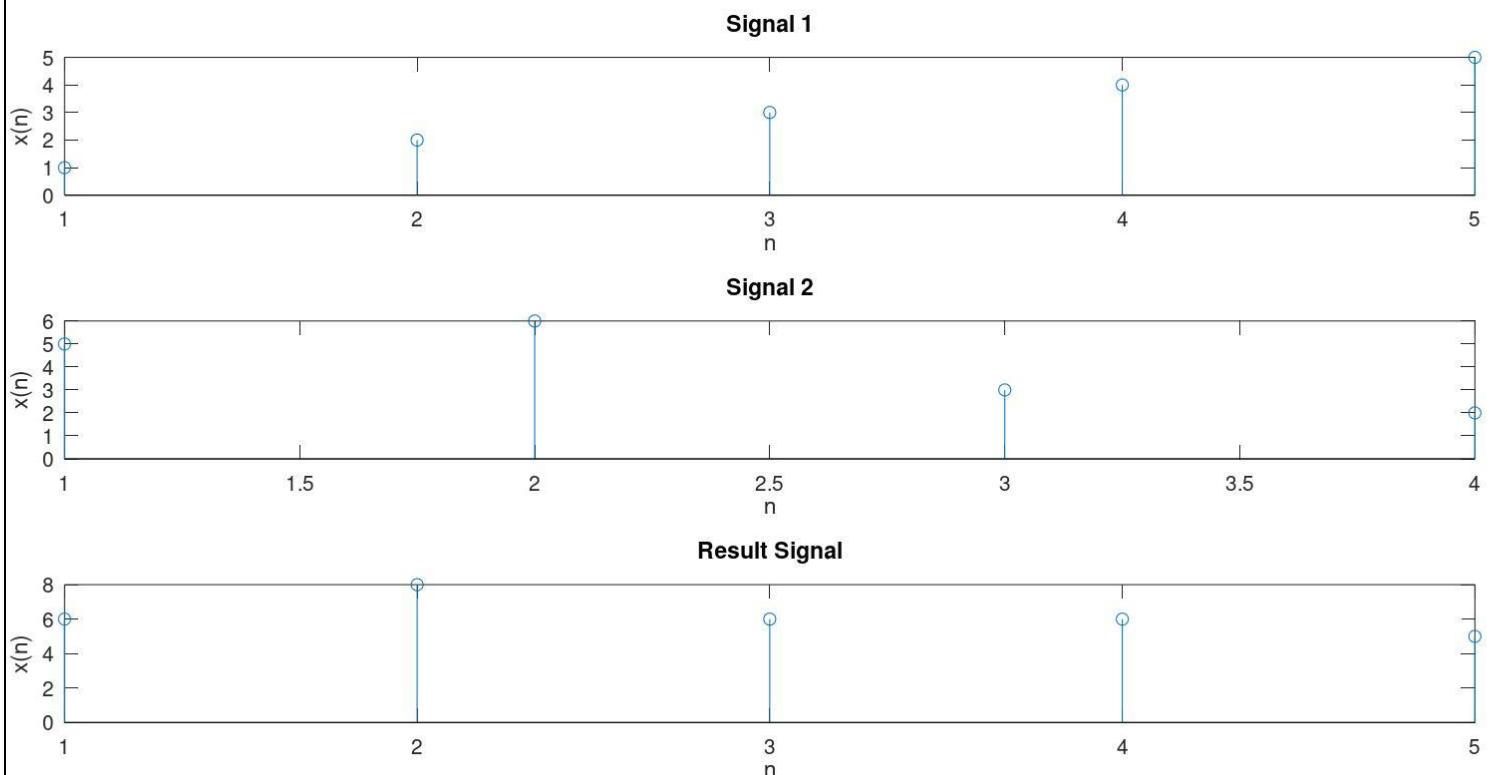
stem(n,y);

title("Result Signal");

xlabel("n");

ylabel("x(n)");
```

Output



Program-3

Aim

Write a MATLAB code for the multiplication of discrete time signal.

Software used

MATLAB/GNU Octave/SciLab

Code

```
x1 = [9,1,4,2,5];
n1 = [1:5];
x2 = [5,6,7,2];
n2 = [1:4];
subplot(3,1,1);
stem(n1,x1);
title("Signal 1");
xlabel("n");
ylabel("x(n)");
subplot(3,1,2);
stem(n2,x2);
title("Signal 2");
xlabel("n");
ylabel("x(n)");
subplot(3,1,3);
```

```
[y,n] = sigmult(x1,n1,x2,n2);

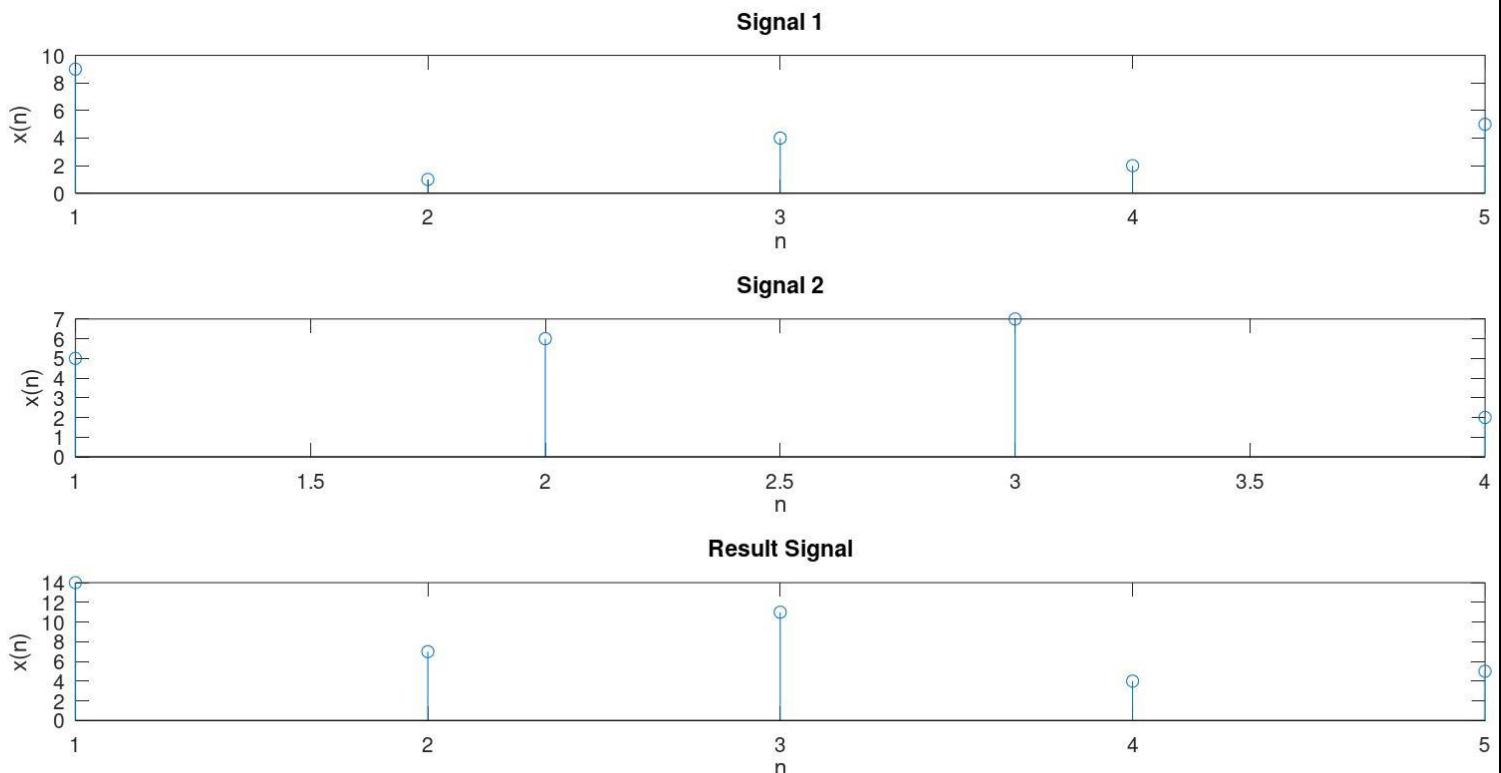
stem(n,y);

title("Result Signal");

xlabel("n");

ylabel("x(n)");
```

Output



Program-4

Aim

Write a MATLAB code for the shifting and folding of discrete time signal.

Software used

MATLAB/GNU Octave/SciLab

Code

```
%sigshift  
x = [6,7,3,4,5];  
m = [1:5];  
k = 1;  
subplot(2,2,1);  
stem(m,x);  
title("Signal-1");  
xlabel("m");  
ylabel("x(m)");  
subplot(2,2,2);  
stem(n,y);  
[y,n] = sigshift(x,m,k);
```

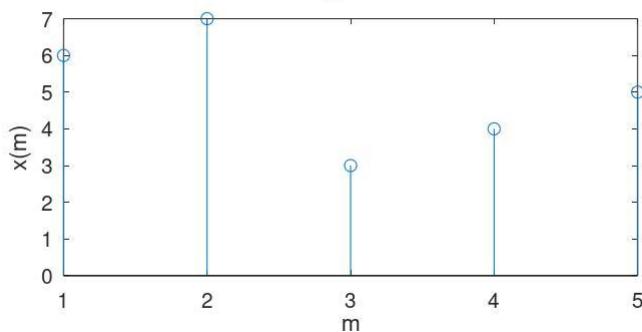
```
stem(n,y);
title("Shifted Signal-1");
xlabel("n");
ylabel("x(n)");
```



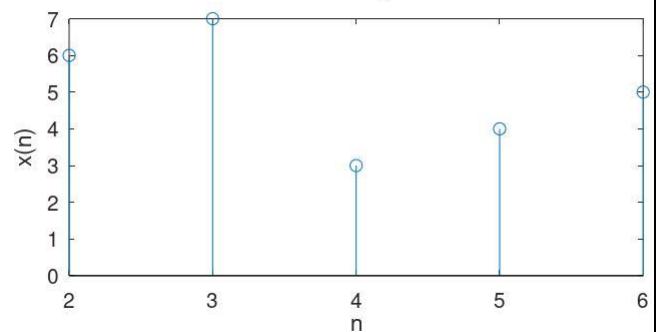
```
%sigfold
x = [5,2,9,4,1];
n = [1:5];
subplot(2,2,3);
stem(n,x);
title("Signal-2");
xlabel("n");
ylabel("x(n)");
subplot(2,2,4);
stem(n,x);
[y,n] = sigfold(x,n);
stem(n,y);
title("Folded Signal-2");
xlabel("n");
ylabel("x(n)");
```

Output

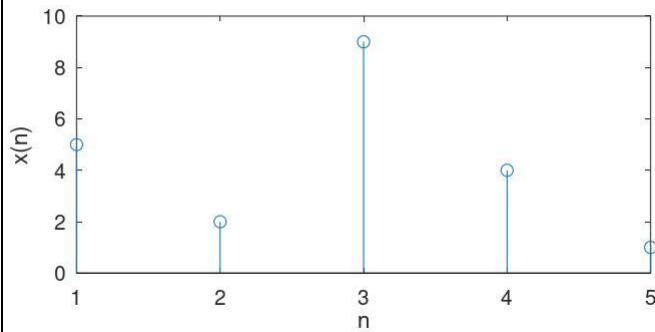
Signal-1



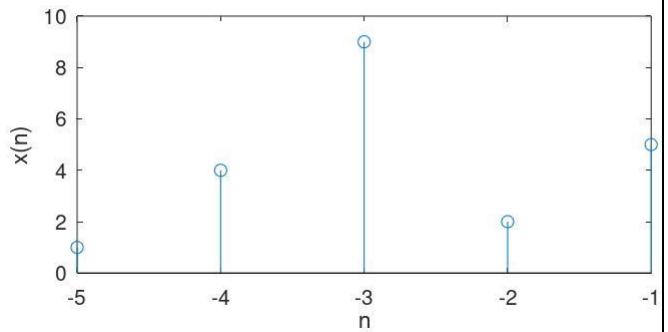
Shifted Signal-1



Signal-2



Folded Signal-2



Program-5

Aim

Write a MATLAB code for the following question:

Generate and plot each of the following sequences over the indicated interval.

a) $x(n)=2\delta(n + 2) - \delta(n - 4)$, $-5 \leq n \leq 5$.

b) $x(n) = n[u(n)-u(n-10)]+10e^{-0.3(n-10)}[u(n-10)-u(n-20)]$,

$$0 \leq n \leq 20.$$

c) $x(n) = \cos(0.04\pi n)+0.2w(n)$, $0 \leq n \leq 50$, where $w(n)$ is a Gaussian random sequence with zero mean and unit variance.

d) $\tilde{x}(n) = \{..., 5, 4, 3, 2, 1, 5, 4, 3, 2, 1, 5, 4, 3, 2, 1, ...\}$; $-10 \leq n \leq 9$.

Software used

MATLAB/GNU Octave/SciLab

Code

```
n=[-5:5];  
x=2*impseq(-2,-5,5)-impseq(4,-5,5);  
subplot(2,2,1);  
stem(n,x);
```

```

title('Sequence in Problem 2.1a');
xlabel('n');
ylabel('x(n)');
n=[0:20];
x1=n.* (stepseq(0,0,20)-stepseq(10,0,20));
x2=10*exp(-0.3*(n-10)).*(stepseq(10,0,20)-
stepseq(20,0,20));
x=x1+x2;
subplot(2,2,3);
stem(n,x);
title('Sequence in Problem 2.1b');
xlabel('n');
ylabel('x(n)');
n=[0:50];
x=cos(0.04*pi*n)+0.2*randn(size(n));
subplot(2,2,2);
stem(n,x);
title('Sequence in Problem 2.1c');
xlabel('n');
ylabel('x(n)');
n=[-10:9];
x=[5,4,3,2,1];
xtilde=x'* ones(1,4);
xtilde=(xtilde(:))';

```

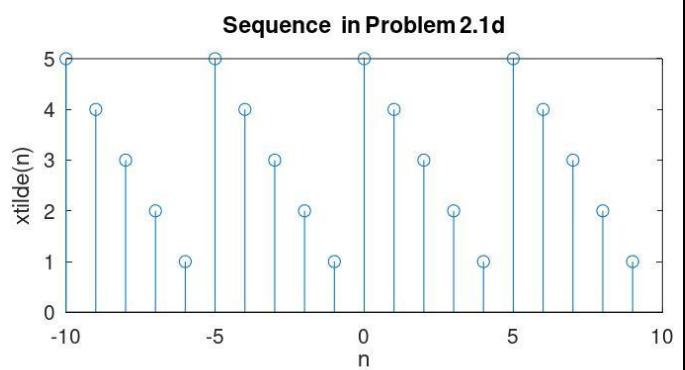
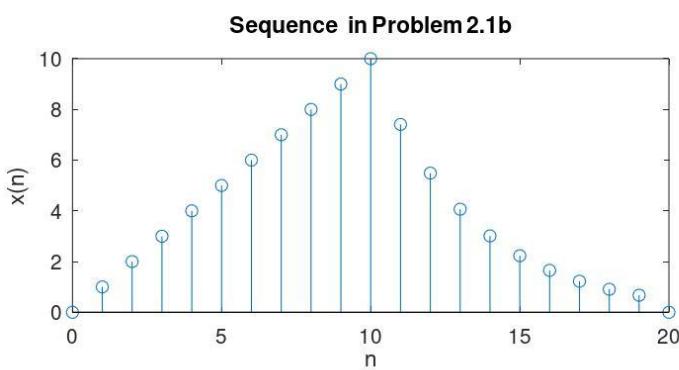
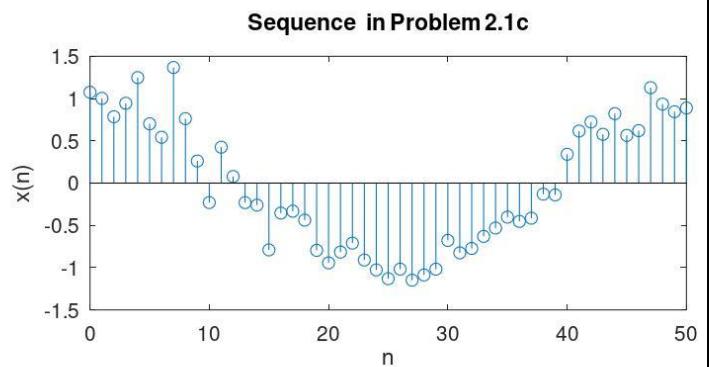
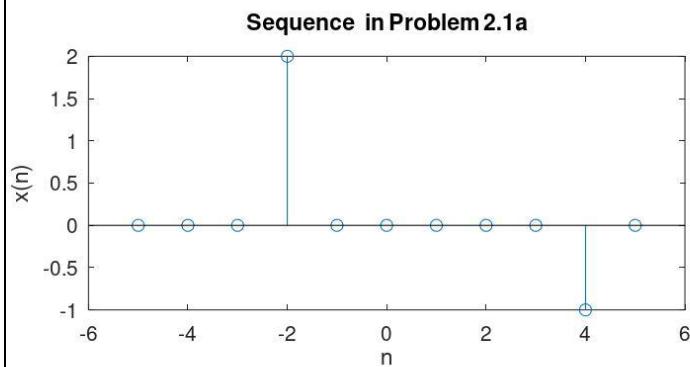
```

subplot(2,2,4);

stem(n,xtilde);
title('Sequence in Problem 2.1d');
xlabel('n');
ylabel('xtilde(n)');

```

Output



Program-6

Aim

Write a MATLAB code for the following question:

Let $x(n) = u(n) - u(n - 10)$. Decompose $x(n)$ into even and odd components.

Software used

MATLAB/GNU Octave/SciLab

Code

```
function [x,n] = stepseq(n0,n1,n2)
% Generates x(n) = u(n-n0); n1 <= n <= n2
%
% [x,n] = stepseq(n0,n1,n2)
%
n = [n1:n2]; x = [(n-n0) >= 0];
end

function [x,n] = impseq(n0,n1,n2)
% Generates x(n) = delta(n-n0); n1 <= n <= n2
%
```

```

% [x,n] = impseq(n0,n1,n2)
%
n = [n1:n2]; x = [ (n-n0) == 0];
end

%MATLAB CODE

clc;
clear all;
close all;
n=[0:10];
x=stepseq(0,0,10)-stepseq(10,0,10);
[xe,xo,m]=evenodd(x,n);
subplot(3,1,1); stem(n,x);
title('Unit Step Sequence')
xlabel('n'); ylabel('x(n)');
axis([-10,10,0,1.2])
subplot(3,1,2);
stem(m,xe);
title('Even Part')
xlabel('n');
ylabel('xe(n)');
axis([-10,10,0,1.2])
subplot(3,1,3);

```

```

stem(m,xo);

title('Odd Part')

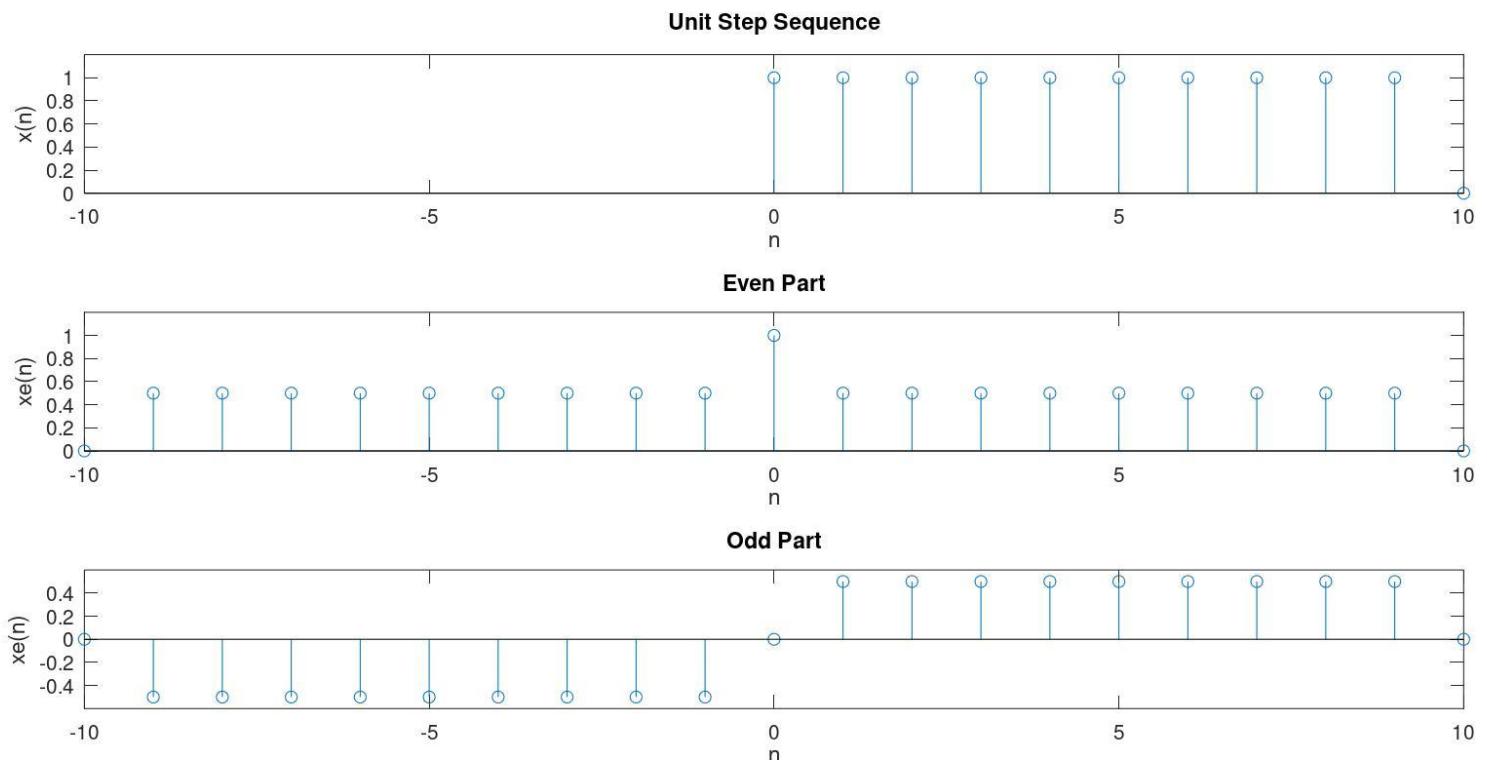
xlabel('n');

ylabel('xe(n)');

axis([-10,10,-0.6,0.6])

```

Output



Program-7

Aim

Write a MATLAB code for the following question:

Given the following two sequences

- a) $x(n) = [3, 11, 7, 0, -1, 4, 2], -3 \leq n \leq 3;$
- b) $h(n) = [2, 3, 0, -5, 2, 1], -1 \leq n \leq 4.$

Determine the convolution $y(n) = x(n) * h(n).$

Software used

MATLAB/GNU Octave/SciLab

Code

```
clc;  
x=[ 3      11     7      0      -1      4      2 ] ;  
h=[ 2      3      0      -5      2      1 ] ;  
y=conv (x,h);  
subplot(3,1,1);  
stem(x);  
title('First signal');  
xlabel('y');
```

```

ylabel('x');

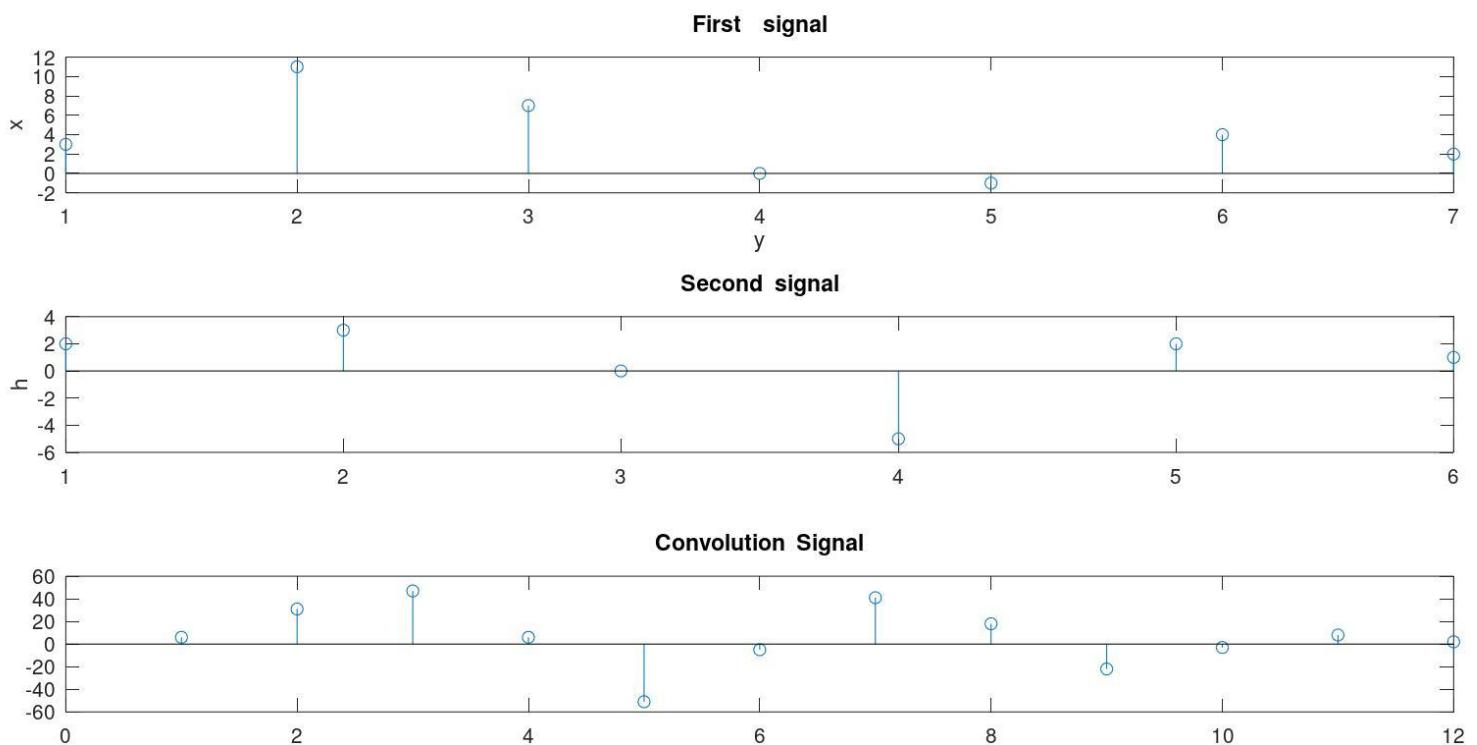
subplot(3,1,2);
stem(h);
title('Second signal');

ylabel('h');

subplot(3,1,3);
stem(y)
title('Convolution Signal');

```

Output



Program-8

Aim

Write a MATLAB code for the following question:

Given the following difference equation

$$y(n) - y(n - 1) + 0.9y(n - 2) = x(n); \forall n$$

- Calculate and plot the impulse response $h(n)$ at $n = -20, \dots, 120$.
- Calculate and plot the unit step response $s(n)$ at $n = -20, \dots, 120$.

Software used

MATLAB/GNU Octave/SciLab

Code

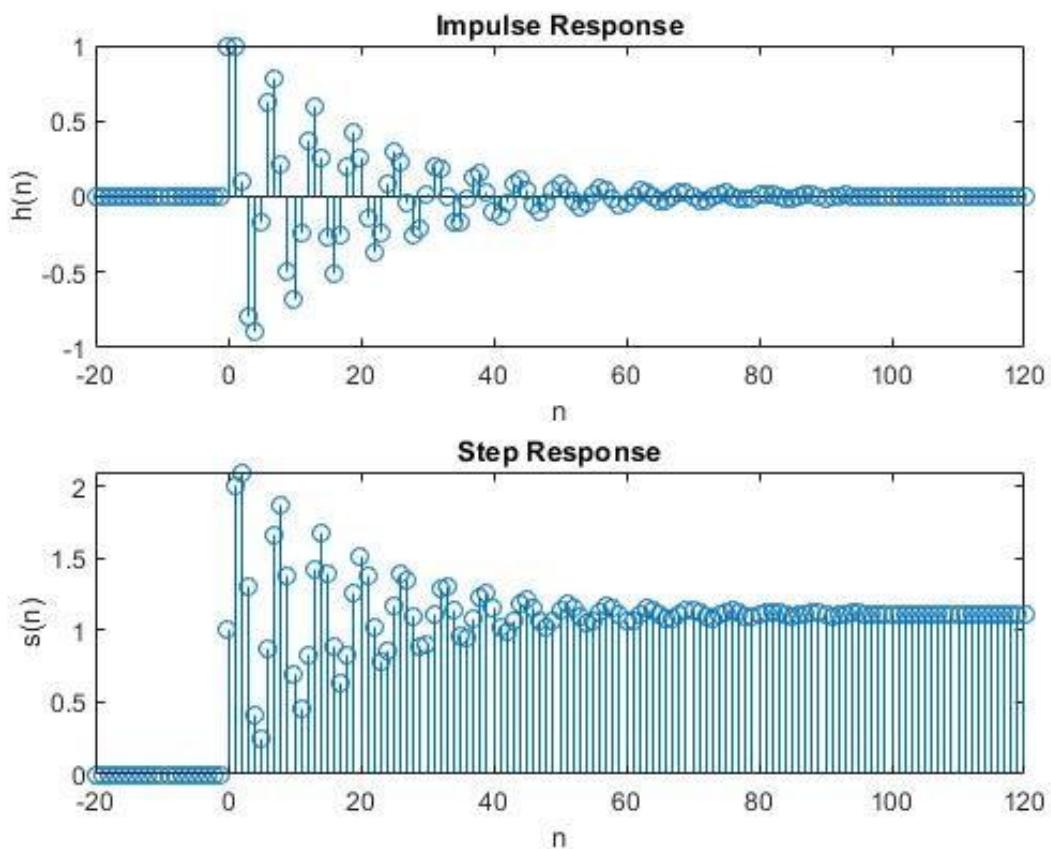
```
clc;  
b= [1];  
a= [1, -1, 0.9];  
b= [1];  
a= [1, -1, 0.9];  
n= [-20:120];  
h= impz(b, a, n);  
subplot(2,1,1);
```

```

stem(n, h);
title('Impulse Response');
xlabel('n');
ylabel('h(n)');
x= stepseq(0, -20,120);
s= filter(b, a, x);
subplot(2,1,2);
stem(n, s);
title('Step Response');
xlabel('n');
ylabel('s(n)');

```

Output



Program-9

Aim

Write a MATLAB code for the following question:

Determine the discrete-time Fourier transform of $x(n) = (0.5)^n u(n)$ and also evaluate $X(e^{j\omega})$ at 501 equispaced points between $[0, \pi]$ and plot its magnitude, angle, real, and imaginary parts.

Software used

MATLAB/GNU Octave/SciLab

Code

```
clc;
clear all;
close all;
w= [0:1:500]*pi/500;
%[0, pi] axis divided into 501 points.
X= exp(1i*w)./(exp(1i*w) - 0.5*ones(1,501));
magX= abs(X); angX = angle(X);
realX= real(X);
imagX= imag(X);
subplot(2,2,1);
```

```
plot(w/pi,magX); grid
xlabel('frequency in pi units');
ylabel('Magnitude');
title('Magnitude Part');

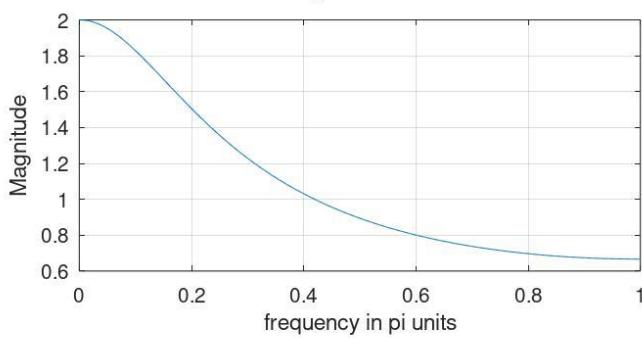
subplot(2,2,3);
plot(w/pi,angX); grid
xlabel('frequency in pi units') ;
ylabel('Radians');
title('Angle Part');

subplot(2,2,2);
plot(w/pi,realX); grid
xlabel('frequency in pi units');
ylabel('Real');
title('Real Part');

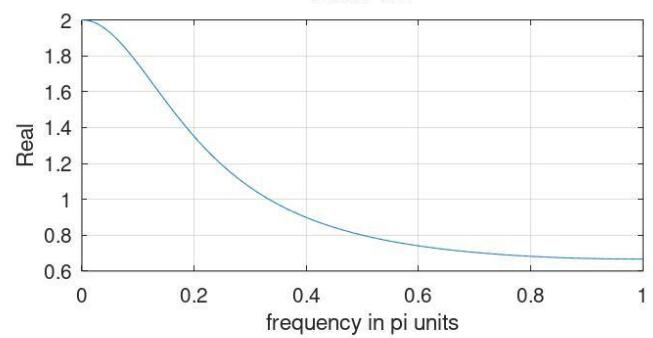
subplot(2,2,4);
plot(w/pi,imagX); grid
xlabel('frequency in pi units');
ylabel('Imaginary')
title('Imaginary Part');
```

Output

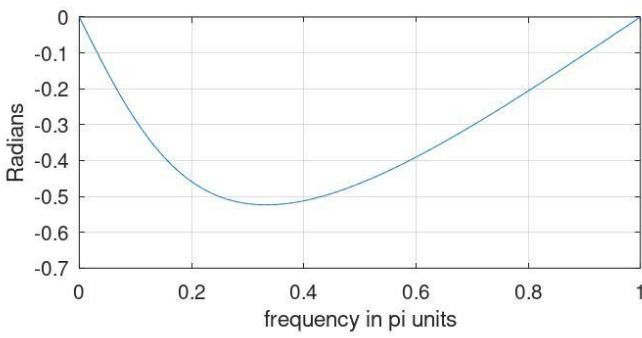
Magnitude Part



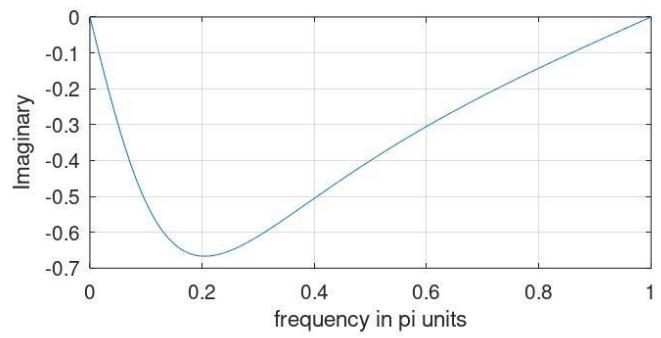
Real Part



Angle Part



Imaginary Part



Program-10

Aim

Write a MATLAB code for the following question:

To study the effect of sampling on the frequency-domain quantities, we will sample $x_a(t) = e^{-1000|t|}$ at 2 different sampling frequencies.

- Sample $x_a(t)$ at $F_s = 5000$ sam/sec to obtain $x_1(n)$. Determine and plot $X_1(e^{j\omega})$.
- Sample $x_a(t)$ at $F_s = 1000$ sam/sec to obtain $x_2(n)$. Determine and plot $X_2(e^{j\omega})$.

Software used

MATLAB/GNU Octave/SciLab

Code

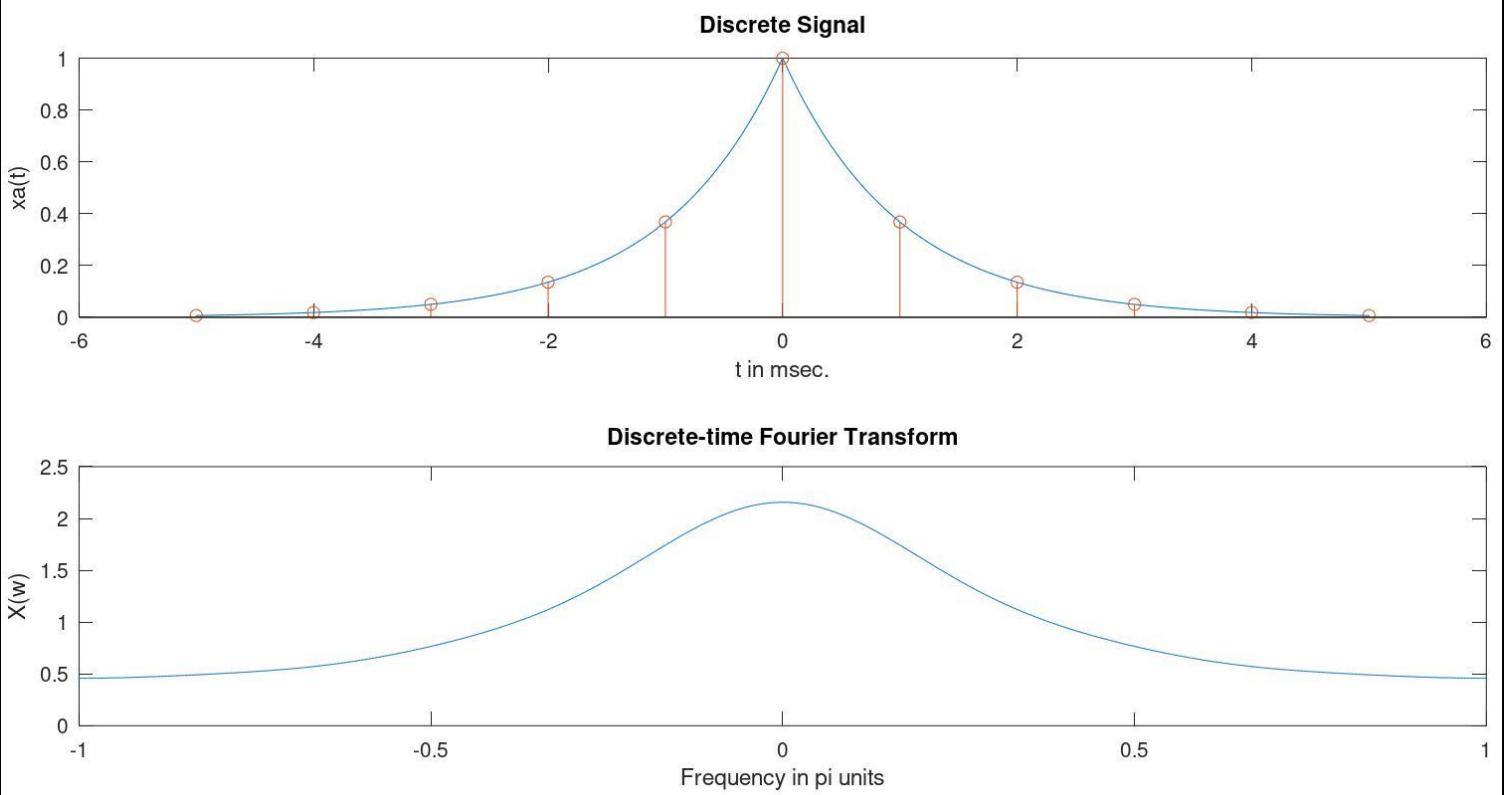
```
clc;  
clear all;  
% Analog Signal  
Dt = 0.00005;  
t = -0.005:Dt:0.005;  
xa = exp(-1000*abs(t));  
% Discrete-time Signal
```

```

Ts = 0.001; n = -5:1:5;
x = exp(-1000*abs(n*Ts));
% Discrete-time Fourier transform
K = 500; k = 0:1:K;
w = pi*k/K;
X = x * exp(-j*n'*w);
X = real(X);
w = [-fliplr(w), w(2:K+1)];
X = [fliplr(X), X(2:K+1)];
subplot(1,1,1)
subplot(2,1,1); plot(t*1000,xa);
xlabel('t in msec.');
ylabel('xa(t)');
title('Discrete Signal'); hold on
stem(n*Ts*1000,x); hold off
subplot(2,1,2); plot(w/pi,X);
xlabel('Frequency in pi units');
ylabel('X(w)');
title('Discrete-time Fourier Transform');

```

Output



Program-11

Aim

Write a MATLAB code for the following question:

Reconstruct $x_a(t)$ from the sample $x_2(n)$ obtained in Experiment-10(b), by sampling $x_a(t) = e^{-1000|t|}$ at $F_S = 1000$ sam/sec.

Software used

MATLAB/GNU Octave/SciLab

Code

```
clc;  
clear all;  
close all;  
% Discrete-time Signal x2(n)  
Ts = 0.001;  
Fs = 1/Ts;  
n = -5:1:5;  
nTs = n*Ts;  
x = exp(-1000*abs(nTs));  
% Analog Signal reconstruction  
Dt = 0.00005;
```

```

t = -0.005:Dt:0.005;

xa = x * sinc(Fs*(ones(length(nTs),1)*t-
nTs'*ones(1,length(t))));

% check

error = max(abs(xa - exp(-1000*abs(t))));

plot(t*1000,xa);

xlabel('t in msec.');

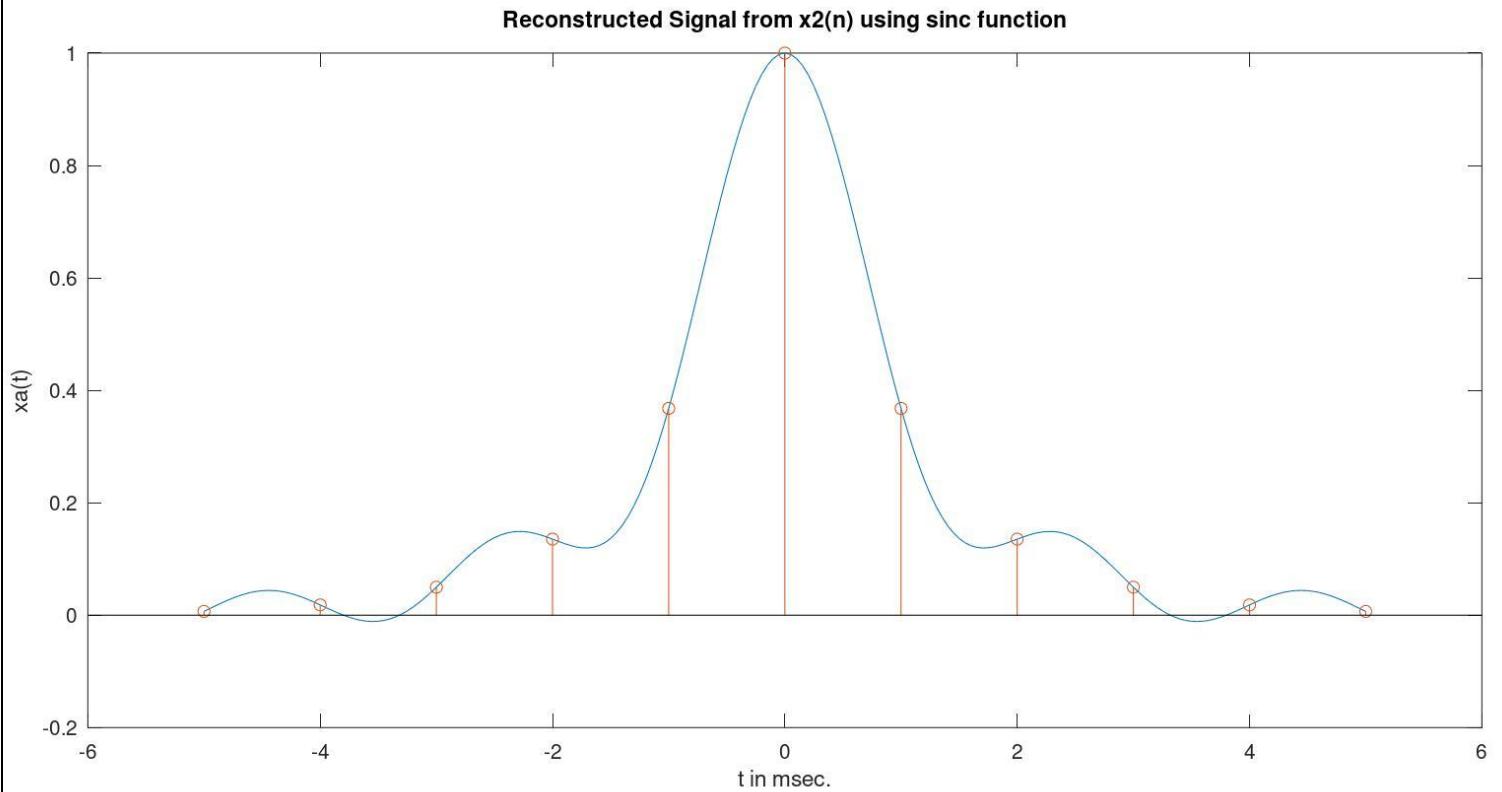
ylabel('xa(t)');

title('Reconstructed Signal from x2(n) using
sinc function'); hold on

stem(n*Ts*1000,x); hold off

```

Output



Program-12

Aim

Write a MATLAB code for the following question:

Given a causal system,

$$y(n) = 0.9y(n - 1) + x(n)$$

- Determine $H(z)$ and sketch its pole-zero plot.
- Plot $|H(e^{j\omega})|$ and $\angle H(e^{j\omega})$.

Software used

MATLAB/GNU Octave/SciLab

Code

```
b = [1, 0];
a = [1, -0.9];
zplane(b,a)
title('Pole-Zero Plot');
xlabel('Real Part');
ylabel('Imaginary Part');
[H,w] = freqz(b,a,100);
magH = abs(H);
```

```

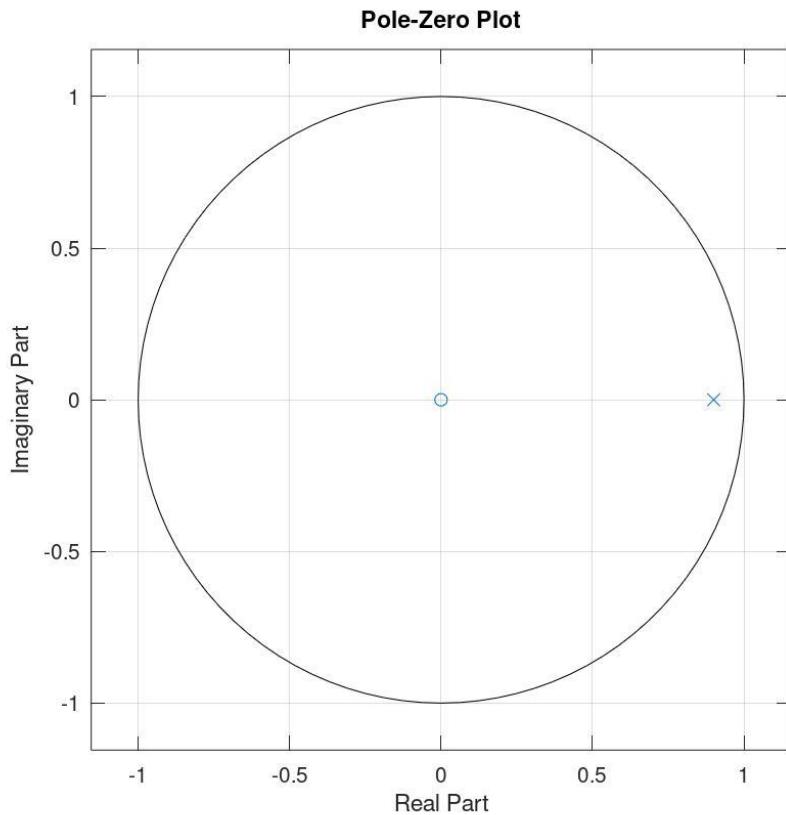
phaH = angle(H);
subplot(2,1,1);
plot(w/pi,magH);grid
xlabel('frequency in pi units');
ylabel('Magnitude');
title('Magnitude Response');

subplot(2,1,2);
plot(w/pi,phaH/pi);grid
xlabel('frequency in pi units');
ylabel('Phase in pi units');
title('Phase Response');

```

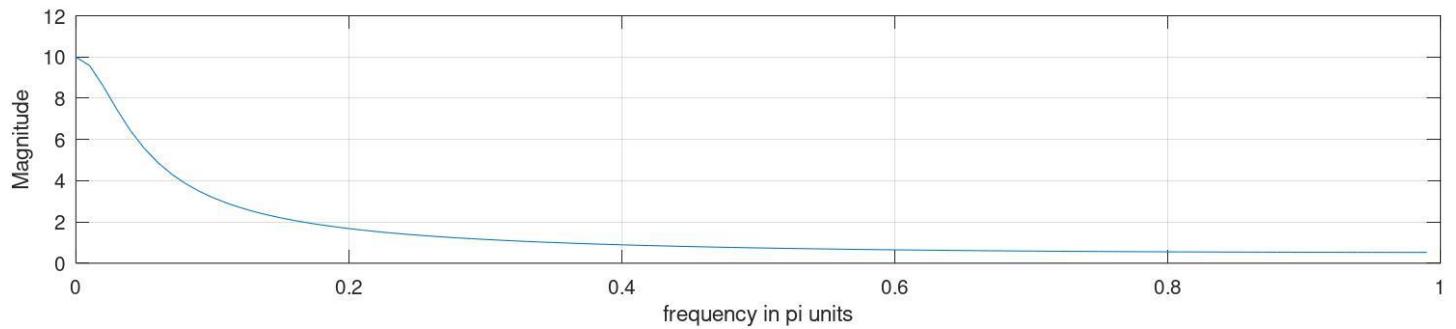
Output

a)

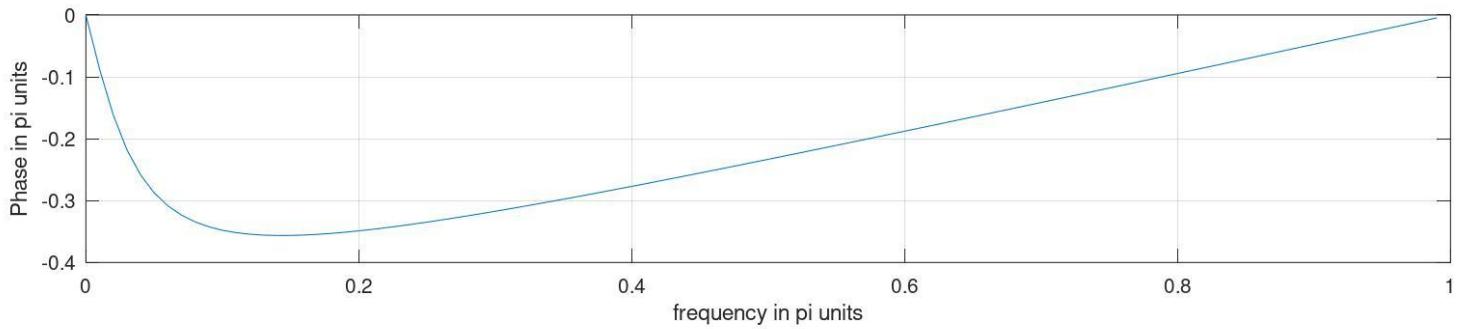


b)

Magnitude Response



Phase Response



Program-13

Aim

Write a MATLAB code for the following question:

A filter is described by the following difference equation:

$$16y(n) + 12y(n - 1) + 2y(n - 2) - 4y(n - 3) - y(n - 4) \\ = x(n) - 3x(n - 1) + 11x(n - 2) - 27x(n - 3) + 18x(n - 4)$$

Determine its cascaded form structure.

Software used

MATLAB/GNU Octave/SciLab

Code

```
function [b0,B,A] = dir2cas(b,a);  
%----DIRECT-form to CASCADE-form conversion----  
b0 = b(1); b = b/b0; a0 = a(1); a = a/a0;  
b0 = b0/a0;  
%  
M = length(b);  
N = length(a);  
if N>M
```

```

b = [b zeros(1,N-M)] ;
elseif M>N

a = [a zeros(1,M-N)] ;
N = M;
else

NM = 0;

end

%
K = floor(N/2); B = zeros(K,3); A = zeros(K,3);

if K*2 == N;

b = [b 0];
a = [a 0];
end

%
broots = cplxpair(roots(b));
aroots = cplxpair(roots(a));
for i=1:2:2*K

Brow = broots(i:1:i+1,:);
Brow = real(poly(Brow));
B(fix((i+1)/2),:) = Brow;
Arow = aroots(i:1:i+1,:);
Arow = real(poly(Arow));
A(fix((i+1)/2),:) = Arow;

```

```
end  
%  
function y = casfiltr(b0,B,A,x);  
%-----CASCADE form realization of IIR and FIR  
%filters-----  
[K,L] = size(B);  
N = length(x);  
w = zeros(K+1,N);  
w(1,:) = x;  
for i = 1:1:K  
w(i+1,:) = filter(B(i,:),A(i,:),w(i,:));  
end
```

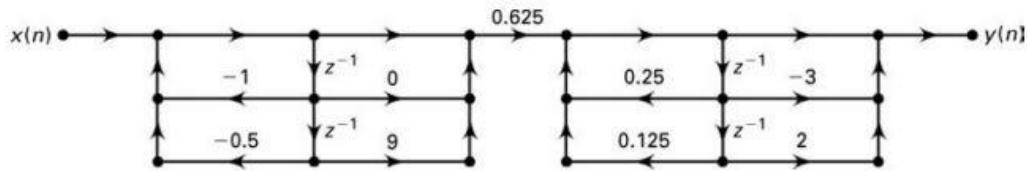
```
%MATLAB CODE  
b= [1 -3 11 -27 18];  
a= [16 12 2 -4 -1];  
[b0,B,A]= dir2cas(b,a)  
delta = impseq(0,0,7)  
format long  
hcas= casfiltr(b0,B,A,delta)  
hdir= filter(b,a,delta)
```

Output

```

b0 = 0.062500
B =
1.0000e+00  9.9920e-16  9.0000e+00
1.0000e+00  -3.0000e+00  2.0000e+00
A =
1.0000  1.0000  0.5000
1.0000  -0.2500  -0.1250
delta =
1 0 0 0 0 0 0 0
hcas =
Columns 1 through 6:
6.25000000000000e-02 -2.34375000000000e-01 8.55468750000007e-01 -2.28417968750004e+00 2.676513671875005e+00 -1.522644042968753e+00
Columns 7 and 8:
2.898406982421875e-01 4.993171691894546e-01
hdir =
Columns 1 through 6:
6.25000000000000e-02 -2.34375000000000e-01 8.55468750000000e-01 -2.28417968750000e+00 2.676513671875000e+00 -1.522644042968750e+00
Columns 7 and 8:
2.898406982421875e-01 4.993171691894531e-01

```



Program-14

Aim

Write a MATLAB code for the following question:

Design a digital FIR lowpass filter with the following specifications:

$$\omega_p = 0.2\pi, R_p = 0.25 \text{ dB}$$

$$\omega_s = 0.3\pi, A_s = 50 \text{ dB}$$

Using Hamming Window function, determine the impulse response and provide a plot of the frequency response of the designed filter.

Software used

MATLAB/GNU Octave/SciLab

Code

```
wp = 0.2*pi; ws = 0.3*pi; tr_width = ws - wp;
M = ceil(6.6*pi/tr_width) + 1
n=[0:1:M-1];
wc = (ws+wp)/2 % Ideal LPF cutoff frequency
hd = ideal_lp(wc,M);
w_ham = (hamming(M))';
h = hd .* w_ham;
[db,mag,pha,grp,w] = freqz_m(h,[1]);
delta_w = 2*pi/1000;
```

```

Rp = -(min(db(1:1:wp/delta_w+1))) % Actual
%Passband Ripple

As = -round(max(db(ws/delta_w+1:1:501))) % Min
%Stopband attenuation

% plots

subplot(2,2,1); stem(n,hd);
title('Ideal Impulse Response');
axis([0 M-1 -0.1 0.3]);
xlabel('n');
ylabel('hd(n)');

subplot(2,2,2); stem(n,w_ham);
title('Hamming Window');
axis([0 M-1 0 1.1]); xlabel('n'); ylabel('w(n)');
subplot(2,2,3); stem(n,h);
title('Actual Impulse Response');
axis([0 M-1 -0.1 0.3]);
xlabel('n');
ylabel('h(n)');

subplot(2,2,4);
plot(w/pi,db);
title('Magnitude Response in dB'); grid
axis([0 1 -100 10]);
xlabel('frequency in pi units');
ylabel('Decibels');

```

Output

$$\text{tr_width} = 0.3142$$

$$M = 67$$

$$WC = 0.7854$$

$$Rp = 0.0394$$

$$As = 52$$

