DIGITAL SIGNAL PROCESSING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY BHAGALPUR

BY Dr. SANDEEP SIR

Name : SANI KUAMR
Branch : ECE
Roll no. : 2001125

EXPERIMENT-1

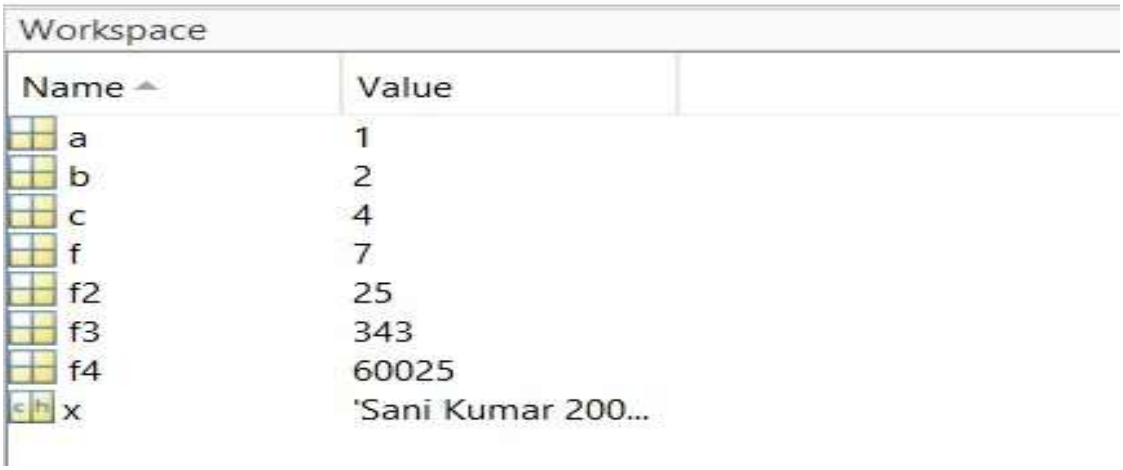
1.) Write MATLAB code to solve/plot:

i) Y= (a+b-c) * (a-b-c) ^2 * (a+b+c)^ 3 , where a=1, b=2, c=4

```
a=1;
b=2;
c=4;
x='Sani Kumar 2001125';
f=(a+b+c);
f3=(a+b+c)*(a+b+c)*(a+b+c)
; f2=(a-b-c)*(a-b-c);
f4=f*f2*f3; display(x);
display(f4);
```

OUTPUT:

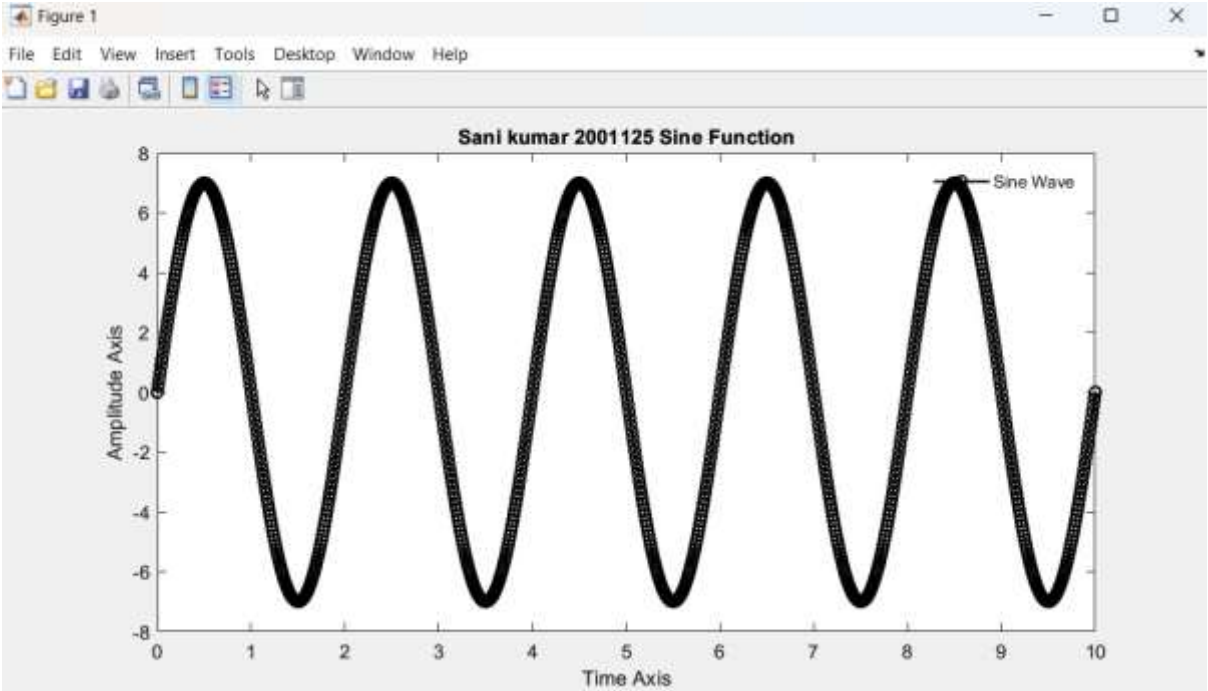
```
x =

    'Sani Kumar 2001125'


f4 =

       60025
```



ii) Sine Wave

```
clc;
close all; clear all; t = 0:0.01:10;
a=7;  f=0.5;  figure;  hold  on;  y  =
a*sin(2*pi*f*t);              plot(t,y,"k-
o","LineWidth",1.3);  title('Sani  kumar
2001125  Sine  Function');  xlabel('Time
Axis');     ylabel('Amplitude     Axis');
legend("Sine     Wave");     box     on;
legend("boxoff"); hold off;
```

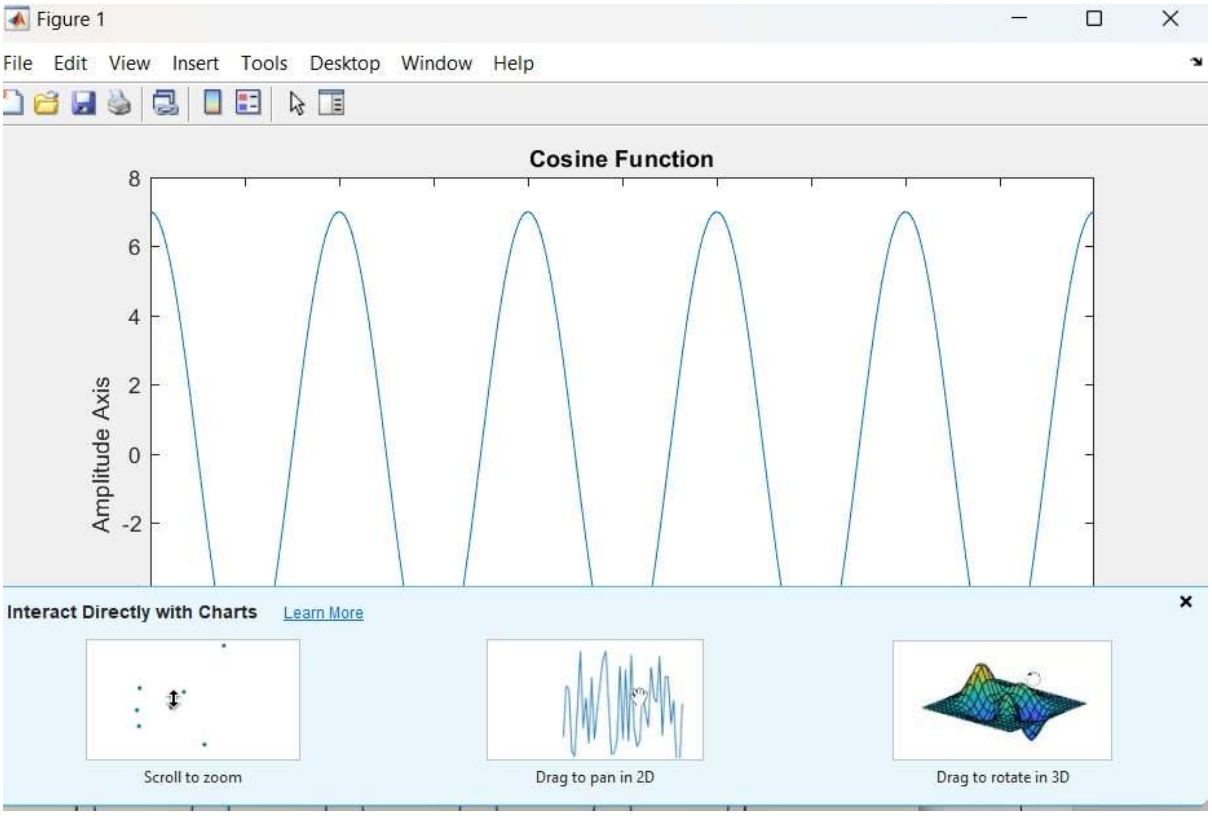
OUTPUT:



iii) Cosine Wave

```
clc
;
close all; clear all;
t = 0:0.01:10; a=7;
 f=0.5; y = a*cos(2*pi*f*t);
plot(t,y); title('Cosine
Function');  xlabel('Time
Axis');  ylabel('Amplitude
Axis');
```

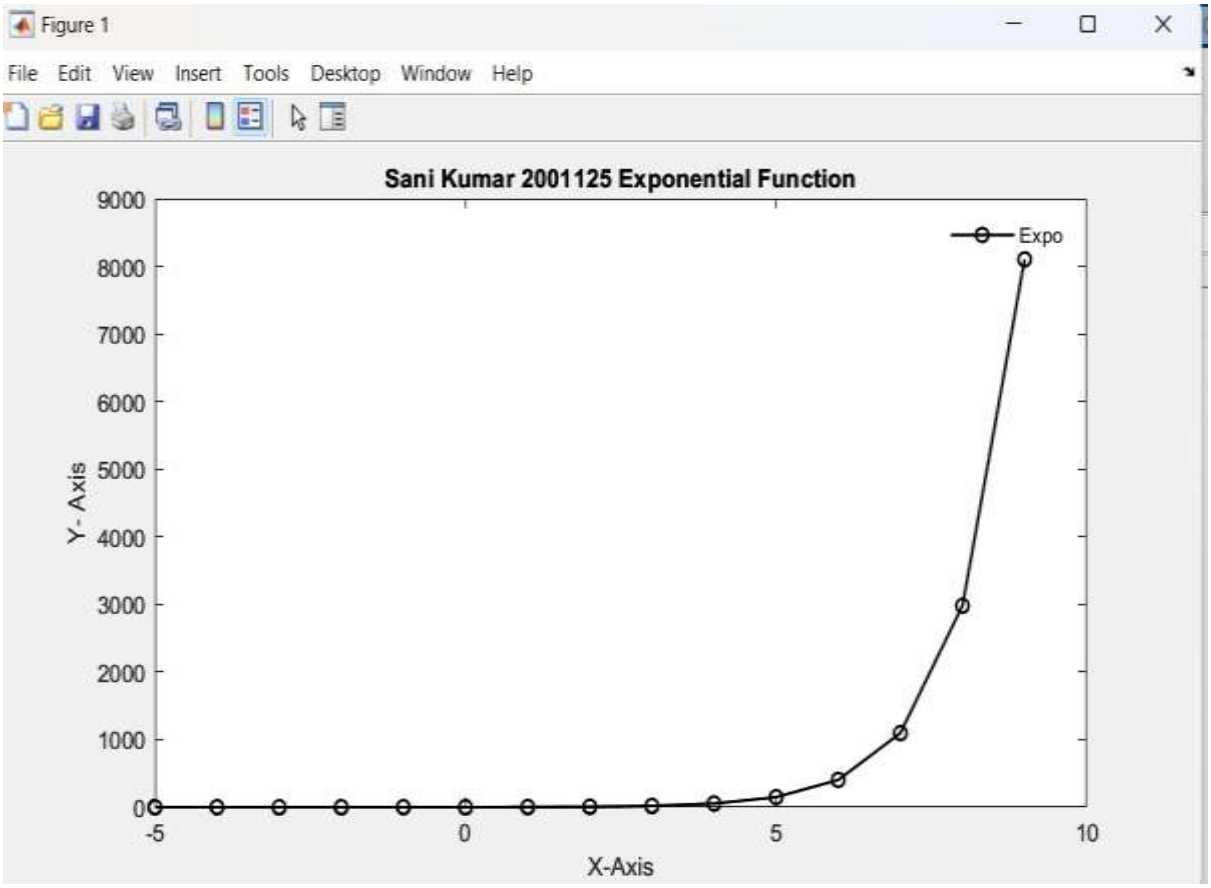
OUTPUT:



iv) e^x

```
clc; clear
all;
close all;
x = -5:1:9
figure;
hold on;
y= exp(x);
plot(x,y,"k-o","LineWidth",1.3);
title('Sani Kumar 2001125 Exponential Function');
xlabel('X-Axis'); ylabel('Y- Axis');
legend("Expo"); box on; legend("boxoff"); hold
off;
```

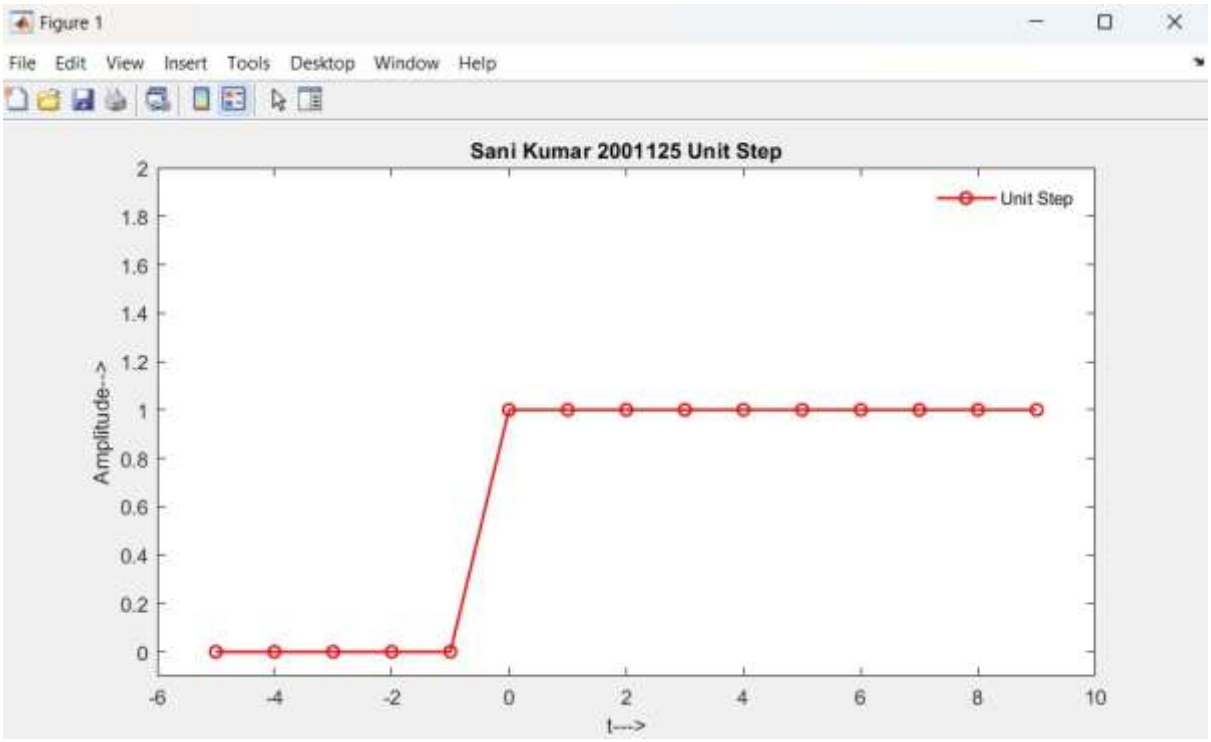
OUTPUT:



v) Unit Step

```
 clc;
clear all;
close all;
t= ( -5 : 1
: 9) ;
figure;
hold on; x
= (t>=0);
plot(t,x,"r-o","LineWidth",1.3);
xlim([t(1)-1 t(end)+1]); ylim([min(x)-
0.1 max(x)+1]); xlabel("t--->");
ylabel("Amplitude-->");
title("Sani Kumar 2001125 Unit
Step") legend("Unit Step"); box on;
legend("boxoff"); hold off;
```

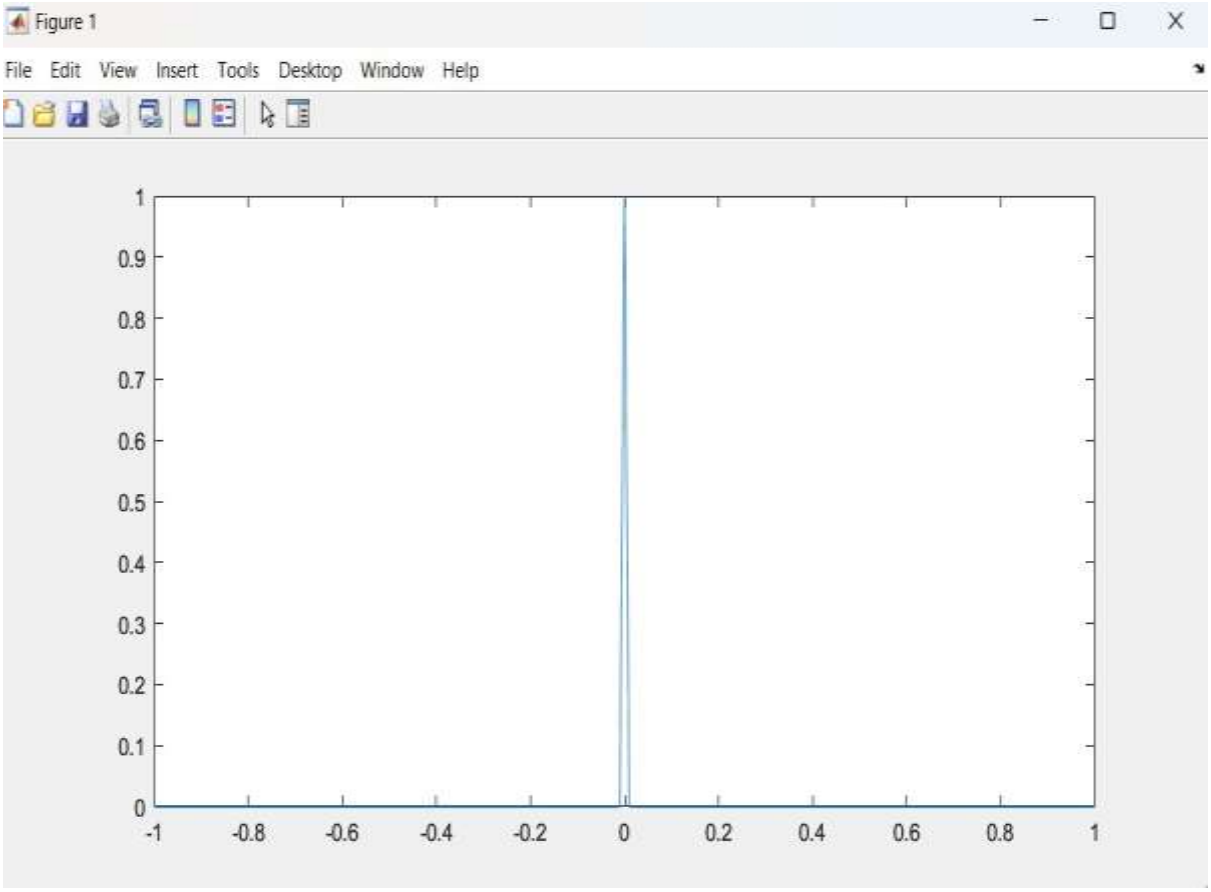
Output:-



vi) Unit Impulse

```
>> clear all;
close all; t=(-
1:0.01:1);
impulse= t==0; >>
unitstep=t>=0;
>> ramp=t.*unitstep;
>> plot(t,[impulse]);
```

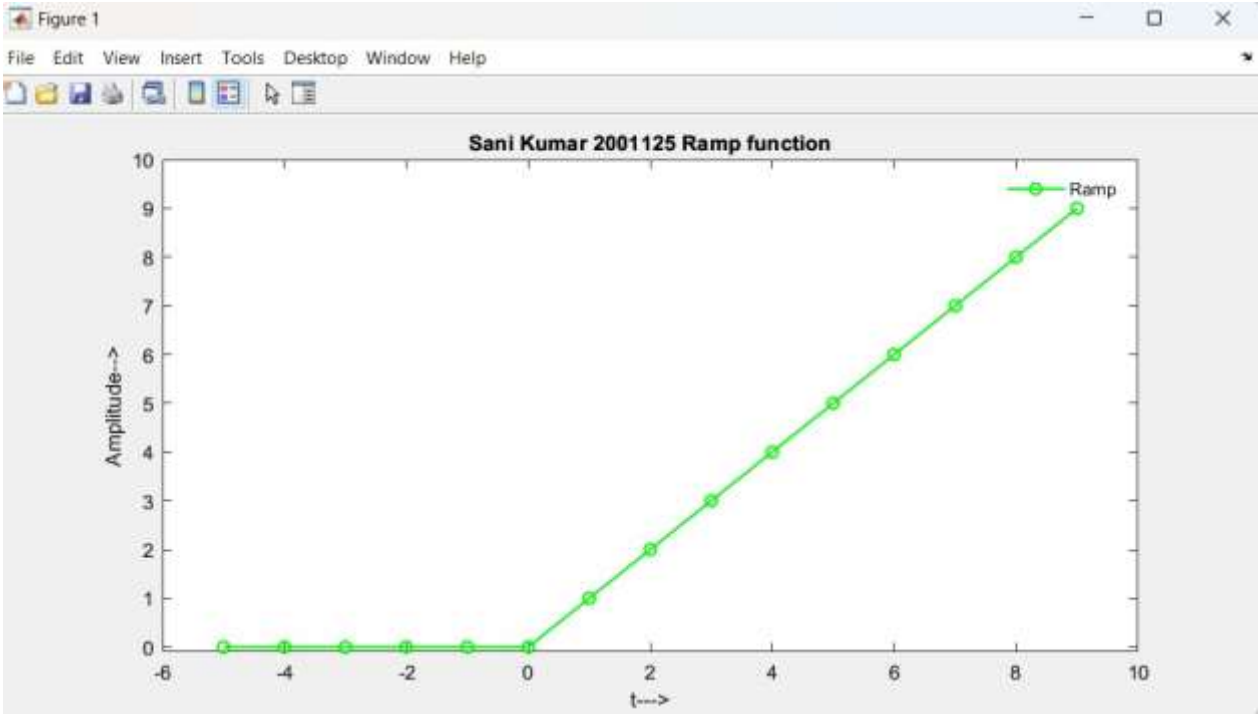
OUTPUT:-



vii) Ramp Function

```
clc;
clear all; close all;
t= (-5:1:9) ; figure; hold on x
=t.*(t>=0); plot(t,x,"g-
o","LineWidth",1.3);
xlim([t(1)-1 t(end)+1]);
ylim([min(x)-0.1 max(x)+1]);
xlabel("t--->"); ylabel("Amplitude-->");
title("Sani Kumar 2001125 Ramp function")
legend("Ramp"); box on; legend("boxoff");
hold off;
```

Output:



EXPERIMENT-2

2)Write MATLAB function to solve/plot:

i) Y= (a-b-c) 3, where a=2, b=3, c=4

```
  %Y= (a-b-c)^3 , where a=2, b=3, c=4
prompt1 = 'What is the value at a? ' ;
prompt2 = 'What is the value at b? '
; prompt3 = 'What is the value at c?
' ; a=input(prompt1);
b=input(prompt2); c=input(prompt3);
display("Require Sol is :");
display((a-b-c) *(a-b-c) *(a-b-c))
```

OUTPUT:

```
What is the value at a? 4
What is the value at b? 5
What is the value at c? 6
    "Require Sol is :"

  -343
```

Workspace

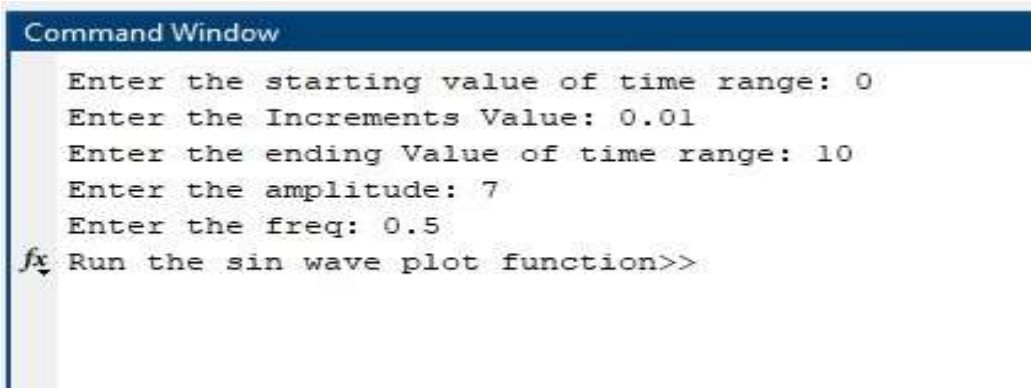
Name	Value
a	4
b	5
c	6
prompt1	'What is the valu...
prompt2	'What is the valu...
prompt3	'What is the valu...
t	*1x15 double*
x	*1x15 double*

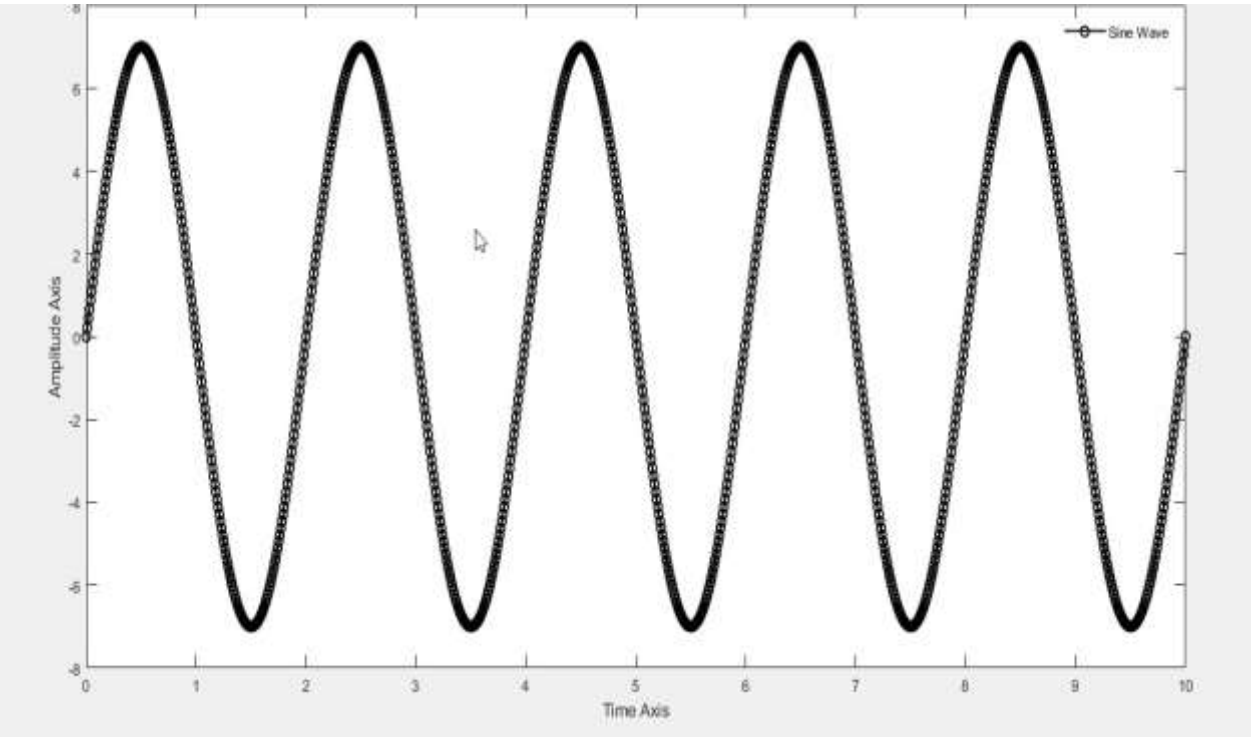
ii) Sine Wave

```
clc; close all; clear all; prompt1 = "Enter the
starting value of time range:
";
prompt2 = "Enter the Increments Value: "; prompt3 =
"Enter the ending Value of time range: "; s =
input(prompt1); i = input(prompt2); e =
input(prompt3); t = s:i:e;
prompt4 = "Enter the amplitude: "; prompt5
= "Enter the freq: "; a=input(prompt4);
f=input(prompt5);     plotSineWave(t,a,f)
function [] =plotSineWave(t,a,f) figure;
hold on; y = a*sin(2*pi*f*t);
plot(t,y,"k-o","LineWidth",1.3); title('
Sani Kumar 2001125 Sine Function');
```

```
xlabel('Time Axis');
ylabel('Amplitude Axis');
legend("Sine Wave"); box
on; legend("boxoff");
hold off;
fprintf("Run the sin wave plot function") end
```

OUTPUT:





iii) $3e^{x}$

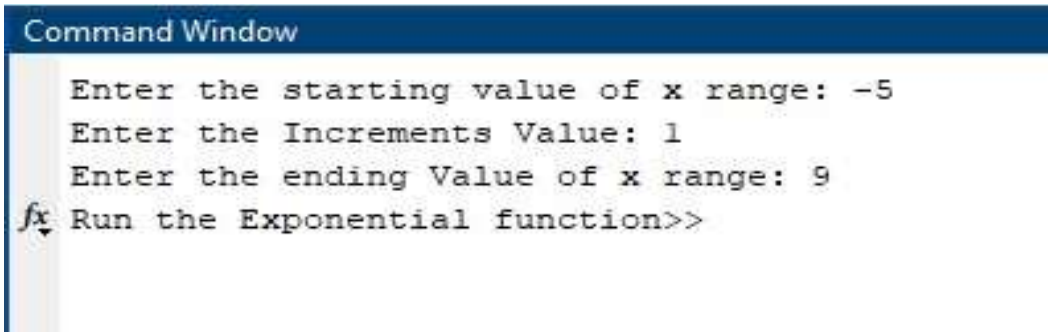
```
clc; clear
all; close
all;
prompt1 = "Enter the starting value of x range:
";
prompt2 = "Enter the Increments Value: ";
prompt3 = "Enter the ending Value of x range:
";
s = input(prompt1); i
= input(prompt2); e =
```

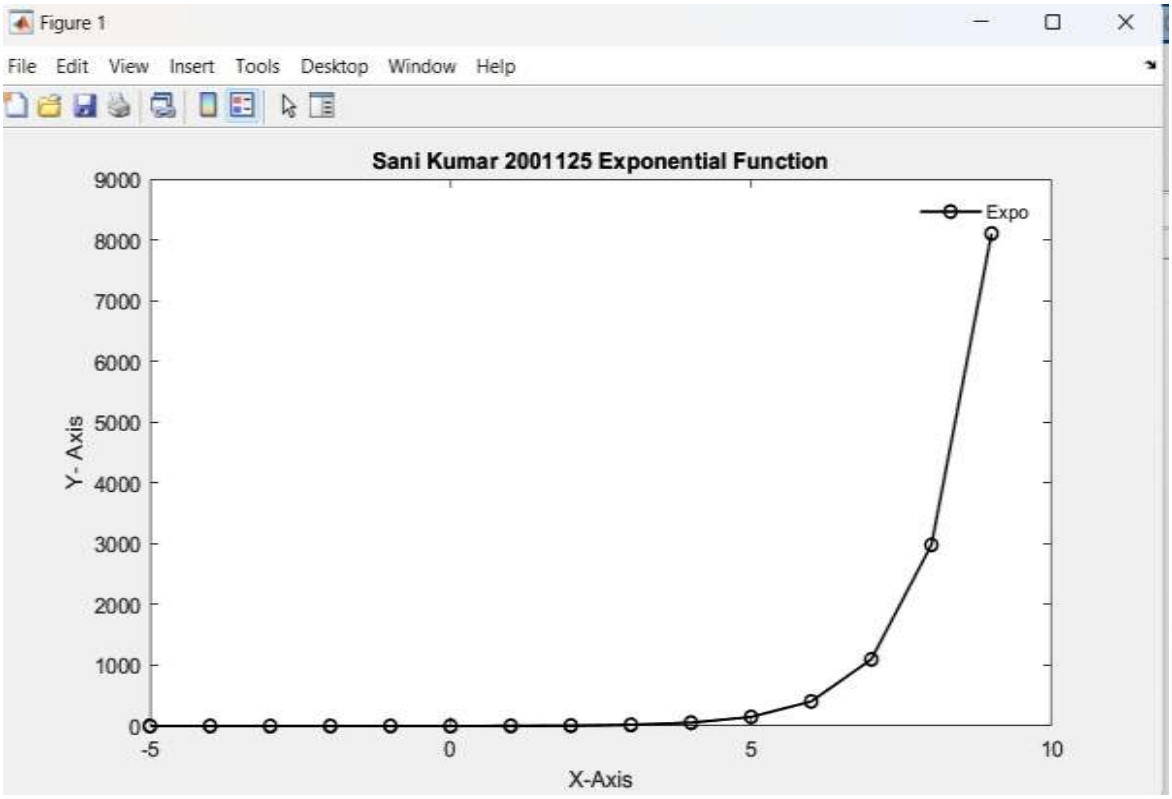
```
input(prompt3); x =
s:i:e;
%Calling Exponential Function exponential(x)
```

% Exponential Plot Function Code.

```
function [] = exponential(x)
figure; hold on; y= exp(x);
plot(x,y,"k-o","LineWidth",1.3);
title(' Sani Kumar 2001125 Exponential Function');
xlabel('X-Axis');        ylabel('Y-        Axis');
legend("Expo"); box on; legend("boxoff"); hold
off;
fprintf("Run the Exponential function")
end
```

OUTPUT:





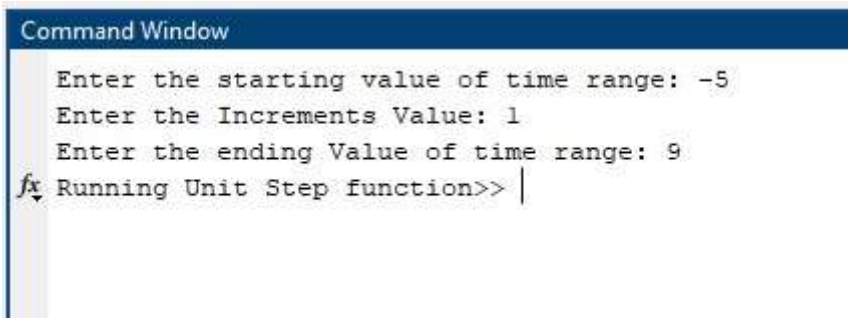
iv) Unit Step

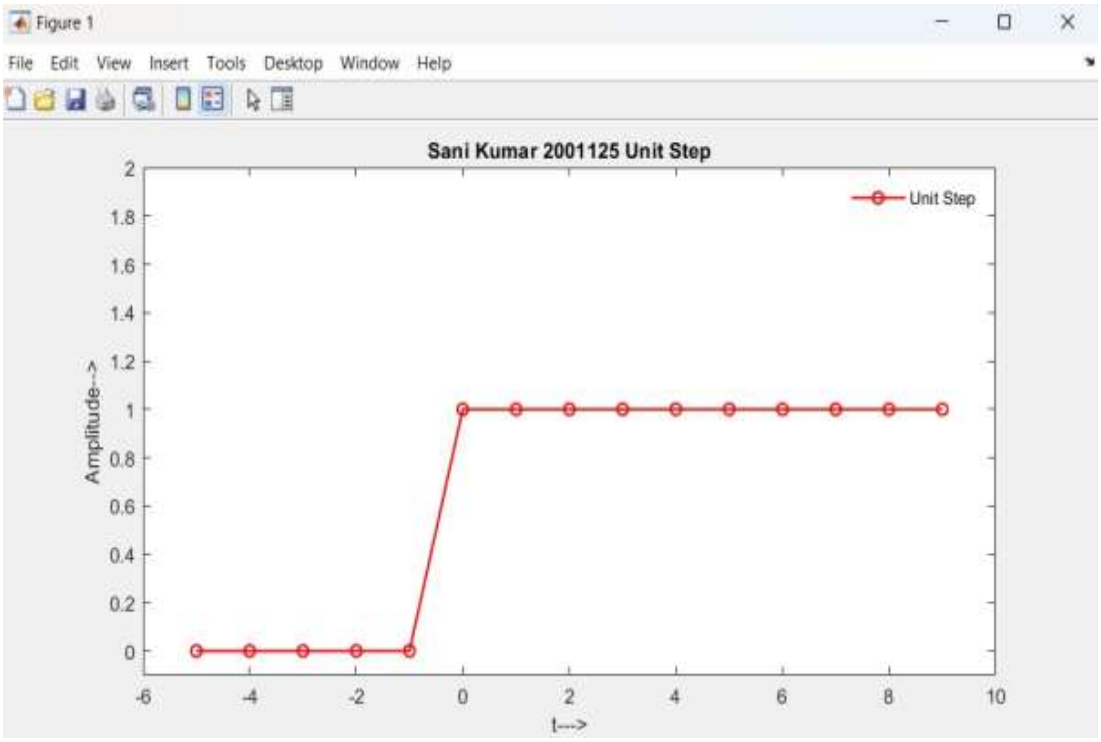
```
clc; clear all; close all; prompt1 = "Enter the
starting value of time range:
";
prompt2 = "Enter the Increments Value: "; prompt3 =
"Enter the ending Value of time range: "; s =
input(prompt1); i = input(prompt2); e =
input(prompt3); t = s:i:e;
% Calling Unit step Plot Function; unitStepPlot(t)
```

% Unit Step Plot Function Code.

```
function []=unitStepPlot(t)
figure; hold on; x =
(t>=0);
plot(t,x,"r-o","LineWidth",1.3);
xlim([t(1)-1 t(end)+1]); ylim([min(x)-0.1
max(x)+1]);
xlabel("t--->");
ylabel("Amplitude-->"); title("Sani
Kumar 2001125 Unit Step")
legend("Unit Step");
box on;
legend("boxoff");
hold off;
fprintf("Running Unit Step function") end
```

OUTPUT:





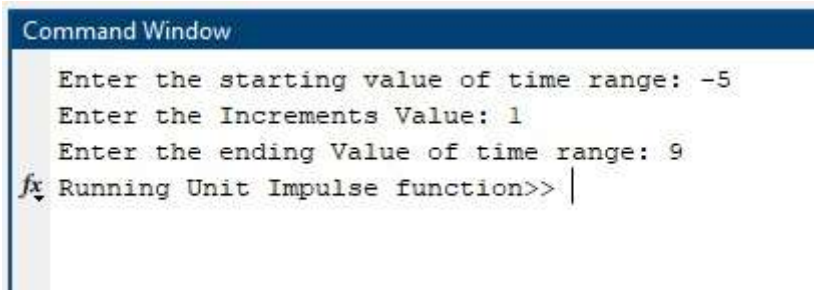
v) Unit Impulse

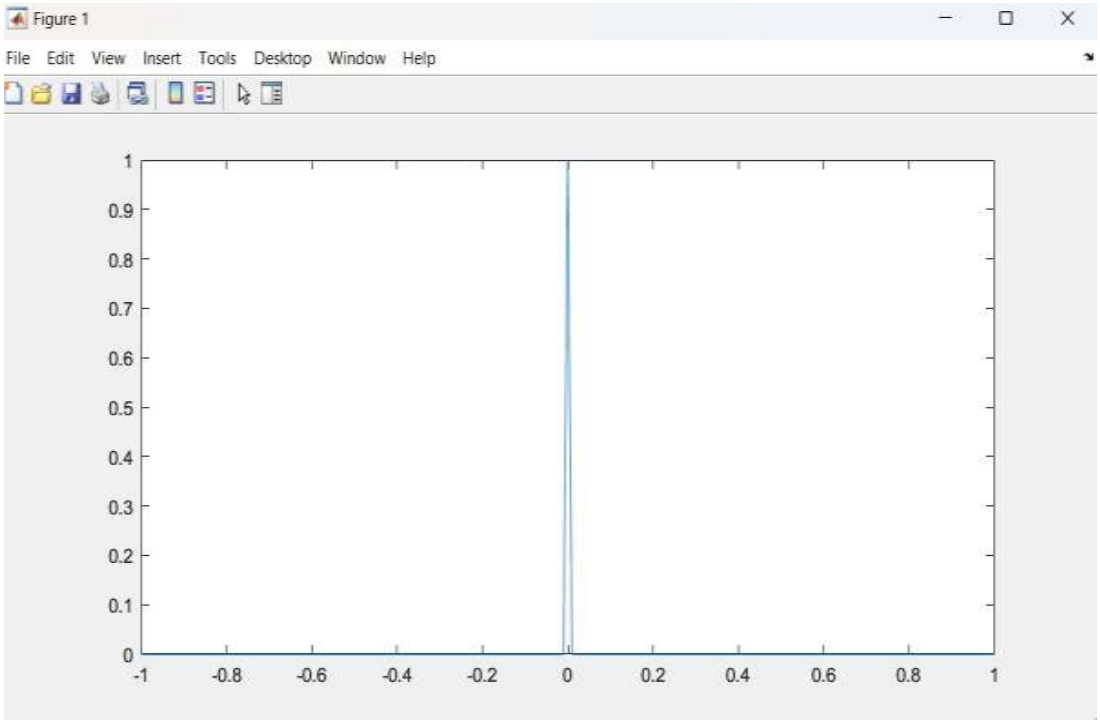
```
clc; clear
all;
close all;
prompt1 = "Enter the starting value of time
range: ";
prompt2 = "Enter the Increments Value: ";
prompt3 = "Enter the ending Value of time
range: "; s = input(prompt1); i =
input(prompt2); e = input(prompt3); t =
s:i:e;
%calling unit impulse function
UnitImpulseSignalPlot(t)
```

% Unit Impulse Plot Function Code.

```
function [] = UnitImpulseSignalPlot(t)
figure; hold on x = (t==0);
plot(t,x,"m-o","LineWidth",1.3);
xlim([t(1)-1 t(end)+1]); ylim([min(x)-
0.1 max(x)+1]); xlabel("t--->");
ylabel("Amplitude-->");
title("Sani Kumar 2001125 Unit Impulse")
legend("Unit impulse");
box on;
legend("boxoff"); hold
off; fprintf("Running
Unit Impulse function")
end
```

OUTPUT:





vi) Ramp Function

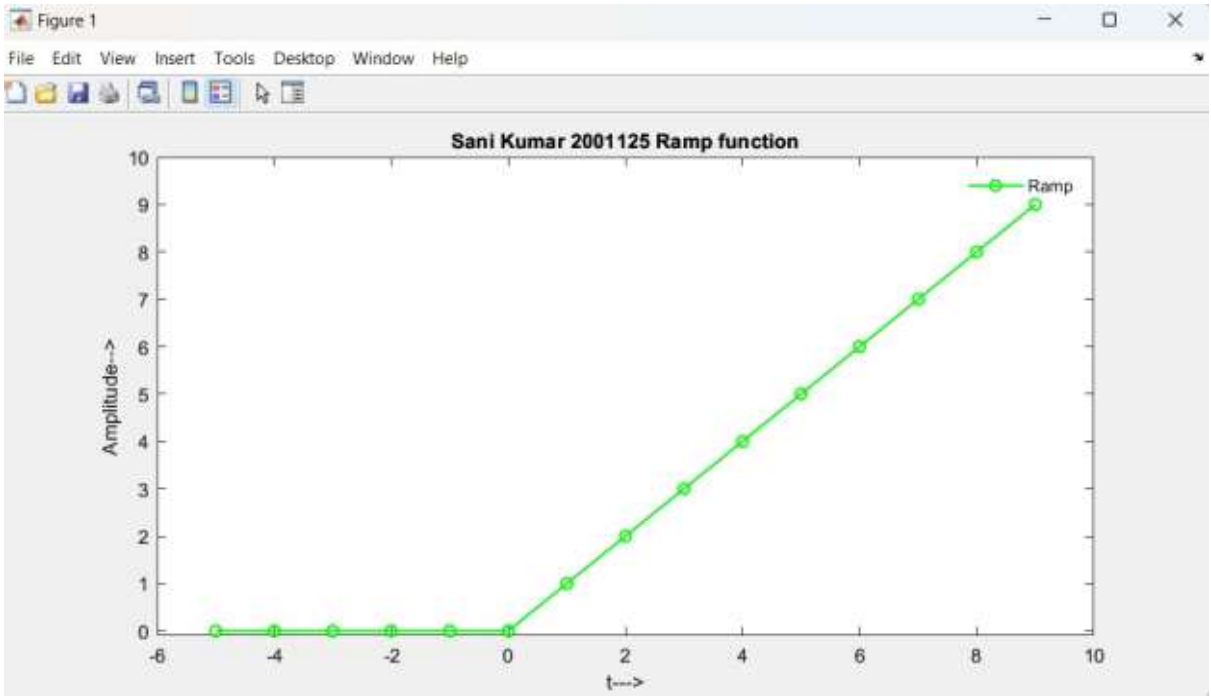
```
 clc;
clear all;
close all;
prompt1 = "Enter the starting value of time
range: ";
prompt2 = "Enter the Increments Value: ";
prompt3 = "Enter the ending Value of time
range: "; s = input(prompt1); i =
input(prompt2); e = input(prompt3); t =
s:i:e;

%calling Ramp signal plot function

rampSignalPlot(t)
```

% Ramp Plot Function Code.

```
function [] = rampSignalPlot(t)
figure; hold on x =t.*(t>=0);
plot(t,x,"g-o","LineWidth",1.3);
xlim([t(1)-1 t(end)+1]); ylim([min(x)-
0.1 max(x)+1]); xlabel("t--->");
ylabel("Amplitude-->");
title("Sani Kumar 2001125 Ramp function")
legend("Ramp"); box
on;
legend("boxoff");
hold off;
fprintf("Running  Ramp function") end
```

OUTPUT:

```
Command Window
Enter the starting value of time range: -5
Enter the Increments Value: 1
Enter the ending Value of time range: 9
Running  Ramp function>>
```



EXPERIMENT-3

1 Verify Sampling theorem through MATLAB coding.

Sampling Theorem: Sampling theorem states that "continues form of a time-variant signal can be represented in the discrete form of a signal with help of samples and the sampled (discrete) signal can be recovered to original form when the sampling signal frequency Fs having the greater frequency value than or equal to the input signal frequency Fm.

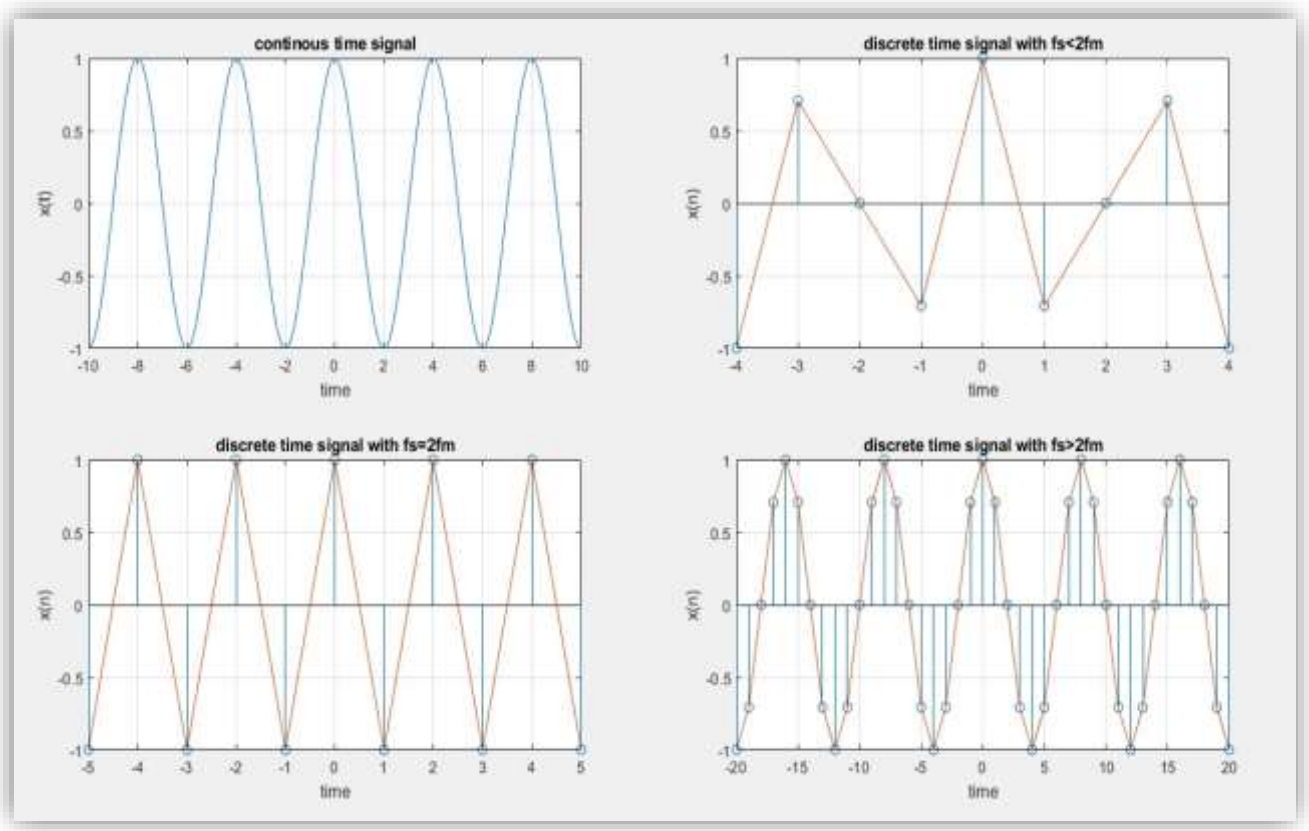
$$Fs \geq 2Fm$$

CODE:

```
t=-10:.01:10;
T=4; fm=1/T;
x=cos(2*pi*fm*t);
subplot(2,2,1);
plot(t,x);
xlabel('time');
ylabel('x(t)')
title('continous time signal')
grid; n1=-4:1:4 fs1=1.6*fm;
fs2=2*fm; fs3=8*fm;
x1=cos(2*pi*fm/fs1*n1);
subplot(2,2,2); stem(n1,x1);
xlabel('time'); ylabel('x(n)')
title('discrete time signal with fs<2fm')
hold on subplot(2,2,2); plot(n1,x1) grid;
n2=-5:1:5;
x2=cos(2*pi*fm/fs2*n2);
subplot(2,2,3);
stem(n2,x2);
xlabel('time');ylabel('x(n)')
title('discrete time signal with
fs=2fm') hold on subplot(2,2,3);
plot(n2,x2) grid; n3=-20:1:20;
x3=cos(2*pi*fm/fs3*n3);
subplot(2,2,4); stem(n3,x3);
```

```
xlabel('time');ylabel('x(n)')
title('discrete time signal with fs>2fm')
hold on subplot(2,2,4); plot(n3,x3) grid;
```

OUTPUT:



EXPERIMENT- 4

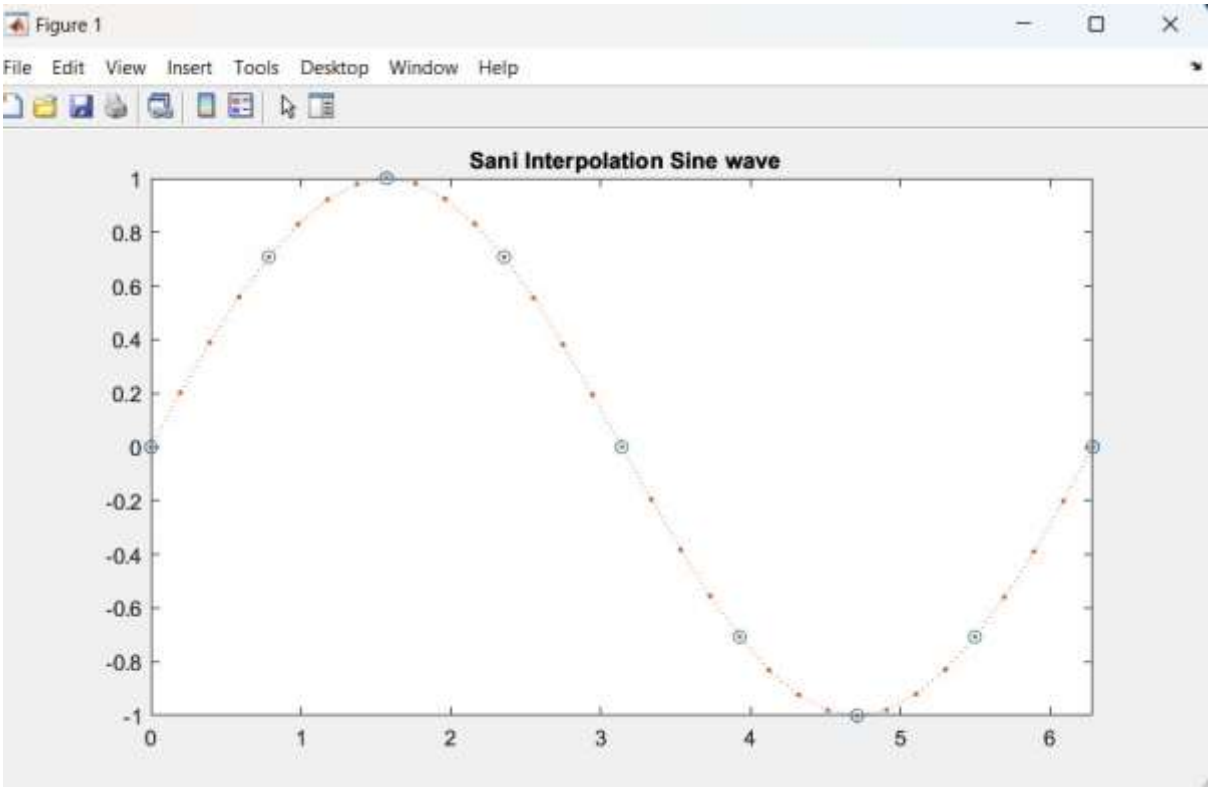
Q4). Perform Interpolation and Decimation of any wave and verify any deformation in wave after performing both actions by same factor by using MATLAB.

Interpolation of Sine Wave: -

CODE:-

```
x = 0:pi/4:2*pi; v
= sin(x); xq =
0:pi/16:2*pi;
%figure
%vq1 = interp1(x,v,xq);
%plot(x,v,'o',xq,vq1,':.');
%xlim([0 2*pi]);
%title('(Default) Linear
Interpolation'); figure
vq2 = interp1(x,v,xq,'spline');
plot(x,v,'o',xq,vq2,':.'); xlim([0
2*pi]);
title(' Sani Interpolation Sine
wave');
```

OUTPUT:

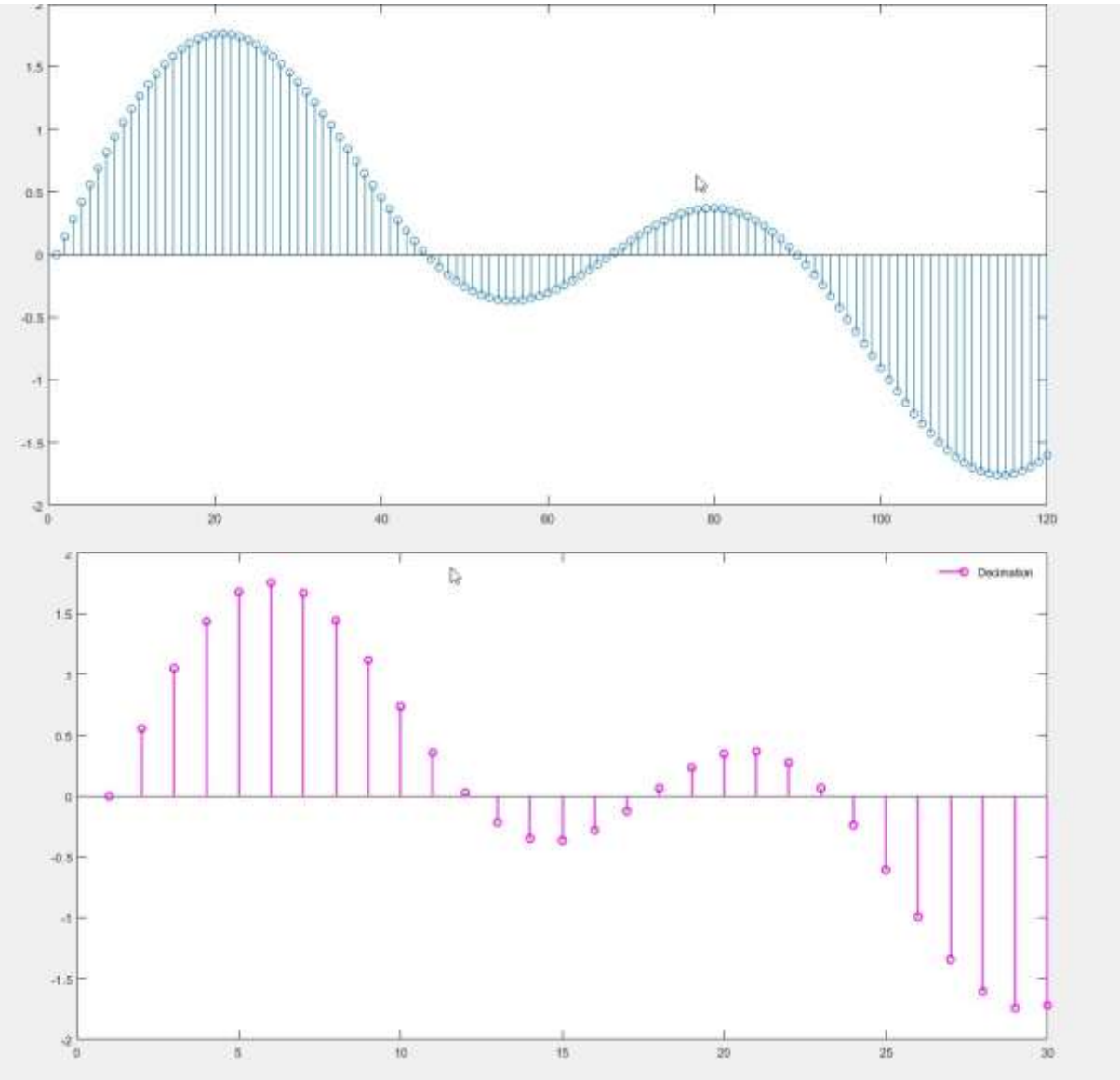


Decimation of Sine Wave: -

CODE:

```
t = 0:.00025:1; % Time vector
x = sin(2*pi*30*t) +
sin(2*pi*60*t); y =
decimate(x,4);
stem(x(1:120)), axis([0 120 -2 2])
% Original signal title(' Sani
Original Signal') figure
stem(y(1:30)) % Decimated signal
title(' Sani Decimated Signal')
```

OUTPUT:-



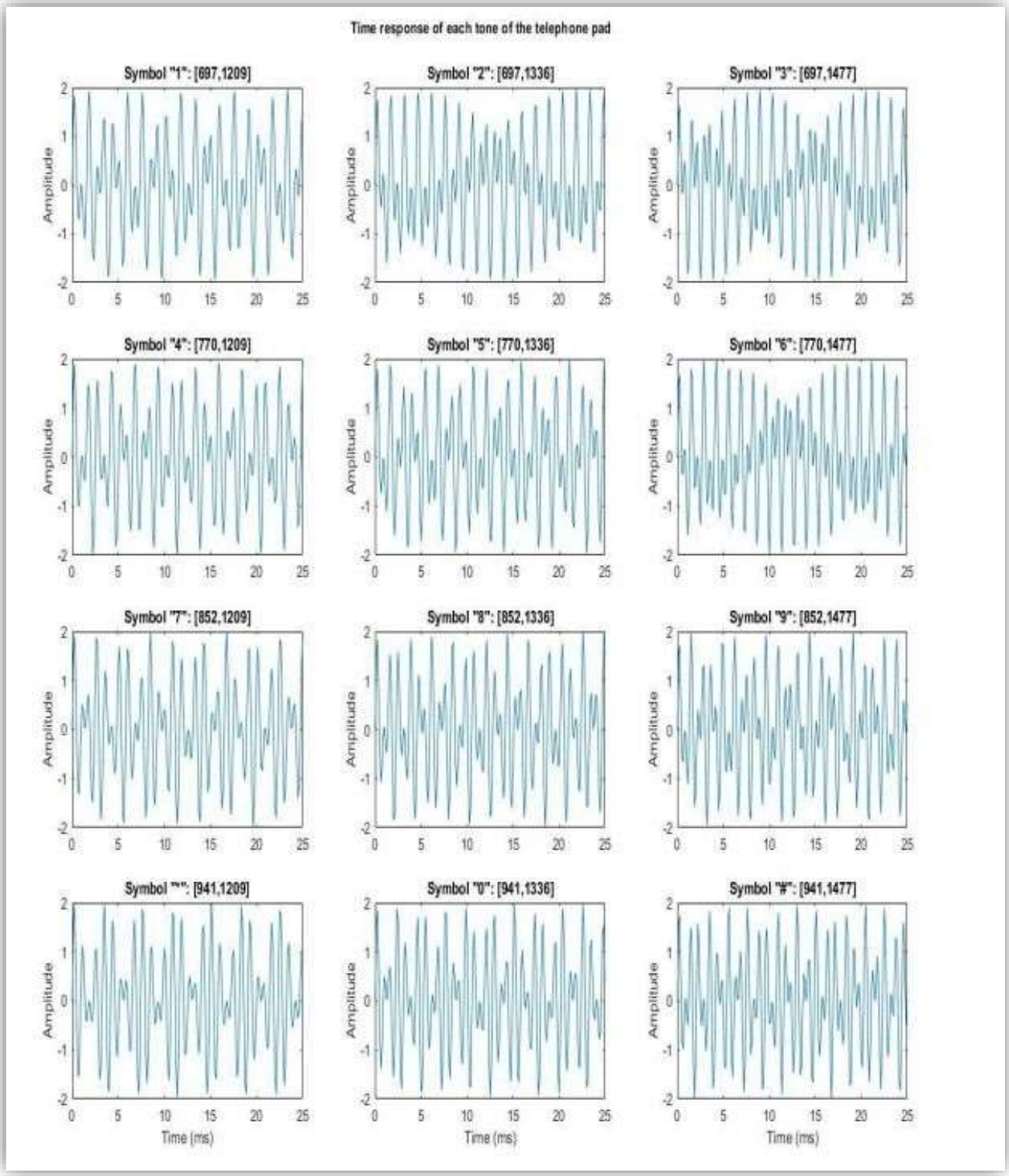
EXPERIMENT-5

Q5. Write MATLAB code to plot graph for Dual Tone Multi Frequency for: 1, 2,3,4,5,6,7,8,9,0, *, #.

CODE:

```
Symbol={'1','2','3','4','5','6','7','8','9','*','0','
#'}; lfg = [697 770 852 941]; % Low frequency
group hfg = [1209 1336 1477]; % High frequency
group f = []; for c=1:4,  for r=1:3,
 f = [ f [lfg(c);hfg(r)] ];
end end
Fs = 8000; % Sampling frequency 8 kHz
N = 800; % Tones of 100 ms t = (0:N-
1)/Fs; % 800 samples at Fs pit =
2*pi*t; tones = zeros(N,size(f,2));
for toneChoice=1:12,  % Generate tone
tones(:,toneChoice) =
sum(sin(f(:,toneChoice)*pit))';
 % Plot tone

subplot(4,3,toneChoice),plot(t*1e3,tones(:,toneChoice
));  title(['Symbol "', symbol{toneChoice},'":
[',num2str(f(1,toneChoice)),',',num2str(f(2,toneChoic
e)),']'])
 set(gca, 'Xlim', [0 25]);
ylabel('Amplitude');
 if toneChoice>9, xlabel('Time (ms)'); end end
set(gcf, 'Color', [1 1 1], 'Position', [1 1 1280
1024]) annotation(gcf,'textbox', 'Position',[0.38
0.96 0.45 0.026],...
 'EdgeColor',[1 1 1],...
 'String', '\bf Time response of each tone of the
telephone pad', ...
 'FitBoxToText','on');
```

OUTPUT:


Time response of each tone of the telephone pad
Symbol "1": [697,1209]
Symbol "2": [697,1336]
Symbol "3": [697,1477]
Symbol "4": [770,1209]
Symbol "5": [770,1336]
Symbol "6": [770,1477]
Symbol "7": [852,1209]
Symbol "8": [852,1336]
Symbol "9": [852,1477]
Symbol "*": [941,1209]
Symbol "0": [941,1336]
Symbol "#": [941,1477]
Amplitude
Time (ms)

EXPERIMENT-6

Q6. (a)Write matlab codes to find convolution, auto and cross correlation of given sequence and analyse output

A= [1 2 5 7]

B= [3 1 2 4]

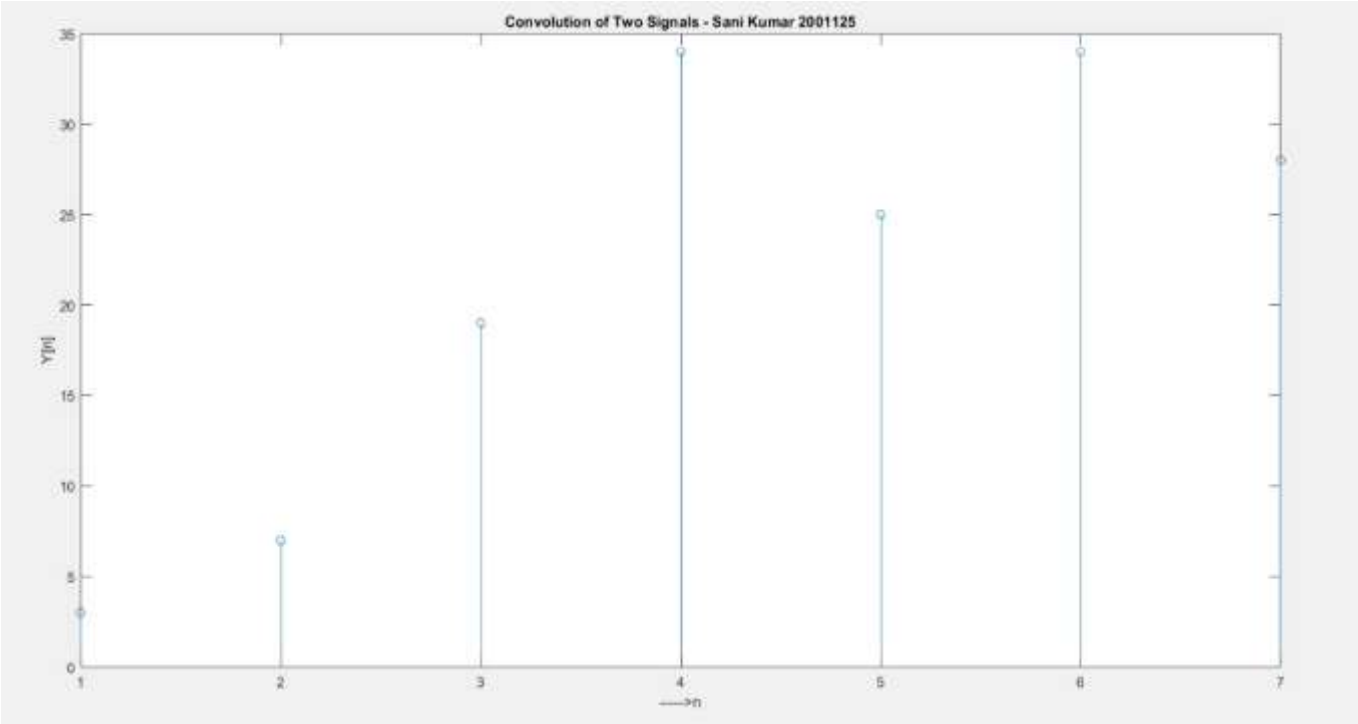
For Convolution-

Code:

```
% linear convolution
close all
x=input('Enter x: ');
h=input('Enter h: ');
m=length(x);
n=length(h);
X=[x,zeros(1,n)];
H=[h,zeros(1,m)];
for i=1:n+m-1
Y(i)=0;
for j=1:i
    Y(i)=Y(i)+X(j)*H(i-j+1);
end
Y
end
stem(Y);
ylabel('Y[n]');
xlabel('----->n');
title('Convolution of Two Signals - Sani Kumar 2001125')
```

Output:

```
Command Window
   xlabel('----->n');
   title('Convolution of Two Signals - Sani Kumar 2001125')
   Enter x: [1 2 5 7]
   Enter h: [3 1 2 4]

   Y =
fx
```



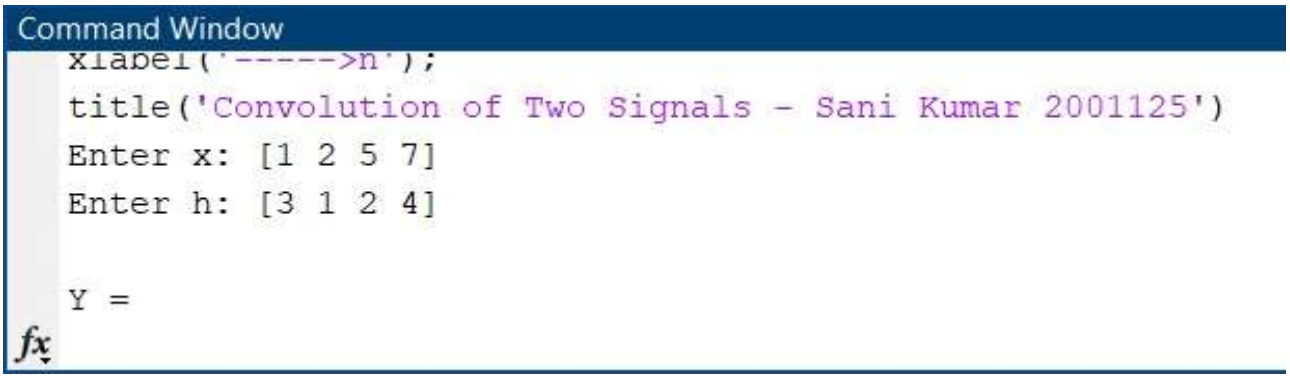
For Auto Correlation-

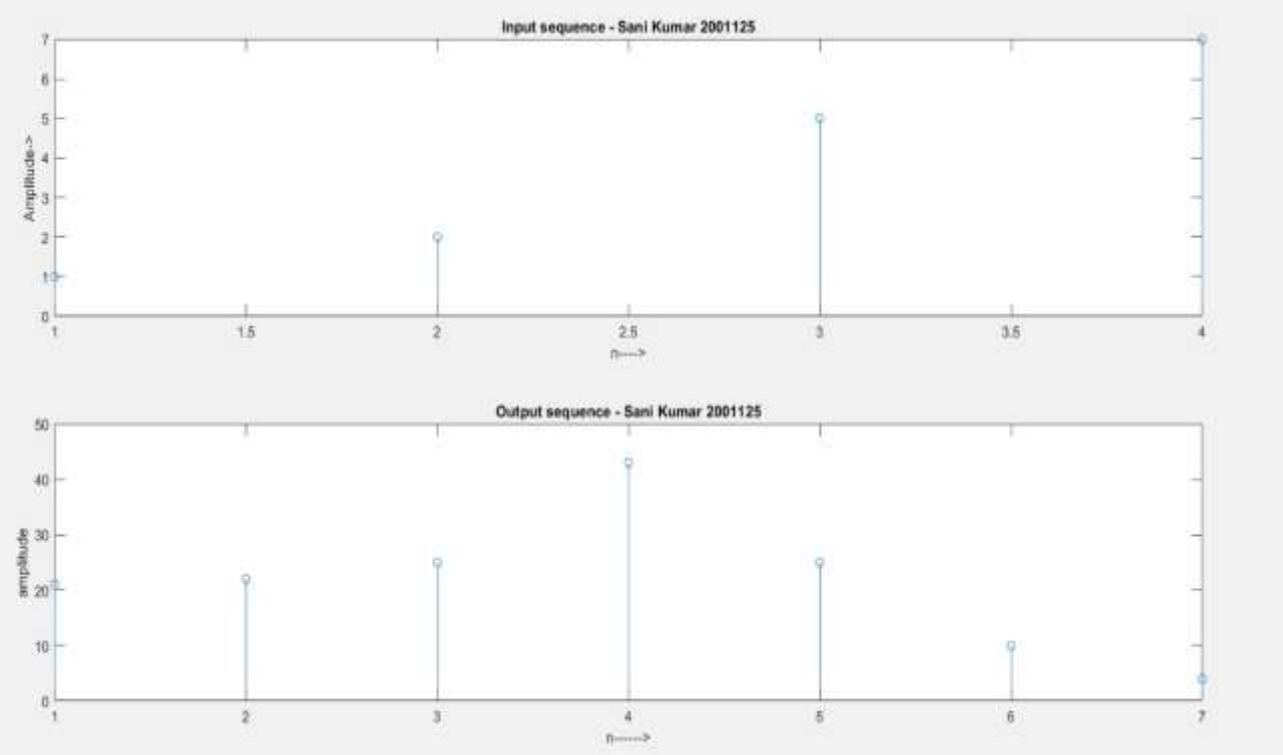
Code:

```
clc;
close all;
x=input('Enter the sequence 1: ');
```

```
y=input('Enter the sequence 2: ');
y=xcorr(x,y);
figure;
subplot(2,1,1);
stem(x);
title('the resultant is ');
ylabel('Amplitude->');
xlabel('n---->');
title('Input sequence - Sani Kumar 2001125');
subplot(2,1,2);
stem(fliplr(y));
ylabel('amplitude');
xlabel('n------>');
title('Output sequence - Sani Kumar 2001125');

fliplr(y);
```

Output :





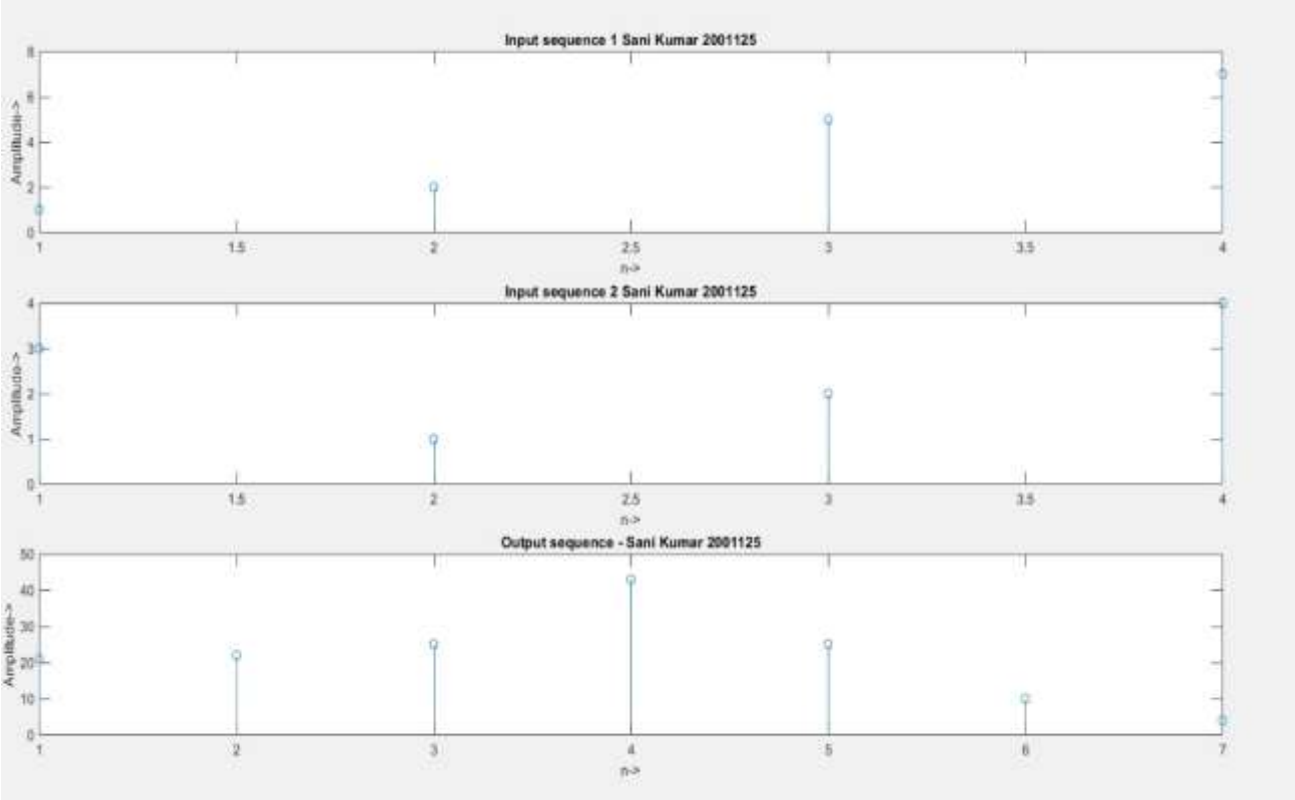
For cross-correlation-

Code:

```
x=input('Enter the sequence 1: ');
h=input('Enter the sequence 2: ');
y=xcorr(x,h);
figure;
subplot(3,1,1);
stem(x);
xlabel('n->');
ylabel('Amplitude->');
title('Input sequence 1 Sani Kumar 2001125');
subplot(3,1,2);
stem(fliplr(y));
stem(h);
xlabel('n->');
ylabel('Amplitude->');
title('Input sequence 2 Sani Kumar 2001125');
subplot(3,1,3);
stem(fliplr(y));
xlabel('n->');
ylabel('Amplitude->');
```

```
title('Output sequence - Sani Kumar 2001125');
disp('The resultant is');
fliplr(y);
```

Output:



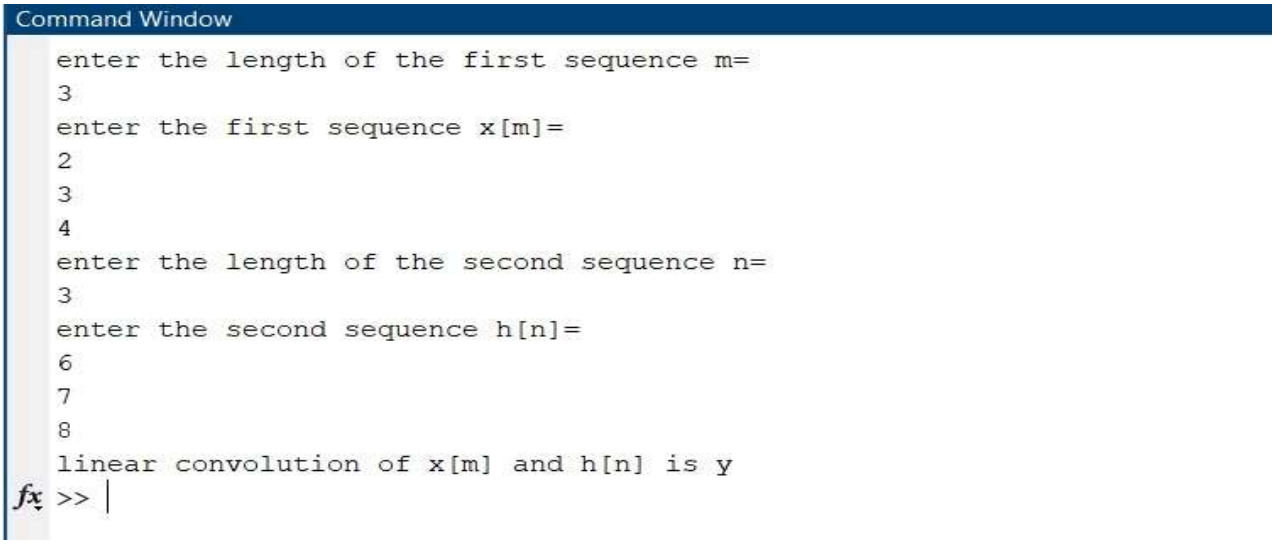
(b) Write Matlab codes to find either convolution or auto correlation or cross-correlation without using inbuilt codes.

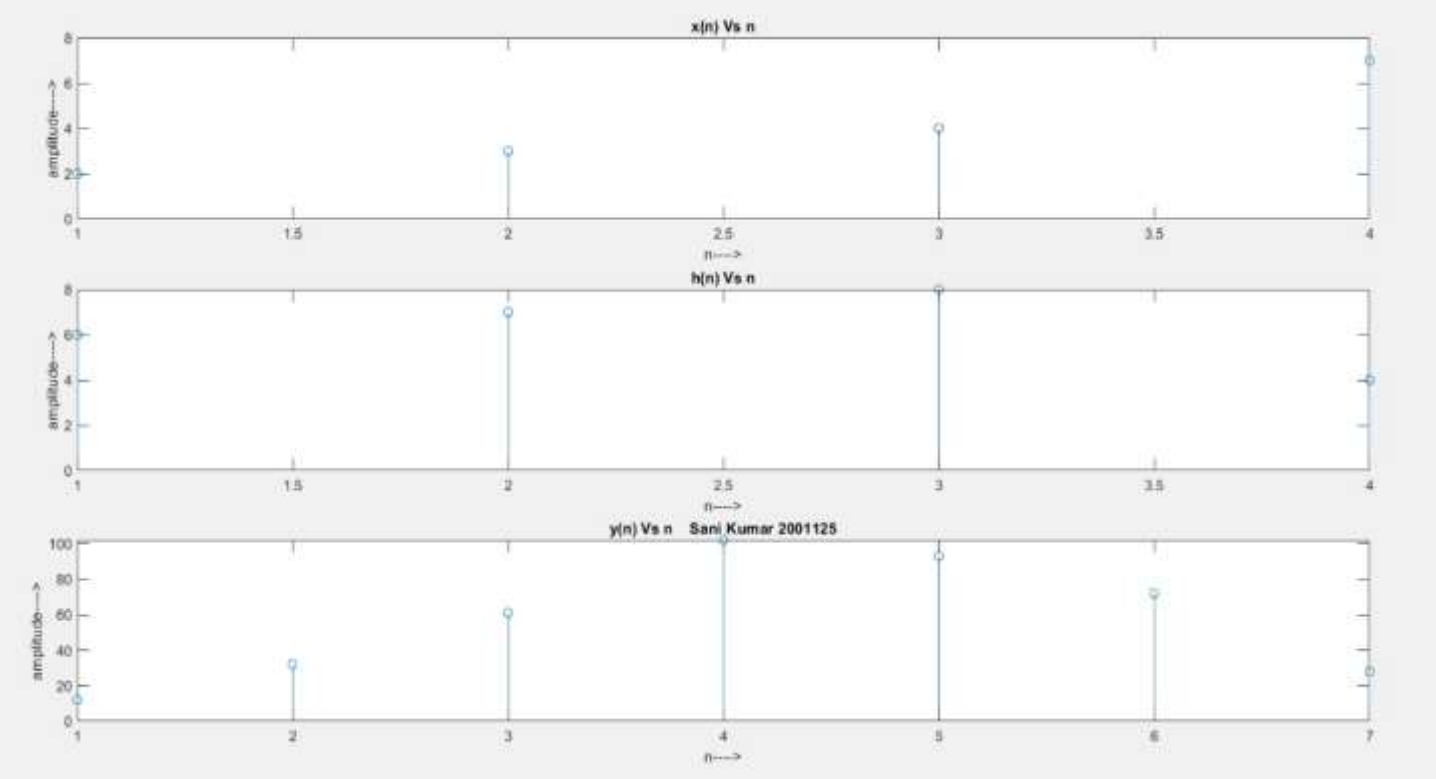
Code:

```
clc;
close;
disp('enter the length of the first sequence m=');
m=input('');
disp('enter the first sequence x[m]=');
for i=1:m
x(i)=input('');
```

```
end
disp('enter the length of the second sequence n=');
n=input('');
disp('enter the second sequence h[n]=');
for j=1:n
h(j)=input('');
end
y=conv(x,h);
figure;
subplot(3,1,1);
stem(x);
ylabel ('amplitude---->');
xlabel('n---->');
title('x(n) Vs n');
subplot(3,1,2);
stem(h);
ylabel('amplitude---->');
xlabel('n---->');
title('h(n) Vs n');
subplot(3,1,3);
stem(y);
ylabel('amplitude---->');
xlabel('n---->');
title('y(n) Vs n    Sani Kumar 2001125');disp('linear
convolution of x[m] and h[n] is y');
```

Output:





EXPERIMENT-7

7) Write a MATLAB code to design IIR Butterworth lowpass, highpass, bandpass and bandstop filter and verify its characteristics for given cutoff frequency: lowpass- 1200 Hz , highpass-1200 Hz bandpass- 1200 Hz to 1800Hz ,bandstop- 1200 Hz to 1800Hz

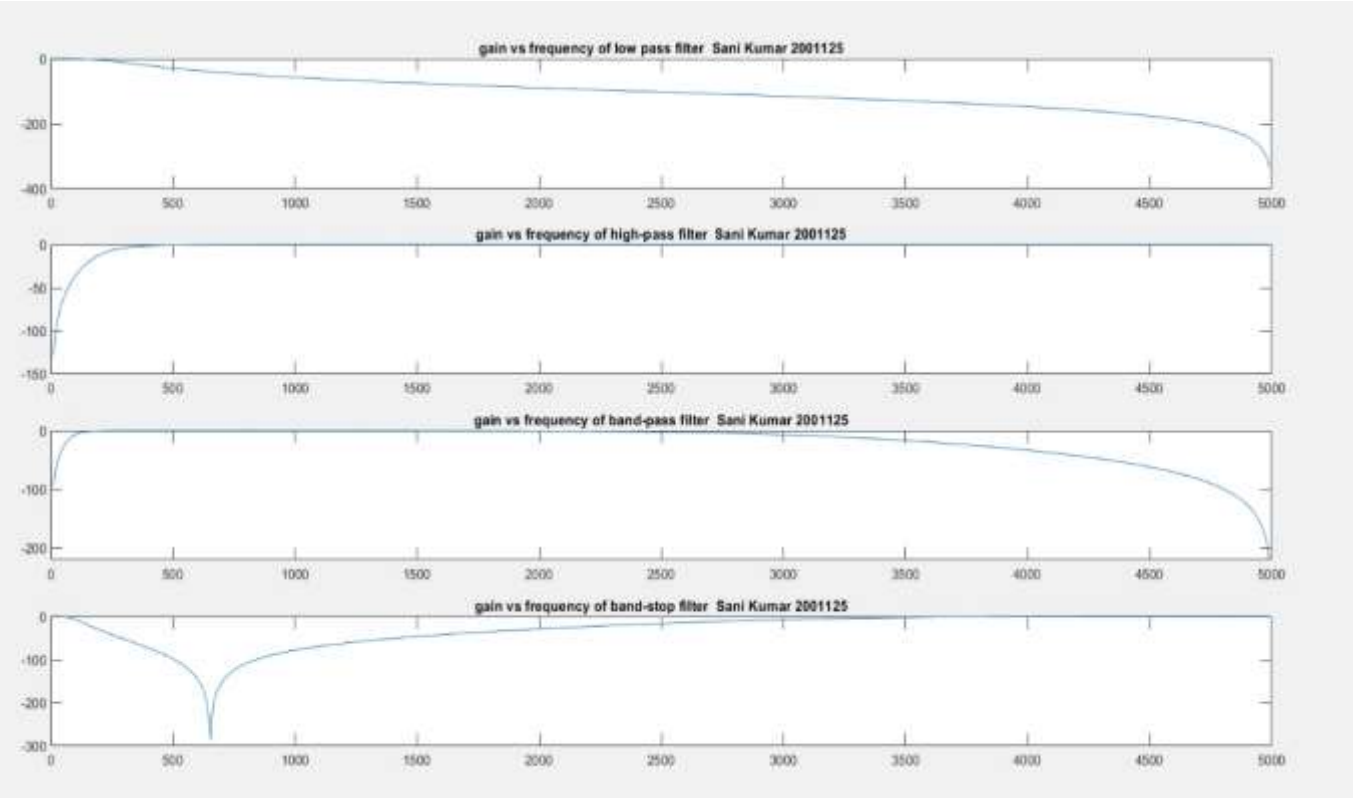
Code:

```
rp=input('enter passband ripple value');
rs=input('enter stopband ripple value');
fp=input('enter passband freq value');
fs=input('enter stopband freq value');
fsp=input('enter sampling freq value');
w1=2*fp/fsp;
w2=2*fs/fsp;
[n,wn]=buttord(w1,w2,rp,rs);
[b,a]=butter(n,wn,'low');
```

```
[h,w]=freqz(b,a,512,fsp);
mag=20*log(abs(h));
subplot(4,1,1);
plot(w,mag);
title('gain vs frequency of low pass filter  Sani Kumar
2001125');
%high pass
[b,a]=butter(n,wn,'high');
[h,w]=freqz(b,a,512,fsp);
mag=20*log(abs(h));
subplot(4,1,2);
plot(w,mag);
title('gain vs frequency of high-pass filter  Sani Kumar
2001125);
%band pass
[n]=buttord(w1,w2,rp,rs);
wn=[w1,w2];
[b,a]=butter(n,wn,'bandpass');
[h,w]=freqz(b,a,512,fsp);
mag=20*log(abs(h));
subplot(4,1,3);
plot(w,mag);

title('gain vs frequency of band-pass filter  Sani Kumar
2001125);
%band stop
[n]=buttord(w1,w2,rp,rs);
[b,a]=butter(n,wn,'stop');
[h,w]=freqz(b,a,512,fsp);
mag=20*log(abs(h));
subplot(4,1,4);
plot(w,mag);
title('gain vs frequency of band-stop filter  Sani Kumar
2001125');
```

Output:

```
enter passband ripple value 3
enter stopband ripple value 50
enter passband freq value 100
enter stopband freq value 3000
enter sampling freq value 10000
fx >>
```



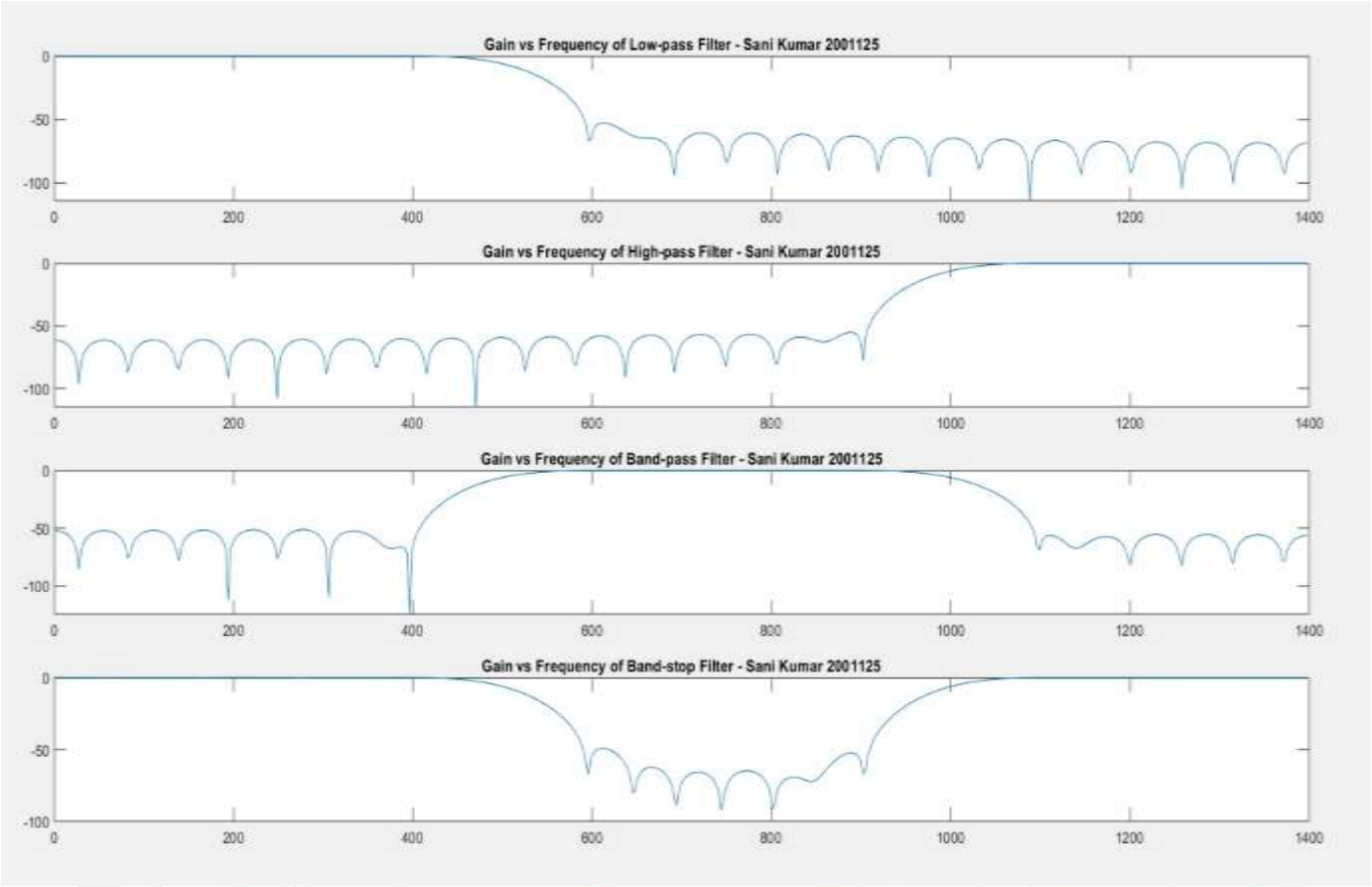
EXPERIMENT-8

8) Write a MATLAB code to design FIR lowpass, high pass, bandpass and band stop filters and verify its characteristics for given cutoff frequency: lowpass- 1500 Hz , highpass-1500 Hz bandpass- 1000 Hz to 2800Hz , band stop- 1000 Hz to 2800Hz

Code:

```
fp=input('enter the passing freq:');
fstop=input('enter the stopping freq:');
fsp=input('enter the sampling freq value:');
n=input('enter the order of the filter:');
w1=2*fp/fsp;
w2=2*fstop/fsp;
%low pass filter
A=fir1(n,w1,'Low');
%freqz(A,1,512,fsp);
[h,w]=freqz(A,1,512,fsp);
mag=20*log10(abs(h));
subplot(4,1,1);
plot(w,mag);
title('Gain vs Frequency of Low-pass Filter - Sani Kumar
2001125');
%high pass filter
A=fir1(n,w2,'high');
%freqz(A,1,512,fsp);
[h,w]=freqz(A,1,512,fsp);
mag=20*log10(abs(h));
subplot(4,1,2);
plot(w,mag);
title('Gain vs Frequency of High-pass Filter - Sani Kumar
2001125');
%band pass filter
wn=[w1,w2];
A=fir1(n,wn,'bandpass');
%freqz(A,1,512,fsp);
[h,w]=freqz(A,1,512,fsp);
mag=20*log10(abs(h));
subplot(4,1,3);
plot(w,mag);
title('Gain vs Frequency of Band-pass Filter - Sani Kumar
2001125');
%band stop filter
A=fir1(n,wn,'stop');
%freqz(A,1,512,fsp);
[h,w]=freqz(A,1,512,fsp);
mag=20*log10(abs(h));
subplot(4,1,4);
plot(w,mag);
title('Gain vs Frequency of Band-stop Filter - Sani kumar
2001125');
```

Output:




```
title('Gain vs Frequency of Band-stop Filter - Sani Kumar 2001125');
enter the passing freq:500
enter the stopping freq:1000
enter the sampling freq value:2800
enter the order of the filter:50
>>
```