AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Faculty of Science and Technology

Course Title: Data Communication

Lab Report-2

Submitted by:	***Submitted to:***
Name: Shifat, Shadril Hassan ID: 20-42451-1 Section: G Program: BSc CSE Semester: Spring 2021-2022 Date: 19 February, 2022	Course Teacher: Tanjil Amin

<u>Class Task:</u>

Similar task can be done where we use a composite signal instead of signals x1 and x2. Suppose our composite signal is

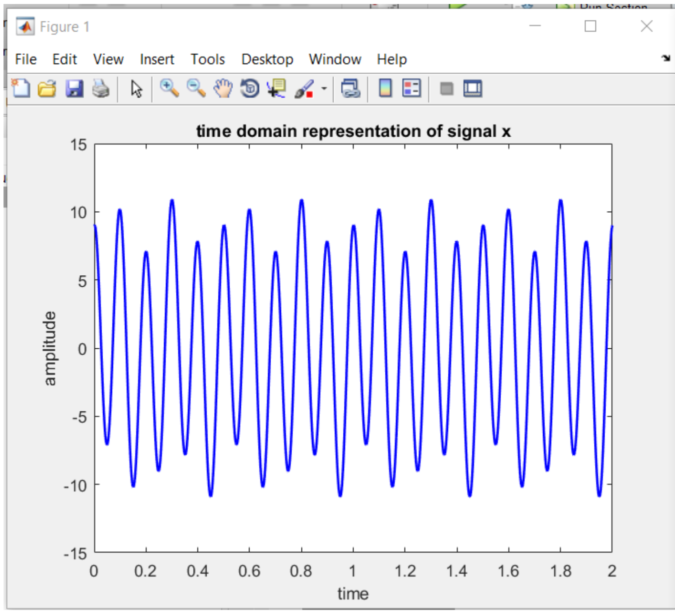
```
signal_x = a1*sin(2*pi*f1*t) + a2*cos(2*pi*f2*t);
Here, a1 = (B + G + H), a2 = (C + E + H), f1 = (G + H + 2), and
f2 = (E + F+ H). [Assume your ID is AB-CDEFG-H]
```

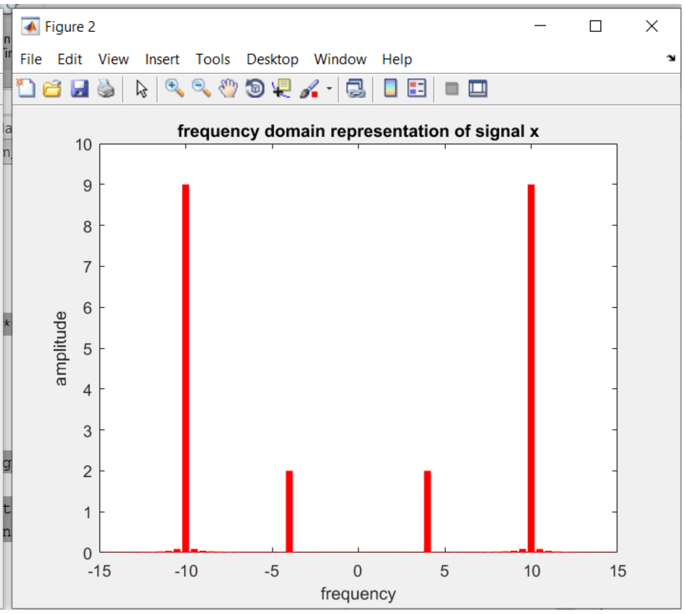
*******Show this signal both in time domain and frequency domain.**

<u>MATLAB Code:</u>

```
clc;
clear all;
close all;
A=2;
B=0;
C=4;
D=2;
E=4;
F=5;
G=1;
H=1;
a1= B+G+H;
a2= C+E+H;
f1= G+H+2;
f2= E+F+H;

fs = 1000; %sampling frequency
t = 0:1/fs:2; % time array
signal_x = a1*sin(2*pi*f1*t) + a2*cos(2*pi*f2*t);
% time domain representation
figure;
plot(t,signal_x,'b','linewidth',1.5);
xlabel('time');
ylabel('amplitude');
title('time domain representation of signal x');
% frequency domain representation
fx = abs(fftshift(fft(signal_x)))/(length(signal_x)/2);
freq = linspace(-fs/2, fs/2, length(signal_x));
figure;
bar(freq, fx,'r','linewidth',1);
xlim([-15 +15]);
xlabel('frequency');
ylabel('amplitude');
title('frequency domain representation of signal x')
```

Output:





Performance Task for Lab Report: (your ID = AB-CDEFG-H)

**Generate a composite signal using three simple signals as,

x1 = a1*cos(2*pi*f1*t), x2 = a2*sin(2*pi*f2*t), x3 = a3*cos(2*pi*f3*t)
signal_x = x1 + x2 + x3

Select the values of the amplitude and frequency as follows: a1 = A + C + 1, a2 = A + D + 2, a3 = A + E + 1, f1 = A + E + 1, f2 = A + D + 2, f3 = A + C + 1.

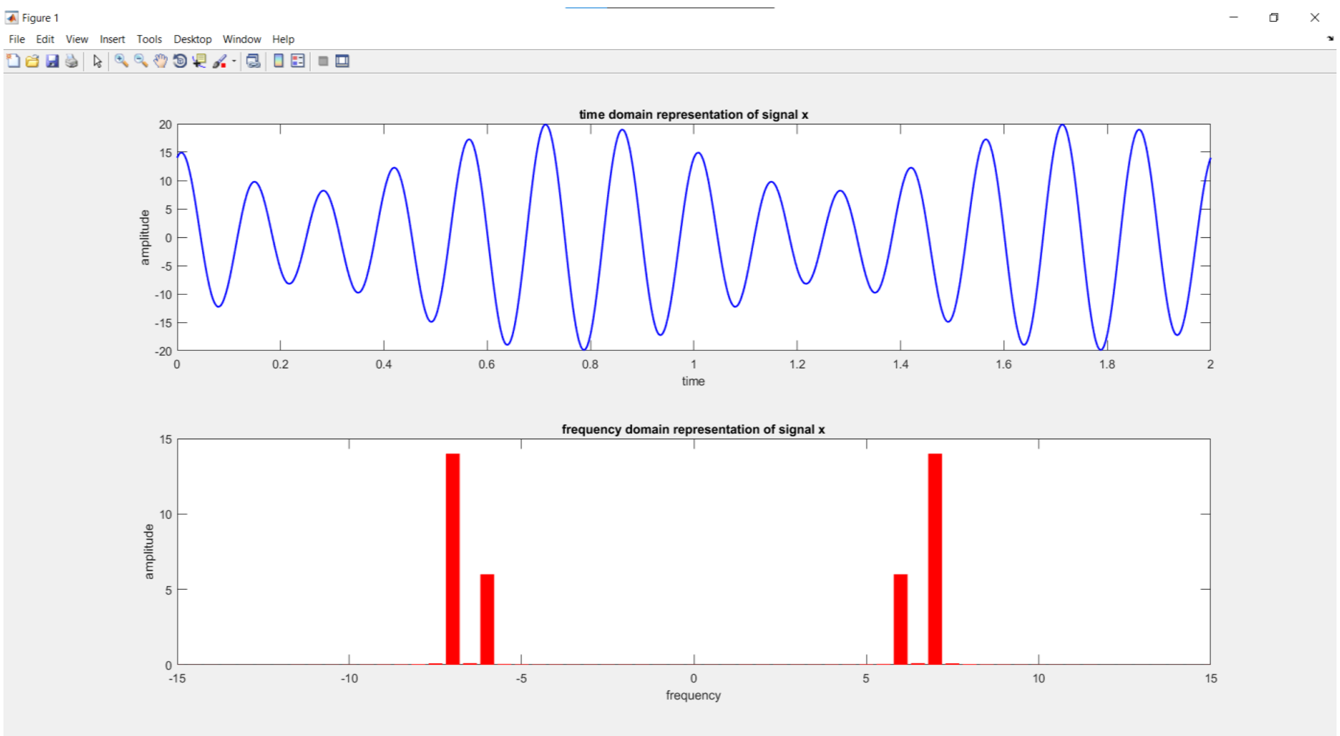
(a) Show time domain and frequency domain representations of signal_x in a single figure window using subplot. Use axis, or xlim, or ylim to appropriately represent the signal.

MATLAB Code:

```
clc;
clear all;
close all;

A=2;
B=0;
C=4;
D=2;
E=4;
```

```
F=5;
G=1;
H=1;

a1= A+C+1;
a2= A+D+2;
a3= A+E+1;
f1= A+E+1;
f2= A+D+2;
f3= A+C+1;

fs = 1000; %sampling frequency
t = 0:1/fs:2; % time array

x1= a1*cos(2*pi*f1*t);
x2= a2*sin(2*pi*f2*t);
x3 = a3*cos(2*pi*f3*t);
signal_x= x1+x2+x3;

% Question (a)

% time domain representation
subplot(2,1,1);
plot(t,signal_x,'b','linewidth',1.5);
xlabel('time');
ylabel('amplitude');
title('time domain representation of signal x');
% frequency domain representation
fx = abs(fftshift(fft(signal_x)))/(length(signal_x)/2);
freq = linspace(-fs/2, fs/2, length(signal_x));
subplot(2,1,2);
bar(freq, fx,'r','linewidth',1.5);
xlim([-15 +15]);
xlabel('frequency');
ylabel('amplitude');
title('frequency domain representation of signal x');
```

Output:



(b) Quantize signal_x in 8 equally distributed levels and provide image for one cycle of the original signal and quantized signal. Use axis, or xlim, or ylim to appropriately represent the signal.

MATLAB Code:

```
clc;
clear all;
close all;

A=2;
B=0;
C=4;
D=2;
E=4;
F=5;
G=1;
H=1;

a1= A+C+1;
```

```
a2= A+D+2;
a3= A+E+1;
f1= A+E+1;
f2= A+D+2;
f3= A+C+1;

fs = 1000; %sampling frequency
t = 0:1/fs:1; % time array
x1= a1*cos(2*pi*f1*t);
x2= a2*sin(2*pi*f2*t);
x3 = a3*cos(2*pi*f3*t);
signal_x= x1+x2+x3;
% Question (b)
p= linspace(-15,15,7);
c= linspace(-20,20,8);
[i,q]= quantiz(signal_x,p,c);

plot(t,signal_x,'*',t,q,'x','linewidth',1.5);
xlabel('time');
ylabel('amplitude');
title('quantization using quantiz() of signal x');
legend('signal x','quantized signal x');
```

Output: