Task03 : Signal Processing II

Name: Wilson Carofilis
Degree: PH
ID: 20986777

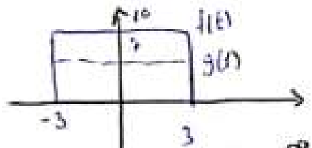
Problem 1: Convolution

(a)

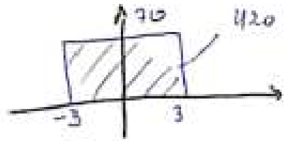
• Problem 1: Convolution, a)

$f(t) = \begin{cases} A; & a > |t| \\ 0; & \text{otherwise} \end{cases}$ $\qquad g(t) = \begin{cases} B; & b > |t| \\ 0; & \text{otherwise} \end{cases}$

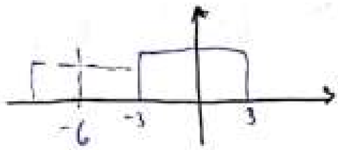
$A = 10 \wedge a = 3$ $\qquad B = 7 \wedge b = 7$



$\Rightarrow f(t) \cdot g(t)$

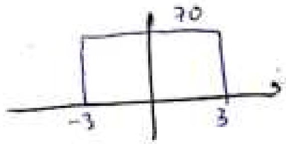


$y(t) = f(t) * g(t) \rightarrow \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau \Rightarrow$ therefore 3 cases.

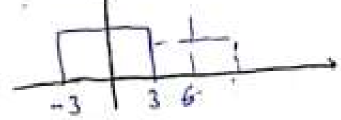


$y(-6) = 0$

and $t < -6$



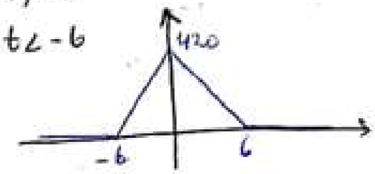
$y(0) = 420$



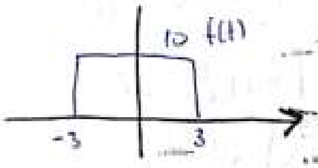
$y(6) = 0$

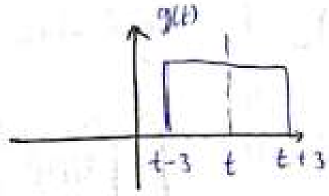
and $t > 6$

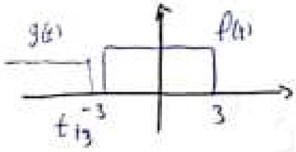
$\Rightarrow$

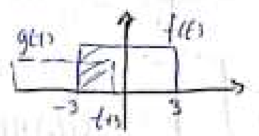


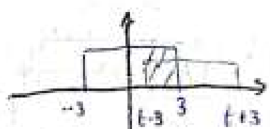
Another way to see this.

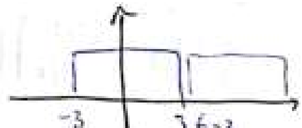












$t+3 < -3$
$t < -6$
$y(t) = 0$

$-3 < t+3 \quad t+3 < 3$
$t > -6 \quad t < 0$

$t-3 < 3 \quad t+3 > 3$
$t < 6 \quad t > 0$

$t-3 > 3$
$t > 6$

$\cancel{\int_{-\infty}^{-6} f(\tau) g(t-\tau)\, d\tau} \rightarrow 0$

$\int_{-3}^{t+3} (10)(7)\, d\tau$

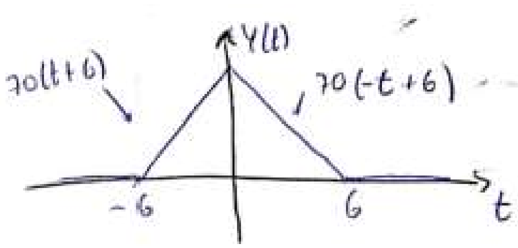
$70[t+3+3]$

$\int_{t-3}^{3} (10)(7)\, d\tau$

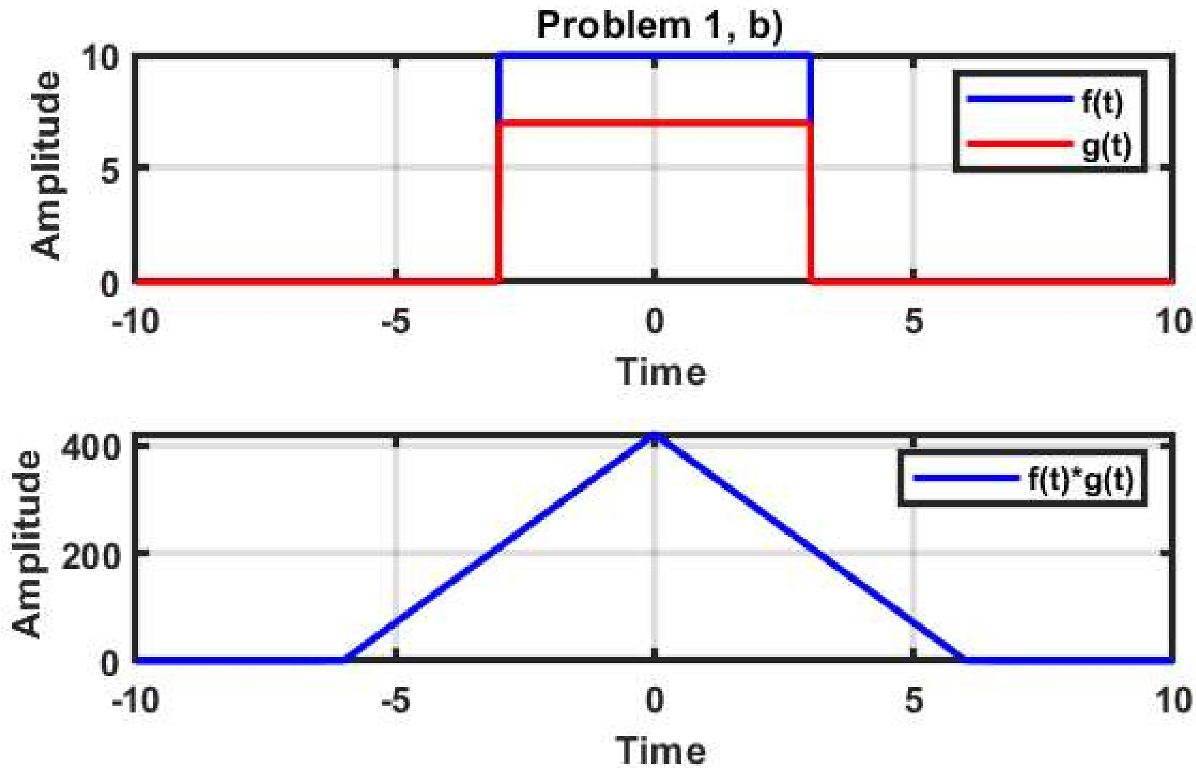
$70[3-t+3]$

$\cancel{\int_{6}^{\infty} f(\tau) g(t-\tau)\, d\tau} \rightarrow 0$

$y(t) = \begin{cases} 70(t+6); & -6 \le t < 0 \\ 70(-t+6); & 0 < t \le 6 \\ 0; & \text{otherwise} \end{cases}$



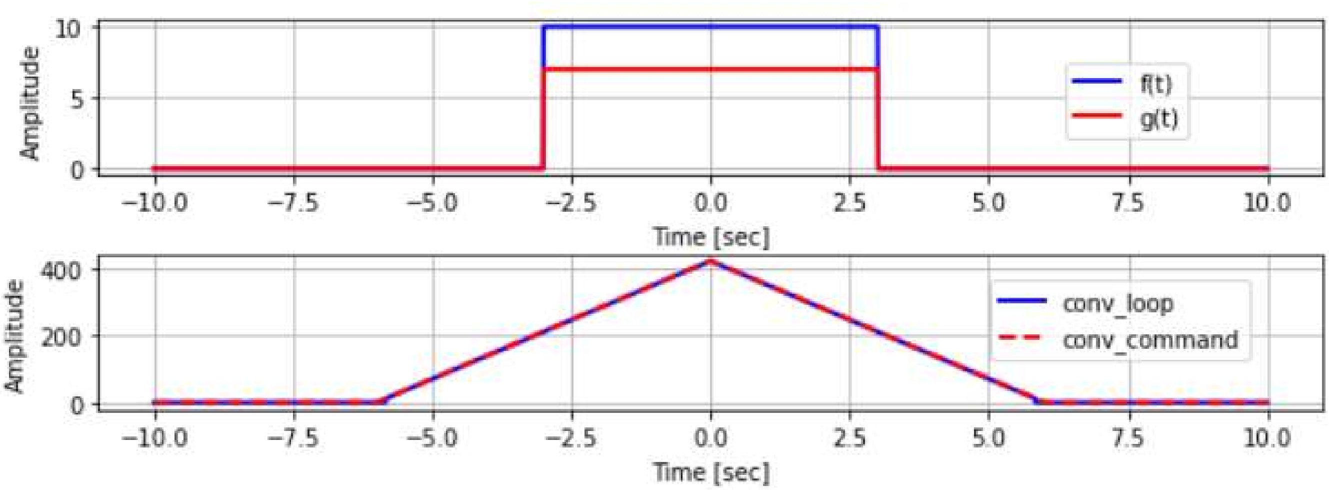
(b)




```
clear; clc ; close all;

syms t

A = 10; a=3;
f_t = @(t) A*(abs(t)<=a) + 0; % defining function f(t)

B = 7; b=3;
g_t = @(t) B*(abs(t)<=b) + 0; % defining function g(t)

t = -10:0.01:10; % time vector

% convolution using loop
count=1; %start index for y(t)
t_y = -10:10;
for ii=t_y
    y(count) = trapz(t,f_t(t).*g_t(ii-t)); % get the area of each shift
    count = count+1; % increase index for y(t)
end
```

```
fig1 = figure(1);
set(fig1,'Position', [100 100 500 300]);
subplot(211)
plot(t,f_t(t),'-b', 'linewidth', 2);hold on;
plot(t,g_t(t),'-r', 'linewidth', 2);
legend('f(t)', 'g(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time')
title('\bf Problem 1, b)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');

subplot(212);
plot(t_y,y,'-b', 'linewidth', 2)
legend('f(t)*g(t)'); axis tight;grid on;
ylabel('\bf Amplitude')
xlabel('\bf Time')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

[Bonus, script done in Python (Jupyter)]

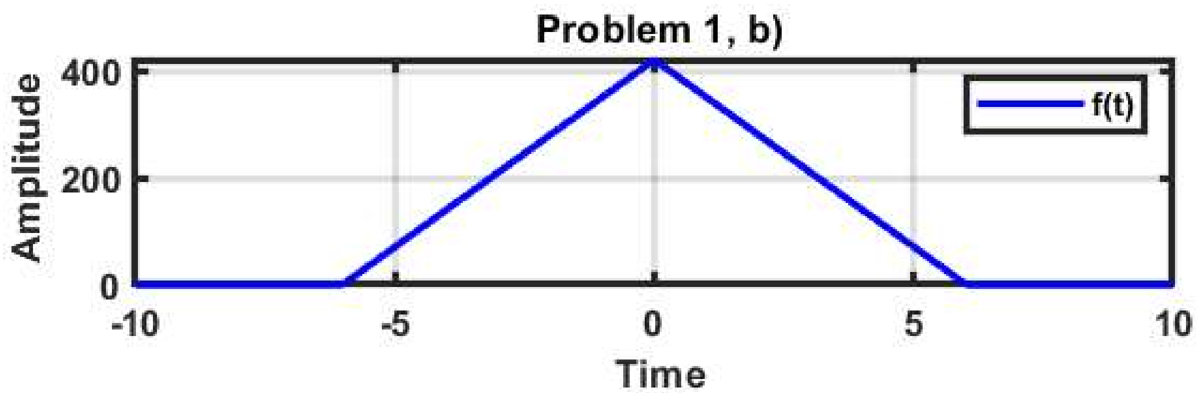



```
#Python script for Assignment 3
#Problem 1: Convolution b)
import numpy as np
import matplotlib.pyplot as plt
import scipy.integrate as integrate

#Parameters for function f(t)
```

```python
a = 3.0; A = 10.0;
f_t = lambda t0: A*(abs(t0)<=a) + 0

#Parameters for function g(t)
b = 3.0; B = 7.0;
g_t = lambda t0: B*(abs(t0)<=b) + 0

NPC = 2001; # Sampling rate/cycle (assumed high rate)
t0 = np.linspace(-10.0,10.0,NPC) # time vector

f = [f_t(t0) for t0 in t0]
g = [g_t(t0) for t0 in t0]
f =np.array(f)
g =np.array(g)
y_con = [] # initiate conv vector

for ii in range(len(t0)):
    t = t0[ii]
    fg1 = lambda t0: f_t(t0)*g_t(t-t0)
    y_con.append(integrate.quad(fg1, -10, 10)[0])

y_con2=np.convolve(f,g, 'same')*0.01

# plt.figure(figsize=(9, 3))
fig, aw = plt.subplots(figsize=(9,3),nrows=2, ncols=1)
aw[0].plot(t0,f,'b-',lw=2,label ='f(t)')
aw[0].plot(t0,g,'r-',lw=2,label ='g(t)')
aw[0].legend(fontsize =10, bbox_to_anchor=(0.9,0.8), loc=1)
aw[0].grid();
aw[0].set_ylabel('Amplitude')
aw[0].set_xlabel('Time [sec]')

aw[1].plot(t0,y_con,'b-',lw=2,label ='conv_loop')
aw[1].plot(t0,y_con2,'r--',lw=2,label ='conv_command')
aw[1].grid();
aw[1].set_ylabel('Amplitude')
aw[1].set_xlabel('Time [sec]')
aw[1].legend(fontsize =10, bbox_to_anchor=(0.95,0.9), loc=1)
```

```
fig.subplots_adjust(hspace =0.5, right =0.98, left=0.15, bottom = 0.1, top = 0.89)
plt.show()
```

(c)




```
f_t_con = conv(f_t(t), g_t(t),'same')*0.01; % obtaining the convolution through "conv"
 command

fig2 = figure(2);
set(fig2,'Position', [100 100 500 150]);
plot(t,f_t_con,'-b', 'linewidth', 2)

legend('f(t)', 'g(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time')
title('\bf Problem 1, c)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

Problem 2: Convolution Theorem

(a)

These equations show the relationship between function in the time and frequency domain. The Fourier transform of the convolution of two functions in the time domain becomes the multiplication of these functions in the frequency domain. On the other hand, the Fourier transform of the multiplication of two functions in the time domain is the convolution of these functions in the frequency domain.

• Problem 2: Convolution theorem, a)

$$\bullet\ \mathcal{F}(x(t) * h(t)) = X(f) . H(f) = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} x(\tau) h(t-\tau) d\tau\right] e^{-i2\pi f t} dt \Rightarrow$$

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-i2\pi f t} dt$$

$$H(f) = \int_{-\infty}^{\infty} h(t) e^{-i2\pi f t} dt$$

let's do a change of variable $\rightarrow$ $p = t - \tau \rightarrow t = \tau + p$

$dt = dp$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(\tau) h(p) d\tau\, e^{-i2\pi f(\tau+p)} dp = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(\tau) h(p) d\tau\, e^{-i2\pi f\tau} e^{-i2\pi f p} dp$$

$$\Rightarrow \underbrace{\int_{-\infty}^{\infty} x(\tau) e^{-i2\pi f\tau} d\tau}_{X(f)} \underbrace{\int_{-\infty}^{\infty} h(p) e^{-i2\pi f p} dp}_{H(f)} \Rightarrow \underline{X(f) . H(f)} \ //$$

$$\bullet\ \mathcal{F}(x(t) . h(t)) = X(f) * H(f) = \int_{-\infty}^{\infty} X(\tau) . H(f-\tau) d\tau$$

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{i2\pi f t} df$$

$$h(t) = \int_{-\infty}^{\infty} H(f) e^{i2\pi f t} df$$

$$= \int_{-\infty}^{\infty} (x(t) . h(t)) e^{-i2\pi f t} dt = \int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty} X(f) e^{i2\pi f t} df\right)\left(\int_{-\infty}^{\infty} H(f) e^{i2\pi f t} df\right) e^{-i2\pi f t} dt$$

$$\Rightarrow \int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty} X(f) e^{i2\pi f_1 t} df_1\right)\left(\int_{-\infty}^{\infty} H(f) e^{i2\pi f_2 t} df_2\right) e^{-2i\pi f t} dt = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(f_1) H(f_2) e^{i2\pi(f_2+f_1-f)t} df_1\, df_2\, dt$$

$$\Rightarrow \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(f_1) H(f_2) \underbrace{\int_{-\infty}^{\infty} e^{-i2\pi(f_2+f_1-f)t} dt}_{\substack{\delta(f_2+f_1-f) \\ \delta(f_2-(f-f_1))}} df_1\, df_2 = \int_{-\infty}^{\infty} X(f_1) df_1 \underbrace{\int_{-\infty}^{\infty} H(f_2)\, \delta(f_2-(f-f_1)) df_2}_{H(f-f_1)}$$

$$\Rightarrow \int_{-\infty}^{\infty} X(f_1) H(f-f_1) df_1 \Rightarrow \underline{X(f) * H(f)} \ //$$

(b)

• Problem 2 : Convolution theorem, b)

$$x(t) = \begin{cases} 1 - |t| & ; |t| < 1 \\ 0 & ; |t| > 1 \end{cases}$$

$$X(f) = \int_{-\infty}^{\infty} [1-|t|] e^{-i2\pi ft} dt = \cancel{\int_{-\infty}^{-1} (0) e^{-i2\pi ft} dt}^{\,0} + \int_{-1}^{0} [1+t] e^{-i2\pi ft} dt + \int_{0}^{1} [1-t] e^{-i2\pi ft} dt + \cancel{\int_{1}^{\infty} (0) e^{-i2\pi ft} dt}^{\,0}$$

$$\Rightarrow \int_{-1}^{0} (e^{-i2\pi ft} + t e^{-i2\pi ft}) dt + \int_{0}^{1} (e^{-i2\pi ft} - t e^{-i2\pi ft}) dt \Rightarrow$$

$$\bullet \; t e^{-i2\pi ft} \rightarrow g(x) = t \longrightarrow g'(x) = 1.dt$$

$$f(x) = e^{-i2\pi ft} \longrightarrow F(x) = \int e^{-i2\pi ft} dt = \frac{-e^{-i2\pi ft}}{i2\pi f}$$

$$F(x) g(x) - \int F(x).g'(x)$$

$$\frac{-t e^{-i2\pi f}}{i2\pi f} + \int \frac{e^{-i2\pi ft}}{i2\pi f} dt = \frac{-t e^{-i2\pi f}}{i2\pi f} - \frac{e^{-i2\pi ft}}{(i2\pi f)^2}$$

$$\Rightarrow \left[\frac{-e^{-i2\pi ft}}{i2\pi f} - \frac{t e^{-i2\pi ft}}{i2\pi f} - \frac{e^{-i2\pi ft}}{(i2\pi f)^2} \right]_{-1}^{0} + \left[\frac{-e^{-i2\pi ft}}{i2\pi f} + \frac{t e^{-i2\pi f}}{i2\pi f} + \frac{e^{-i2\pi f}}{(i2\pi f)^2} \right]_{0}^{1}$$

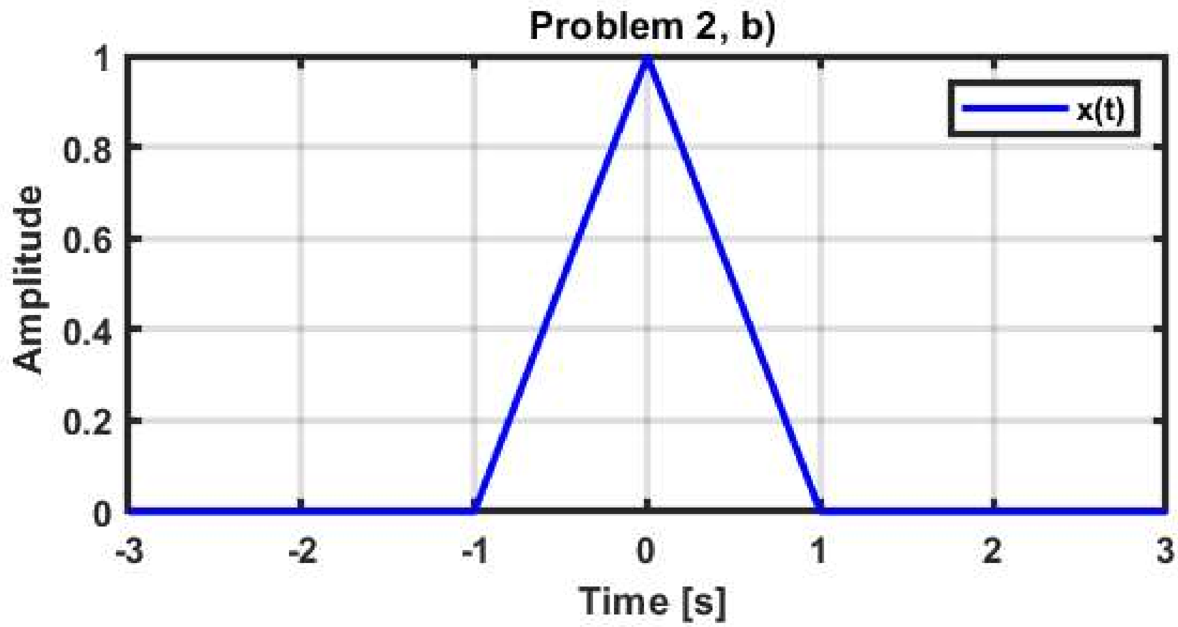
$$\Rightarrow \left[\cancel{-\frac{1}{i2\pi f}} - \frac{1}{(i2\pi f)^2} + \cancel{\frac{e^{i2\pi f}}{i2\pi f}} - \cancel{\frac{e^{i2\pi f}}{i2\pi f}} + \frac{e^{i2\pi f}}{(i2\pi f)^2} \right] + \left[\cancel{\frac{-e^{-i2\pi f}}{i2\pi f}} + \cancel{\frac{e^{-i2\pi f}}{i2\pi f}} + \frac{e^{-i2\pi f}}{(i2\pi f)^2} + \cancel{\frac{1}{i2\pi f}} - \frac{1}{(i2\pi f)^2} \right]$$

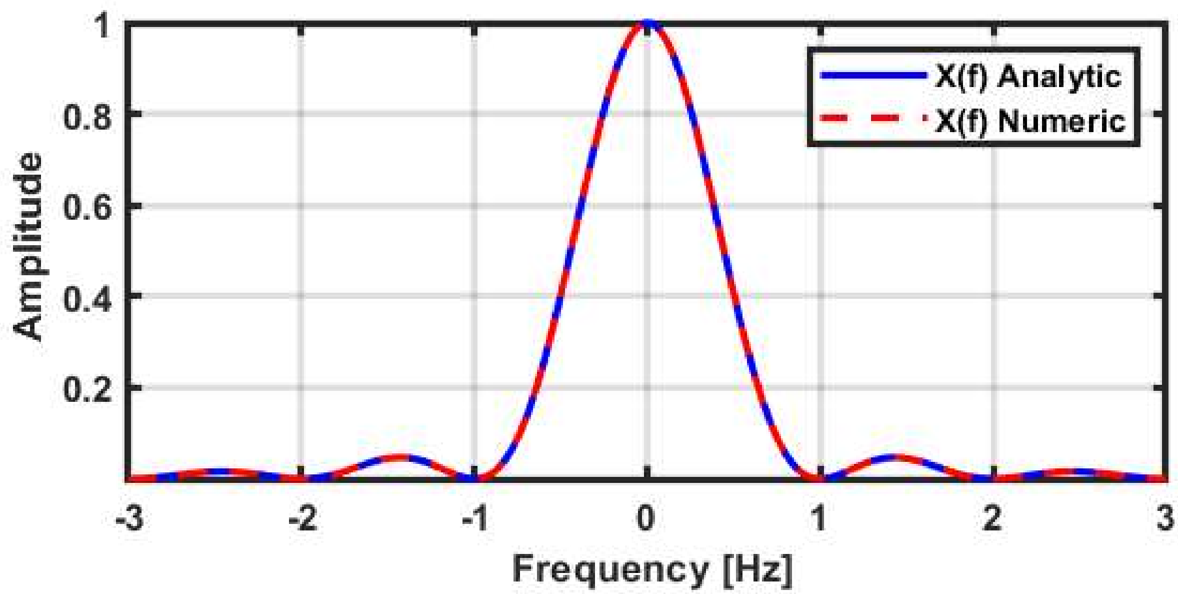
$$\Rightarrow -\frac{1}{(i2\pi f)^2} + \frac{e^{i2\pi f}}{(i2\pi f)^2} + \frac{e^{-i2\pi t}}{(i2\pi f)^2} - \frac{1}{(i2\pi f)^2} = \frac{-2}{(i2\pi f)^2} + \frac{e^{i2\pi f}}{(i2\pi f)^2} + \frac{e^{-i2\pi f}}{(i2\pi f)^2} = \frac{-2 + e^{-i2\pi f} + e^{i2\pi f}}{(i2\pi f)^2}$$

$$\Rightarrow \frac{-2 + [\cos(2\pi f) - i\sin\cancel{(2\pi f)}] + [\cos(2\pi f) + i\sin\cancel{(2\pi f)}]}{(i2\pi f)^2} = \frac{2[\cos(2\pi f) - 1]}{(i2\pi f)^2}$$

$$\Rightarrow \frac{2[\cancel{1} - 2\sin^2(\pi f) \cancel{- 1}]}{(i2\pi f)^2} = \frac{\cancel{-4}\sin^2(\pi f)}{\cancel{(i)^2 (2)^2}(\pi f)^2} = \frac{\sin^2(\pi f)}{(\pi f)^2} = \underbrace{\left(\frac{\sin(\pi f)}{\pi f}\right)}_{\mathrm{sinc}(f)} \cdot \underbrace{\left(\frac{\sin(\pi f)}{\pi f}\right)}_{\mathrm{sinc}(f)}$$

$$\Rightarrow \underline{\mathrm{sinc}(f)} \; //$$






```
syms t f

x_t = @(t) (1-abs(t)).*(abs(t)<1); % defining function x(t)
t=-3:0.01:3; % time vector

% Fourier transfrom analytic
F_xt = @(f) (sinc(f)).^2; % sinc function
f= -3:0.01:3; % frequency vector

% Fourier transfrom numeric
F0_xt = @(t) x_t(t).*exp(-i*2*pi*f*t);
F2_xt = @(f) integral(F0_xt,-inf,inf,'ArrayValued',true);
```

```
fig3 = figure(3);
set(fig3,'Position', [100 100 500 500]);
subplot(211)
plot(t,x_t(t),'-b', 'linewidth', 2), hold on
legend('x(t)', 'g(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
title('\bf Problem 2, b)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');

subplot(212)
plot(f,F_xt(f),'-b', 'linewidth', 2), hold on
plot(f,F2_xt(f),'--r', 'linewidth', 2)
legend('FT Analytic', 'FT Numeric');
grid on;
axis tight
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold')
```

(c)

The Fourier transform of the x(t) function results in a sinc square of f. If we remember the Fourier transform of a rectangular function is a sinc function in the frequency domain. In problem 1 we demonstrated that the convolution of two rectangular function with the same limits results in a triangular function (similar to Problem 2 a). So when performing the fourier transform of a triangular function, which does not have a regular increment, this result in a sinc fucntion power to 2.

Problem 3: Discrete Fourier Transform 1

(a)

X(f) refers to the Fourier transform of the original continuous-time function x(t), while Xs(f) is the Fourier transform of the sample time signal x_s(t). On the other hand, X(k) is the discrete Fourier transform of the sampled signal x_s(t). In other words, X(f) is continuous in the frequency domain and is related to a continuous original time signal; Xs(f) is continuous in the frequency domain but is related to a discrete-sample time signal; X(k) is discrete in the frequency domain and is obtained from X(f) evaluated at f = k/(N*delta), where k is an integer.

(b)

These two graphs explain the aliasing effect. On the graph of the left is set a large sample frequency so that the Nyquist frequency is much higher than the frequencies of the original time signal, meaning that no aliasing is produced. When the Fourier transform is applied to the sampled signal this looks identical to the original frequency signal. On the other hand, the plot on the right shows a time signal sampled with a very low sampling ratio (i.e., low frequency). In that case, the Nyquist frequency is lower than the frequency range of the original signal, meaning that aliasing is generated.

(c)

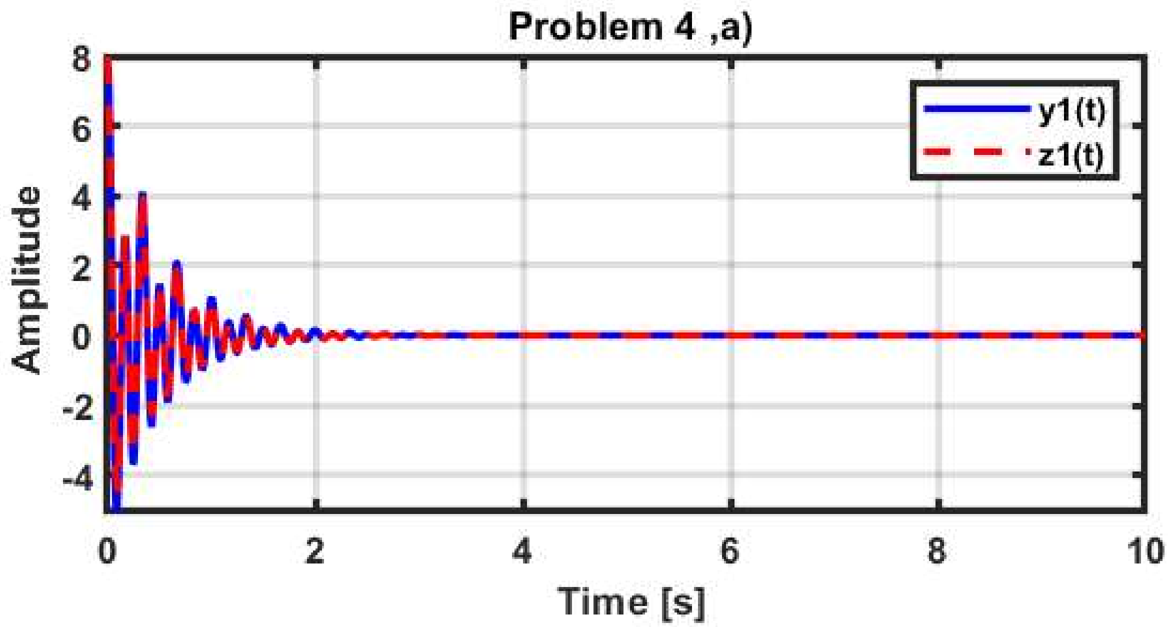
This relationship presents a property of Xs(f). Xs(f + r/delta) = Xs(f) is equivalent to the relationship of the periodic signal x(t) = x(t + nTp). Therefore, we can say that Xs(f+r/delta) = Xs(f) is also periodic although the signal xs(t) is discrete; the period is 1/delta = fs (impulse period) since r is an integer. Xs is the fourier transform of the sample time signal xs(t) becoming into a periodic signal in the frequency domain.

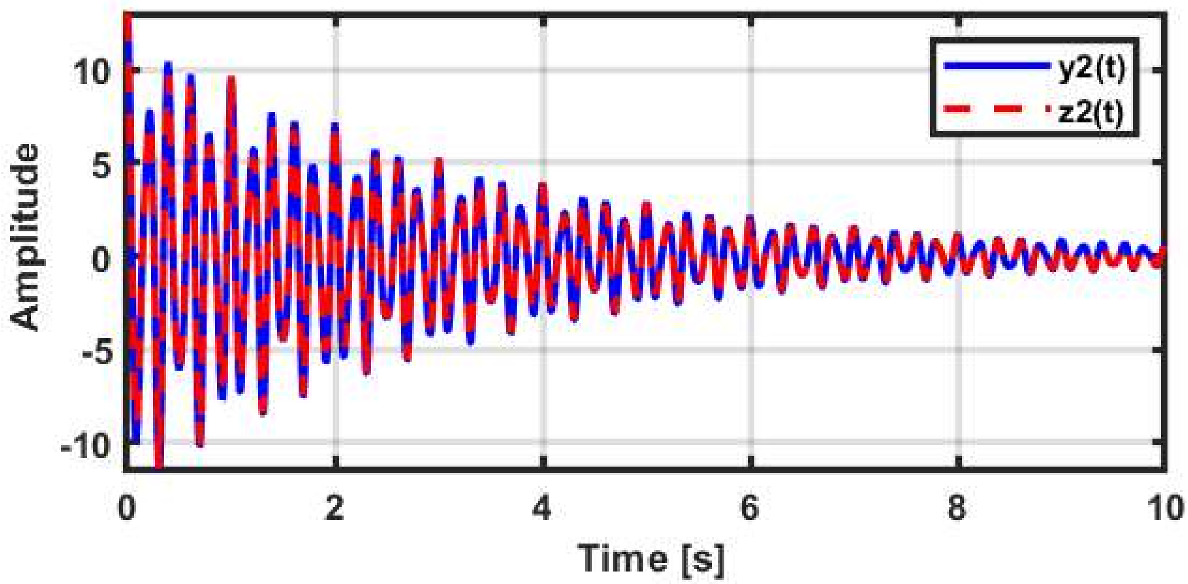
(d)

This relationship indicates how Xs(f) and X(f) are related. Xs(f) the Fourier transform of the sampled signal can be obtained by performing the Fourier transform of the original signal x(t) times the impulse train i(t), meaning that Xs(f) is equal to the convolution of X(f)*I(f) in the frequency domain. Furthermore, this convolution can be represented as a summation where X(f) is shifted every 1/delta = fs (sampling frequency), meaning that the original wave is repeated at every sampling rate. However, when my signal has a frequency component around my Nyquist frequency [1/(2delta)], aliasing is produced. When measuring the signal for frequencies around the Nyquist frequency these frequencies are wrapped around, this is aliasing frequency happening.

Problem 4: Discrete Fourier Transform 2

(a)

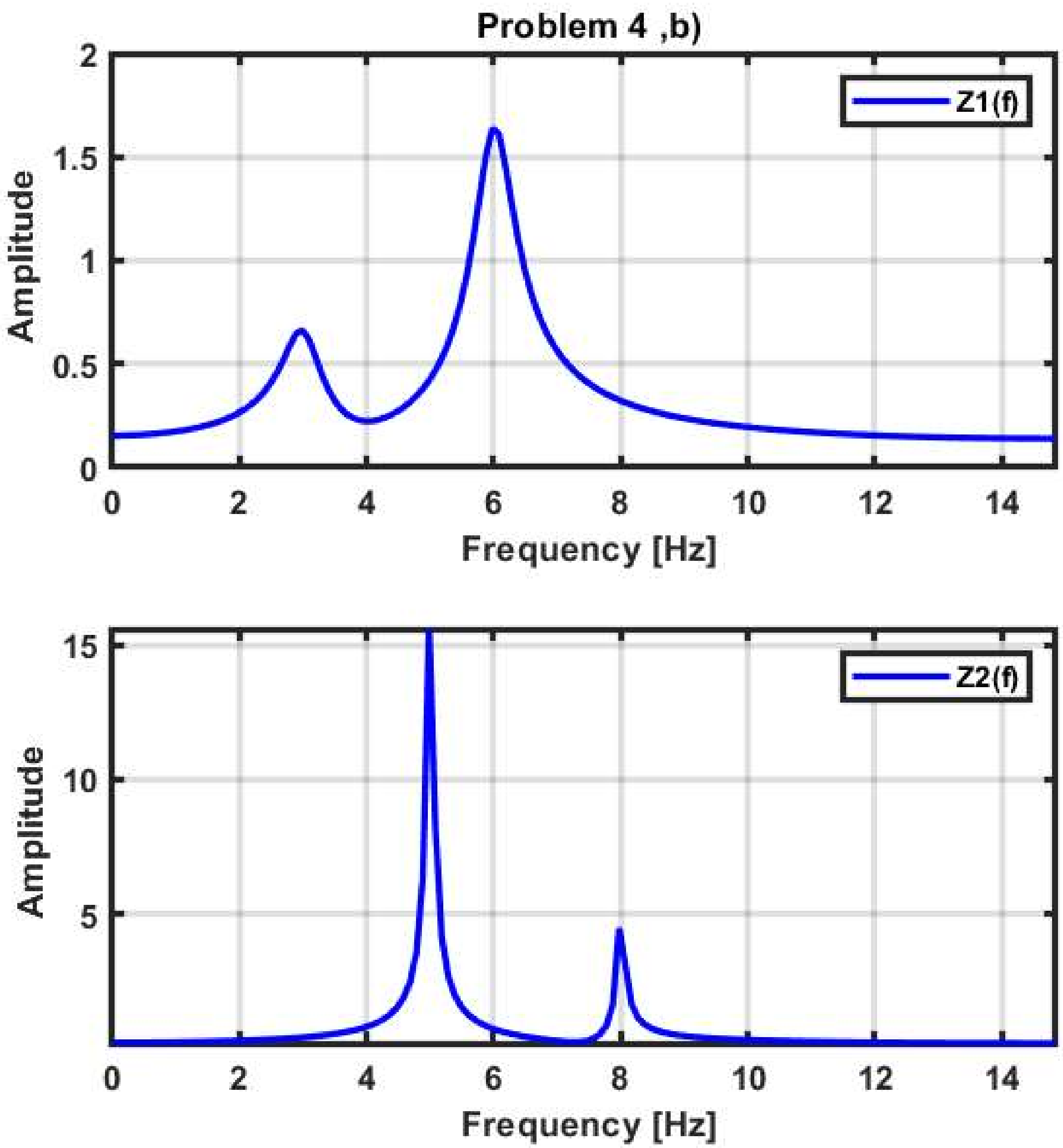





```
syms t
a=2;b=2;c=6;f1=3;f2=6;
y1= @(t) exp(-a*abs(t)).*(b*cos(2*pi*f1*t)+c*cos(2*pi*f2*t)); % defining function y1
a=0.3;b=10;c=3;f1=5;f2=8;
y2= @(t) exp(-a*abs(t)).*(b*cos(2*pi*f1*t)+c*cos(2*pi*f2*t)); % defining function y2

t=0:0.01:10; % time vector
fs = 30; % sampling frequency
Ts = 1/fs; % sampling rate
t0=0:Ts:10; % sampling time vector
```

```
fig4 = figure(4);
set(fig4,'Position', [100 100 500 500]);
subplot(211)
plot(t,y1(t),'-b', 'linewidth', 2), hold on
plot(t0,y1(t0),'--r', 'linewidth', 2)
legend('y1(t)', 'z1(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
title('\bf Problem 4 ,a)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');

subplot(212)
plot(t,y2(t),'-b', 'linewidth', 2), hold on
plot(t0,y2(t0),'--r', 'linewidth', 2)
legend('y2(t)', 'z2(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

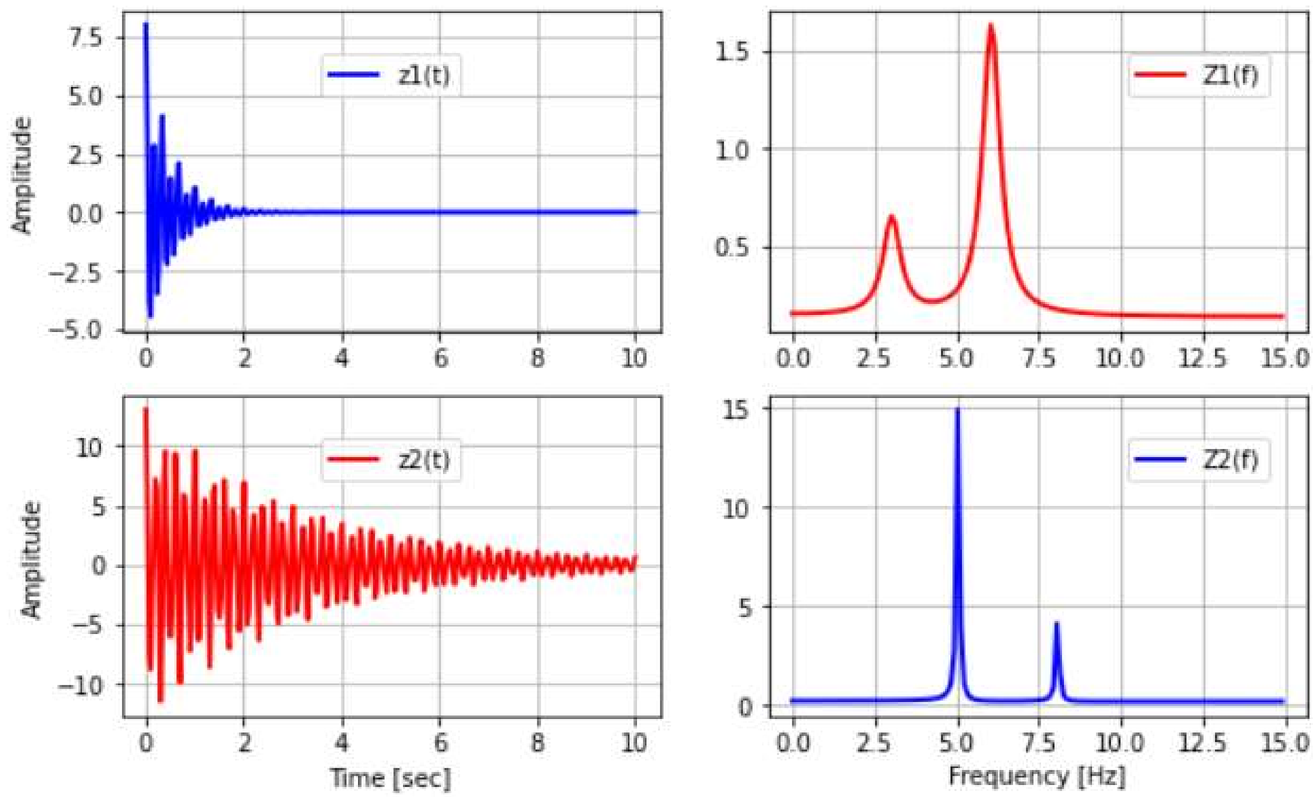
(b)




```
% Discrete Fourier Transform
F_z1 = fft(y1(t0),numel(t0)); % fft for y1 (Z1)
F_z2 = fft(y2(t0),numel(t0)); % fft for y2 (Z2)

f = 1/(numel(t0)*Ts) * (0:numel(t0)/2-1); % frequency vector

fig5 = figure(5);
set(fig5,'Position', [100 100 500 500]);
subplot(211)
plot(f,1/fs*abs(F_z1(1:numel(t0)/2)),'-b', 'linewidth', 2)
legend('Z1(f)'); axis tight;grid on;
```

```
ylabel('\bf Amplitude');
ylim([0 2])
xlabel('\bf Frequency [Hz]')
title('\bf Problem 4 ,b)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');

subplot(212)
plot(f,1/fs*abs(F_z2(1:numel(t0)/2)),'-b', 'linewidth', 2)
legend('Z2(f)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

[Bonus, script done in Python (Jupyter)]



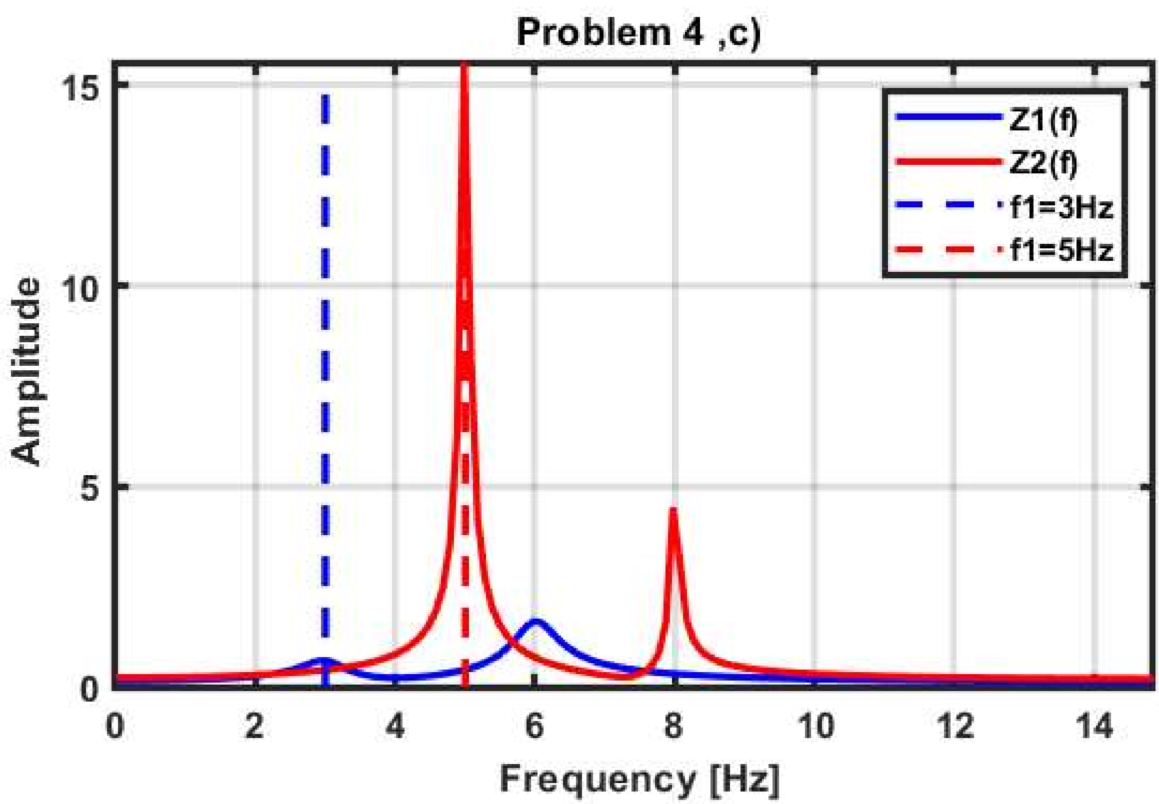

```
#Python script for Assignment 3
#Problem 4: Discrete Fourier Transform 2, b)
import numpy as np
import matplotlib.pyplot as plt


a=2;b=2;c=6;f1=3;f2=6;
```

```
y1 = lambda t: np.exp(-a*abs(t))*(b*np.cos(2*np.pi*f1*t)+c*np.cos(2*np.pi*f2*t)) # def
ininf function d1

a1=0.3;b1=10;c1=3;f11=5;f21=8;
y2 = lambda t: np.exp(-a1*abs(t))*(b1*np.cos(2*np.pi*f11*t)+c1*np.cos(2*np.pi*f21*t))
# defininf function d1

fs = 30 # sampling frequency
Ts = 1/fs # sampling rate
NP = int(10/Ts)
t0 = np.linspace(0,10,NP) # time vector

#Fast Fourier Transform
Fy1 = np.fft.fft(y1(t0),len(t0))/fs
Fy2 = np.fft.fft(y2(t0),len(t0))/fs

f = np.linspace(0,int(len(t0)/2-1),int(len(t0)/2)-1)
f = [f/(len(t0)*Ts) for f in f]

fig1, aw1 = plt.subplots(figsize=(9,5.5),nrows=2, ncols=2)
aw1[0,0].plot(t0,y1(t0),'b-',lw=2,label ='z1(t)')
aw1[0,0].legend(fontsize =10, bbox_to_anchor=(0.5,0.9), loc=9)
aw1[0,0].grid();
aw1[0,0].set_ylabel('Amplitude')

aw1[1,0].plot(t0,y2(t0),'r-',lw=2,label ='z2(t)')
aw1[1,0].legend(fontsize =10, bbox_to_anchor=(0.5,0.9), loc=9)
aw1[1,0].grid();
aw1[1,0].set_ylabel('Amplitude')
aw1[1,0].set_xlabel('Time [sec]')

aw1[0,1].plot(f,Fy1[0:int(len(t0)/2)-1].real,'r-',lw=2,label ='Z1(f)')
aw1[0,1].legend(fontsize =10, bbox_to_anchor=(0.8,0.9), loc=9)
aw1[0,1].grid();

aw1[1,1].plot(f,Fy2[0:int(len(t0)/2)-1].real,'b-',lw=2,label ='Z2(f)')
aw1[1,1].legend(fontsize =10, bbox_to_anchor=(0.8,0.9), loc=9)
aw1[1,1].grid();
```

```
aw1[1,1].set_xlabel('Frequency [Hz]')

fig.subplots_adjust(hspace =0.75,wspace =0.5, right =0.98, left=0.15, bottom = 0.1, to
p = 0.89)
plt.show()
```

(c)

The frequency curve of Z2(f) is narrower than that Z1(f), we can see this pattern when observing f1. For instance, Z1(f1=3) is wider than that Z2(f1 =5). This effect is related to the inverse spreading property.

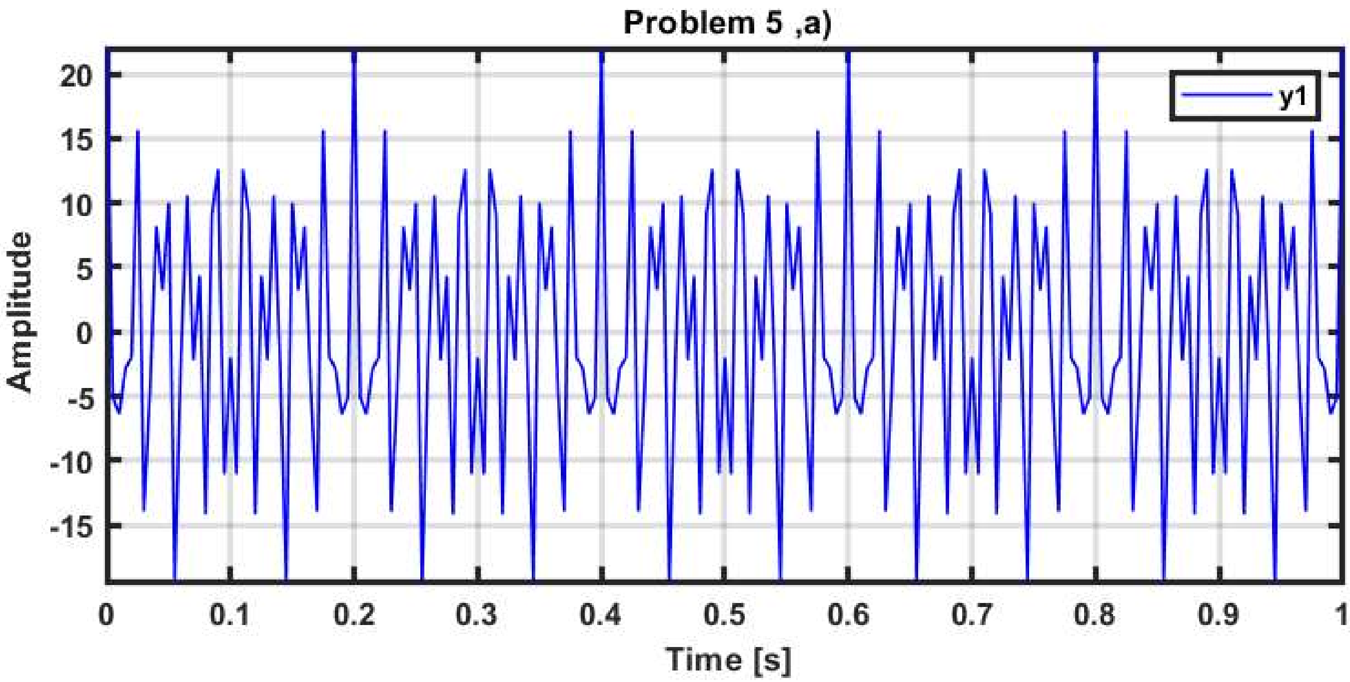



```
fig6 = figure(6);
set(fig6,'Position', [100 100 500 300]);
plot(f,1/fs*abs(F_z1(1:numel(t0)/2)),'-b', 'linewidth', 2), hold on
plot(f,1/fs*abs(F_z2(1:numel(t0)/2)),'-r', 'linewidth', 2)
plot([3 3],[0 15],'--b', 'linewidth', 2)
plot([5 5],[0 15],'--r', 'linewidth', 2)

legend('Z1(f)','Z2(f)','f1=3Hz','f1=5Hz'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
title('\bf Problem 4 ,c)')
```

```
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

Problem 5: Discrete Fourier Transform 3

(a)

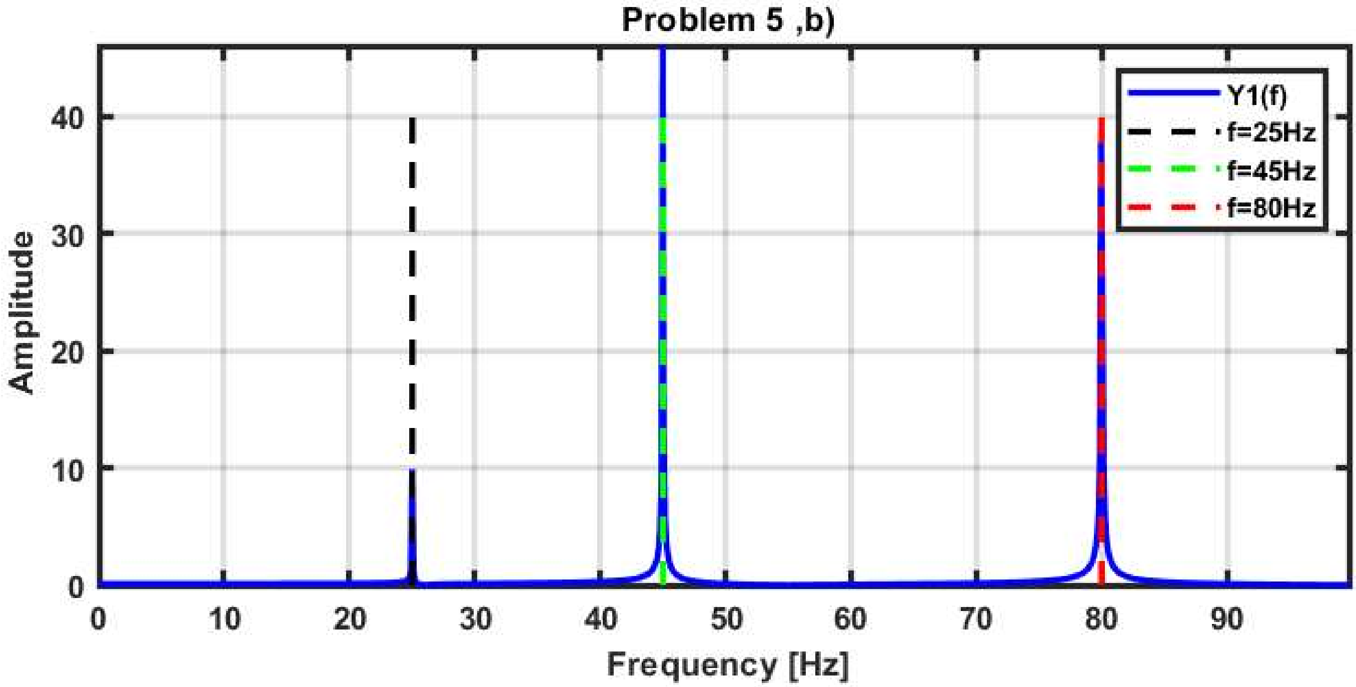



```
syms t

A1=2;A2=10.0;A3=10.0; % define parameters for function
y = @(t) A1*cos(2*pi*25*t) + A2*cos(2*pi*45*t) + A3*cos(2*pi*80*t); % define function

fs = 200; % sampling frequency
Ts =  1/fs; % sampling rate or period
t0 = 0:Ts:10 % time vector

fig7 = figure(7);
set(fig7,'Position', [100 100 700 300]);
plot(t0,y(t0),'-b', 'linewidth', 1)
legend('y1'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
xlim([0 1])
title('\bf Problem 5 ,a)')
```

```
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

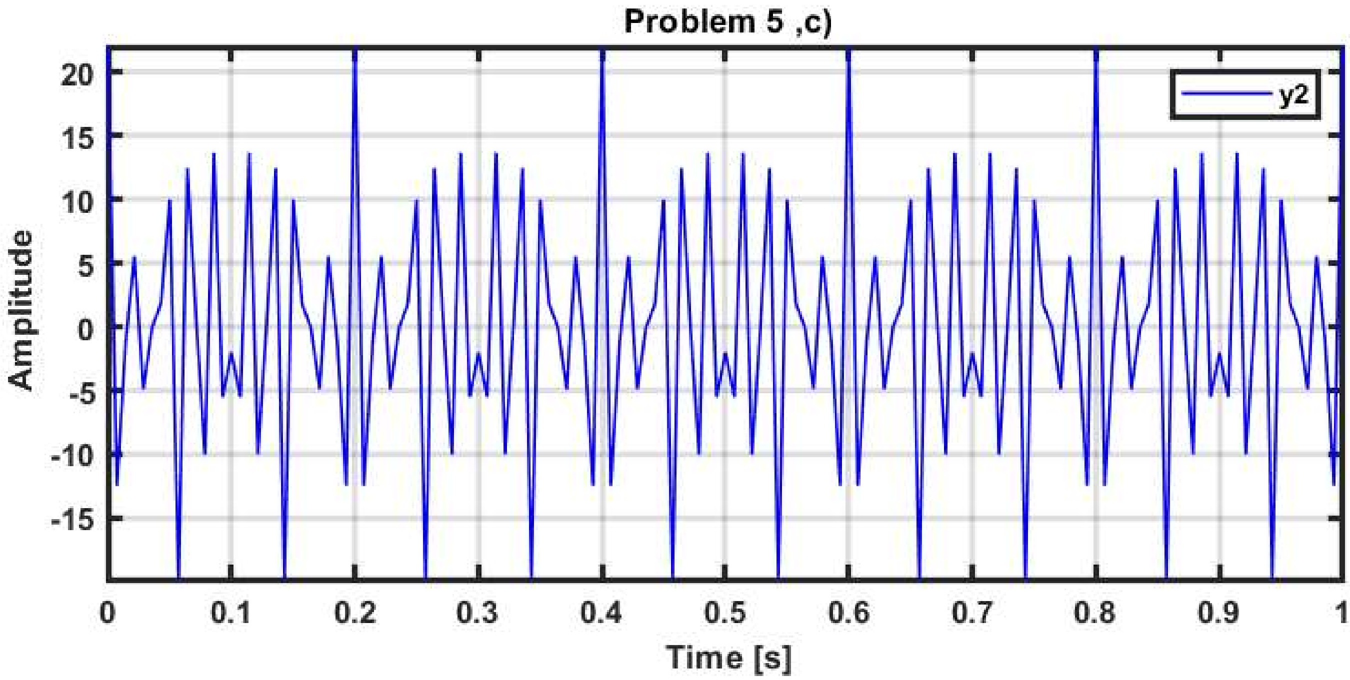
(b)

Yes, we are able to capture all three frequencies with a sampling frequency of 200 Hz since the Nyquist frequency (100 Hz) is much higher than the three frequencies (25, 45, and 80 Hz).




```
% Discrete Fourier Transform
F_y1 = fft(y(t0),numel(t0)); % fft for y1
f = 1/(numel(t0)*Ts) * (0:numel(t0)/2-1); % frequency vector

fig8 = figure(8);
set(fig8,'Position', [100 100 700 300]);
plot(f,1/fs*abs(F_y1(1:numel(t0)/2)),'-b', 'linewidth', 2), hold on
plot([25 25],[0 40],'--k', 'linewidth', 2)
plot([45 45],[0 40],'--g', 'linewidth', 2)
plot([80 80],[0 40],'--r', 'linewidth', 2)

legend('Y1(f)','f=25Hz','f=45Hz','f=80Hz'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
title('\bf Problem 5 ,b)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

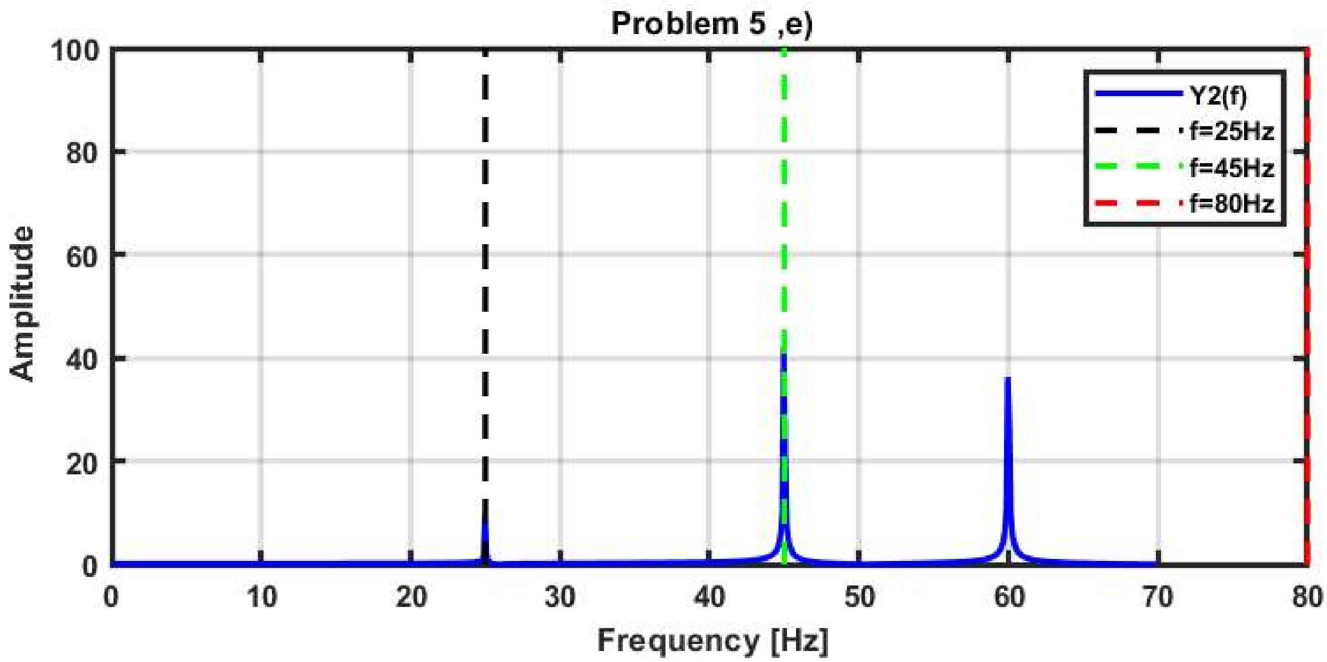
(c)




```
fs = 140; % sampling frequency
Ts =  1/fs; % samplig rate
t0 = 0:Ts:10; % time vector

fig9 = figure(9);
set(fig9,'Position', [100 100 700 300]);
plot(t0,y(t0),'-b', 'linewidth', 1)
legend('y2'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
xlim([0 1])
title('\bf Problem 5 ,c)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

(d)

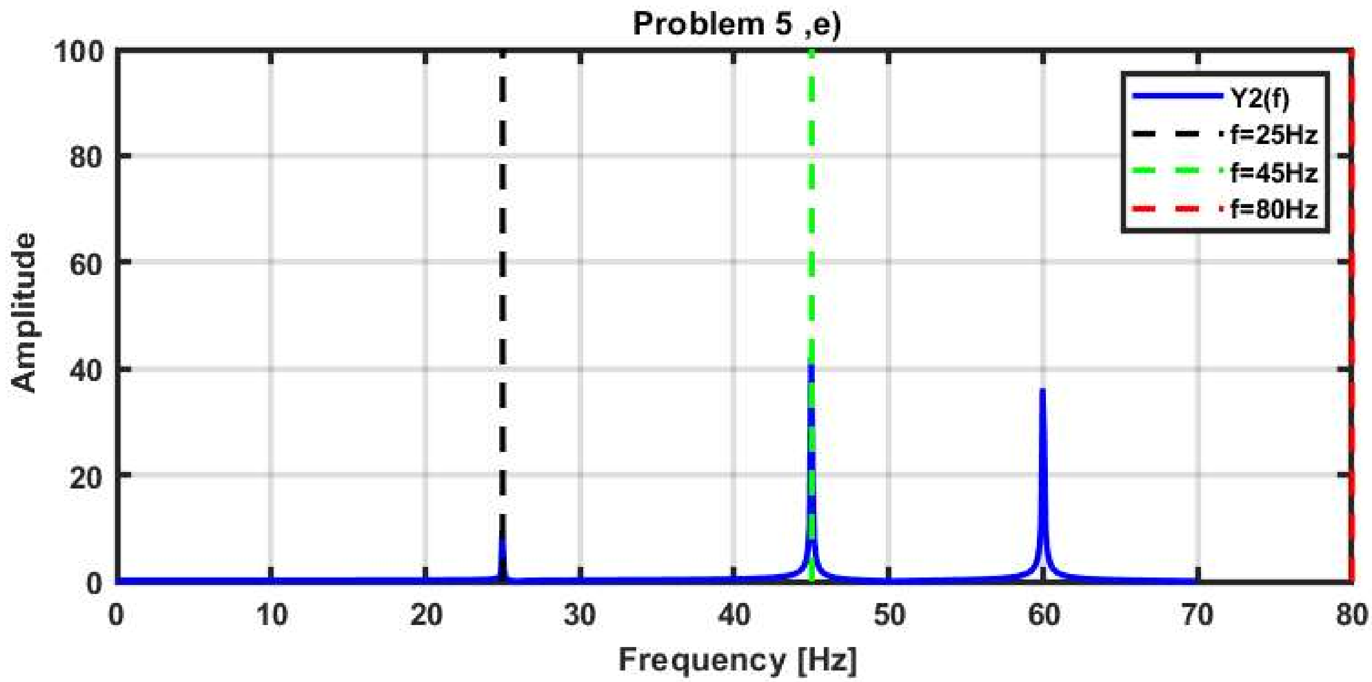
With a sampling frequency of 140 Hz we are not able to capture all three frequencies, only 25 and 45 Hz can be measured since these are lower than the Nyquist frequency (70Hz). Additionally, aliasing is produced, generating an alias frequency of 60 Hz.




```
% Discrete Fourier Transform
F_y2 = fft(y(t0),numel(t0)); % fft for y1
f = 1/(numel(t0)*Ts) * (0:numel(t0)/2-1); % frequency vector

fig10 = figure(10);
set(fig10,'Position', [100 100 700 300]);
plot(f,1/fs*abs(F_y2(1:numel(t0)/2)),'-b', 'linewidth', 2), hold on
plot([25 25],[0 100],'--k', 'linewidth', 2)
plot([45 45],[0 100],'--g', 'linewidth', 2)
plot([80 80],[0 100],'--r', 'linewidth', 2)
xlim([0 100])
legend('Y2(f)','f=25Hz','f=45Hz','f=80Hz'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
title('\bf Problem 5 ,e)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

(e)

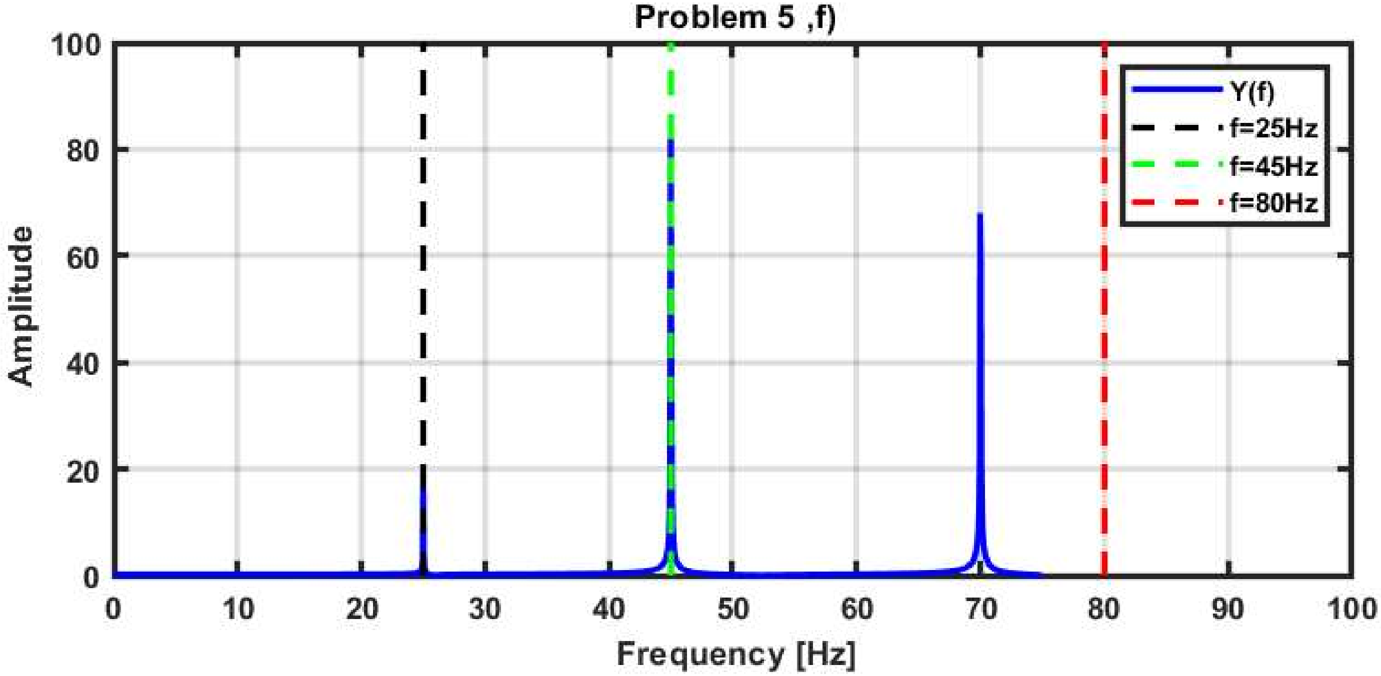
Again, I am not able to measure all three frequencies only 25 and 45 Hz since the sampling frequency has not changed (140 Hz). Therefore, even though the time signal has a longer duration the Nyquist frequency remains the same, and thereby the frequency of 80 Hz cannot be captured.




```
fs = 140; % sampling frequency
Ts =  1/fs; % rate
t0 = 0:Ts:100; % time vector
% Discrete Fourier Transform
F_y3 = fft(y(t0),numel(t0)); % fft for y
f = 1/(numel(t0)*Ts) * (0:numel(t0)/2-1); % frequency vector

fig11 = figure(11);
set(fig11,'Position', [100 100 700 300]);
plot(f,1/fs*abs(F_y3(1:numel(t0)/2)),'-b', 'linewidth', 2), hold on
plot([25 25],[0 300],'--k', 'linewidth', 2)
plot([45 45],[0 300],'--g', 'linewidth', 2)
plot([80 80],[0 300],'--r', 'linewidth', 2)

legend('Y(f)','f=25Hz','f=45Hz','f=80Hz'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
title('\bf Problem 5 ,e)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
xlim([0 100])
```

(f)

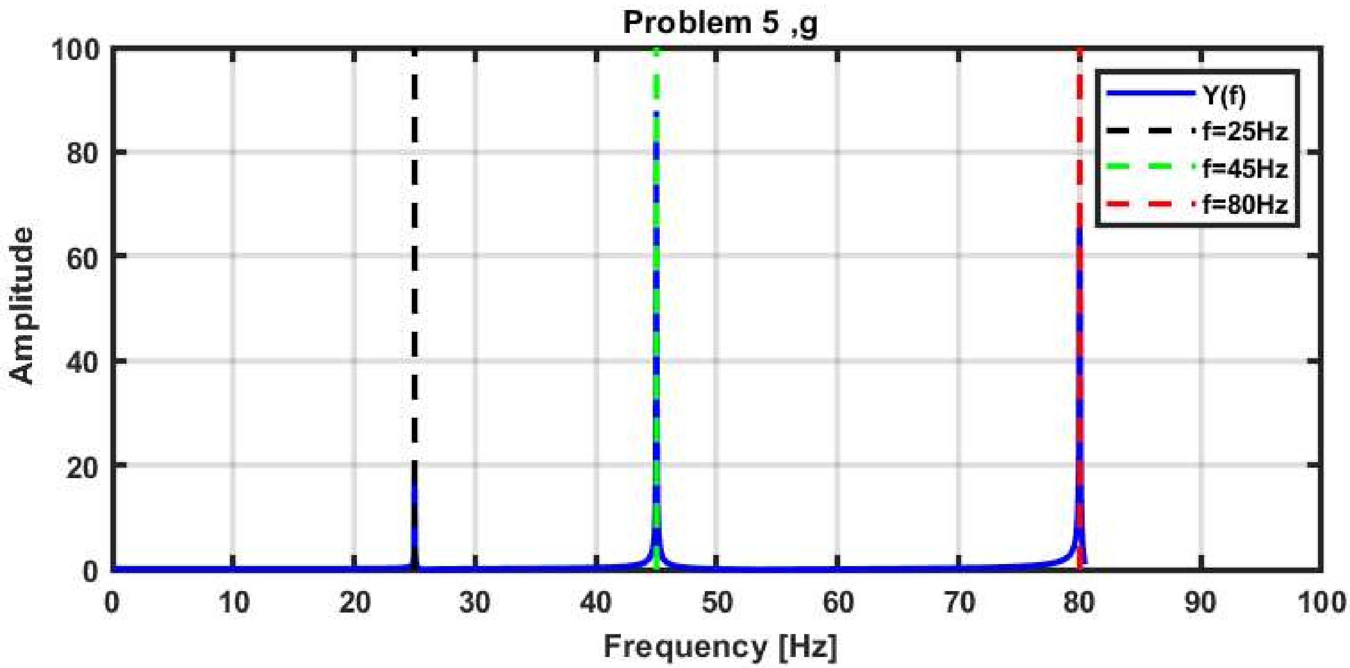
The sampling frequency is a slighter bit higher than before but the Nyquist frequency is 75 Hz which is still lower than 80Hz. Therefore, even though we sample for 20 seconds we can only capture 25 and 45 Hz but not 80 Hz. Additionally, aliasing is happening since a new frequency is captured 70 Hz (alias frequency).




```
fs = 150; % sampling frequency
Ts =  1/fs; % rate
t0 = 0:Ts:20; % time vector
% Discrete Fourier Transform
F_y4 = fft(y(t0),numel(t0)); % fft for y
f = 1/(numel(t0)*Ts) * (0:numel(t0)/2-1); % frequency vector

fig12 = figure(12);
set(fig12,'Position', [100 100 700 300]);
plot(f,1/fs*abs(F_y4(1:numel(t0)/2)),'-b', 'linewidth', 2), hold on
plot([25 25],[0 100],'--k', 'linewidth', 2)
plot([45 45],[0 100],'--g', 'linewidth', 2)
plot([80 80],[0 100],'--r', 'linewidth', 2)

legend('Y(f)','f=25Hz','f=45Hz','f=80Hz'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
title('\bf Problem 5 ,f)')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
xlim([0 100])
```

(g)

Now we are able to measure all three frequencies since the Nyquist frequency is 80.5Hz which si higher than 80Hz.

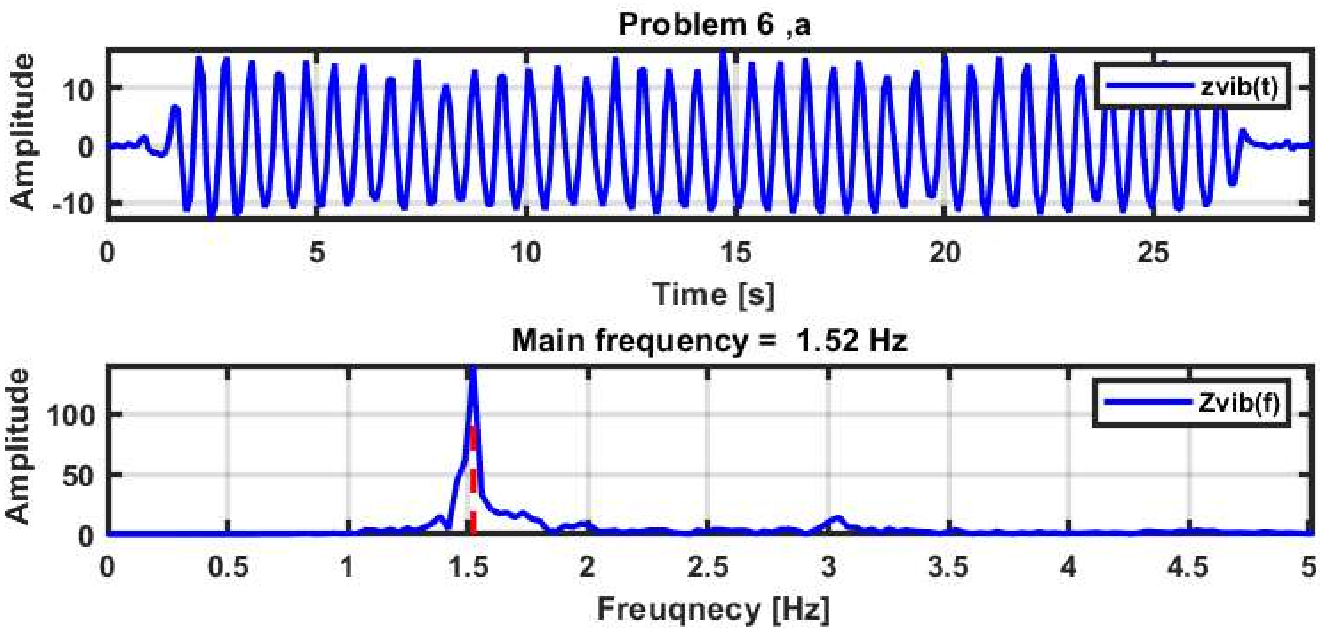



```
fs = 161; % sampling frequency
Ts =  1/fs; % rate
t0 = 0:Ts:20; % time vector
% Discrete Fourier Transform
F_y5 = fft(y(t0),numel(t0)); % fft for y
f = 1/(numel(t0)*Ts) * (0:numel(t0)/2-1); % frequency vector

fig13 = figure(13);
set(fig13,'Position', [100 100 700 300]);
plot(f,1/fs*abs(F_y5(1:numel(t0)/2)),'-b', 'linewidth', 2), hold on
plot([25 25],[0 100],'--k', 'linewidth', 2)
plot([45 45],[0 100],'--g', 'linewidth', 2)
plot([80 80],[0 100],'--r', 'linewidth', 2)
legend('Y(f)','f=25Hz','f=45Hz','f=80Hz'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Frequency [Hz]')
title('\bf Problem 5 ,g')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
xlim([0 100])
```

Problem 6: Frequency Analysis

(a)

The main frequency of this wave is 1.52Hz.

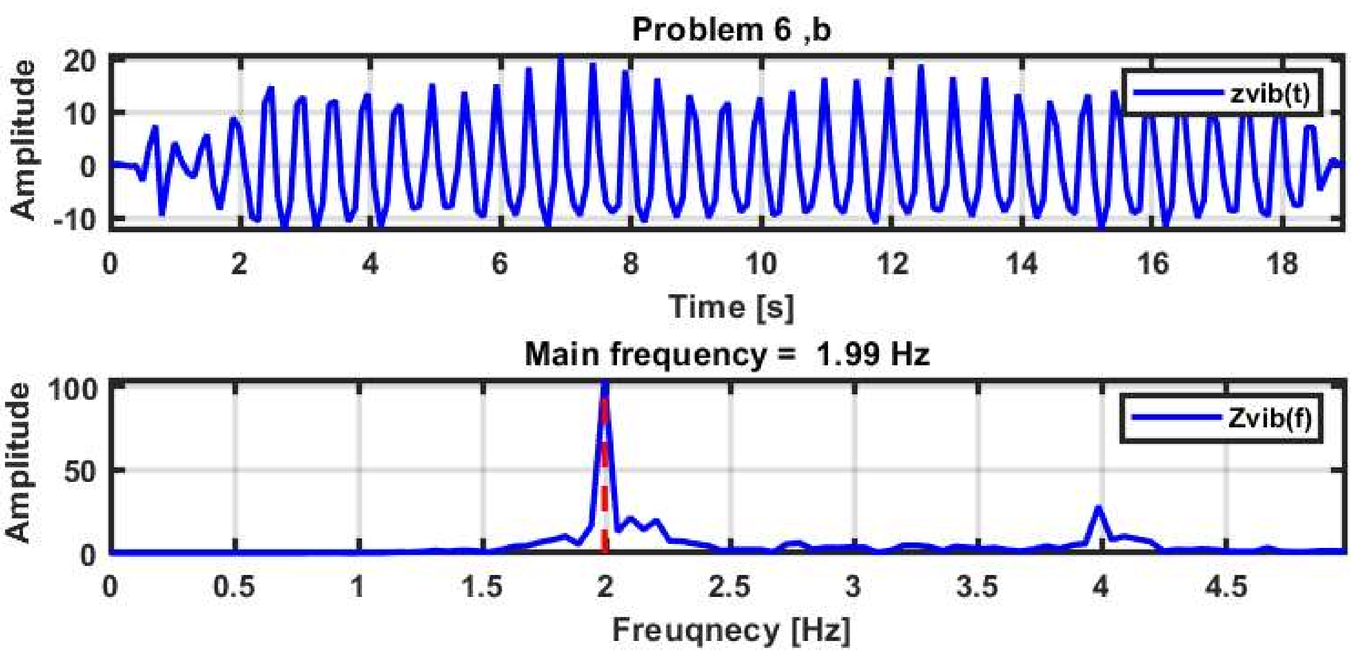



```
load('data1.mat'); % load file
fs = 10.1355; % samping frequency
Ts = 1/fs ; % rate or dt
t0 = [0:Ts:numel(zvib)*Ts-Ts] % assuming a dt of 0.05

F_zvib = fft(zvib,numel(zvib)); % fft for zvib
f = 1/(numel(F_zvib)*Ts) * (0:numel(F_zvib)/2-1); % frequency vector
fvector = 1/fs*abs(F_zvib(1:numel(zvib)/2)); % plot positive values
ind_max = find(fvector==max(fvector)); % find index of max frequency
fmax= f(ind_max);

fig14 = figure(14);
set(fig14,'Position', [100 100 700 300]);
subplot(211)
plot(t0 ,zvib,'-b', 'linewidth', 2)
legend('zvib(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
title('\bf Problem 6 ,a')
```

```
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');

subplot(212)
plot(f,fvector,'-b', 'linewidth', 2), hold on
plot([fmax fmax],[0 100],'--r', 'linewidth', 2)
legend('Zvib(f)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Freuqnecy [Hz]')
title(sprintf('Main frequency =  %.2f Hz',fmax))
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```

(b)

The main frequency of this wave is 1.99Hz.




```
load('data2.mat'); % load file
fs = 10.1192; % samping frequency
Ts = 1/fs ; % rate or dt
t0 = [0:Ts:numel(zvib)*Ts-Ts] % assuming a dt of 0.05

F_zvib = fft(zvib,numel(zvib)); % fft for zvib
f = 1/(numel(F_zvib)*Ts) * (0:numel(F_zvib)/2-1); % frequency vector
fvector = 1/fs*abs(F_zvib(1:numel(zvib)/2)); % plot positive values
ind_max = find(fvector==max(fvector));  % find index of max frequency
fmax= f(ind_max);
```

```
fig15 = figure(15);
set(fig15,'Position', [100 100 700 300]);
subplot(211)
plot(t0 ,zvib,'-b', 'linewidth', 2)
legend('zvib(t)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Time [s]')
title('\bf Problem 6 ,b')
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');

subplot(212)
plot(f,fvector,'-b', 'linewidth', 2), hold on
plot([fmax fmax],[0 100],'--r', 'linewidth', 2)
legend('Zvib(f)'); axis tight;grid on;
ylabel('\bf Amplitude');
xlabel('\bf Freuqnecy [Hz]')
title(sprintf('Main frequency =  %.2f Hz',fmax))
set(gca,'fontsize',10,'linewidth',2,'fontweight','bold');
```