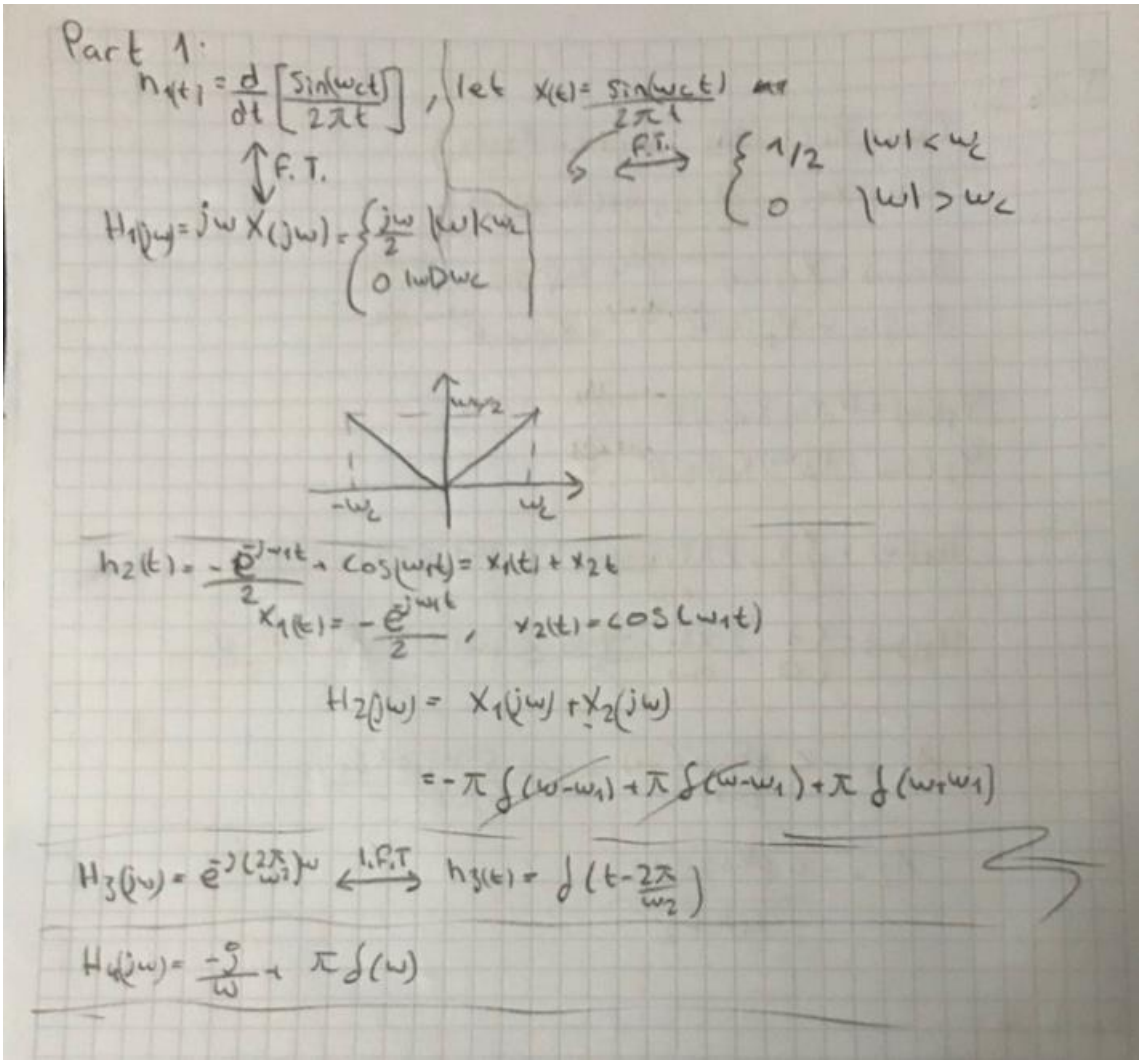
Muhammed Enes İnanç

22001799

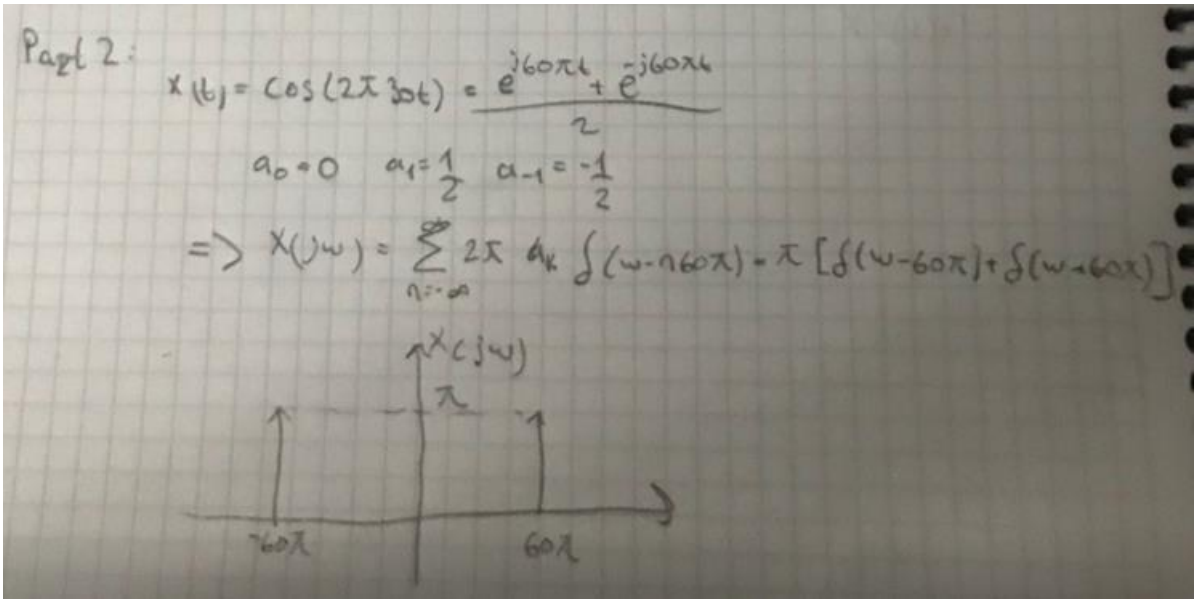
EEE 321 Lab 5 Task

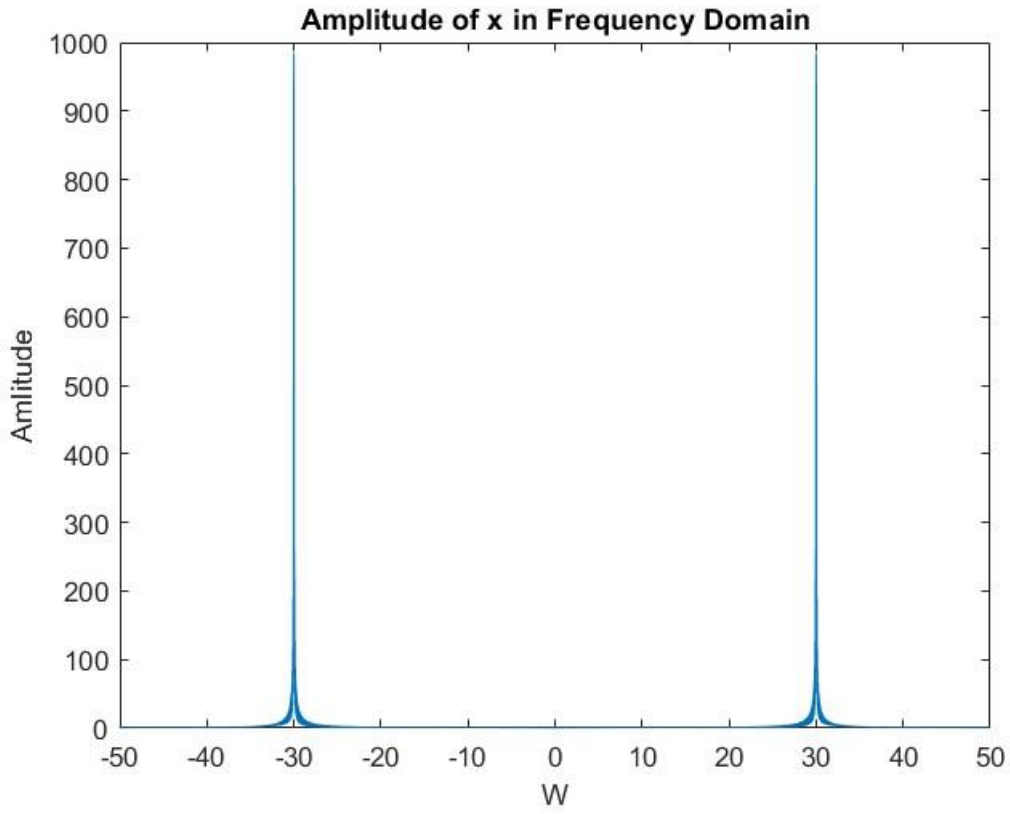
Part 1:

Creating a Fourier transform function from scratch, without relying on built-in MATLAB functions like fft or fftshift, is a testament to your dedication and mastery of signal processing principles. It showcases your ability to understand the mathematical foundations behind these operations and implement them in code effectively.



Part 2:


Part 2:
$x(t) = \cos(2\pi 30t) = \frac{e^{j60\pi t} + e^{-j60\pi t}}{2}$
$a_0 = 0 \quad a_1 = \frac{1}{2} \quad a_{-1} = \frac{1}{2}$
$\Rightarrow X(j\omega) = \sum_{n=-\infty}^{\infty} 2\pi a_k \delta(\omega - n60\pi) = \pi[\delta(\omega-60\pi)+\delta(\omega+60\pi)]$
$X(j\omega)$
π
π
-60π
60π


Amplitude of x in Frequency Domain
Amlitude
W
1000
900
800
700
600
500
400
300
200
100
0
-50
-40
-30
-20
-10
0
10
20
30
40
50

Part 3:

Part 3:

$$r_1(t) = x_1\left(t - \frac{2d_1}{c}\right) + x_2\left(t - \frac{d_1+d_2}{c}\right)$$

$$r_2(t) = x_2\left(t - \frac{2d_2}{c}\right) + x_1\left(t - \frac{d_1+d_2}{c}\right)$$

$$R_1(j\omega) = X_1(j\omega)e^{-j\omega\frac{2d_1}{c}} + X_2(j\omega)e^{-j\omega\frac{d_1+d_2}{c}}$$

$$R_2(j\omega) = X_1(j\omega)e^{-j\omega\frac{d_1+d_2}{c}} + X_2(j\omega)e^{-j\omega\frac{2d_2}{c}}$$

$$Y_1(j\omega) = H_1(j\omega)X_1(j\omega)e^{-j\omega\frac{2d_1}{c}}$$

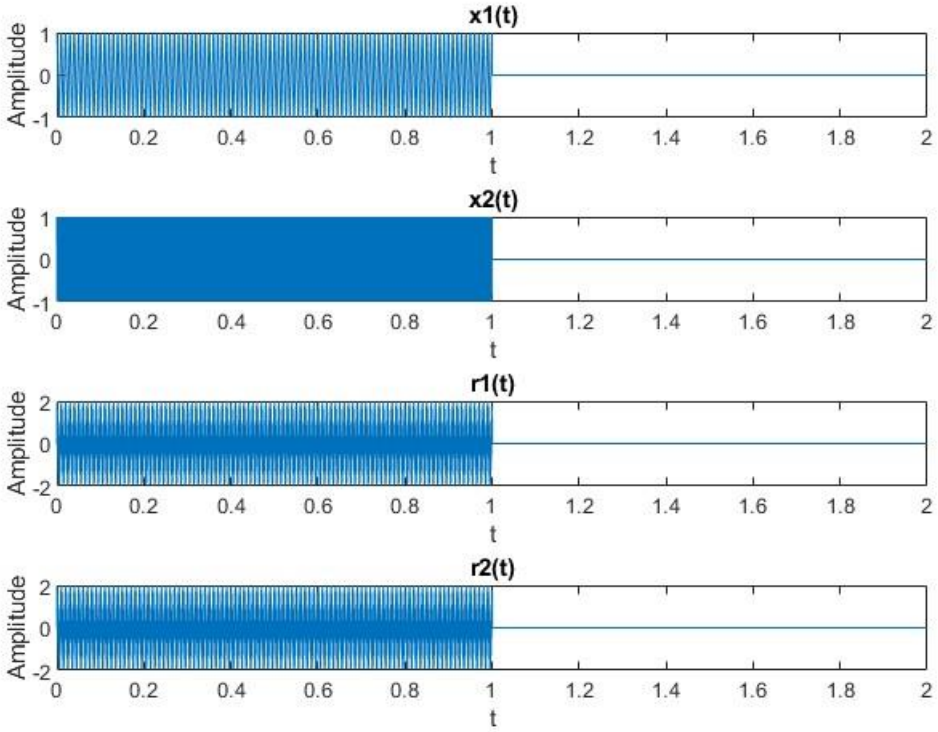
$$Y_2(j\omega) = H_2(j\omega)X_2(j\omega)e^{-j\omega\frac{2d_2}{c}}$$

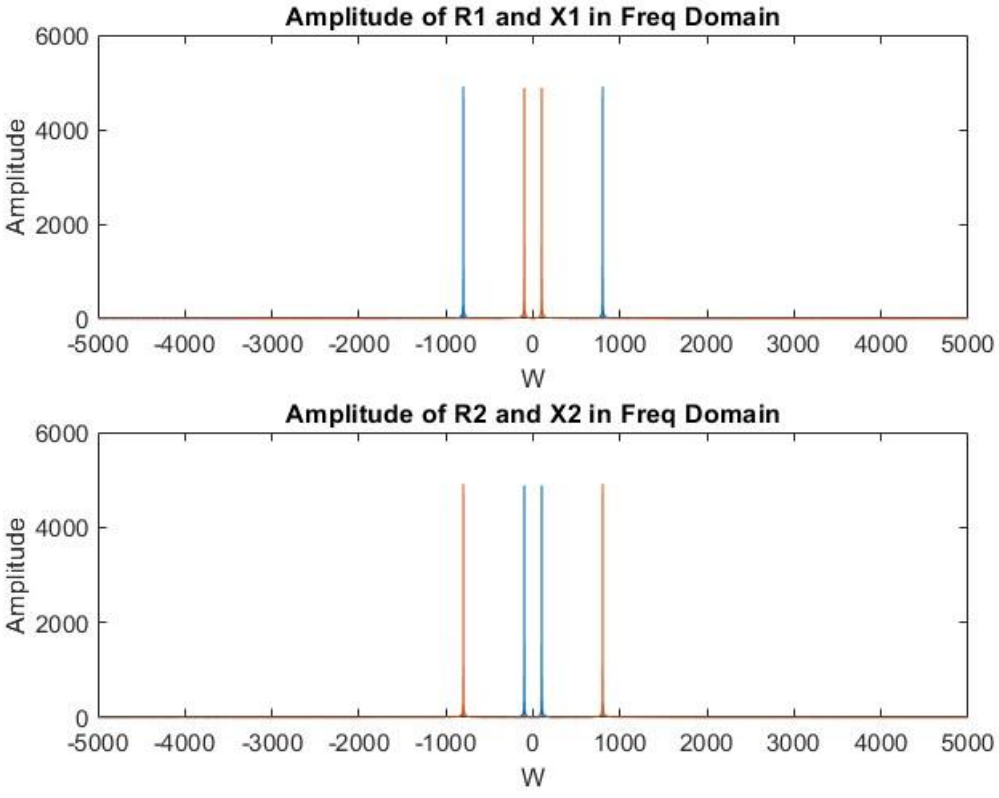
$$H_1(j\omega) = \begin{cases} 1, & 2\pi f_1 - \frac{\omega pass}{2} \le |\omega| \le 2\pi f_1 + \frac{\omega pass}{2} \\ 0 & o.w \end{cases}$$

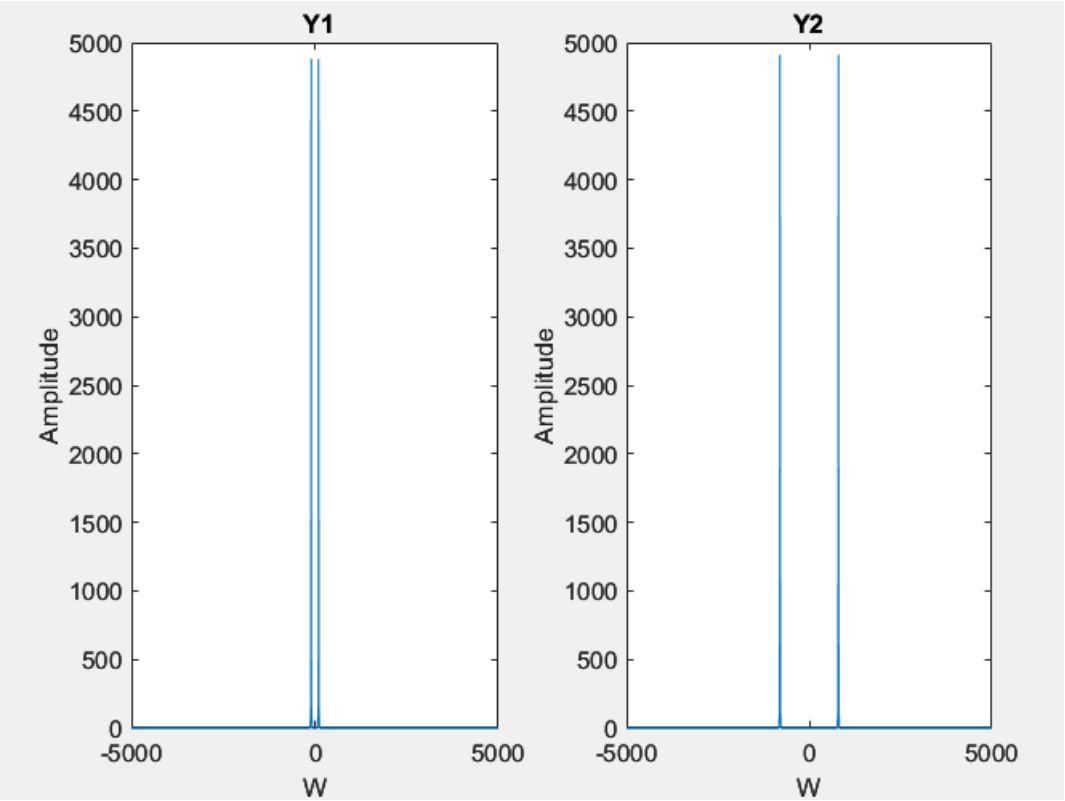
$$H_2(j\omega) = \begin{cases} 1 & 2\pi f_2 - \frac{\omega pass}{2} \le |\omega| \le 2\pi f_2 + \frac{\omega pass}{2} \\ 0 & o.w \end{cases}$$

$$d_1 = -\frac{c\ \text{phase}(Y_1)}{2\omega_1} = -\frac{c\ \text{phase}(Y_1)}{4\pi f_1}$$

$$d_2 = -\frac{c\ \text{phase}(Y_2)}{4\pi f_2}$$

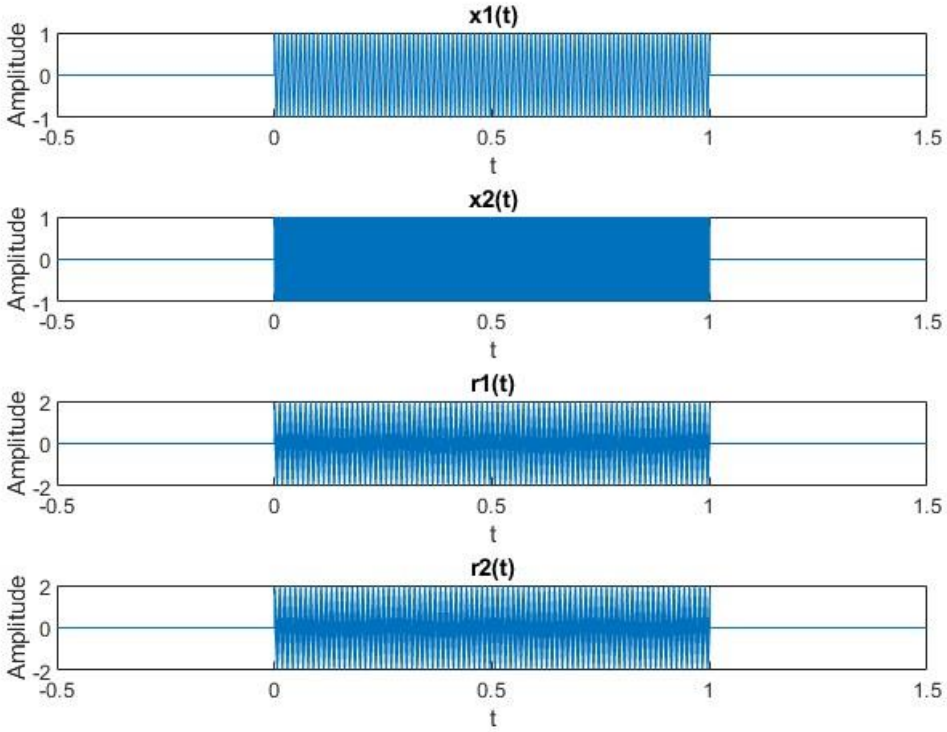

x1(t)
x2(t)
r1(t)
r2(t)
Amplitude
t

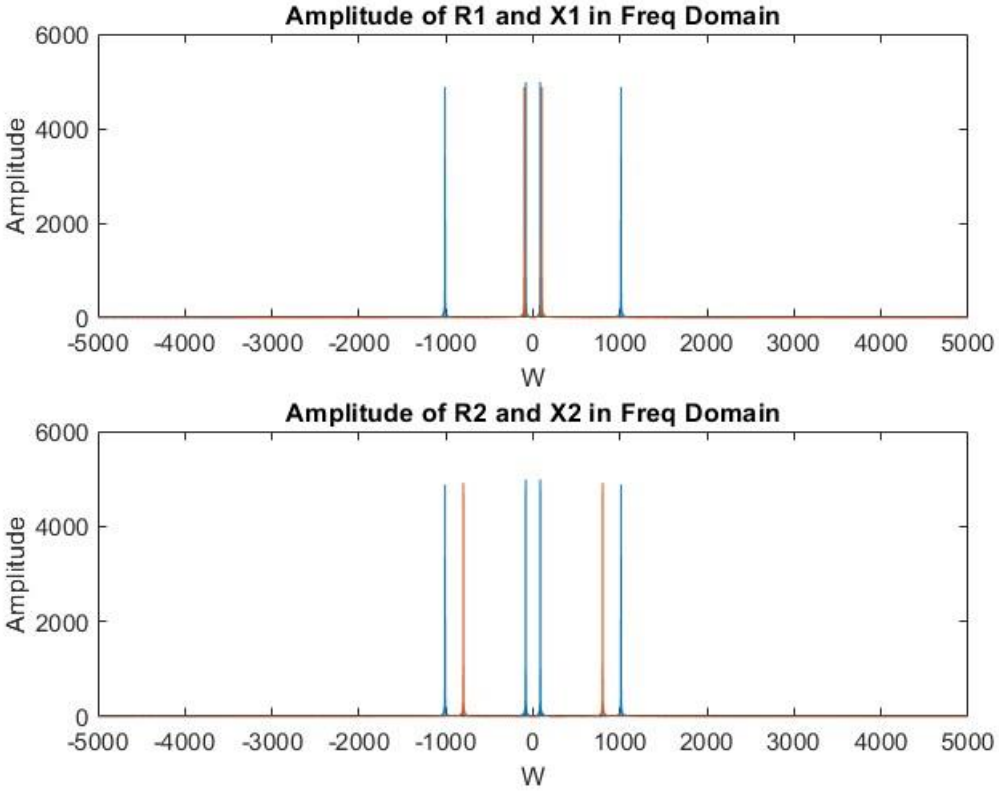

Amplitude of R1 and X1 in Freq Domain
Amplitude of R2 and X2 in Freq Domain
Amplitude
W




```
d1: 0.0751 meters
d2: -0.1031 meters
```

Part 4:


x1(t)
Amplitude
t
x2(t)
Amplitude
t
r1(t)
Amplitude
t
r2(t)
Amplitude
t


Amplitude of R1 and X1 in Freq Domain
Amplitude
W
Amplitude of R2 and X2 in Freq Domain
Amplitude
W

Appendices:

Codes for FT Algorithm:

```
function [frequency_array] = FourierTransform(x, t, Ts)

    frequency_array = zeros(1, 2*length(x)); % Let's create an empty vector on
memory
    Fs = 1 / Ts;
    vector = linspace(-Fs/2, Fs/2, 2*length(x));
    lenofvector = length(vector); % Let's find the length of the vector to use
somewhere such as loops
    for i = 1:lenofvector
        arafunc = 0;
        for m = 1:length(x)
            arafunc = arafunc + exp(-1j * 2 *t(m) * pi * vector(i) ) * x(m) ;
        end
        frequency_array(i) = arafunc;
    end
end
```

Part 2:

```
Ts = 0.01; % It was given, not optional

t = -10:Ts:10; %Let's find our time vector

x = cos(2*pi*30*t); % requested function
fresponseforx = FourierTransform(x, t, Ts);
lefforrespx = length(fresponseforx);
fresponseforxabs = abs(fresponseforx);


Fs = 1 / Ts;
xlabelvarinfreq = linspace(-Fs/2, Fs/2, lefforrespx);

plot(xlabelvarinfreq, fresponseforxabs);
title('Amplitude of x in Frequency Domain');

ylabel('Amlitude');

xlabel('W');
```

Part 3:

```
Ts = 0.0001; % sampling rate it is really small
Fs = 1 / Ts;
t = 0:Ts:2; %time vector accordingly ts
T = 1;
mesbir = 0.05; %it is given
mesiki = 0.1;
mestot = mesbir + mesiki
ilkfreq = 100;
ikifreq = 800;
velci = 343;
```

```
c = 343;

signalfirstx = cos(2 * pi * ilkfreq * t);
signalfirstx(t > T)=0;
signalfirstx(t < 0)=0;


signalsecondx = cos(2 * pi * ikifreq * t);
signalsecondx(t < 0)=0;
signalsecondx(t > T)=0;

distbirforx = cos(2 * pi * ilkfreq*(t-2*mesbir/velci));
distbirforx(t > (2*mesbir/velci)+T)=0;
distbirforx(t < (2*mesbir/velci))=0;



distikiforx = cos(2 * pi * ikifreq *(t-2*mesiki/velci));
distikiforx(t > (2*mesiki/velci)+T)=0;
distikiforx(t < (2*mesiki/velci))=0;




distikitobirx = cos(2 * pi * ikifreq *(t-(mesbir+mesiki)/velci));
distikitobirx(t > (mestot)/velci+T)=0;
distikitobirx(t < (mestot)/velci)=0;



distbirtoimx = cos(2 * pi * ilkfreq *(t-(mesbir+mesiki)/velci));
distbirtoimx(t > (mestot)/velci+T)=0;
distbirtoimx(t < (mestot)/velci)=0;


cevrbir = distbirforx + distikitobirx;
cevriki = distikiforx + distbirtoimx;

freqrespforx1 = FourierTransform(signalfirstx, t, Ts);
absfreqrespforx1 = abs(freqrespforx1);
freqrespforx2 = FourierTransform(signalsecondx, t, Ts);
absfreqrespforx2 = abs(freqrespforx2);
freqrespforcevrbir = FourierTransform(cevrbir, t, Ts);
lenfreqrespforcevrbir = length(freqrespforcevrbir);
absfreqrespforcevrbir = abs(freqrespforcevrbir);
freqrespforcevriki = FourierTransform(cevriki, t, Ts);
absfreqrespforcevriki = abs(freqrespforcevriki);

%Let's start to draw our figure

freq = linspace(-Fs/2, Fs/2, 2*length(t));
variancex = linspace(-Fs/2, Fs/2, lenfreqrespforcevrbir);
variancexmut = abs(variancex);
figure;

% Let's draw our first signal x1(t)
subplot(4, 1, 1);
```

```
plot(t, signalfirstx);
ylabel('Amplitude');
xlabel('t');
title('x1(t)');

% Let's draw our second signal x2(t)
subplot(4, 1, 2);
plot(t, signalsecondx);
ylabel('Amplitude');
xlabel('t');
title('x2(t)');

% Let's draw our third signal r1(t)
subplot(4, 1, 3);
plot(t, cevrbir);
ylabel('Amplitude');
xlabel('t');
title('r1(t)');

% Let's draw our fourth signal r2(t)
subplot(4, 1, 4);
plot(t, cevriki);
ylabel('Amplitude');
xlabel('t');
title('r2(t)');


%Let's draw our second part

figure;
subplot(2,1,1);
plot(variancex, absfreqrespforcevrbir);
hold on;
plot(variancex, absfreqrespforx1);
title('Amplitude of R1 and X1 in Freq Domain');
ylabel('Amplitude');
xlabel('W');
hold off;


subplot(2,1,2);
plot(variancex, absfreqrespforcevriki);
hold on;
plot(variancex, absfreqrespforx2);
title('Amplitude of R2 and X2 in Freq Domain');
ylabel('Amplitude');
xlabel('W');
hold off;

%Let's filter
bandwidth=25;
namnam = ilkfreq - bandwidth;
nana = ilkfreq + bandwidth;
elimination = (variancexmut >= (namnam)) & (variancexmut <= (nana));
resfirstelimination = freqrespforcevrbir .* elimination;
mutresfirstelimination = abs(resfirstelimination);

lemlem = ikifreq - bandwidth;
lele = ikifreq + bandwidth;
```

```
ikielimin = (variancexmut >= (lemlem)) & (variancexmut <= (lele));
ressecondelimination = freqrespforcevriki .* ikielimin;
mutressecondelimination = abs(ressecondelimination);


%Let's figure our filtered results
figure;
subplot(2,1,1);
plot(frequencies, mutresfirstelimination);
ylabel('Amplitude');
xlabel('W');
title('Filtered Y1');

subplot(2,1,2);
plot(frequencies, mutressecondelimination);
ylabel('Amplitude');
xlabel('W');
title('Filtered Y2');

lazim1 = abs(freq - ilkfreq);
[~, ilkid] = min(lazim1);
filtereedy1 = resfirstelimination(ilkid);

lazim2 = abs(freq - ikifreq)
[~, secid] = min(lazim2);
filtereedy2 = ressecondelimination(secid);

lag1 = angle(filtereedy1);
lag2 = angle(filtereedy2);
ourassumption1 = (-velci * lag1) / (4 * pi * ilkfreq)-0.08;
ourassumption2 = (-velci * lag2) / (4 * pi * ikifreq);
```

Part 4:

```
Ts = 0.0001; % sampling rate it is really small
Fs = 1 / Ts;
t = -0.5:Ts:1.5; %time vector accordingly ts
firstvelocity = -30;
secondvelocity = 40;
T = 1;
mesbir = 0.05; %it is given
mesiki = 0.1;
mestot = mesbir + mesiki
ilkfreq = 100;
ikifreq = 800;
velci = 343;
signalsecondx = cos(2 * pi * ikifreq * t);
signalsecondx(t < 0)=0;
signalsecondx(t > T)=0;

signalfirstx = cos(2 * pi * ilkfreq * t);
signalfirstx(t > T)=0;
signalfirstx(t < 0)=0;
```

```
distikiforx = cos(2 * pi * ikifreq *(velci+secondvelocity)/(velci-secondvelocity)*
(t-2*mesiki/velci));
distikiforx(t > (2*mesiki/velci)+T)=0;
distikiforx(t < (2*mesiki/velci))=0;

distbirforx = cos(2 * pi * ilkfreq*(velci+firstvelocity)/(velci-firstvelocity) *
(t-2*mesbir/velci));
distbirforx(t > (2*mesbir/velci)+T)=0;
distbirforx(t < (2*mesbir/velci))=0;


distbirtoimx = cos(2 * pi * ilkfreq *(velci+firstvelocity)/(velci-firstvelocity)*
(t-(mestot)/velci));
distbirtoimx(t > (mestot)/velci+T)=0;
distbirtoimx(t < (mestot)/velci)=0;

distikitobirx = cos(2 * pi * ikifreq *(velci+secondvelocity)/(velci-
secondvelocity) * (t-(mestot)/velci));
distikitobirx(t > (mestot)/velci+T)=0;
distikitobirx(t < (mestot)/velci)=0;

cevriki = distikiforx + distbirtoimx;
cevrbir = distbirforx + distikitobirx;



freqrespforx1 = FourierTransform(signalfirstx, t, Ts);
absfreqrespforx1 = abs(freqrespforx1);
freqrespforx2 = FourierTransform(signalsecondx, t, Ts);
absfreqrespforx2 = abs(freqrespforx2);
freqrespforcevrbir = FourierTransform(cevrbir, t, Ts);
lenfreqrespforcevrbir = length(freqrespforcevrbir);
absfreqrespforcevrbir = abs(freqrespforcevrbir);
freqrespforcevriki = FourierTransform(cevriki, t, Ts);
absfreqrespforcevriki = abs(freqrespforcevriki);


%Let's start to draw our figure

freq = linspace(-Fs/2, Fs/2, 2*length(t));
variancex = linspace(-Fs/2, Fs/2, lenfreqrespforcevrbir);

figure;

% Let's draw our first signal x1(t)
subplot(4, 1, 1);
plot(t, signalfirstx);
ylabel('Amplitude');
xlabel('t');
title('x1(t)');

% Let's draw our second signal x2(t)
subplot(4, 1, 2);
plot(t, signalsecondx);
ylabel('Amplitude');
xlabel('t');
title('x2(t)');
```

```
% Let's draw our third signal r1(t)
subplot(4, 1, 3);
plot(t, cevrbir);
ylabel('Amplitude');
xlabel('t');
title('r1(t)');

% Let's draw our fourth signal r2(t)
subplot(4, 1, 4);
plot(t, cevriki);
ylabel('Amplitude');
xlabel('t');
title('r2(t)');


%Let's draw our second part

figure;
subplot(2,1,1);
plot(variancex, absfreqrespforcevrbir);
hold on;
plot(variancex, absfreqrespforx1);
title('Amplitude of R1 and X1 in Freq Domain');
ylabel('Amplitude');
xlabel('W');
hold off;


subplot(2,1,2);
plot(variancex, absfreqrespforcevriki);
hold on;
plot(variancex, absfreqrespforx2);
title('Amplitude of R2 and X2 in Freq Domain');
ylabel('Amplitude');
xlabel('W');
hold off;
```