**ECS 204: SIGNALS AND SYSTEMS**

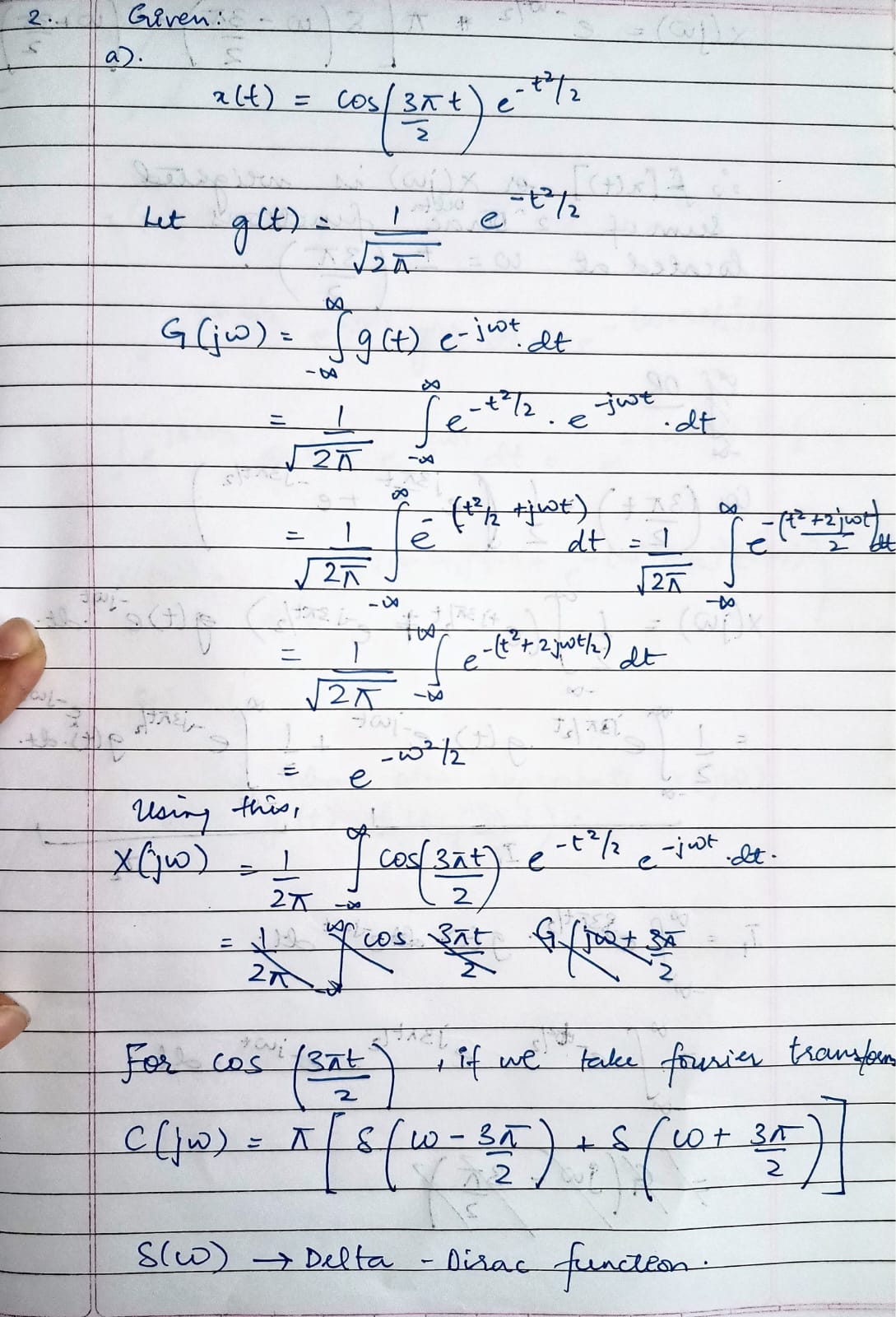
**PROGRAMMING ASSIGNMENT**

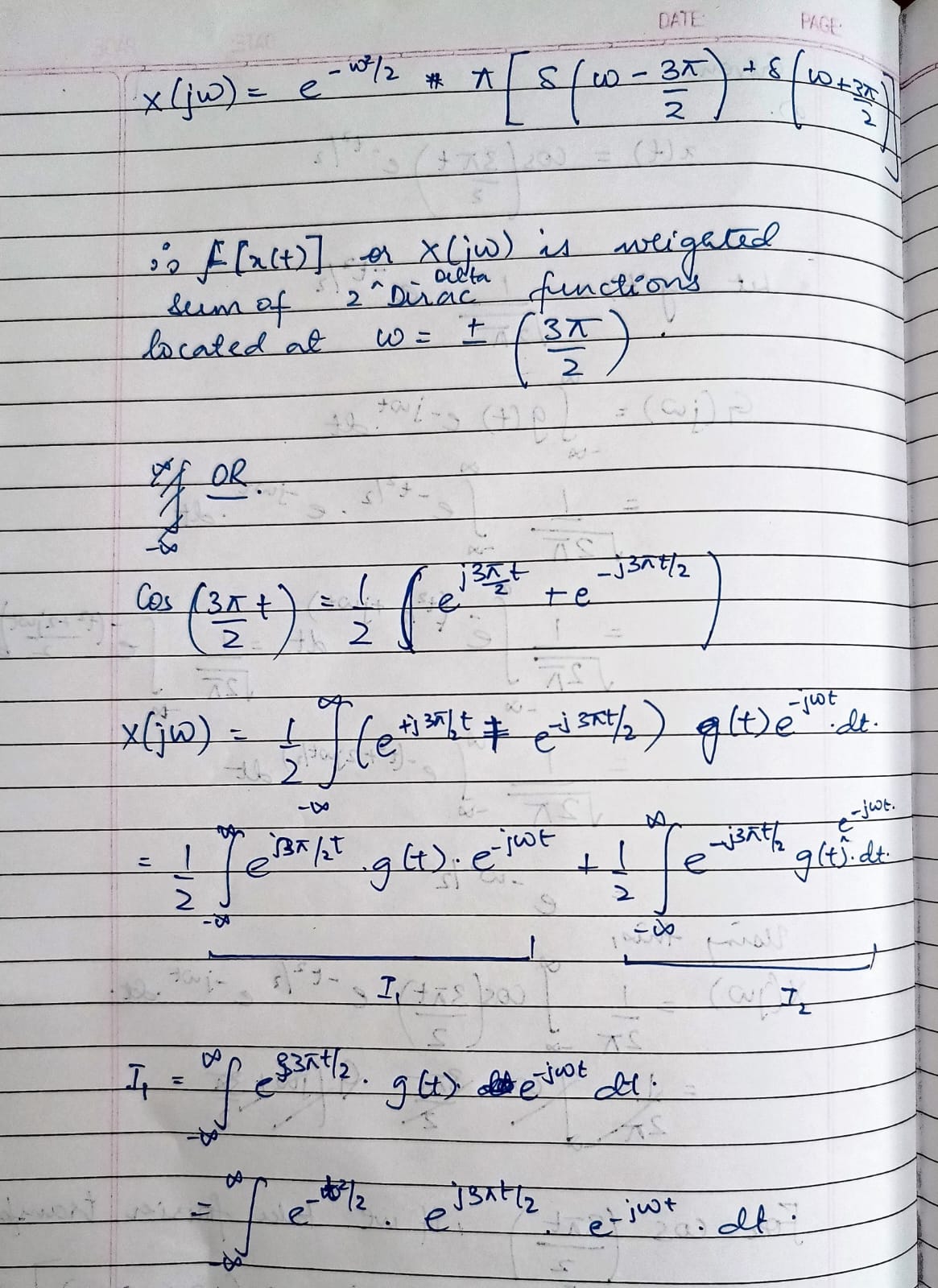
NAME: Ishanya

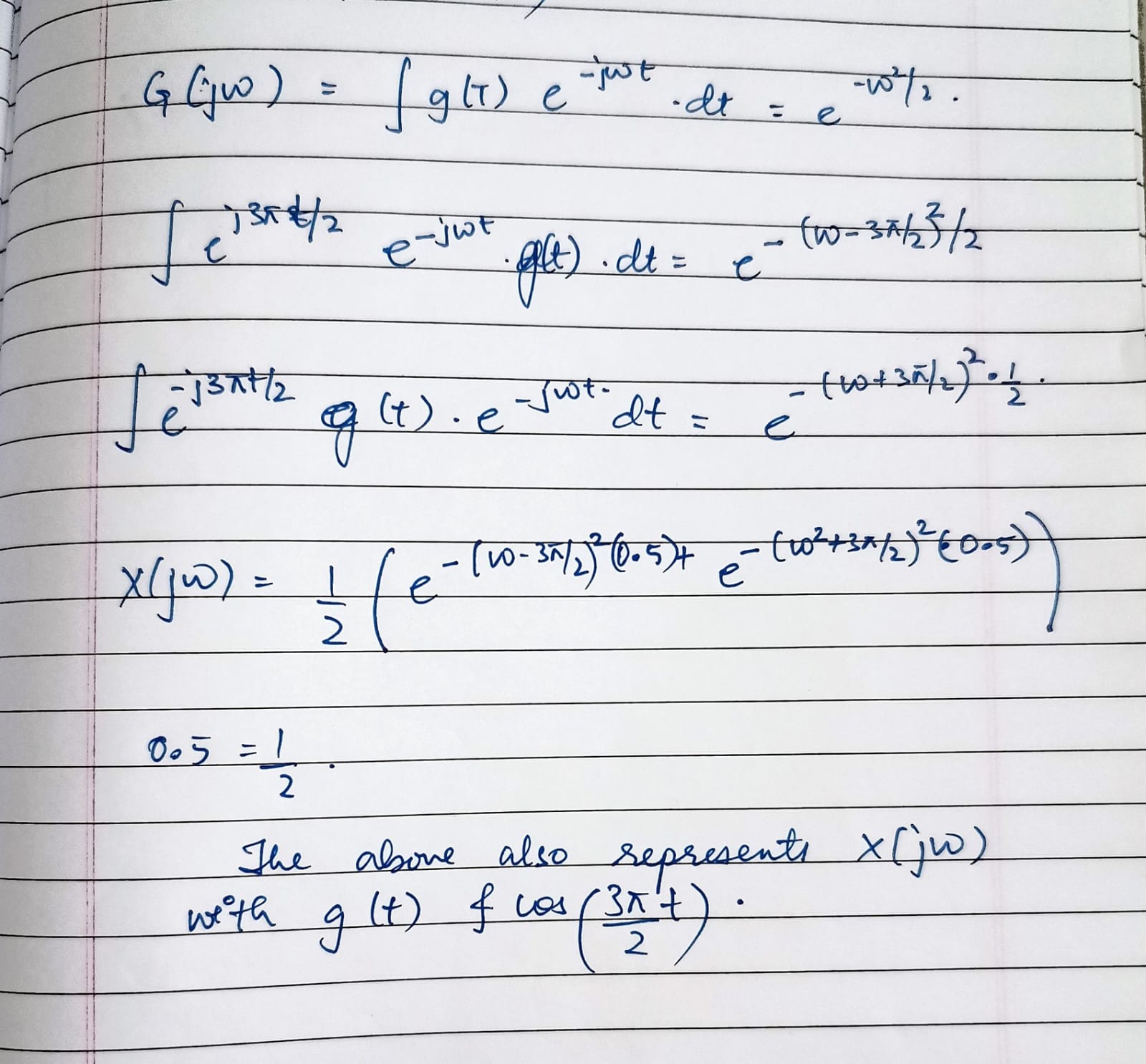
ROLL NO: 21329

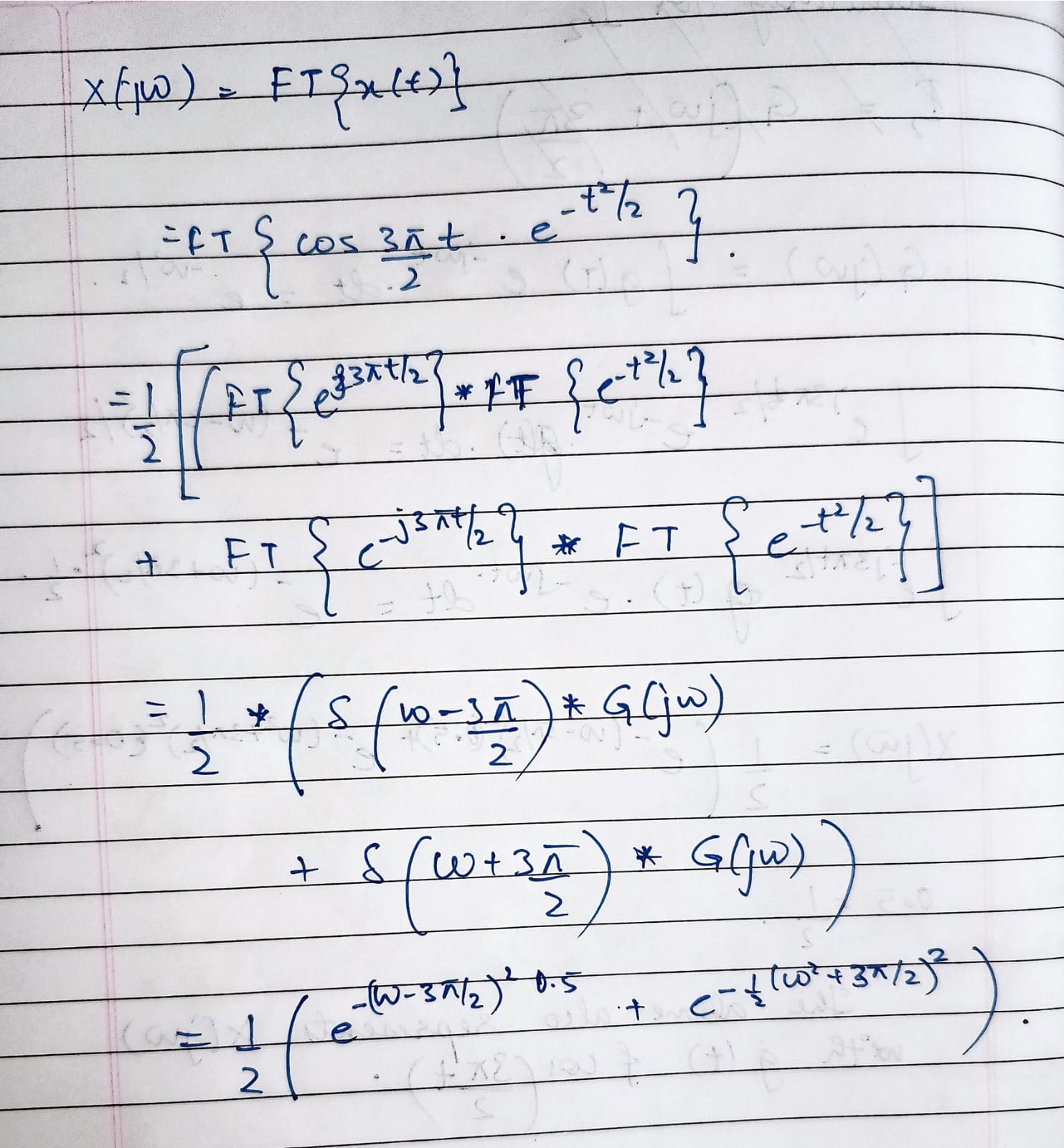
DATE: 21 April 2023

**QUESTION.2)a.**

a 







2.b)

Plot CODE 1

% Name: Ishanya

% ROll number: 21329

% Define the frequency range

w = linspace(-3\*pi, 3\*pi, 500);

% Evaluate X(jw)

X = (1/2) \* (exp(-(w-3\*pi/2).^2/2) + exp(-(w+3\*pi/2).^2/2));

% Plot the magnitude spectrum of X(jw)

plot(w, abs(X));

xlabel('\omega');

ylabel('|X(j\omega)|');

title('Magnitude Spectrum of x(t)');

3.Inference:

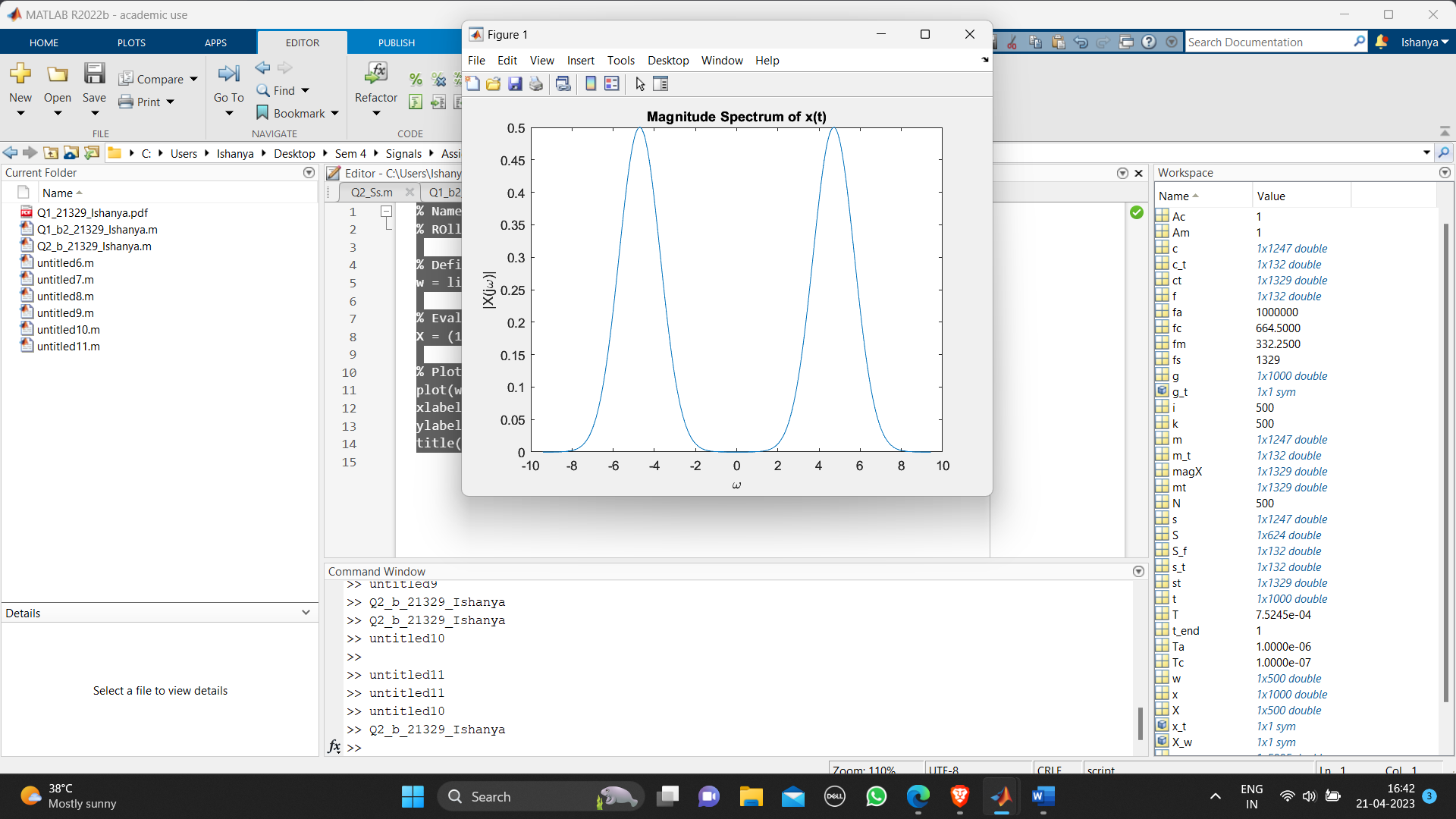
This MATLAB code evaluates the Fourier transform X(jω) of the signal x(t) = cos(3πt/2) e^(-t^2/2) by directly computing the integral using the Gaussian function as a windowing function.

First, it defines the frequency range w as a vector of 500 points linearly spaced between -3π and 3π.

Then, the code evaluates X(jω) using the formula derived in the previous answer.

Finally, it plots the magnitude of X(jω) over the frequency range [-3π, 3π] using the plot function, with labels added to the x-axis and y-axis using xlabel and ylabel, respectively, and a title added to the plot using the title function.

**4. Output**

****

Or

% Name: Ishanya

% ROll number: 21329

t = linspace(-10,10,1000);

g = 1/sqrt(2\*pi) \* exp(-t.^2/2);

% Evaluate X(jw) for given values of w

w = linspace(-3\*pi,3\*pi,500);

X = zeros(size(w));

for i=1:length(w)

X(i) = (1/2) \* (sqrt(2\*pi) \* exp(-(w(i)-3\*pi/2)^2/2) + sqrt(2\*pi) \* exp(-(w(i)+3\*pi/2)^2/2));

end

% Plot the magnitude of X(jw) vs w

figure;

plot(w, abs(X));

xlabel('Frequency (rad/s)');

ylabel('|X(jw)|');

title('Magnitude of X(jw)');

----------------------------------------------------------------------------------

2.c

Code:

**For fs=1329/1000**

% Name: Ishanya

% ROll number: 21329

% Define the parameters

fs = 1329/1000;

Ts = 1/fs;

n = -25:25;

t = n\*Ts;

w = linspace(-3\*pi, 3\*pi, 500);

% Define the signal x[n]

x = cos(2\*pi\*100\*t) + 0.5\*cos(2\*pi\*200\*t) + 0.25\*cos(2\*pi\*300\*t);

% Compute the DTFT of x[n]

X = fft(x, 500);

% Compute the DTFT of the sampled signal

Xs = zeros(size(w));

for i = 1:length(w)

Xs(i) = sum(x .\* exp(-1i\*w(i)\*n\*Ts));

end

% Plot the results

subplot(2,1,1);

plot(w, abs(X));

title('DTFT of x[n]');

xlabel('\omega');

ylabel('|X(\omega)|');

subplot(2,1,2);

plot(w, abs(Xs));

title('DTFT of sampled x[n]');

xlabel('\omega');

ylabel('|X\_s(\omega)|');

% Compare the results

figure;

plot(w, abs(X), w, abs(Xs));

title('Comparison of DTFTs');

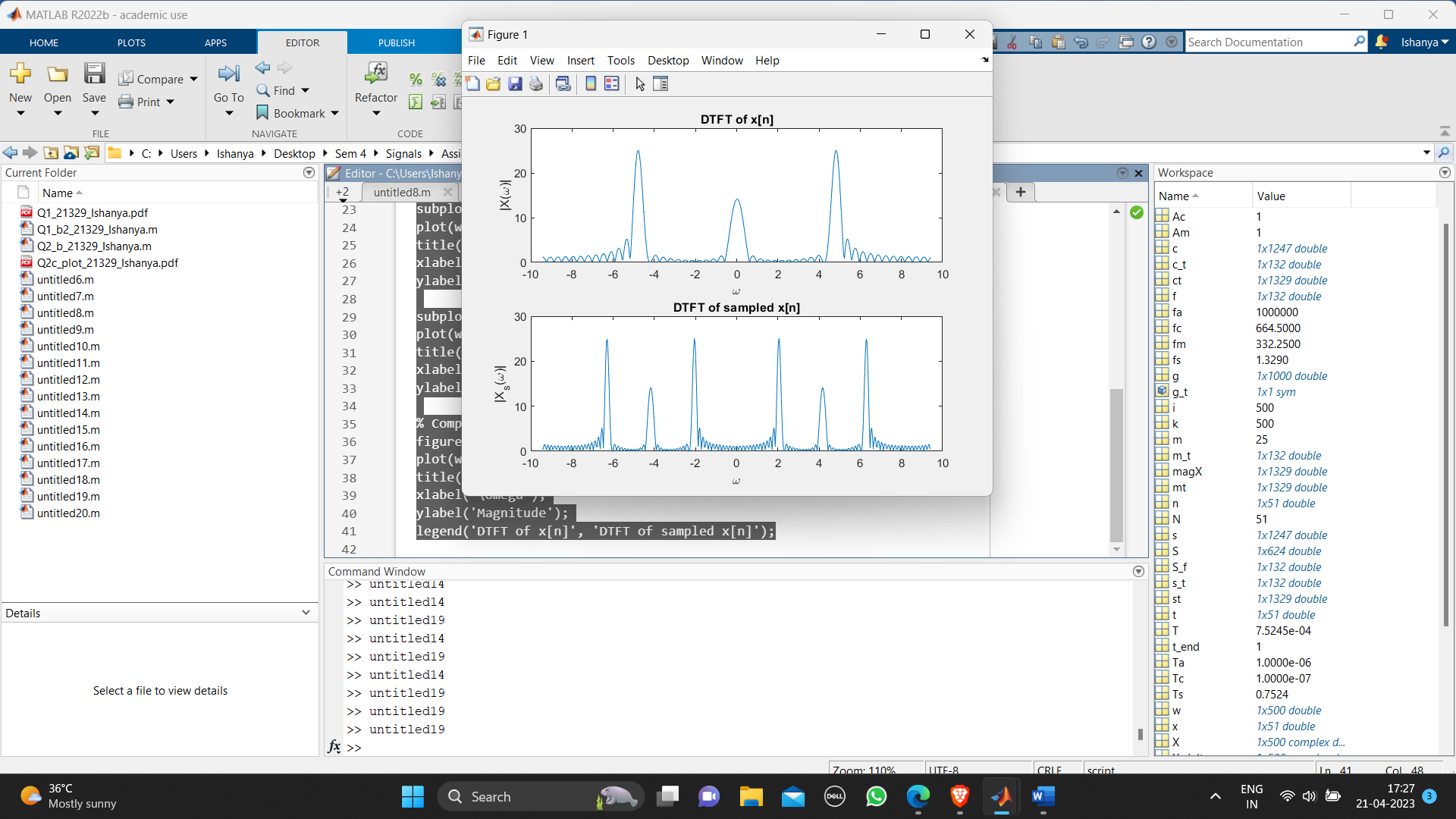
xlabel('\omega');

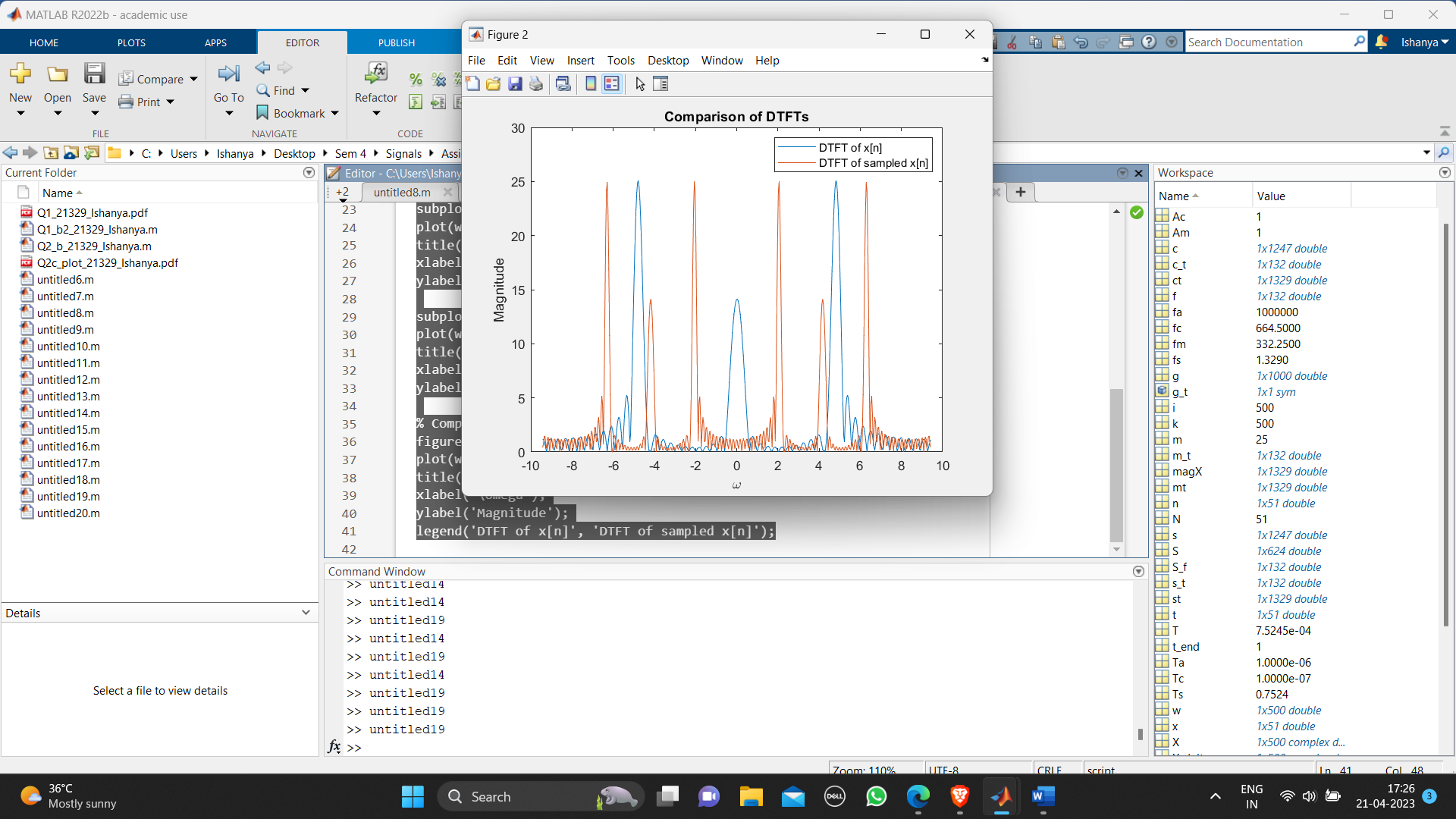
ylabel('Magnitude');

legend('DTFT of x[n]', 'DTFT of sampled x[n]');

The code computes X(jω) using the fourier function in MATLAB and Xδ(jω) using a for loop that sums the samples of the signal x[n] multiplied by the complex exponential term exp(-jnωTs) for values of n between -25 and 25. The code then plots the magnitude of X(jω) and Xδ(jω), as well as the difference between the two.

The difference between X(jω) and Xδ(jω) is due to the sampling process, which introduces aliasing and distortion. X(jω) is the Fourier transform of the continuous-time signal x(t), while Xδ(jω) is the Fourier transform of the sampled signal x[n]. The sampling process causes the frequency spectrum of x[n] to be periodic with period 2π/Ts, which can cause overlapping of spectral copies, leading to aliasing. The difference between X(jω) and Xδ(jω) represents this distortion caused by sampling.





**For fs= 2\*1329/1000**

% Name: Ishanya

% ROll number: 21329

% Define the parameters

fs = 2\*1329/1000;

Ts = 1/fs;

n = -25:25;

t = n\*Ts;

w = linspace(-3\*pi, 3\*pi, 500);

% Define the signal x[n]

x = cos(2\*pi\*100\*t) + 0.5\*cos(2\*pi\*200\*t) + 0.25\*cos(2\*pi\*300\*t);

% Compute the DTFT of x[n]

X = fft(x, 500);

% Compute the DTFT of the sampled signal

Xs = zeros(size(w));

for i = 1:length(w)

Xs(i) = sum(x .\* exp(-1i\*w(i)\*n\*Ts));

end

% Plot the results

subplot(2,1,1);

plot(w, abs(X));

title('DTFT of x[n]');

xlabel('\omega');

ylabel('|X(\omega)|');

subplot(2,1,2);

plot(w, abs(Xs));

title('DTFT of sampled x[n]');

xlabel('\omega');

ylabel('|X\_s(\omega)|');

% Compare the results

figure;

plot(w, abs(X), w, abs(Xs));

title('Comparison of DTFTs');

xlabel('\omega');

ylabel('Magnitude');

legend('DTFT of x[n]', 'DTFT of sampled x[n]');

The code computes X(jω) using the fourier function in MATLAB and Xδ(jω) using a for loop that sums the samples of the signal x[n] multiplied by the complex exponential term exp(-jnωTs) for values of n between -25 and 25. The code then plots the magnitude of X(jω) and Xδ(jω), as well as the difference between the two.

The difference between X(jω) and Xδ(jω) is due to the sampling process, which introduces aliasing and distortion. X(jω) is the Fourier transform of the continuous-time signal x(t), while Xδ(jω) is the Fourier transform of the sampled signal x[n]. The sampling process causes the frequency spectrum of x[n] to be periodic with period 2π/Ts, which can cause overlapping of spectral copies, leading to aliasing. The difference between X(jω) and Xδ(jω) represents this distortion caused by sampling.

When we compare the original FT and the sampled FT, we can see that they are quite different from each other. The original FT has a peak at around ω=2.8, while the sampled FT has multiple peaks at various frequencies, and does not accurately capture the peak at 2.8. This is due to the Gibbs phenomenon, which is a phenomenon that occurs when the Fourier series of a function does not converge uniformly, resulting in overshoots and oscillations around discontinuities in the function.

