Full Name: ______________________________

EEL 3135 (Spring 2025) – Lab #06

Question #1: *(DTFT of Filters)*

Download `EEL3135_lab06_comment.m` from Canvas, replace each of the corresponding comments with the corresponding descriptions. This is designed to show you how to visualize and analyze the frequency response of FIR filters by Discrete-Time Fourier Transform (DTFT) in MATLAB.

Note: You should run the code to help you understand how it works and help you write your comments. You will use elements of this MATLAB code for the rest of the lab assignment.

Question #2: *(DTFT of Common Functions)*

Question #1 provides a `DTFT(x,w)` function to compute the frequency-domain of signal `x` across normalized angular frequencies `w`. Note that this is *exactly* the same as your `FreqResponse` function from the previous lab. This question explores the frequency domains of common signals. Use the `DTFT` function to compute the frequency-domain magnitudes and phases of the following signals. Plot the time-domain signal, the magnitude response, and the phase response. Use Question #1 as a guide for plotting.

For each question, use only 25 values for n corresponding to $n = 0$ to $n = 24$.

For each question, answer if the data is predominantly low frequency, high frequency, or neither.

Note: Add all code into skeleton `eel3135_lab06_skeleton.m` from Canvas. Include all code (and functions) in this one file so that everything is published to a single PDF.

(a) $x[n] = \delta[n-1] + \delta[n-2] + \delta[n-3]$

(b) $x[n] = \delta[n-2] + \delta[n-3] + \delta[n-4]$

(c) $x[n] = (-0.9)^n u[n]$

(d) $x[n] = (-0.9)^n \cos((3\pi/4)n) u[n]$

(e) $x[n] = (-0.9)^n \cos((\pi/4)n) u[n]$

(f) $x[n] = (-1)^n u[n]$

Question #3: *(DTFT Properties)*

This question explores the properties of the DTFT. Use the `DTFT` function to compute the frequency-domain magnitudes and phases of the following signals. Plot the time-domain signal, the magnitude response, and the phase response. Use Question #1 as a guide for plotting. Each question uses the following signal:

$$x[n] = [(25/46) - (21/46)\cos((\pi/6)n)]\,(u[n] - u[n-12])$$

For each question, use only 40 values for n corresponding to $n = 0$ to $n = 39$.

For each question, describe how each system changes the frequency domain.

(a) $y[n] = 4x[n]$

(b) $y[n] = 4x[n-8]$

(c) $y[n] = x[n]\cos((\pi/4)n)$

(d) $y[n] = x[n] * x[n]$

(e) $y[n] = |x[n]|^2$

Question #4: *(Nulling Filter)*

In this problem, you will be given audio signals with noise (`noisy.wav` and `noisy2.wav`). Your goal is to design and apply nulling filters (a.k.a., notch or bandstop filters) to remove the noise.

(a) For `noisy.wav`, use the `DTFT` function to calculate the DTFT of the audio file. Plot its frequency-domain magnitude. Note: your magnitude plot should have at least 10000 points to properly identify the noise (it may take a minute or so to compute).

(b) For `noisy.wav`, identify the frequency that is contaminated with noise (Note: you can use "Tools→Data Tips" to identify the values of specific points in the plot). **Answer in your comments:** List this frequency in normalized angular frequency and continuous-time cyclic frequency.

(c) Consider the following nulling FIR filter:

$$y[n] = x[n] - 2\cos(\widehat{w}_0)x[n-1] + x[n-2], \tag{1}$$

This filter can remove the normalized frequency specified by $\widehat{w}_0$. For `Noisy.wav`, design this filter to remove the frequency that you identified in (b). Use `subplot` and the `DTFT` function to plot the magnitude response and phase response of the filter.

(d) Apply (i.e., convolve) the nulling filter to the noisy audio data. Use the `DTFT` function to plot the magnitude response of the filtered audio data. **Answer in your comments:** What are the differences between the original audio and the output of the nulling filter?

(e) Listen to the output of the nulling filter. **Turn in a .wav file of this audio output along with your PDF.**

(f) For `noisy2.wav`, there are multiple contaminated frequencies. Use a similar approach and adapt your strategy (based on what you previously learned) to remove every frequency and identify the underlying audio. **Turn in a .wav file of this audio output with your PDF.**