Full Name: ____________________

EEL 3135 (Spring 2021) – Lab #11 **Due: 11:59 PM ET, Mar. 13 - 19 (On Lab Day)**

Question #1: *(Implementing the DFT)*

Download `EEL3135_lab11_comment.m` from Canvas, replace each of the corresponding comments with the corresponding descriptions. This is designed to let you get more familiar with DFT.

Question #2: *(DFT and DTFT)*

In this question, you create a DFT function and compare results with the DTFT function. Keep in mind that the DFT can be thought of as a sampled version of the DTFT. Consider the following discrete-time signal:

$$x[n] = [0.75 + \cos((\pi/20)n) + \cos((\pi/15)n) + \cos(\pi n + 2\pi/3)]\,[u[n] - u[n-60]]$$

(a) Take the DTFT function from previous labs and modify it into the function `X = DFT(x)`, which inputs a signal `x` and outputs its DFT `X` with the same length.

Hint 1: The DTFT and DFT are defined as (where N is the signal length):

$$\text{DTFT: } X(\widehat{\omega}) = \sum_{n=0}^{N-1} x[n]e^{-j\widehat{\omega}n} \qquad , \qquad \text{DFT: } X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\left(\frac{2\pi}{N}\right)kn}$$

Hint 2: The `w` variable in the skeleton code's `DFT` function should be in normalized angular frequency $\widehat{\omega}$ and can be determined by directly comparing the DFT and DTFT equations.

(b) Use `plot` to plot the magnitude of the DTFT. Be sure to label axes with proper units.

(c) Now plot the magnitude of the 60-length DFT of $x[n]$ on top of the DTFT. See the skeleton code on how to best plot the DFT on top of the DTFT plot. Note: `w_DFT` is the same as `w` in the DFT function.

(d) Now plot the magnitude of the 55-length DFT (i.e., remove the last 5 values from the original signal) of $x[n]$ on top of the DTFT.

(e) Now plot the magnitude of the 65-length DFT (i.e., add 5 zeros to the original signal) of $x[n]$ on top of the DTFT.

(f) Now plot the magnitude of the 200-length DFT of $x[n]$ on top of the DTFT.

(g) **Answer in your comments:** Based on the last several questions, what is the relationship between the DTFT and the DFT? Under what conditions will the theoretical DTFT and DFT have the same result?

Question #3: *(DFT and IDFT)* In this question, you create the inverse DFT function and use it with the DFT function to perform convolution. Consider the following discrete-time signal:

$$x[n] = u[n-10] - u[n-45]$$

(a) Modify the DFT function from Question #2 to create an inverse DFT function `x = IDFT(X)`, which inputs a DFT-transformed signal `X` and outputs the time signal `x`.

Hint: The inverse DFT is defined as

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{+j\left(\frac{2\pi}{N}\right)kn}$$

(b) Use `stem` to plot the $x[n]$ for $0 \leq n \leq 99$.

(c) Use `conv` to compute $y[n] = x[n] * x[n]$ and then use `stem` to plot the result for $0 \leq n \leq 100$.

(d) Use $N = 100$ length `DFT` and `IDFT` functions to compute $y[n] = x[n] * x[n]$ (i.e., use the convolution property) and then use `stem` to plot the result.

(e) Use $N = 60$ length `DFT` and `IDFT` functions to compute $y[n] = x[n] * x[n]$ and then use `stem` to plot the result.

(f) **Answer in your comments:** What is the difference in the last two solutions? Why does the difference exist?

Question #4: *(DFT and FFT)*

This question will focus on computation time differences between `DFT` and `FFT`. Choose a song at least 3 minutes long for this problem and include it in your submission. Load it into MATLAB using `audioread`. Note that most audio files will be stereo, so you need to make sure that you only use one column of the audio data for this part of the lab (e.g., set `x = x(:,1);`). This site has a large archive of free music that you can choose from: `https://freemusicarchive.org/static`.

(a) Use `DFT` to plot the magnitude of the DFT of only the first 10000 samples of the audio. Use `tic` and `toc` (read `help tic` and `help toc`) to measure the length of time it takes to compute the DFT, in seconds. Display the result with the `disp` function.

(b) Use `fft` (read `help fft`) to plot the magnitude of the DFT of only the first 10000 samples of the audio. Use `tic` and `toc` to measure the length of time it takes to compute the FFT, in seconds. Display the result with the `disp` function.

(c) **Answer in your comments:** Are there differences in the magnitude results? If so, why?

(d) **Answer in your comments:** How much faster is the `FFT` algorithm compared with the `DFT` in this scenario?

(e) We will not perform this test, but using `DFT` on an entire 3 minute audio (sampled at 44.1 kHz) should take approximately 500 hours. So instead, use `FFT` to plot the magnitude of the DFT of the **entire** audio signal. Use `tic` and `toc` to measure the length of time it takes to compute the FFT, in seconds. Display the result with the `disp` function.