[Connor Dupuis]

Table of Contents

[Friday 1:55pm] - [28944] - [Naoki Sawahashi]

QUESTION 2

DO NOT REMOVE THE LINE BELOW MAKE SURE 'eel3135_lab11_comment.m' IS IN THE SAME DIRECTORY AS THIS FILE

```
clear; close all; clc;
type('eel3135_lab11_comment.m')
```

```
%% Question 1 Comment Code
clear all; clc; close all;

fs = 16000;
tt = 0:1/fs:0.01-1/fs;

w0 = 888*pi;
w1 = 3520*pi;
w2 = 14080*pi;
```

```
x = 0.5 + cos(w0*tt) + cos(w1*tt + pi/4) + cos(w2*tt + 2*pi/3);

N = [50,length(x),500,1000];

X1 = fft(x,N(1));
X2 = fft(x,N(2));
X3 = fft(x,N(3));
X4 = fft(x,N(4));

% Answer in your comments: How is the DFT (using the FFT algorithm)
% calculated when N is larger than the length of X?
% (Hint: read help FFT)
% If N is greater than the length of the signal, then x is padded with
% trailing zeros to length N.

% Answer on your comments: How is the DFT (using the FFT algorithm)
% calculated  when N is smaller than the length of X?
% If N is less than the length of the signal, then fft ignores the
 remaining
% signal values past the nth entry and returns the truncated result

% Answer in your comments: Why would you not want to always make N as
 large
% as possible?
% Increasing N increases the amount of time it takes to calculate the
% result.

b1 = -ceil((N(1)-1)/2):floor((N(1)-1)/2);
b2 = -ceil((N(2)-1)/2):floor((N(2)-1)/2);
b3 = -ceil((N(3)-1)/2):floor((N(3)-1)/2);
b4 = -ceil((N(4)-1)/2):floor((N(4)-1)/2);

% Answer in your comments: What frequencies do the peaks represent?
%

f1 = b1*fs/N(1);
f2 = b2*fs/N(2);
f3 = b3*fs/N(3);
f4 = b4*fs/N(4);

w1 = 2*pi/N(1)*b1;
w2 = 2*pi/N(2)*b2;
w3 = 2*pi/N(3)*b3;
w4 = 2*pi/N(4)*b4;

figure
subplot(221)
plot(b1,fftshift(abs(X1)))
hold on;
plot(b1,fftshift(abs(X1)), '.', 'markersize', 8)
hold off;
xlim([-N(1)/2 N(1)/2]); ylim([0 100]);
xlabel('Bin Number')
```

```
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(1)))

subplot(222)
plot(b2,fftshift(abs(X2)))
hold on;
plot(b2,fftshift(abs(X2)), '.', 'markersize', 8)
hold off;
xlim([-N(2)/2 N(2)/2]); ylim([0 100]);
xlabel('Bin Number')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(2)))

subplot(223)
plot(b3,fftshift(abs(X3)))
hold on;
plot(b3,fftshift(abs(X3)), '.', 'markersize', 8)
hold off;
xlim([-N(3)/2 N(3)/2]); ylim([0 100]);
xlabel('Bin Number')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(3)))

subplot(224)
plot(b4,fftshift(abs(X4)))
hold on;
plot(b4,fftshift(abs(X4)), '.', 'markersize', 8)
hold off;
xlim([-N(4)/2 N(4)/2]); ylim([0 81]);
xlabel('Bin Number')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(4)))



figure
subplot(221)
plot(f1,fftshift(abs(X1)))
hold on;
plot(f1,fftshift(abs(X1)), '.', 'markersize', 8)
hold off;
xlim([-fs/2 fs/2]); ylim([0 81]);
xlabel('Frequency (Hz)')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(1)))

subplot(222)
plot(f2,fftshift(abs(X2)))
hold on;
plot(f2,fftshift(abs(X2)), '.', 'markersize', 8)
hold off;
xlim([-fs/2 fs/2]); ylim([0 81]);
xlabel('Frequency (Hz)')
ylabel('Magnitude')
```

```
title(sprintf('Length %i DFT',N(2)))

subplot(223)
plot(f3,fftshift(abs(X3)))
hold on;
plot(f3,fftshift(abs(X3)), '.', 'markersize', 8)
hold off;
xlim([-fs/2 fs/2]); ylim([0 81]);
xlabel('Frequency (Hz)')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(3)))

subplot(224)
plot(f4,fftshift(abs(X4)))
hold on;
plot(f4,fftshift(abs(X4)), '.', 'markersize', 8)
hold off;
xlim([-fs/2 fs/2]); ylim([0 81]);
xlabel('Frequency (Hz)')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(4)))




figure
subplot(221)
plot(w1,fftshift(abs(X1)))
hold on;
plot(w1,fftshift(abs(X1)), '.', 'markersize', 8)
hold off;
xlim([-pi pi]); ylim([0 81]);
xlabel('Normalized Frequency (rad/s)');
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(1)))

subplot(222)
plot(w2,fftshift(abs(X2)))
hold on;
plot(w2,fftshift(abs(X2)), '.', 'markersize', 8)
hold off;
xlim([-pi pi]); ylim([0 81]);
xlabel('Normalized Frequency (rad/s)');
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(2)))

subplot(223)
plot(w3,fftshift(abs(X3)))
hold on;
plot(w3,fftshift(abs(X3)), '.', 'markersize', 8)
hold off;
xlim([-pi pi]); ylim([0 81]);
xlabel('Normalized Frequency (rad/s)');
```

```
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(3)))

subplot(224)
plot(w4,fftshift(abs(X4)))
hold on;
plot(w4,fftshift(abs(X4)), '.', 'markersize', 8)
hold off;
xlim([-pi pi]); ylim([0 81]);
xlabel('Normalized Frequency (rad/s)')
ylabel('Magnitude')
title(sprintf('Length %i DFT',N(4)))

% Answer in your comments: How does the length of the DFT affect the
% magnitude? Be sure to zoom in on the bases of the peaks in each DFT
% magnitude plot.
% Increasing the length of the DFT increases the amount of samples.
```

Question 2

This question will cover the creation of their own DFT function by modifying their DTFT function Keep in mind that the DFT can be thought of as a sampled version of the DTFT

```
n=0:59;
x = 0.75 + cos(pi*n/20) + cos(pi*n/15) + cos(pi*n + 2*pi/3);

w_DTFT = linspace(0, 2*pi-pi/5000, 10000);
X_DTFT = DTFT(x, w_DTFT);

% NOTE: USE THE FOLLOWING COMMENTED LINE FOR PLOTTING THE DFT ATOP THE
 DTFT
% (YOU NEED TO DEFINE w_DFT), THIS WILL MAKE THE PLOTS EASIER TO
 INTERPRET
```

Question 2a

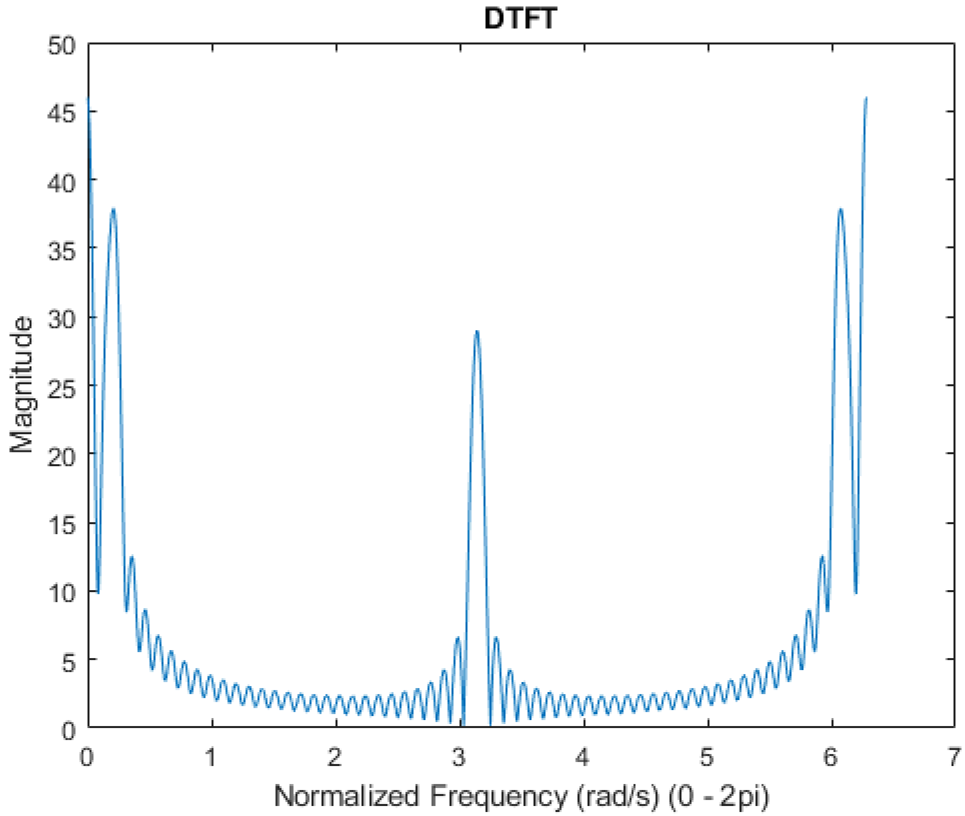
Take your DTFT function from previous labs, and modify it to output the DFT given an input X

```
w_DFT = ((2*pi)/length(n))*(n);
X_DFT = DFT(x);
```

Question 2b

Given x and w_DTFT above, plot the magnitude of the DTFT. Be sure to label axes with units and title the plot.

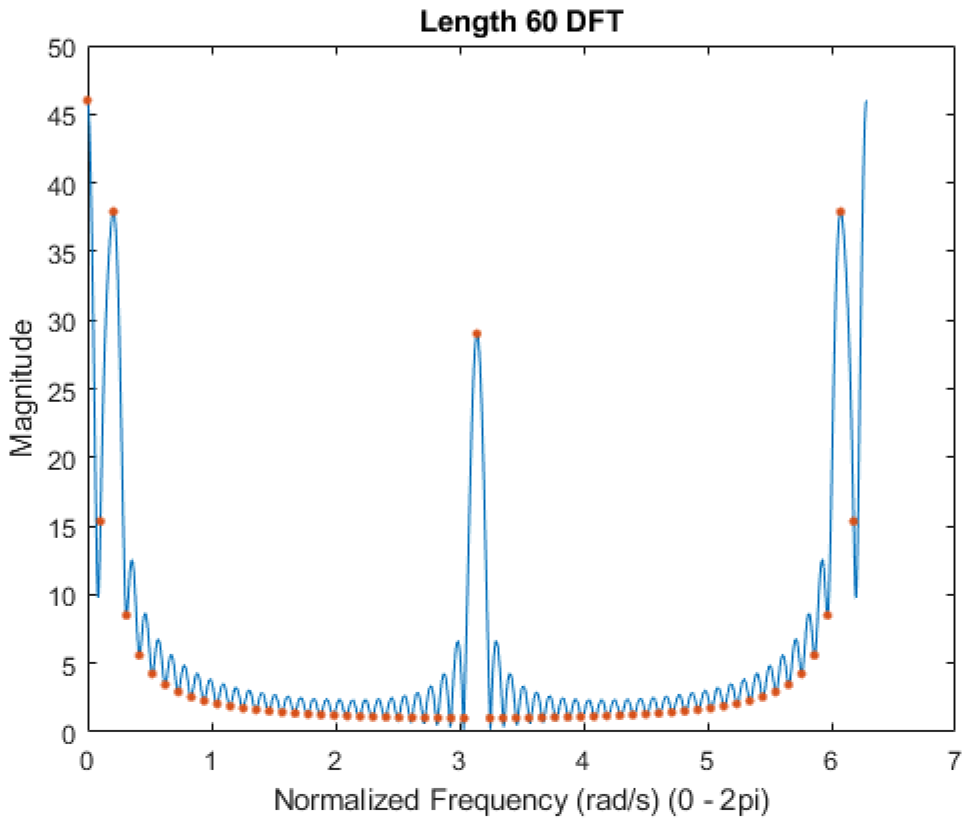
```
figure
plot(w_DTFT,abs(X_DTFT));
xlabel('Normalized Frequency (rad/s) (0 - 2pi)');
ylabel('Magnitude')
title('DTFT')
```



Question 2c

Now plot the magnitude of the 60-length DFT of x. Use hold on and hold off to plot the DFT on top of the DTFT plot

```
figure
plot(w_DTFT,abs(X_DTFT));
hold on; plot(w_DFT,abs(X_DFT),'.', 'markersize', 10); hold off;
xlabel('Normalized Frequency (rad/s) (0 - 2pi)');
ylabel('Magnitude')
title('Length 60 DFT')
```



Question 2d

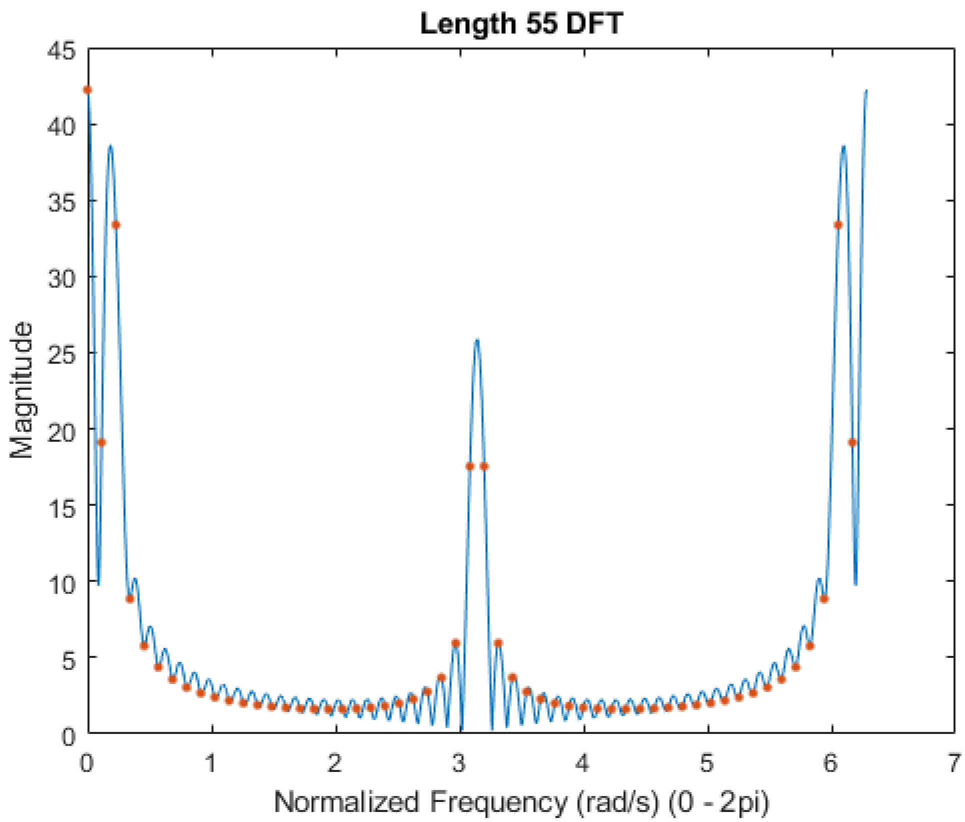
Now plot the magnitude of the 55-length DFT (i.e. remove the last 5 values) of x. Use hold on and hold off to plot the DFT on top of the DTFT plot

```
x5 = x(1:55);

X_DTFT = DTFT(x5, w_DTFT);

w_DFT = ((2*pi)/length(x5))*(0:length(x5)-1);
X_DFT = DFT(x5);

figure
plot(w_DTFT,abs(X_DTFT));
hold on; plot(w_DFT,abs(X_DFT),'.', 'markersize', 10); hold off;
xlabel('Normalized Frequency (rad/s) (0 - 2pi)');
ylabel('Magnitude')
title('Length 55 DFT')
```



Question 2e

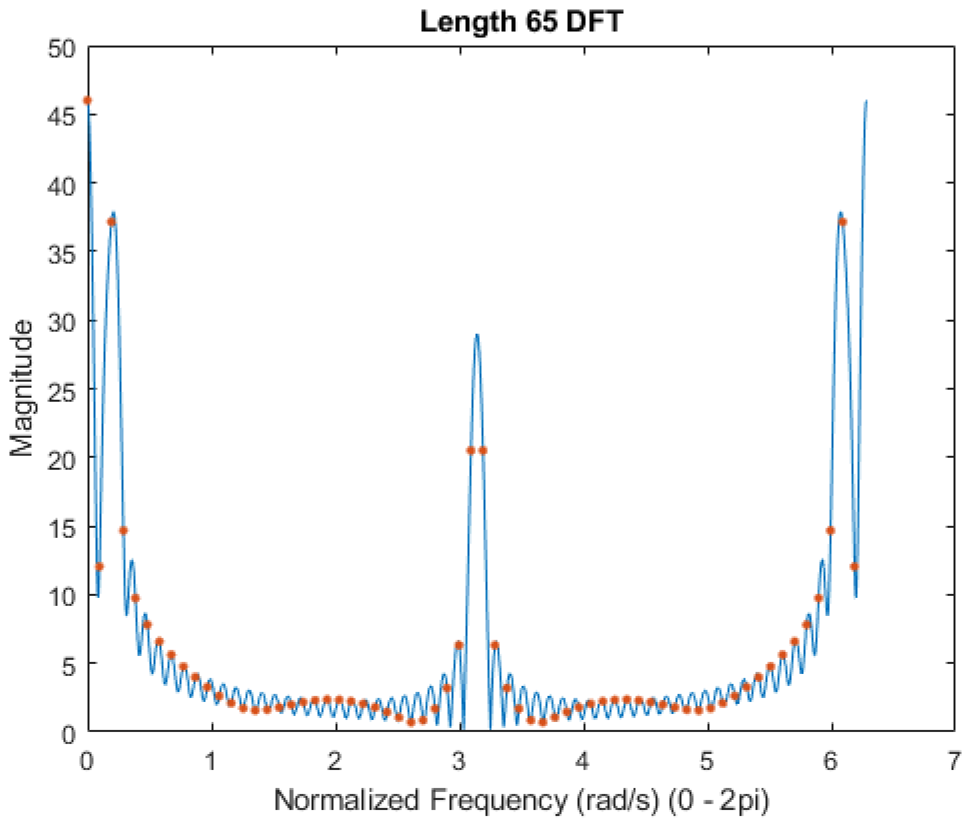
Now plot the magnitude of the 65-length DFT (i.e. include 5 zeros values) of x. Use hold on and hold off to plot the DFT on top of the DTFT plot

```
x0 = [x zeros(1,5)];

X_DTFT = DTFT(x0, w_DTFT);

w_DFT = ((2*pi)/length(x0))*(0:length(x0)-1);
X_DFT = DFT(x0);

figure
plot(w_DTFT,abs(X_DTFT));
hold on; plot(w_DFT,abs(X_DFT),'.', 'markersize', 10); hold off;
xlabel('Normalized Frequency (rad/s) (0 - 2pi)');
ylabel('Magnitude')
title('Length 65 DFT')
```



Question 2f

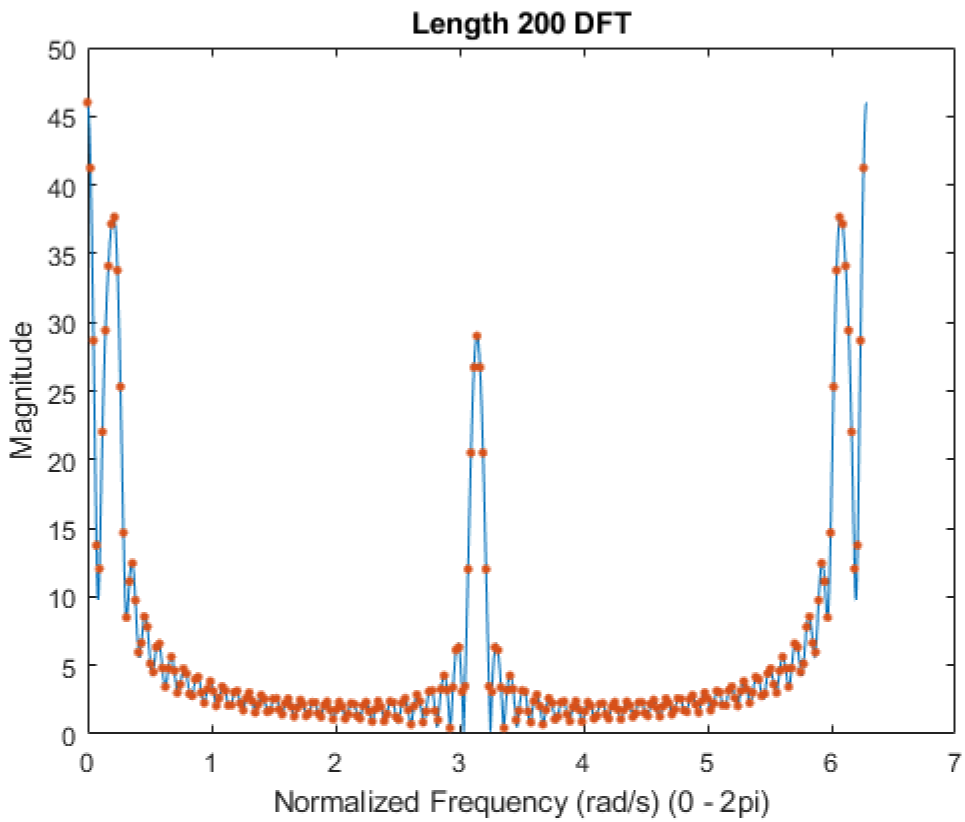
Now plot the magnitude of the 200-length DFT of x. Use hold on and hold off to plot the DFT on top of the DTFT plot

```
x200 = [x zeros(1,200)];

X_DTFT = DTFT(x200, w_DTFT);

w_DFT = ((2*pi)/length(x200))*(0:length(x200)-1);
X_DFT = DFT(x200);

figure
plot(w_DTFT,abs(X_DTFT));
hold on; plot(w_DFT,abs(X_DFT),'.', 'markersize', 10); hold off;
xlabel('Normalized Frequency (rad/s) (0 - 2pi)');
ylabel('Magnitude')
title('Length 200 DFT')
```



Question 2g

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 (i.e. when w_hat is continuous) and DFT have the same result?

```
% The DFT is a sampled DTFT. If the length of the sample (N) is
 inifinite,
% then the DFT would equal the DTFT.
```

Question 3

This question will cover the creation of your own DFT function by modifying your DTFT function. Keep in mind that the DFT can be thought of as a sampled version of the DTFT. In this problem. consider the following discrete-time signal: x[n] = u[n-10] - u[n-45]

```
clear all; clc;
x = [zeros(1,10) ones(1,35)];
```

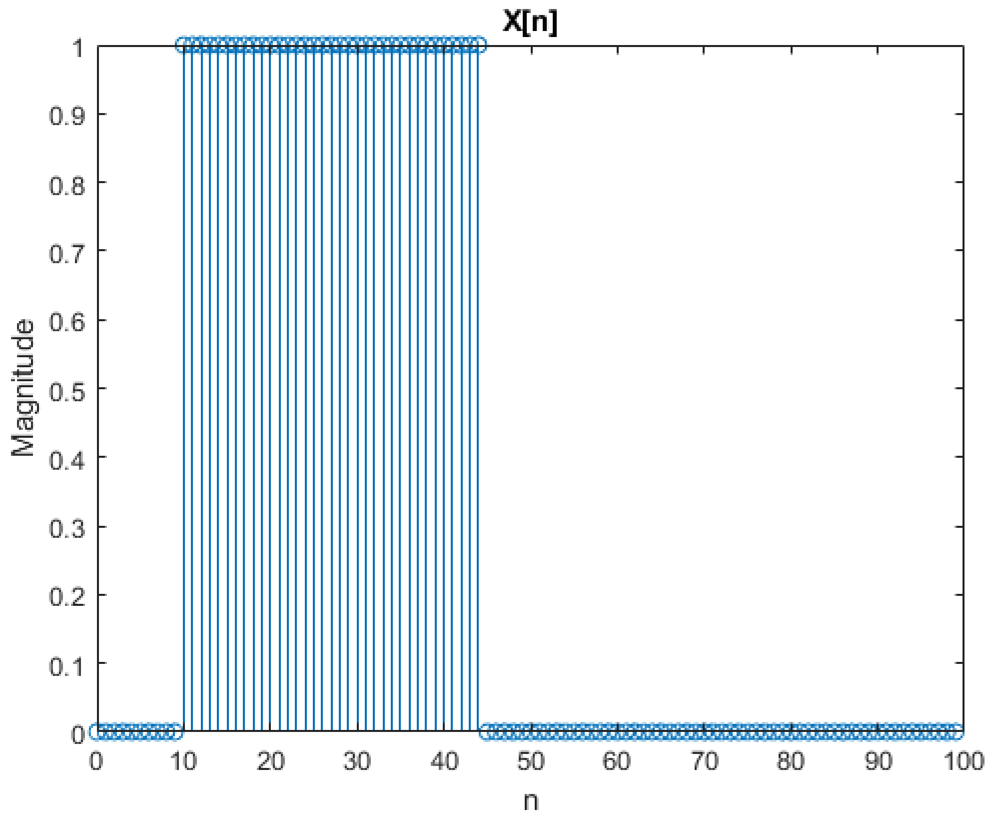
Question 3a

Modify the DFT question from Question 2 to create an inverse DFT function x = IDFT(x), which inputs a DFT-transformed signal X and outputs the time-domain signal x. Hint: the inverse DFT is defined in the lab document.

Question 3b

Use stem to plot x[n] for n from 0 to 99, inclusive

```
x99 = [x zeros(1,55)];
figure
stem(0:length(x99)-1,x99);
xlabel('n');
ylabel('Magnitude')
title('X[n]')
```

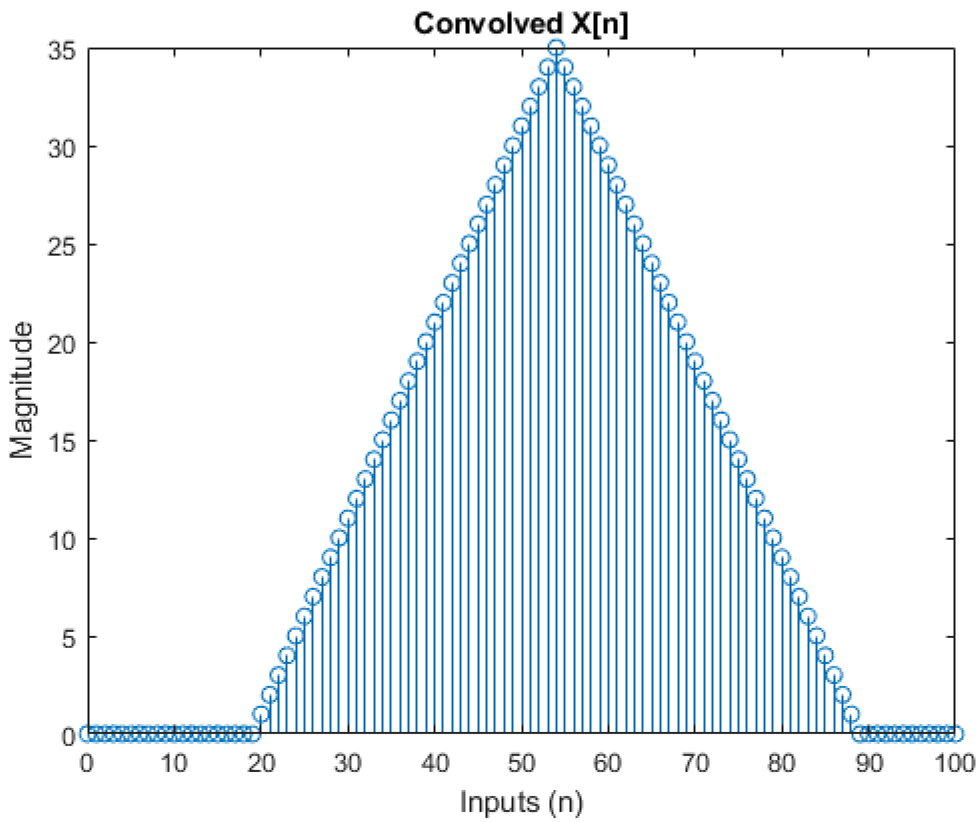


Question 3c

Use conv to compute y[n] = (x[n] convolved with x[n]) and then use stem to plot the result for n from 0 to 100, inclusive

```
xc = conv(x,x);
xc100 = [xc, zeros(1,12)];

figure
stem(0:length(xc100)-1,xc100);
xlabel('Inputs (n)');
ylabel('Magnitude')
title('Convolved X[n]')
```



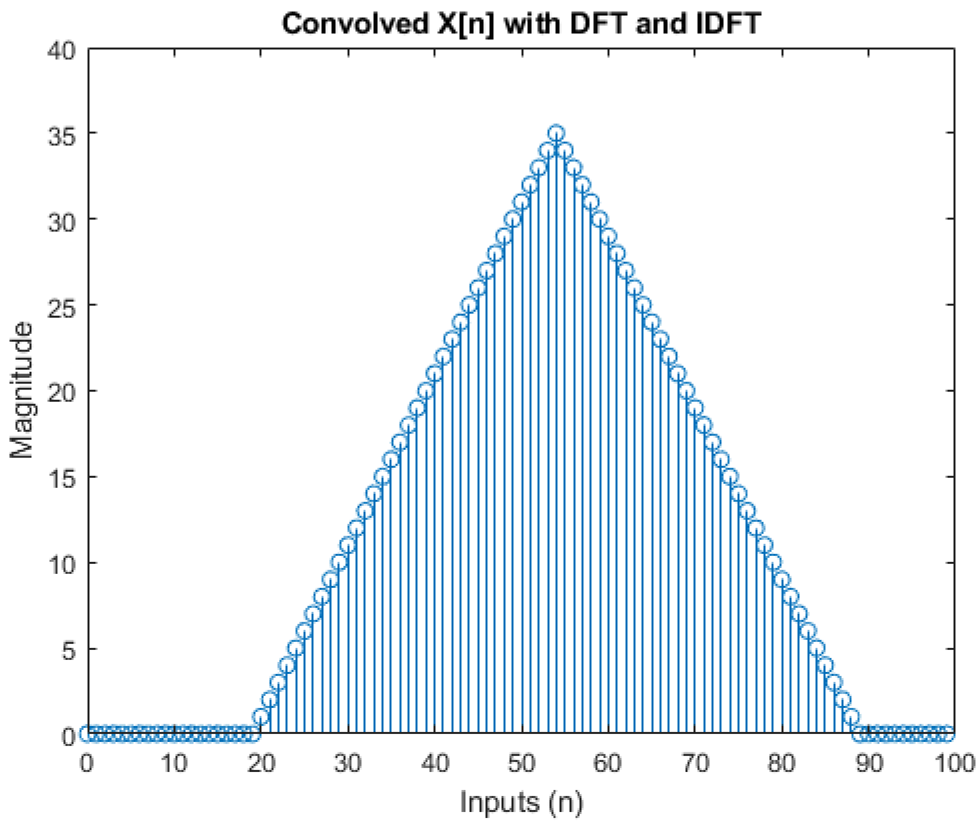
Question 3d

Use N = 100 length DFT function to compute y[n] = (x[n] convolved with x[n]) in the frequency domain, then convert back to the time domain. Use stem to plot the result.

```
xd = DFT(x99);
xd = xd.*xd;
xi = IDFT(xd);

figure
stem(0:length(xi)-1, xi);
xlabel('Inputs (n)');
ylabel('Magnitude')
title('Convolved X[n] with DFT and IDFT')
```

Warning: Using only the real component of complex data.



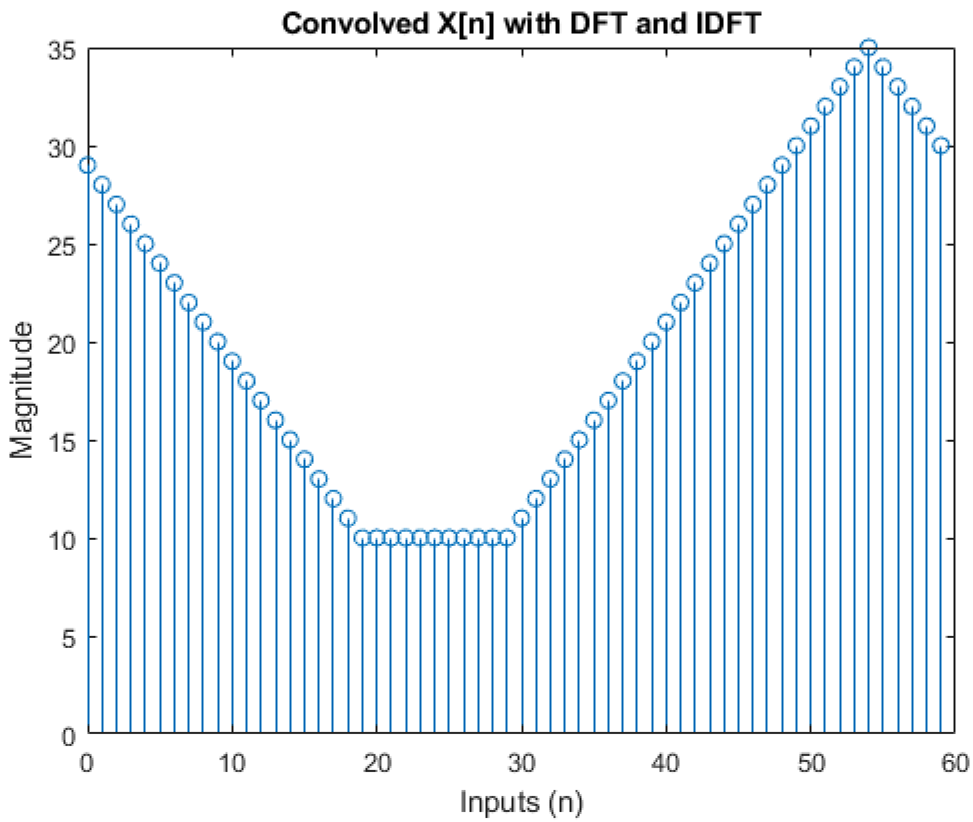
Question 3e

Use N = 60, and repeat 3d

```
x60 = [x, zeros(1,15)];
xd = DFT(x60);
xd = xd.*xd;
xi = IDFT(xd);

figure
stem(0:length(xi)-1, xi);
xlabel('Inputs (n)');
ylabel('Magnitude')
title('Convolved X[n] with DFT and IDFT')
```

Warning: Using only the real component of complex data.



Question 3f

Answer in your comments: What is the difference in the last two solutions? Why does this difference exist?

```
% In the second problem we are only sampling the first 60 points from
 the
% convolution. The original convolution is 90 points long, but when we
 use
% the DFT, we are only using 60. We have to wrap from point 0 to
% point 60, then it would repeat again from point 0.
```

Question 4

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

```
clear all; clc;
[x,fs] = audioread('Wisteria.mp3');
x = x(:,1);
```

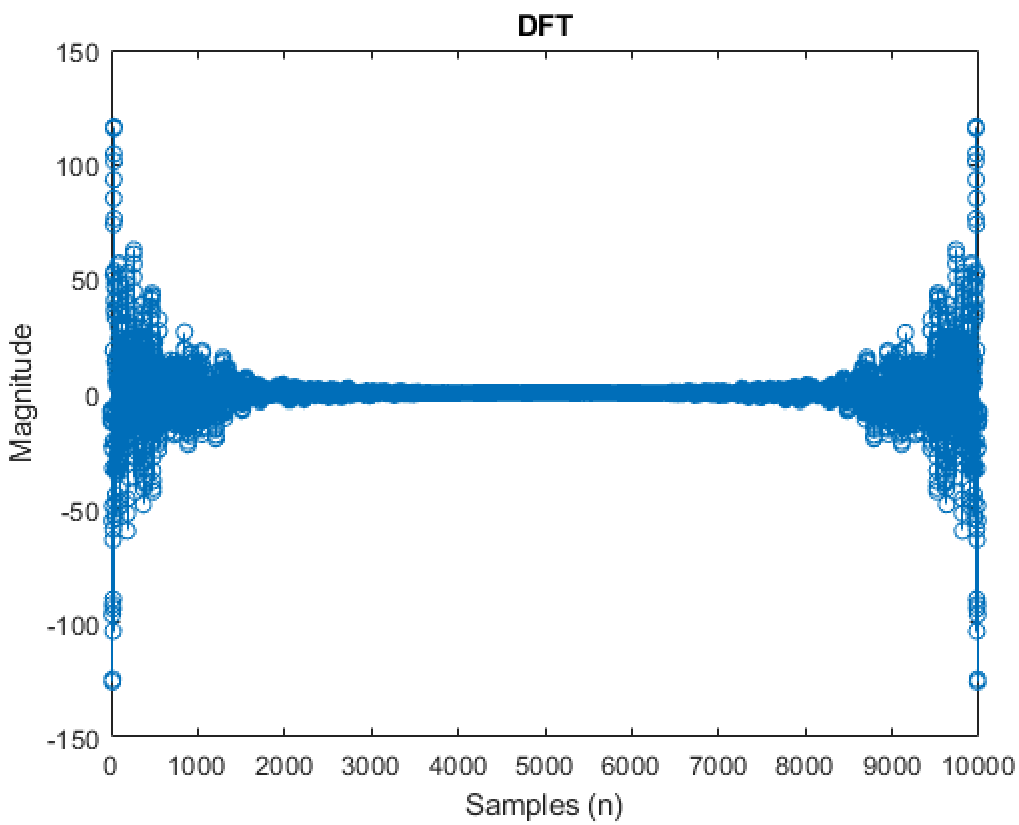
Question 4a

Use DFT 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 DFT. Display the result with the disp function.

```
TSTART = tic;
xd = DFT(x(1:10000));
figure
stem(0:length(xd)-1, xd);
xlabel('Samples (n)');
ylabel('Magnitude')
title('DFT')
T = toc(TSTART);

fprintf('Elapsed time in seconds for DFT (10000 samples): %d seconds.
\n', T);
disp(T);
```

```
Warning: Using only the real component of complex data.
Elapsed time in seconds for DFT (10000 samples): 1.148930e+00 seconds.
    1.1489
```



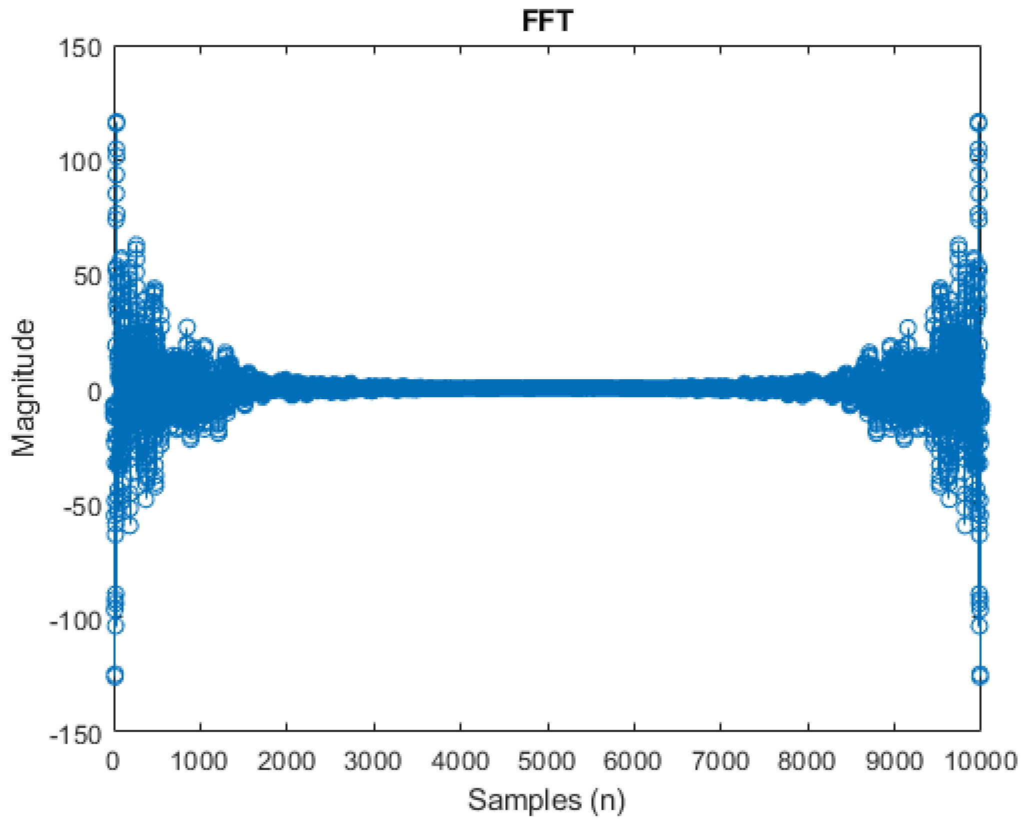
Question 4b

Use fft (read help fft) to plot the magnitude of the DFT of only the first 10000 samles of the audio. Use tic and toc to measure the length of time it takes to compute the DFT. Display the result with the disp function.

```
TSTART = tic;
figure
xf = fft(x,10000);
stem(0:length(xf)-1, xf);
xlabel('Samples (n)');
ylabel('Magnitude')
title('FFT')
T = toc(TSTART);

fprintf('Elapsed time in seconds for FFT (10000 samples): %d seconds.
\n', T);
disp(T);
```

```
Warning: Using only the real component of complex data.
Elapsed time in seconds for FFT (10000 samples): 8.707840e-02 seconds.
    0.0871
```



Question 4c

Answer in your comments: Are there any differences in the results? If so, why?

```
% There did not seem to be any difference in magnitude.
```

Question 4d

Answer in your comments: How much faster is the FFT algorithm compared with the DFT in this scenario?

```
% The FFT is roughly 15-20 times faster than DFT.
```

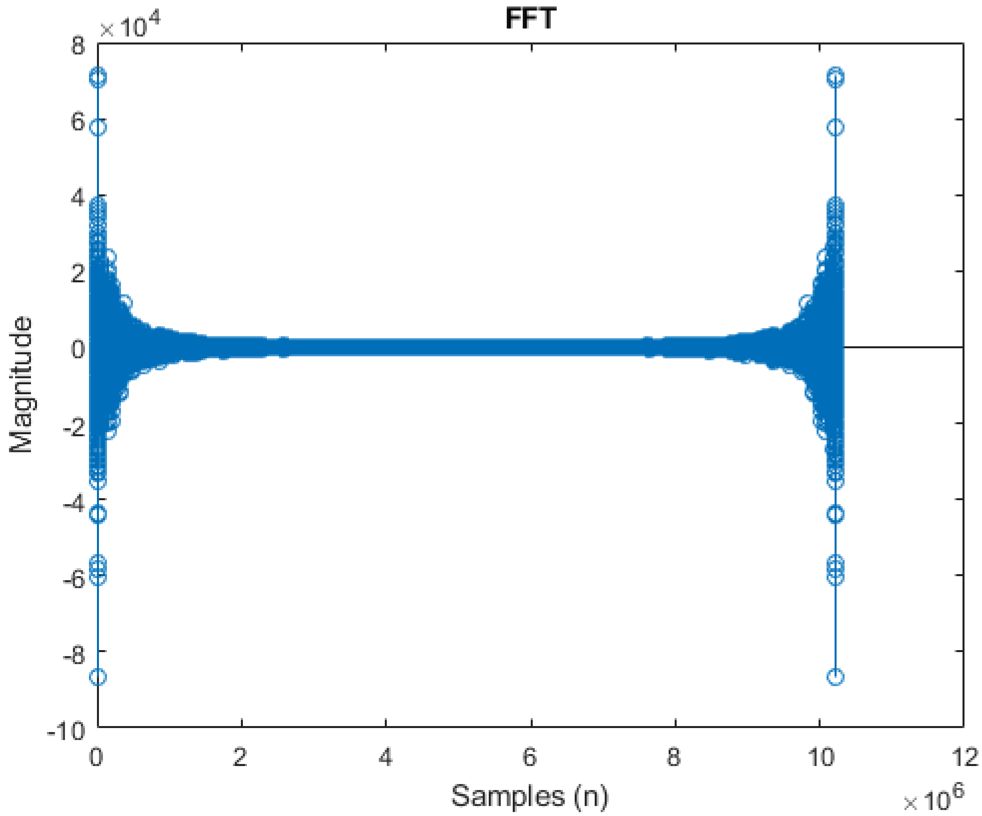
Question 4e

Now use fft to plot the magnitude of the DFT of the ENTIRE audio signal.

```
TSTART = tic;
figure
xf = fft(x);
stem(0:length(xf)-1, xf);
xlabel('Samples (n)');
ylabel('Magnitude')
title('FFT')
T = toc(TSTART);

fprintf('Elapsed time in seconds for FFT (entier song): %d seconds.
\n', T);
disp(T);
```

```
Warning: Using only the real component of complex data.
Elapsed time in seconds for FFT (entier song): 4.144093e-01 seconds.
    0.4144
```



Functions provided for the lab

```
function H = DTFT(x,w)
% DTFT(X,W)  compute the Discrete-time Fourier Transform of signal X
% acroess frequencies defined by W.

    H = zeros(1, length(w));
    for nn = 1:length(x)
        H = H + x(nn).*exp(-1j*w.*(nn-1));
    end

end

function X = DFT(x)
% DFT(x)  compute the N-point Discrete Fourier Transform of signal x
% Where N is the length of signal x
    N = length(x);
    w = ((2*pi)/N)*(0:N-1);
    X = zeros(1, length(w));

    for nn = 1:N
        X = X + x(nn).*exp(-1j*w.*(nn-1));
    end
end
```

```
function x = IDFT(X)
% IDFT(x)  compute the N-point Inverse Discrete Fourier Transform of
 signal
% X where N is the length of signal X
    N = length(X);
    w = ((2*pi)/N)*(0:N-1);
    x = zeros(1, length(w));
    for nn = 1:length(X)
        x = x + X(nn).*exp(1j*w.*(nn-1));
    end
    x=x/N;
end
```

Published with MATLAB® R2020a