DARA Training 2024
Digital Signal Processing Exercises
Lecturer: Mr. J. Heystek Grobler

(This document is from Prof TG Swart from the University of Johannesburg, Department of Electrical and Electronic Engineering Science, and acknowledges that it from the course Signals and Systems 3A)

The purpose of this exercise is to use Matlab to investigate the Fourier series and signal processing.

Question 1:

Write a Matlab function, ***four_ser***, that takes as input a vector of complex exponential Fourier series coefficients (Xn), a fundamental frequency (f0) and a vector of time (t). The function should calculate the output signal using the truncated Fourier series, where

$$y(t) = \sum_{n=-L}^{L} X_n e^{j2\pi f_0 n t}$$

The length of the vector Xn should be 2L+1, and L can be calculated from the length of Xn by using the Matlab function length. Remember that Matlab only works with positive indices, whereas the Fourier series representation expects negative and positive n-values. Write your function in such a way that you take this limitation into account.

Set A = 1 and T_0 = 8 which results in f_0 = 1/8.
Select L = 4

The coefficient equations will have:
$X_{-4} = -A/15\pi$
$X_{-3} = 0$
$X_{-2} = -A/3\pi$
$X_{-1} = -iA/4$
$X_{-0} = A/\pi$
$X_1 = -iA/4$
$X_2 = -A/15\pi$
$X_3 = 0$
$X_4 = -A/15\pi$

```
Xn = [-A/(15*pi) 0 -A/(3*pi) 0.25*j*A A/pi -0.25*j*A -A/(3*pi) 0 -A/(15*pi)];
f0 = 1/T0;
t = 0:0.01:4*T0;
y = four_ser(Xn, f0, t);
plot(t,y);
```

If the function works correctly, the figure should show a signal that is starting to look like a half-rectified sine wave.

Question 2:

Use the sound wave "horn.mat". The sampling rate, fs, for this sound is 11025 Hz. Use "Load" to load the file. Matlab will read the sound values into the variable data and the sampling rate into the variable fs. Now play the sound using the sound function in Matlab (use help sound if you are not sure how to use the sound function). Plot a small section of the sound file to show that the sound is periodic, then use this to estimate the value of the fundamental period, T0, thereby getting an estimate for the fundamental frequency, f0. Report these values.

Use the following code to view the amplitude spectrum of this sound:

```
N = 16384;                     % size to be used for calculating fft
x = fft(data, N);              % determine the fft of the signal
xmag = abs(x);                 % determine the magnitude of the signal
xang = angle(x);               % determine the phase of the signal
f = fs * (0:N/2)/N;            % scale to the correct frequency axis
plot(f, xmag(1:(N/2)+1));      % plot the magnitude for positive frequencies
xlabel('f [Hz]')
ylabel('Magnitude of X(f)')
figure;
plot(f, xang(1:(N/2)+1));      % plot the phase for positive frequencies
xlabel('f [Hz]')
ylabel('Phase of X(f)')
```

You should now see the amplitude spectrum of the sound signal (the double sided amplitude spectrum was calculated, but only the positive side is plotted). We are interested in the peaks in the spectrum. Use Matlab's "data cursor" tool to read the magnitude and frequency values for the first 10 peaks, and write these down. Use the frequency values to determine an estimate for the fundamental frequency, f_0. Note that the fundamental frequency of the sound is obtained by looking at the spacing between the peaks in the amplitude spectra.