Homework Assignment #4

Due Date: Friday, 25 October, 2013 at 11:59pm (MDT)

What to Submit

Please submit **TWO** appropriately-named **pdf files** via Learning Suite: (1) one containing your solutions for the WRITTEN exercises and (2) the other for the PROGRAMMING exercises (include the necessary plots and all code that you've written to complete the exercises).

WRITTEN EXERCISES

1) Show that the Fourier Transform is linear. (You'll need to use the Fourier Transform integral to show this.)

2) Compute by hand the DFT (Discrete Fourier Transform) of the signal f = [1,1,1,1]. Make sure to show your work.

PROGRAMMING EXERCISES

Please complete the following exercises in your programming environment of choice (nominally MATLAB or Python/NumPy):

PART A: Implement the Discrete Fourier Transform

1) You will first implement the Discrete Fourier Transform on your own. It's not as complicated as it seems, just follow what we did in class (there's pseudocode).

   Hint: You can verify that you get the same results as the `fft()` routine in Matlab or Python (it's in the `numpy.fft` module and note that Python uses `j` instead of `i` for the imaginary axis, the book does this too). In both Matlab and Python there will be a factor of M difference (they don't do the 1/M normalization).

PART B: Simple sines and cosines

1) Generate and display a one-dimensional sine wave with 128 samples that makes 8 cycles over this length:

   f[t] = sin(2 pi s t / N) where s=8 and N = 128

   (This means that you should write your own code that will create an array of length N and fill in each element of the array with the appropriate value for f[t] as t goes from 0 to N-1.)

   - Apply the Fourier Transform to this sinusoid and plot the Real, Imaginary, Magnitude, and Phase parts of the result.

- Explain each of the four parts. (Be careful to pay attention to the range of values displayed and to be careful interpreting what is plotted for approximately 0 divided by approximately 0.)

2) Do the same thing for a cosine wave.
3) Do the same things again, but this time for the sum of your sine and cosine functions.
4) Play with the relative weightings of the sine and cosine parts and see what happens. Report your findings in your write-up.

PARTS C-F use signals contained in these files (***for MATLAB***) and (***for Python***). In Matlab, you simply call `load('HW4_signals.mat')` and you'll then access the different signals of the variable `signals` by field (e.g. `signals.rect`). In Python, you'll want to first `import pickle` and then you load the variable signals with `signals = pickle.load(open('HW4_signals.pkl','r'))` and then access the different signals as `signals['rect']`, etc.

PART C: Rect Function

1) Compute the Fourier Transform of the Rectangular Pulse (*`rect`*) and plot the magnitude and Power Spectrum.
2) Explain (mathematically) in your write-up why you get this result. Use what you know about the Fourier Transform of a rectangular pulse.

PART D: Gaussian Function

1) Compute the Fourier Transform of the Gaussian (*`gaussian`*) and plot the magnitude and Power Spectrum.
2) Show (mathematically) in your write-up why you get this result.

PART E: Frequency Analysis

1) Write a program that uses the Fourier Transform to identify the *n* most significant frequencies in a signal (other than the constant component). Your program should read in the signal as input and print out the dominant frequencies as output. Hint: remember that a particular frequency will be echoed in both the positive and negative frequencies of the Fourier Transform, so be careful not to inadvertently report the same frequency twice.
2) Test your program to find the three most significant frequencies in the 1-D signal found in (*`test`*). You should also plot this signal to see what it looks like in the time domain, but you don't need to include this plot in your write-up.

PART F: Discover the Transfer Function *H(u)*

Let the input *f(t)* = the Rectangular Pulse (`rect`) and the output *g(t)* = the output (`output`).

1) Find the transfer function *H(u)* of the system used to produce this output and plot the magnitude portion of it.
2) Apply the inverse Fourier Transform to the transfer function and plot the real part of the results. Does this bear any relationship to the input and output signals?

   Hint: be careful of division by 0! (Treat 0/0 as 0.)

Programming Exercise Write-up

Please prepare a written write-up (preferably submitted as a PDF) which includes the following:

- All code that you wrote for the exercises (parts B-F).
- The relevant plots and answers to questions from parts C-F.
- A brief explanation of what you did, any challenges you encountered, things that were difficult, unclear, etc.
- How long did the different parts (written, programming) of the assignment take you?