Note: In all problems, include all your plots and code files. It might help to use MATLAB live scripts.

Problem 1 *FFT Basics in MATLAB* 10 points

Use MATLAB's `fft` routine to find the spectrum of a waveform consisting of two sinusoids with frequencies of 200 and 400 Hz. Use 512 samples for each waveform and sampling rate of 1 kHz. Take the Fourier transform of the waveform and plot the magnitude of the full spectrum (see Example 3.10 of CSSB). Remember to generate a frequency vector so the horizontal axis is properly scaled. Use `fftshift` if necessary.

Problem 2 *FFT and noise levels* 15 points

Repeat the same exercise as in Problem 1 but with noise added to the signal. Use the routine `sig_noise` (download from Ed's resources tab) to add noise at different levels and compare.

(a) First add noise to both the signals so that the SNR is -8 dB. Generate the frequency spectrum as in Problem 1 but ignoring the DC component and showing only the positive frequency spectrum. Display this as `subplot(1, 2, 2)`.

(b) Now add noise to both the signals so that the SNR is -16 dB. Generate the frequency spectrum as above but now in `subplot(1, 2, 2)`.

Compare them side by side and remark on whether it is possible to still distinguish different frequencies.

Problem 3 *Revisit an old friend* 10 points

Recall the pesky plot from the last homework (Problem 2) that required two pages of calculations? Take the `fft` of that and plot the magnitude spectrum. Assume that the period, $T = 1$ second and and the sampling rate is 500 Hz. The spectrum of this curve falls off rapidly, so plot only the first 20 points plus the DC term using `scatter` plot.

Problem 4 *A cardiac waveform* 10 points

Read Example 3.12 of CSSB and note how they make use of the `unwrap` command in MATLAB. Take the Fourier Transform of the `ecg` variable in the file `ECG_60s.mat`. The sampling frequency used is 250 Hz. Plot both the magnitude and phase (unwrapped) spectrum up to 20 Hz and do not include the DC term. Find the average heart rate using the strategy found in Example 3.12.

Hint: Due to the periodic nature of the DFT, the magnitude spectrum will have multiple peaks. However, the largest low frequency peak will correspond to the cardiac cycle.

Problem 5 *Chirp or the sweep function* 10 points

Read about the chirp waveform here. Download the wave file from the bottom of that page. Use MATLAB's `audioread` function to read it into memory. Plot the magnitude spectrum of this chirp wave (the sampling frequency is 2 kHz). Resample the wave at 1 kHz and plot the magnitude spectrum side by side with the previous plot. What changed?

Problem 6 *Power spectrum I* 15 points

Use MATLAB routine `sig_noise` to generate two arrays, one 128 points long and the other 512 points long. Include two closely spaced sinusoids having frequencies of 320 and 340 Hz with an SNR of -12 dB. Calculate and plot the power spectrum. Repeat the execution of the program several times and remark on whether the frequencies can be distinguished.

Hint: Note that `sig_noise` assumes the sampling frequency is 1 kHz. The call to `sig_noise` will look like: `sig_noise([320, 340], -12, N)` where `N` is either 128 or 512.

Problem 7 *Averaged power spectrum* 15 points

Read Example 4.5 from CSSB in detail. Download the `welch.m` routine from the CSSB website's or from the lecture notes. Load the file `eeg_data.mat` that contains EEG data at a sampling frequency of 50 Hz. Analyze these data using the unaveraged power spectral technique mentioned in the lecture notes and the windowed averaging technique using the `welch` routine. Find a segment/window length that smooths the background spectrum, but still retains any important spectral peaks. Use a 99% overlap.

Problem 8 *A question about bandwidth* 15 points

Read example 4.7 in CSSB in detail. Download the file `broadband2.mat` which is sampled at 1 kHz. Find the effective bandwidth of the signal `x` using methods in that example. Find the power spectrum using `welch` routine and a window size ranging between 50 to 200 points and the maximum overlap. Find the window size that appears to give the best estimate of the bandwidth.

Hint: Window sizes that produce the best looking power spectrum may not result in the most accurate estimate of the bandwidht. Can you explain why?