1. The radix-2 FFT we discussed in class can roughly summarized as follows:

function $y = \mathbf{fft}(x, n)$
  if $n = 1$
    $y \longleftarrow x$
  else
    $m \longleftarrow n/2$
    $\omega \longleftarrow \exp(-2\pi i/n)$
    $\Omega \longleftarrow \mathrm{diag}(1, \omega, \ldots, \omega^{m-1})$
    $z_T \longleftarrow \mathbf{fft}(x(0:2:n-1), m)$
    $z_B \longleftarrow \Omega\mathbf{fft}(x(1:2:n-1), m)$
    $y \longleftarrow \begin{bmatrix} z_T + z_B \\ z_T - z_B \end{bmatrix}$
  end

Implement this radix-2 FFT scheme into a computer program. Employ your program in the following applications:

- A common use of the FFT is to find the frequency components of a signal buried in a noisy time domain signal. Follow the steps below to see how your code can help for spectral analysis.
  (a) Consider data sampled at $1000Hz$. Start by forming a time axis for the data, running from $t = 0$ until $t = .255$ in step of 1 *millisecond*.
  (b) Form a signal containing $50Hz$ and $120Hz$, that is,
  $$x(t) = \sin(2\pi * 50t) + \sin(2\pi * 120t).$$
  (c) Add some random noise from a normal distribution with a standard deviation of 2 to produce a noisy signal, that is,
  $$y(t) = x(t) + 2 * normal(t).$$
  (d) Plot the graphs of $x(t)$ and $y(t)$. Is it easy to identify the frequency components from looking at the original signal and the noisy signal?
  (e) Find the discrete Fourier transform Y of the noisy signal y.
  (f) The power spectral density $P(y)$ is a vector where $P_i(y) := Y_i * \overline{Y}_i$. The quantity $P(y)$ is a measurement of the energy at various frequencies.
  (g) Form a frequency axis, running from $0Hz$ to $1000Hz$ in step of $\frac{1000}{256}Hz$.
  (h) Plot the power spectral density against the frequency axis. What have you observed? Give some mathematical justification why it is what you have observed.

(i) Repeat the above experiment with some other different signal functions or different magnitude of noises.

- Another application of the Fourier analysis is for smoothing data. Following the steps below to learn this application.

  (a) Let the function
  $$y(x) = \frac{x(400 - x^2)}{400}$$
  represent the actual values of a certain physical phenomenon. Plot the graph of $y(x)$ over the interval $[0, 20]$.

  (b) Suppose some random errors from a normal distribution has contaminated $y(x)$ to become $z(x)$. Plot the graph of $z(x)$.

  (c) Suppose the function $z(x)$ is extended to an odd function of period 40. Use the FFT you have developed to compute the coefficients of the sine Fourier series of $z(x)$.

  (d) Do you observe some rapidly decreasing terms in the Fourier series?

  (e) Create a new function $w(x)$ by truncating the rapidly decreasing terms in the Fourier series of $z(x)$. Plot the graph of $w(x)$.

  (f) Plot the errors $|y(x)-z(x)|$ and $|y(x)-w(x)|$ against x over the interval $[0, 20]$.

  (g) Repeat the same procedure except that $z(x)$ is extended to an even function. Compare the performance of the two extensions.

  (h) Repeat the experiment for some other functions.

2. The following is a practice on filter design using the Signal Processing Toolbox available from MATLAB:

   (a) From within MATALB, start **sptool**.

   (b) Use the **Import** option under the **File** menu of sptool to load from disk the data file **voice.mat** into sptool. The file contains two variables: Assign the variable **sch** to **Data**, calling the signal **sig1** and the variable **Fs** to the **Sampling Frequency**.

   (c) Use the **View** option under the column **Signals** to see a plot of the waveform **sig1**. You can listen to **sig1** by choosing **Play** under **Options** of the waveform window. The file **sig1** is a woman speaking along with an annoying tone-like signal due to a narrow-band FM noise signal.

   (d) Use the **Create** option under the column **Spectra** of the sptool window to see the spectrum of **sig1**. Use the **FFT** option under **Method** in the spectrum window.

   (e) Use the mouse to move cursor 1 (the solid line) or cursor 2 (the dashed line) to the large peak in the middle of the spectrum. The frequency of the peak will be shown on the appropriate x coordinate on the right. The peak should be in the vicinity of $2000Hz$. Examine the peak. This is due to the FM noise. From the spectrum, decide the specifications (in terms of cutoff frequencies and attenuation)

of a **bandstop filter** to filter the signal **sig1** with the intention to remove the FM noise.

(f) Return to the sptool window, chose **New Design** under the column **Filters** to design a filter that meets your specifications. Choose any type of **Algorithm** that strikes your fancy. You can use the **Overlay Spectrum** option in the filter window to overlay the signal's spectrum on the filter's frequency response. This will allow you to compare the filter transfer against the signal spectrum.

(g) Hit **Apply** button to filter **sig1** using the filter you designed to create **sig2**. Listen to the new signal and examine its spectrum see how well the FM noise has been removed.

(h) In your report, give the specifications of your designed filter.