1. Sketch the spectrum of the signal

$$x(t) = \sin(24\pi t)\left(\sum_{k=-3}^{3} \frac{1}{1+j2\pi k} e^{jk2\pi t}\right) .$$

(Suggestion: use Euler's Formula on the sine term.)

2. The signal

$$x(t) = \cos(2\pi(440)t + \pi/3)\cos(2\pi(880)t)\sin(2\pi(110)t)$$

turns out to be periodic.

(a) Find its fundamental frequency and fundamental period.
(b) Find its Fourier series without integrating. (Suggestion: go "full Euler" on it). Give formulas for the Fourier coefficients a_k for all integers k. Most of them will be zero.
(c) Sketch the spectrum of $x(t)$.

3.

(a) Sketch over the range $-7 \le t \le 7$ the sawtooth signal $x(t)$ given for t in the interval $[0,1)$ by the formula $x(t) = t$.
(b) MATLAB has a built-in function `sawtooth(t)` that gives you a sawtooth of period 2π with max/min amplitudes ± 1. Find a, b, and $c > 0$ such that you get $x(t)$ in part (a) from the MATLAB expression `a+b*sawtooth(c*t)`.
(c) Find the Fourier coefficients for $x(t)$ from part (a). You can probably guess what a_0 is. To find a_k for $k \neq 0$, you'll have to integrate by parts (it's easy).
(d) For each $k > 0$, pair the $\pm k$-terms in the Fourier series

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk2\pi t}$$

to obtain a series expansion

$$x(t) = a_0 + \sum_{k=1}^{\infty} b_k \sin(k2\pi t) .$$

4. From the book, Problem P-3.12.

5. We'll build off this problem during the upcoming lab, so please pay attention to it. Suppose you have a periodic continuous-time signal and you want to use MATLAB to compute its Fourier coefficients. MATLAB works only with discrete objects, so the computation will be approximate in any event, but how might you set up Fourier-coefficient computation so a discrete matrix/vector operation implements it, even approximately? The formulas for Fourier coefficients are continuous-time integrals.

(a) Suppose $T > 0$ and $g(t)$ is some continuous-time function. Let N be a large positive integer and let $T_s = T/N$ and $f_s = 1/T_s$. I claim that

$$\frac{1}{T}\int_0^T g(t)dt \approx \frac{T_s}{T}\sum_{n=0}^{N-1} g(nT_s) = \frac{1}{N}\sum_{n=0}^{N-1} g(nT_s) ,$$

and the approximation is better the larger N is. Explain in words what's going on here. Remember Riemann approximations to integrals? Staircase approximations to $g(t)$? Integral is the area under the curve? Maybe draw a picture.

(b) Suppose $x(t)$ is a continuous-time periodic signal with fundamental frequency f_o in Hertz and fundamental period $T_o = 1/f_o$. Let a_k be the kth Fourier coefficient for $x(t)$. We know that

$$a_k = \frac{1}{T_o}\int_0^{T_o} x(t)e^{-jk2\pi f_o t}dt \; .$$

Let $T_s = T_o/N$, where N is large, and let $f_s = 1/T_s$. Deduce from (a) that

$$a_k \approx \frac{1}{N}\sum_{n=0}^{N-1} x(nT_s)e^{-jk2\pi f_o nT_s} \approx \frac{1}{N}\sum_{n=0}^{N} x(nT_s)e^{-jk2\pi f_o nT_s} \; ,$$

where the second approximation just adds an Nth term to the sum that makes negligible contribution to a_k (why?) when N is large.

(c) Suppose $K > 0$ is an integer, and we want to compute a_k for $-K \le k \le K$. In MATLAB, you want to set up computations as vector/matrix operations and minimize `for` loops. So let's set up this computation as follows:

- We want to solve for a $(2K+1)$-dimensional row vector whose entries are the a_k, arrayed left to right in order of increasing k. Call that vector X.
- Let x be the $(N+1)$-dimensional row vector whose entries left to right are $x(nT_s)$ arrayed in order of increasing n, $0 \le n \le N$.
- I claim that you can construct a $(2K+1 \times N+1)$ matrix W so that $X^T = Wx^T$. Think of indexing the rows of W top-to-bottom using k-values running from $-K$ to K. What will the (k,n) entry of the matrix W be? Reconcile the expression

$$a_k = \sum_{n=0}^{N}[W]_{kn}x(nT_s) \quad \text{for} \quad -K \le k \le K$$

with

$$a_k = \frac{1}{N}\sum_{n=0}^{N} x(nT_s)e^{-jk2\pi f_o nT_s} \quad \text{for} \quad -K \le k \le K$$

to figure this out.

- Note that the (k,n)-element of W is of the form $(\text{something})^{-kn}/N$. What is "something"? Think about how to build W using one line of MATLAB that employs matrix multiplications, elementwise powers, etc. Here's a hint. The MATLAB expression `[-1:1]'*[0:3]` performs the computation

$$\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}\begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 0 & -1 & -2 & -3 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix} .$$

- Think about how to write a MATLAB function that takes as inputs T_o, the $(N+1)$-dimensional row vector of samples of a T_o-periodic signal $x(t)$, with samples taken every T_o/N seconds on the interval $[0, T_o]$, and $K > 0$, and whose output is the row vector of (approximate) Fourier coefficients a_k of $x(t)$ with $-K \le k \le K$.
- Note that your function works for computing Fourier coefficients of any $x(t)$ with fundamental period T_o sampled at frequency $f_s = T_o/N$ between 0 and T_o.