Wherever relevant, use $\alpha = 1 + \text{mod}(x, 3)$, where x is the last three digits of your registration number. Wherever relevant, plot signals with normalised axes, with an appropriate resolution for time and with appropriate labels and legends.

Problem 1. (Sampling and frequency-domain aliasing)

1. Plot the following waveforms in one single plot for $1/\alpha$ second duration:
   (a) Cosine function with amplitude α and frequency 5α Hz.
   (b) Cosine function with amplitude $\alpha/2$ and frequency 6α Hz.
   (c) Cosine function with amplitude $\alpha/4$ and frequency 10α Hz.
2. Plot the summation of all the three functions in another figure.
3. Sample the cumulative signal with the following sampling frequency and plot the discrete-time waveforms (using stem) in another figure (as subplots).
   (a) $F_s = 14\alpha$ samples/second.
   (b) At the Nyquist rate of the signal.
   (c) At a sampling rate such that 6α Hz is aliased to 3α Hz.
4. Perform a linear interpolation (you can use plot command which performs this) on the sampled responses for the different sampling rates. Do you observe any difference between the reconstructed waves for the three different sampling rates? Comment on the same.
5. Draw the energy density spectrum for 3(a), 3(b) and 3(c) using FFT. It is known that the case of 3(a) and 3(c) will have an alias. Which frequency(ies) is/are aliased and to what value?

Problem 2. (Generating digital music)
Generate a sequence containing the tones corresponding to *"Do Re Mi Fa So La Ti Do"* as done in the previous experiment. Append the signals together and save the resulting signal as a single *.wav* file. Use different sampling rates to generate the sequence and listen to the audio and comment on the differences.

Problem 3. (Resampling)
Load *Track00α.wav* and generate different *.wav* files with several values of the sampling rate (for example, half the original sampling rate, 1/3rd of the original sampling rate etc.) and see the effect of this different sampling rate on the audio. Try: It is easy to achieve downsampling. Is it possible to upsample the signal? If yes, suggest a method such that the frequency content in the signal does not change.

==========