Wherever relevant, use $\alpha = 1 + \mathrm{mod}(x, 3)$, 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. (Computing DFT)
Generate a unit amplitude sinusoidal signal of frequency 15α Hz for a duration of 2 seconds with a sampling rate of 120 samples/sec.

1. Plot magnitude of the DFT of the first 120 samples of the signal against frequency in Hertz.
2. On the same figure, plot the magnitude of the DFT of the first 130 samples of the signal. Report your observations.
3. Report $N \neq 120$ such that the DFT of the first N points of the signal matches with the DFT of the first 120 samples of the signal.

Problem 2. (Resolution of DFT)
Generate a signal $x_a(t) = 0.1\sin(B\pi t) + \cos(B\pi t)$ and sample it with a rate of 200 samples/sec for a duration of 10 seconds. Plot the DFT of the signal for (i) 215 samples, (ii) 415 samples, (iii) 1115 samples (iv) 1515 samples (v) 1915 samples, in 5 separate figures. Report your observations.
For $\alpha = 1$; $A = 120, B = 126$, for $\alpha = 2$; $A = 140, B = 146$, for $\alpha = 3$; $A = 160, B = 166$.

Problem 3. (Resolution of DFT with windowing)
Repeat Problem 2 by windowing the time-domain signal using: a Hamming window if $\alpha = 1$ or a Hanning window if $\alpha = 2$ or a Blackman window if $\alpha = 3$, i.e., multiply the signal with the corresponding window of the same length, and then compute the DFT. Report your observations. Compare with the results obtained in Problem 2.

Problem 4. (Frequency estimation using windowing)
Estimation of the frequency components using time-domain windowing technique: Load the file `Exp4Data`α`.txt`. This has 500 samples of a dual tone. Express the values of frequency in terms of F_s (up to 3 decimal places).

1. Include time-domain windowing operation using Hamming window. Estimate the two frequency components from these results. *Help:* It is a common practice to pad zeros to improve the "interpolation" of the plot and take a higher-point DFT, say 10000 samples. This can be done by doing `fft(x,10000)` and normalizing the x-axis by 10000).
2. Repeat the above using a rectangular window.