Chapter 2  PROGRAMMING EXERCISE

Implementing the DFT in C++

The discrete Fourier transform (DFT) is a function that takes audio data in the time domain and transforms it to the frequency domain. Given a vector of $N$ audio samples (called $f$), the DFT returns $N/2$ valid frequency components (stored in output vector $F$). The frequency components are complex numbers whose magnitudes are distributed between 0 and the Nyquist frequency. The equation for the DFT is this:

$$F_n = \frac{1}{N} \left( \sum_{k=0}^{N-1} f_k \cos \frac{2\pi nk}{N} - if_k \sin \frac{2\pi nk}{N} \right)$$

Equation 1 The discrete Fourier transform (DFT)

Your assignment is to implement the DFT using based on the equation above. Note that a summation can be implemented using a loop and a “running total.” You can separate the running totals of the real and imaginary parts of the equation.

You may want to verify your implementation by running your DFT on an audio segment with known frequency components, like a C major (with notes C, E, and G). Then compare your results with the results you get running MATLAB’s `fft` function of the audio segment.