Exercise 1.1 Let \( x = (x_0, x_1, \ldots, x_{n-1}) \in \mathbb{R}^n \) and let \( y = (y_0, y_1, \ldots, y_{n-1}) \in \mathbb{C}^n \) be such that \( y_{n-j} = y_j^* \), for \( j = 1, \ldots, n-1 \). (Here \( z^* \) is the conjugate of \( z \in \mathbb{C} \), i.e., if \( z = a + ib \), where \( a, b \in \mathbb{R} \), then \( z^* = a - ib \).)

Exercise 1.2 Let \( x = (f(0), f(\frac{1}{12}), \ldots, f(\frac{31}{12})) \in \mathbb{R}^{32} \), where \( f(x) = \sin(13(2\pi x)) + 7\sin(3(2\pi x) + \frac{\pi}{4}) + 5\cos(7(2\pi x)) \). Compute \( DFT(x) \). (Hint: no complicated calculations are necessary. Use the relations \( \cos x = \frac{1}{2}(e^{ix} + e^{-ix}) \), \( \sin x = \frac{1}{2i}(e^{ix} - e^{-ix}) \), and the fact that Fourier basis is an orthonormal basis.)

Exercise 1.3 The chirp transform of a vector \( (x_0, x_1, \ldots, x_{n-1}) \in \mathbb{C}^n \), with respect to an arbitrary complex number \( z \in \mathbb{C} \), is defined as follows: \( y_k = \sum_{j=0}^{n-1} x_j z^{jk} \), for \( k = 0, 1, \ldots, n-1 \).

(a) Show that the DFT is a special case of the chirp transform. (For which \( z \)?)

(b) Use the relation \( y_k = z^{k^2/2} \sum_{j=0}^{n-1} (x_j z^{j/2})(z^{-(k-j)^2/2}) \) to express the chirp transform as a convolution. (Use caution. In the sum given, \( j \) ranges from 0 to \( n-1 \), for every value of \( k \), while this is not the case for the non-cyclic convolution.)

(c) Show that the convolution of two vectors of length \( n \) can be computed in \( O(n \log n) \) time for every value of \( n \), not necessarily a power of 2.

(d) Use the previous results to show that the DFT of a vector of length \( n \) can be computed in \( O(n \log n) \) time for any value of \( n \).

Exercise 1.4 The cross-correlation of \( x = (x_0, x_1, \ldots, x_{n-1}) \) and \( y = (y_0, y_1, \ldots, y_{m-1}) \) is defined to be \( z = (z_{n-1}, \ldots, z_0) \) such that \( z_k = \sum_j x_{j+k}y_j \). (The sum here is over \( j \) such that \( 0 \leq j + k < n \) and \( 0 \leq j < m \).) Show that the cross-correlation of two vectors of lengths \( n \) and \( m \), respectively, where \( m \leq n \), can be computed in \( O(n \log m) \) time.

Exercise 1.5 Let \( T \) be a text of length \( n \) and let \( P \) be a pattern of length \( m \) over a finite and small alphabet \( \Sigma \). Let \( D \) be a \(|\Sigma| \times |\Sigma| \) matrix such that \( D(a, b) \) specifies the similarity or dissimilarity of \( a, b \in \Sigma \). For every \( k = 0, 1, \ldots, n-m-1 \) define \( d_k = \sum_{j=0}^{m-1} D(T[k+j], P[j]) \) to be the total pattern-text dissimilarity when the 0-th pattern character is aligned with the \( k \)-th text character.

(a) Show that \( d_k \), for \( k = 0, 1, \ldots, n-m-1 \), can be computed in \( O(|\Sigma| n \log m) \) time.

(b) Suppose now that \( \Sigma \subset \mathbb{Z} \), i.e., that each character is actually an integer, and that \( D(a, b) = ab \), for every \( a, b \in \Sigma \). How fast can the \( d_k \)’s be computed?

(c) Suppose that we again have \( \Sigma \subset \mathbb{Z} \) but this time \( D(a, b) = (a-b)^2 \), for every \( a, b \in \Sigma \). How fast can the \( d_k \)’s be computed?

Exercise 1.6 (a) Given a text \( T \) of length \( n \) and a pattern \( P \) of length \( m \) over the alphabet \( \{0, 1, \ldots, m-1\} \) such that each character appears in \( P \) no more than \( c \) times, describe an algorithm that computes an array \( M \) of length \( n \), where \( M[k] \) is the number of matches when \( P \) is aligned with \( T[k : k + m - 1] \). The running time of the algorithm should be \( O(nc) \). (Hint: For each character in \( T[i] \) in \( T \), increment entries in \( M \) to which this character contributes a match.)

(b) Combine the algorithm in (a) with an FFT-based algorithm to obtain an algorithm that computes the array \( M \) in \( O(n \sqrt{m \log m}) \) for any pattern \( P \) of length \( m \). (Hint: Treat separately characters that appear many times in \( P \) and characters that appear only a few times in \( P \).)