CLASSICAL AND QUANTUM COMPUTING
EXERCISE IX

Given a sequence

\[ r(0), r(1), \ldots, r(p). \]

Using this sequence we can form the \( p \times p \) matrix

\[
R := \begin{pmatrix}
    r(0) & r(1) & r(2) & \ldots & r(p-1) \\
    r(1) & r(0) & r(1) & \ldots & r(p-2) \\
    r(2) & r(1) & r(0) & \ldots & r(p-3) \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    r(p-1) & r(p-2) & r(p-3) & \ldots & r(0)
\end{pmatrix}.
\]

The matrix is symmetric and all the elements along the diagonal are equal. We assume that \( r(0) \neq 0 \) and \( R \) is invertible. Let

\[
A := \begin{pmatrix}
    a_1 \\
    a_2 \\
    \vdots \\
    a_p
\end{pmatrix}, \quad P := \begin{pmatrix}
    r(1) \\
    r(2) \\
    \vdots \\
    r(p)
\end{pmatrix}.
\]

Then the linear equation

\[ RA = P \]

can be solved using Durbin’s algorithm which is a recursive procedure. The algorithm is as follows:

\[
E(0) := r(0) \\
k_1 = r(1)/E(0) \\
k_i = \left[ r(i) - \sum_{j=1}^{i-1} a_j(i-1)r(i-j) \right] /E(i-1), \quad i = 2, 3, \ldots, p \\
a_i(i) = k_i, \quad i = 1, 2, \ldots, p \\
a_j(i) = a_j(i-1) - k_ia_{i-j}(i-1), \quad \begin{cases} i = 2, 3, \ldots, p \\ j = 1, 2, \ldots, i - 1 \end{cases} \\
E(i) = (1 - k_i^2)E(i-1), \quad i = 1, 2, \ldots, p
\]
After solving these equations recursively, the solution of the linear equation is given by

\[ a_j = a_j(p), \quad j = 1, 2, \ldots, p \]

Write a C++ or Java program that implements this algorithm. Test your program with the sequence

\[ r(0) = 1, \quad r(1) = 0.5, \quad r(2) = 0.2 \]

This problem plays a role in linear predictive coding.