Algorithm 1 Cooley-Tukey Fast Fourier Transform (FFT)

Require: Input data array x of length $N = 2^n$, where n is a non-negative integer.

Ensure: FFT result in array X of length N.

function FFT(x, N)
    if $N = 1$ **then**
        $X[0] \leftarrow x[0]$
    else
        Split x into even and odd parts:
        $x_{\text{even}}[k] \leftarrow x[2k]$ for $k = 0, 1, \ldots, N/2 - 1$
        $x_{\text{odd}}[k] \leftarrow x[2k + 1]$ for $k = 0, 1, \ldots, N/2 - 1$
        Compute FFT of even and odd parts:
        FFT($X_{\text{even}}, N/2$)
        FFT($X_{\text{odd}}, N/2$)
        Combine results:
        for $k = 0$ to $N/2 - 1$ **do**
            $X[k] \leftarrow X_{\text{even}}[k] + W_N^k \cdot X_{\text{odd}}[k]$
            $X[k + N/2] \leftarrow X_{\text{even}}[k] - W_N^k \cdot X_{\text{odd}}[k]$
        end for
    end if
end function