Complexity of algebraic computation

Homework assignment #3

1. Prove that \( \omega \) is an \( n \)th primitive root of unity if and only if the following holds.

\[
\omega^n = 1 \quad \text{and for each prime divisor } p \text{ of } n, \ \omega^{n/p} \neq 1.
\]

2. What is the number of primitive \( n \)th roots of unity?

3. For which positive integers \( n \) there are primitive \( n \)th roots of unity in the following fields?
   - The \( q \)-element finite field.
   - \( \mathbb{R} \).
   - \( \mathbb{Q}(i) \).

4. Prove that \( W(\omega^{-1}) \) results in a permutation of rows of \( W(\omega) \). Describe the permutation and determine for which values of \( n \) it is even and odd.

5. Modify the discrete Fourier transform algorithm for the case where \( n \) is a power of 3.

6. Modify the discrete Fourier transform algorithm for the case where \( n \) is of the form \( 2^\ell 3^m \).