max planck institut informatik

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Michael Sagraloff **Winter term 2014/15**

Computer Algebra

https://resources.mpi-inf.mpg.de/departments/d1/teaching/ws14/ComputerAlgebra

Assignment sheet 2 due: Wednesday, November 12

Exercise 1: Discrete Fourier transform (4 points)

Let $F = \mathbb{Z}/29\mathbb{Z}$.

1. Find a primitive 4th root of unity $\omega \in F$ and compute its inverse $\omega^{-1} \in F$.
2. Find the matrices DFT_ω and DFT_ω^{-1}, and check that their product is $4\, I_4$, where I_4 denotes the identity matrix in $F^{4\times 4}$.

Exercise 2 ($\star$): Existence of primitive roots in prime fields (4 bonus points)

Denote by $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ the finite field with p elements for some prime p, and let $n \in \{1, \ldots, p-1\}$. Show that $\mathbb{F}_p$ contains a primitive n-th root of unity if and only if n divides $p-1$, and conclude that the multiplicative group $\mathbb{F}_p^\times$ of $\mathbb{F}_p$ is cyclic.

Hints: 1. Use (without proof) **Fermat's little theorem:** *If $p \in \mathbb{N}$ is prime and $a \in \mathbb{Z}$ arbitrary, then*

$$a^p \equiv a \mod p.$$

In particular, if $a \in \{1, \ldots, p-1\}$, then

$$a^{p-1} \equiv 1 \mod p.$$

2. Let $q \in \mathbb{N}$ be a divisor of $p-1$ and $q = q_1^{e_1} \cdots q_r^{e_r}$ its prime factorization. For $a \in \mathbb{F}_p^\times$, we denote by $\mathrm{ord}(a) := \min\{i \in \mathbb{N}_{>0} : a^i = 1\}$ the order of a in $\mathbb{F}_p^\times$.

   Prove the following facts:

   - $\mathrm{ord}(a) = q$ if and only if $a^q = 1$ and $a^{q/q_i} \neq 1$ for $i = 1, \ldots, r$.
   - For each i, $\mathbb{F}_p^\times$ contains an element a_i with $q_i^{e_i} \mid \mathrm{ord}(a_i)$. Conclude that there is an element b_i with $\mathrm{ord}(b_i) = q_i^{e_i}$.
   - If $a, b \in \mathbb{F}_p^\times$ are elements of coprime orders, then $\mathrm{ord}(ab) = \mathrm{ord}(a)\,\mathrm{ord}(b)$.
   - $\mathbb{F}_p^\times$ contains an element of order q.

3. Keep on going if you cannot prove one of the hints. Depending on your background in algebra, you may also want to try out other ways to solve this exercise.

Exercise 3: Fast polynomial multiplication (4 points)

The complex number $\omega = e^{2\pi \mathrm{i}/8} \in \mathbb{C}$ is a primitive 8th root of unity. Let $f = 5x^3 + 3x^2 - 4x + 3$ and $g = 2x^3 - 5x^2 + 7x - 2 \in \mathbb{C}[x]$, and run the Fast Convolution algorithm on this example to calculate the coefficients of the product $f \cdot g$. Use the "classical" method for linear polynomials as the basecase multiplication. Use ω only symbolically, that is by the fact that $\omega^4 = -1$.

Exercise 4: Precision demand of FFT and Fast Convolution (4 points)

Provide the missing details in the proof of Theorem I.4.8. In particular, this means proving the following statements for FFT and Fast Convolution performed with fixed point interval arithmetic:

(a) Let $h \in \mathbb{C}[x]$ be a polynomial of degree less than n and with coefficients of absolute value less than 2^τ, and let $\omega \in \mathbb{C}$ be a primitive n-th root of unity. Show that an absolute precision of size $L + O(\tau + \log n)$ is sufficient to compute $\mathrm{DFT}_\omega(h)$ to an absolute error of less than 2^{-L} by means of FFT.

(Hint: Use that $|h(\omega^i)| \leq n \cdot 2^\tau$ for all $i = 0, \ldots, n-1$ when applying Lemma I.3.2.4 in Step 4 of the FFT.)

(b) Let f and $g \in \mathbb{C}[x]$ be polynomials of degree less than n and with coefficients of absolute value less 2^τ. Show that an absolute precision of size $L + O(\tau + \log n)$ is sufficient to compute $f \cdot g$ to an absolute error of less than 2^{-L} using the Fast Convolution algorithm.

(Hint: Prove that all values occurring in the Fast Convolution algorithm have an absolute value bounded by $2^{O(\tau + \log n)}$.)

Exercise 5: Fast integer division (4 points)

Provide proofs for Lemma I.4.10 and Theorem I.4.11.