max planck institut informatik

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Michael Sagraloff **Winter term 2017/18**

Computer Algebra

https://www.mpi-inf.mpg.de/departments/algorithms-complexity/teaching/winter17/comp-alg/

Assignment sheet 10 due: Monday, January 15

Exercise 1: Chinese remaindering for integers I (4 points)

1. Determine the smallest positive integer x satisfying

$$x \equiv 4 \mod 7, \qquad x \equiv 5 \mod 11, \qquad x \equiv 6 \mod 13.$$

2. How many integers x between 0 and 10^6 are common solutions of the following congruences?

$$x \equiv 3 \mod 13, \qquad x \equiv 4 \mod 15, \qquad x \equiv 5 \mod 17.$$

Exercise 2: Chinese remaindering for integers II (4 points)

The Chinese remainder algorithm allows us to recover a non-negative integer m, with $0 \le m < \prod_{i=1}^{k} p_i$, from the modular images $m \bmod p_i$. Describe a method to recover an integer $m \in \mathbb{Z}$ with $-\frac{1}{2}\prod_{i=1}^{k} p_i < m < \frac{1}{2}\prod_{i=1}^{k} p_i$ from the modular images $m \bmod p_i$ and give a proof.

Exercise 3: Small primes polynomial GCD

Give a randomized Las Vegas-method with expected runtime $\tilde{O}(n(n+L))$ for the computation of $\gcd(f,g) \in \mathbb{Z}[x]$ for $f, g \in \mathbb{Z}[x]$ of degree less than n and with integer coefficients of length less than L. That is, your algorithm must always give correct result but its actual running time might be worse than the the expected running time $\tilde{O}(n(n+L))$.

Hint: Use the fact that a randomly chosen prime is likely a lucky prime and that only $O(n+L)$ lucky primes are needed to recover $\gcd(f,g)$ from its modular images.

Exercise 4: Computing a small separating linear form for points on integer grids (4 points + 4 bonus points)

Let $X = \{x_1, \ldots, x_n\} \subset \mathbb{Z}$ be a set of integers with $|x_i| < 2^L$ and let d be an integer with $d \ge 2$. We consider the problem of computing a *separating linear form* of "small size" for X^d. More precisely, compute coefficients a_k such that the linear map

$$s_a : \mathbb{Z}^d \to \mathbb{Z}, \quad x \mapsto a_1 x_1 + \cdots + a_d x_d$$

is injective on X^d, that is

$$s_a(x_{i_1}, \ldots, x_{i_d}) = \sum_{k=1}^{d} a_k \cdot x_{i_k} \neq \sum_{k=1}^{d} a_k \cdot x_{k_k} = s_a(x_{j_1}, \ldots, x_{j_d})$$

for all pairs of distinct d-tuples $(x_{i_1}, \ldots, x_{i_d}) \neq (x_{j_1}, \ldots, x_{j_d})$ in X^d.
"Small size" means that the coefficients a_k of s_a have bitsize bounded by $\tilde{O}(d(\log L + \log n))$.

Give an algorithm which solves this task in a polynomial number (in n, d and L) of bit operations and provide a runtime analysis.

Hint: Determine primes $p_1, \ldots, p_d$ such that

$$(x_{i_1} \bmod p_1, \ldots, x_{i_d} \bmod p_d) \neq (x_{j_1} \bmod p_1, \ldots, x_{j_d} \bmod p_d)$$

for all distinct d-tuples $(x_{i_1}, \ldots, x_{i_d}) \neq (x_{j_1}, \ldots, x_{j_d})$ in X^d.