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 9 due: Monday, January 8

Exercise 1: Bounds on the intermediate values in the EEA (4 points)

We consider the Extended Euclidean Algorithm for integer polynomials $f, g \in \mathbb{Z}[x]$ of degree bounded by n and coefficients of absolute value less than 2^τ as presented in the lecture. As usual, define $s_i, t_i, \rho_i, r_i, q_i$ as

$$\begin{aligned} \rho_0 &:= \mathrm{LC}(f), \quad r_0 := \mathrm{normal}(f), \quad s_0 := \rho_0^{-1}, \quad t_0 := 0, \\ \rho_1 &:= \mathrm{LC}(g), \quad r_1 := \mathrm{normal}(g), \quad s_1 := 0, \qquad t_1 := \rho_1^{-1} \end{aligned}$$

and, for $1 \le i \le \ell$ (with ℓ the index such that $r_\ell \neq 0$ and $r_{\ell+1} = 0$),

$$\begin{aligned} q_i &:= r_{i-1} \operatorname{quo} r_i, \qquad \rho_{i+1} := \mathrm{LC}(r_{i-1} \operatorname{rem} r_i), \qquad r_{i+1} := \mathrm{normal}(r_{i-1} \operatorname{rem} r_i), \\ s_{i+1} &:= (s_{i-1} - q_i s_i)/\rho_{i+1}, \qquad t_{i+1} := (t_{i-1} - q_i t_i)/\rho_{i+1}. \end{aligned}$$

Show that there exist integers μ_i of length $O(n(\tau + \log n))$ such that $\mu_i \cdot \rho_i$ and $\mu_i \cdot q_i$ are integers (integer polynomials) of length (with coefficients of length) $O(n(\tau + \log n))$!

Proceed as follows:

1. Use that a comparable result has already been shown for s_i, t_i, and r_i!
2. Recall that

$$R_i = \begin{pmatrix} s_i & t_i \\ s_{i+1} & t_{i+1} \end{pmatrix} = R_0 \cdot \prod_{j=1}^{i} Q_j, \quad \text{where}$$

$$R_0 = \begin{pmatrix} s_0 & t_0 \\ s_1 & t_1 \end{pmatrix} \quad \text{and} \quad Q_j = \begin{pmatrix} 0 & 1 \\ \rho_{j+1}^{-1} & -q_j \rho_{j+1}^{-1} \end{pmatrix}$$

   and, in particular, $\begin{vmatrix} s_i & t_i \\ s_{i+1} & t_{i+1} \end{vmatrix} = (-1)^{i-1}(\rho_0 \cdots \rho_i)^{-1}$. Use these identities to derive a bound on the length of the numerator and denominator of ρ_i.
3. Prove that $f = q \cdot g$ with $f, g \in \mathbb{Z}[x]$ and $q \in \mathbb{Q}[x]$ implies that there exists a $\lambda \in \mathbb{Z}$ with $|\lambda| < 2^\tau$ such that

$$\lambda \cdot q \in \mathbb{Z}[x] \quad \text{and} \quad \|\lambda q\|_\infty = 2^{O(n+\tau)}.$$

4. Use the fact that $r_{i-1} = q_i r_i + \rho_{i+1} r_{i+1}$ and the previous result to derive a bound on the size of q_i.

Exercise 2: Modular GCD computation (4 points)

Let $f, g \in \mathbb{Z}[x]$ be integer polynomials of degree bounded by n and coefficients of absolute value less than 2^τ, let p be prime such that $p \nmid \mathrm{LC}(f)$ and $p \nmid \mathrm{LC}(g)$, and define $d := \deg \gcd(f, g)$ to be the degree of the GCD of f and g.

1. Show that

$$\gcd(f, g) \equiv \gcd(\bar{f}, \bar{g}) \mod p \quad \textit{if and only if} \quad p \nmid \mathrm{sres}_d(f, g),$$

   where $\bar{f}$ and $\bar{g}$ are the modular images of f and g in $\mathbb{Z}/p\mathbb{Z}[x]$.

2. Develop a modular algorithm to compute *under guarantee* the degree d of $\gcd(f, g) \in \mathbb{Z}[x]$ and determine its bit complexity in terms of n and τ.

Exercise 3: Applications of subresultants (4 points)

(a) Let

$$f = x^3 + 4x^2 - 2ax - a^2 \quad \text{and}$$
$$g = x^2 - 2a^2.$$

Choose a such that $\deg \gcd(f, g) = 1$.

(b) Determine the gcd of

$$f = x^2 + \Big(\frac{1}{10}\sqrt{5} - \frac{3}{10}\Big)x + \Big(\frac{3}{50}\sqrt{5} - \frac{7}{50}\Big) \quad \text{and}$$
$$g = 4x^2 + \Big(-\frac{1}{10}\sqrt{5} + \frac{3}{10}\Big)x + \Big(\frac{1}{25}\sqrt{5} - \frac{4}{25}\Big).$$

Exercise 4: Errors in the Script (many points)

Detect errors in the script! Depending on the error you will receive the following amount of points for each error:

- Typo: 0.25 point
- Any kind of language error (verified by a native speaker): 0.25 point
- Error in a calculation: 0.5 point
- Serious error (bad argument, etc.): 2 points

In addition, any reasonable recommendation regarding the presentation (explanation, style, etc.) is rewarded with 0.5 point.