Université de Bordeaux **Computational Number Theory – N1MA9W11**
Master 2 MATH 2013–2014

EXERCISES, SESSION n° 2

Exercise 1 – Implement the following algorithms for $T \in \mathbb{F}_q[X]$, where $q = p^f$ and $\deg T = d$:

1) Split $T = vW^p$, where $v, W \in \mathbb{F}_q[X]$, v squarefree.

2) core(T), product of the monic irreducible divisors of T. From now on, we assume that T is monic and separable.

3) Distinct degree factorization : $T = f_1 \dots f_d$, where f_i is a product of distinct monic irreductible polynomials of degre i.

4) Assuming T is an f_i as above, split it into irreducible factors of degre i.

5) Berlekamp algorithm.

Problem II (Multipoint evaluation)

Let R be a commutative ring and $m_0, \dots, m_{n-1}$ in $R[X]$, non-constant, where $n = 2^k$. For $0 \leqslant i \leqslant k$, and $0 \leqslant j < 2^{k-i}$, define

$$M_{i,j} = \prod_{0 \leqslant \ell < 2^i} m_{j2^i+\ell}.$$

1) Write down a natural tree whose vertices at level i are labelled by the $M_{i,j}$.

2) Compute all $M_{i,j}$ in $\widetilde{O}\left(\sum_{i<n} \deg m_i\right)$ basic operations in R. [*For $A, B \in R[X]$, we can compute $A \times B \in R[X]$ in $\widetilde{O}(\deg A + \deg B)$ operations in R.*]

3) When all m_i have degree 1, compare with the naive algorithm which would compute only $M_{k,0}$ with successive multiplications by a factor of degree 1.

4) Let $T \in R[X]$ with $\deg T < n = 2^k$ and $u_0, \dots, u_{n-1}$ in R. Let $m_i = X - u_i$ and assume all $M_{i,j}$ are precomputed, show that the following algorithm correctly computes $T(u_0), \dots, T(u_{n-1})$ in $\widetilde{O}(n)$ operations in R.

Algorithm 1. Multipoint evaluation

1: If $n = 1$, return T.
2: Let $r_0 \leftarrow T \operatorname{rem} M_{k-1,0}$. Compute recursively $r_0(u_0), \dots, r_0(u_{n/2-1})$.
3: Let $r_1 \leftarrow T \operatorname{rem} M_{k-1,1}$. Compute recursively $r_1(u_{n/2}), \dots, r_1(u_{n-1})$.
4: Return the concatenation of the outputs.

5) Show that a polynomial of arbitrary degree n can be evaluated at n points in $\widetilde{O}(n)$ operations in R. Compare with successive applications of Horner's scheme. Compare with the FFT algorithm.

Problem III (The iterated Frobenius algorithm)

Let $\mathbb{F}_q$ be a finite field of *odd* characteristic, $T \in \mathbb{F}_q[X]$ of degree n and let $\mathcal{A} = \mathbb{F}_q[X]/(T)$.

1) In this question, we assume that T is a product of distinct irreducible polynomials of degree d. We want to recover those factors.

a) Use the map $a \mapsto a^{(q^d-1)/2}$ over $\mathcal{A}$ to write down a splitting algorithm.

b) Show that the average depth of the "splitting tree" is $O(\log(n/d))$.

c) Show that your algorithm splits T completely in $\widetilde{O}(dn \log q)$ expected operations in $\mathbb{F}_q$.

2) Let $F : x \mapsto x^q$ be the Frobenius endomorphism of $\mathcal{A}$. We write $\overline{\alpha}$ for the class of $\alpha \in \mathbb{F}_q[X]$ in $\mathcal{A}$.

a) Show that $F(\overline{\alpha}) = \alpha(\overline{X}^q)$ in $\mathcal{A}$ for all $\alpha \in \mathbb{F}_q[X]$.

b) Show the following algorithm is correct and uses $\widetilde{O}(n^2)$ operations in $\mathbb{F}_q$.

Algorithm 2. Iterated Frobenius (von zur Gathen & Shoup)

Input: $T \in \mathbb{F}_q[X]$ of degree n, $D \in \mathbb{Z}_{>0}$ with $D \leqslant n$, $\overline{X}^q$, and $\overline{\alpha}$ in $\mathcal{A}$.
Output: $\overline{\alpha}, \overline{\alpha}^q, \ldots, \overline{\alpha}^{q^D}$.

Let $\overline{t_0} \leftarrow \overline{X}$, $\overline{t_1} \leftarrow \overline{X^q}$ and $\ell \leftarrow \lceil \log_2 D \rceil$.
for $i = 1, \ldots, \ell$ **do** $\{$*Compute* $\overline{t_k} = \overline{X}^{q^k}$ *for all* $k \leqslant D$.$\}$
    Call the multipoint evaluation algorithm to compute the $\overline{t_{2^{i-1}+j}} = t_{2^{i-1}}(\overline{t_j})$, for $1 \leqslant j \leqslant 2^{i-1}$.
Call the multipoint evaluation algorithm to compute and return the $\alpha(\overline{t_k})$, $1 \leqslant k \leqslant D$.

---

3) Using the Iterated Frobenius algorithm with $D = n - 1$, and a number of gcds and divisions, explain how to find the products of all irreducible factors of degree d of T, for $d = 1, \ldots, n$. Show your algorithm runs in time $\widetilde{O}(n^2 + n \log q)$.

4) Using the identity $\frac{q^d-1}{2} = (1 + q + \cdots + q^{d-1})\frac{q-1}{2}$, improve the computation of $\alpha^{(q^d-1)/2}$ in the splitting algorithm in 1) so that it uses an expected number of $\widetilde{O}(n \log q)$ operations in $\mathbb{F}_q$.

5) Show that the expected number of operations in $\mathbb{F}_q$ used by the complete factorization algorithm based on the Iterated Frobenius is $\widetilde{O}(n^2 + n \log q)$.