Université Bordeaux 1 **MAT913 – Master**
Mathématiques Année 2012–2013

Exam. 2006 Jan 10th, 14h – 18h.

Handwritten lecture notes are allowed as well as the course typescript. You may compose in either English or French. Do not hesitate to comment on the questions, partial credit will be given.

Exercise I (A broken knapsack public-key cryptosystem)

Let $(a_i)_{i\geqslant 0}$ be a sequence in $\mathbb{Z}_{>0}$ such that $a_i > \sum_{j<i} a_j$ for all i. Let $\varepsilon_0, \ldots, \varepsilon_{n-1}$ be a secret sequence of bits in $\{0,1\}$, which we want to transmit avoiding eavesdroppers. Choose $N \geqslant a_n$ a large integer, then m uniformly at random in $(\mathbb{Z}/N\mathbb{Z})^*$.

- The secret key, known to the decoder only, is m.
- The public key, known universally, is $(b_i) := (a_i m \operatorname{rem} N)$, together with N. We use it to encrypt the message $\varepsilon_0, \ldots, \varepsilon_{n-1}$ as $M = \sum_{i<n} \varepsilon_i b_i \pmod N$.

1) Explain how the secret key is used to decrypt the message. [*Recover the integer* $\sum_{i<n} \varepsilon_i a_i$ *in* $\mathbb{Z}_{>0}$ *first.*]

2) Using the ideas in the algebraicity test and van Hoeij's knapsack, suggest a lattice-based attack which might well recover the ε_i from the knowledge of M and the public key only. (No proof that the attack will succeed is required.)

Problem II (Multipoint evaluation)

Let R be a commutative ring and $m_0, \ldots, m_{n-1}$ 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) In the special case $n = 8$, write down a natural tree whose vertices at level i are labelled by the $M_{i,j}$, $j = 0, \ldots, 2^{3-i} - 1$.

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 $u_0, \ldots, u_{n-1}$ in R. Let $m_i = X - u_i$ and assume all $M_{i,j}$ are precomputed, show that the following algorithm correctly computes $T(u_0), \ldots, T(u_{n-1})$ in $\widetilde{O}(n)$ operations in R.

Algorithme 1. Multipoint evaluation

1: If $n = 1$, return T.
2: Let $r_0 \leftarrow T \operatorname{rem} M_{k-1,0}$. Compute recursively $r_0(u_0), \ldots, r_0(u_{n/2-1})$.
3: Let $r_1 \leftarrow T \operatorname{rem} M_{k-1,1}$. Compute recursively $r_1(u_{n/2}), \ldots, r_1(u_{n-1})$.
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 and $T \in \mathbb{F}_q[X]$ of degree n.

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) Adapt the algorithm seen for $d = 1$ to the general case [*use the map* $a \mapsto a^{(q^d-1)/2}$ *over* $\mathbb{F}_q[X]/(T)$].

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 $R = \mathbb{F}_q[X]/(T)$. We write $\overline{\alpha}$ for the class of $\alpha \in \mathbb{F}_q[X]$ in R.

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

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

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

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

1: Let $\overline{t_0} \leftarrow \overline{X}$, $\overline{t_1} \leftarrow \overline{X^q}$ and $\ell \leftarrow \lceil \log_2 D \rceil$.
2: **pour** $i = 1, \ldots, \ell$ **faire** $\quad$ {*Compute* $\overline{t_k} = \overline{X}^{q^k}$ *for all* $k \leqslant D$.}
3: Call the multipoint evaluation algorithm to compute the $\overline{t_{2^{i-1}+j}} = t_{2^{i-1}}(\overline{t_j})$, for $1 \leqslant j \leqslant 2^{i-1}$.
4: 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 $\widetilde{O}(n^2 + n \log q)$.

Open exercise IV (to be done last)

Carmichael numbers are non-primes N which satisfy $a^N \equiv a \pmod N$ for all $a \in \mathbb{Z}$; infinitely many exist. Show there exist finitely many integers N such that $a^{N+1} \equiv a \pmod N$ for all $a \in \mathbb{Z}$ and give the complete list.