Université Bordeaux 1 — **MAT823 – Calcul Formel**
Mathématiques — Année 2012–2013

Devoir Surveillé, 08 Mars 2006
Durée 2 heures. Documents interdits

Exercice 1 – Soit G un groupe cyclique d'ordre n, dont la factorisation est $n = p_1^{e_1} \dots p_k^{e_k}$.

1) On tire un élément de G uniformément au hasard. Quelle est la probabilité pour que ce soit un générateur de G ?

2) Proposer et évaluer une stratégie pour trouver un générateur du groupe G.

Exercice 2 – Soient f et g les polynômes suivants :

$$f = X^5 + X^4 + X^3 - X^2 - X + 1, \quad g = X^3 + X^2 + X + 1$$

Pour $p = 3$ et $p = 5$, calculer leur PGCD (f, g) dans $\mathbb{F}_p[X]$, ainsi que des polynômes s et t de $\mathbb{F}_p[X]$ tels que $sf + tg = (f, g)$.

Exercice 3 – On rappelle que les nombres de Fibonacci sont définis par $F_0 = 0$, $F_1 = 1$ et pour $n \geqslant 2$, $F_n = F_{n-1} + F_{n-2}$. Soit $h = \sum_{n \geqslant 0} F_n X^n \in \mathbb{Q}[[X]]$, la série formelle dont les coefficients sont les nombres de Fibonacci.

1) Montrer que h est une fraction rationnelle que l'on calculera.

2) En utilisant la décomposition en éléments simples de h, trouver une expression de F_n pour tout n en fonction du « nombre d'or » ($\varphi = (1 + \sqrt{5})/2$) et de son inverse.

Exercice 4 –

1) Soit $p \in \mathbb{N}_{>0}$, $f, g_0 \in \mathbb{Z}$ tels que $fg_0 \equiv 1 \pmod{p}$. On fixe $\ell > 0$. Écrire un algorithme, utilisant l'inversion de Newton, qui calcule $g \in \mathbb{Z}$ tel que $fg \equiv 1 \pmod{p^\ell}$.

2) On rappelle que le coût d'une opération dans $\mathbb{Z}$ se mesure en terme du nombre d'opérations sur les chiffres dans une base B fixée (opération élémentaire). Estimer le nombre d'opérations élémentaires dans $\mathbb{Z}$ utilisées.

3) Soit R un anneau commutatif et unitaire. On considère l'algorithme suivant

---

Algorithme 1. Racine modulo p^ℓ de $\varphi \in R[X]$

Entrées: $\varphi \in R[X]$, $p \in R$, $\ell \in \mathbb{N}_{>0}$, $g_0, s_0 \in R$, tels que

$$\varphi(g_0) \equiv 0 \pmod{p}, \quad \text{et} \quad s_0\varphi'(g_0) \equiv 0 \pmod{p}.$$

Sorties: $g \in R$ tel que $\varphi(g) \equiv 0 \pmod{p^\ell}$ et $g \equiv g_0 \pmod{p}$.

1: $r \leftarrow$ le plus petit entier supérieur ou égal à $\log_2 \ell$.
2: **pour** $i = 1, \cdots, r - 1$ **faire**
3: calculer $g_i, s_i \in R$ tels que

$$g_i \equiv g_{i-1} - \varphi(g_{i-1})s_{i-1} \pmod{p^{2^i}}, \quad s_i \equiv 2s_{i-1} - \varphi'(g_i)s_{i-1}^2 \pmod{p^{2^i}}.$$

4: Renvoyer $g = g_r$.

Appliquer cet algorithme pour $R = \mathbb{Z}$, $p = 5$ pour déterminer une solution $g \neq \pm 1$ de l'équation $g^4 \equiv 1 \pmod{625}$.

4) [*Facultatif*] Démontrer que cet algorithme est correct.

Université Bordeaux 1
Mathématiques

MAT823 – Calcul Formel
Année 2012–2013

Devoir Surveillé, 08 Mars 2006 (**Correction**)

Exercice 1 –

1) Le nombre de générateurs de G est $\varphi(n)$. Donc la probabilité pour qu'un élément de G tiré uniformément au hasard soit un générateur est $\varphi(n)/n$.

2) On écarte le cas trivial $n = 1$. Soit α un élément de G. On peut suggérer au moins trois stratégies :

(1) α est un générateur de G si et seulement si pour tout $i \in \{1, \ldots, n-1\}$, $\alpha^i \neq 1$. Ceci requiert $O(n)$ multiplications dans G par essai, à condition de calculer α^i comme $\alpha^{i-1} \times \alpha$. En tirant α uniformément au hasard, en moyenne

$$\frac{n}{\varphi(n)} = \prod_{i=1}^{k} \left(1 - \frac{1}{p_i}\right)^{-1} \leqslant 2^k$$

essais suffisent, mais la majoration par 2^k n'est pas très satisfaisante. (Avec un peu de théorie analytique des nombres, on peut montrer que $n/\varphi(n) = O(\log\log n)$.)

(2) α est un générateur de G si et seulement si pour tout $i \in \{1, \ldots, k\}$, $\alpha^{n/p_i} \neq 1$. On peut estimer le coût naïvement comme ci-dessus, auquel cas on ne constate pas d'amélioration, ou calculer α^{n/p_i} en $O(\log(n/p_i))$ multiplications, soit $O(k \log n)$ multiplications par essai, après sommation sur i. Comme $k = O(\log n)$, l'amélioration du test est conséquente. Le nombre moyen d'essais n'est pas affecté.

(3) On peut diminuer le nombre d'essais et simplifier l'analyse en remarquant que si $\alpha^{n/p_i} \neq 1$, alors $g_i := \alpha^{n/p_i^{e_i}}$ est un générateur du p_i-groupe de Sylow de G et que le produit des g_i est un générateur de G. (Dans un groupe abélien, le produit de deux éléments d'ordres premiers entre eux est le produit de leurs ordres.)

Si α dans G est choisi uniformément au hasard, alors $g_i := \alpha^{n/p_i^{e_i}}$ est choisi uniformément au hasard dans le p_i-Sylow de G. (En effet, en posant $q := p_i^{e_i}$, l'application $x \mapsto x^{n/q}$ est une bijection de G/G^q dans le p_i-Sylow, donc le nombre d'éléments α s'envoyant sur un élément x du p_i-Sylow ne dépend pas de x. La distribution est bien uniforme.)

L'élément g_i est donc un générateur avec probabilité $1 - 1/p_i \geqslant 1/2$ d'après la première question. L'espérance du nombre de α à tester avant de trouver g_i est donc $O(1)$, et le coût du test reste $O(\log(n/p_i))$ multiplications dans G. L'espérance du coût *total* est $O(k \log n + k) = O(\log n)^2$ multiplications.

Précisons l'argument probabiliste : soit X_i la variable aléatoire donnant le nombre d'essais (indépendants) avant d'obtenir un g_i engendrant le p_i-Sylow de G. Pour alléger les notations, posons $X = X_i$ et $p = p_i$. On a

$$\mathbb{E}(X) = \sum_{j \geqslant 1} \mathbb{P}(X = j) j = \sum_{j \geqslant 1} \frac{1}{p^{j-1}} \left(1 - \frac{1}{p}\right) j = \left(1 - \frac{1}{p}\right)^{-1} \leqslant 2.$$

(Espérance d'une loi géométrique.) L'espérance du nombre de multiplications pour obtenir tous les g_i est bien $\mathbb{E}(\sum_i X_i \log(n/p_i)) = O(k \log n)$.

Exercice 2 – L'algorithme d'Euclide montre que le PGCD est $2X+2$ dans le premier cas et $3X^2+4X+1$ dans le second cas. On a

$$(2X+1)f + (X^3+2X^2+1)g = 2X+2 \quad \text{dans } \mathbb{F}_3[X],$$
$$f - X^2 g = 3X^2+4X+1 \quad \text{dans } \mathbb{F}_5[X].$$

Exercice 3 –

1) La relation de récurrence sur les nombres de Fibonacci montre que

$$h = X + Xh + X^2 h, \quad \text{c'est à dire que} \quad h = \frac{-X}{X^2+X-1}.$$

2) On a $X^2+X-1 = (X+\varphi)(X-\varphi^{-1})$. Donc

$$h = \frac{-\varphi}{X+\varphi} + \frac{-\varphi^{-1}}{X-\varphi^{-1}}.$$

On en déduit

$$h = \sum_{n \geqslant 0} \left(\frac{X^n}{\varphi^{-n}} + (-1)^{n+1} \frac{X^n}{\varphi^n} \right), \quad \text{soit} \quad F_n = \varphi^n + (-1)^{n+1} \varphi^{-n}.$$

Exercice 4 –

1) Par exemple :

Algorithme 2. inverse modulo p^ℓ

Entrées: $f \in \mathbb{Z}$, p premier, $\ell \in \mathbb{N}_{>0}$, $g_0 \in \mathbb{Z}$ tel que $fg_0 \equiv 1 \pmod{p}$.
Sorties: $g \in \mathbb{Z}$ tel que $fg \equiv 1 \pmod{p^\ell}$.

1: $r \leftarrow$ le plus petit entier supérieur ou égal à $\log_2 \ell$.
2: **pour** $i = 1, \ldots, r$ **faire**
3: calculer $g_i \in \mathbb{Z}$ tel que $g_i \equiv 2g_{i-1} - fg_{i-1}^2 \pmod{p^{2^i}}$.
4: Retourner $g = g_r$.

L'algorithme est correct. En effet, montrons par récurrence sur i que $fg_i \equiv 1 \pmod{p^{2^i}}$ pour tout $i \geqslant 0$. Pour $i = 0$, on a bien $fg_0 \equiv 1 \pmod{p}$. En supposant l'égalité vraie au rang $i-1$, on a

$$1 - fg_i \equiv 1 - f(2g_{i-1} - fg_{i-1}^2) \equiv (1 - fg_{i-1})^2 \equiv 0 \pmod{p^{2^i}}.$$

En particulier, $fg_r \equiv 1 \pmod{p^{2^r}}$, et donc aussi modulo p^ℓ puisque $2^r \geqslant \ell$.

2) On suppose que $0 \leqslant f < p^\ell$. On remarque que $\ell \leqslant 2^r < 2\ell$, donc $r = O(\log \ell)$. Sa valeur se calcule en $O(r)$ additions d'entiers inférieurs à ℓ, donc en utilisant $O(\log \ell)^2$ opérations élémentaires.

La i-ème boucle utilise $O(2^r \log p)^2$ opérations élémentaires (arithmétique naïve) puisque tous les opérandes sont $< p^{2^r}$, donc de taille $O_B(2^r \log p)$ et que chaque boucle effectue $O(1)$ additions, multiplications et divisions euclidiennes. Sommant sur $i \leqslant r$, on obtient $O\big((\ell \log p)^2 \log \ell\big)$ opérations élémentaires, le calcul de r étant négligeable.

Remarquons que le calcul de l'inverse par l'algorithme d'Euclide étendu appliqué à f et p^ℓ utilise *moins* d'opérations élémentaires ! Par contre ce nouvel algorithme est intéressant quand B est de la forme p^{2^k} : dans ce cas, les divisions par p^{2^i}, $i \geqslant k$, sont des troncatures n'utilisant pas

d'opération élémentaire ; comme dans le cas des séries formelles vu en cours. (Un cas particulier important est $p = 2$ et $B = 2^{32}$.)

Dans ce cas, les opérandes de la i-ème boucle sont $< p^{2^i}$ et on obtient $O(\ell \log p)^2$ opérations élémentaires en sommant sur i. Et même $\widetilde{O}(\ell \log p)$ opérations si la multiplication est asymptotiquement rapide (Schönhage-Strassen).

3) Soit $\varphi = X^4 - 1$. On peut démarrer avec $g_0 = 2$, puisque $\varphi(g_0) \equiv 0 \pmod 5$ et $\varphi'(g_0) \equiv 2 \pmod 5$. Donc $s_0 \equiv 3 \pmod 5$. On a

$$g_1 \equiv 7 \pmod{25}, \quad s_1 \equiv 8 \pmod{25}$$
$$g \equiv 182 \pmod{625}$$

4) Démontrons par récurrence sur i que, pour tout $i \geqslant 0$, on a

- $\varphi'(g_i)s_i \equiv 1 \pmod{p^{2^i}}$,
- $\varphi(g_i) \equiv 0 \pmod{p^{2^i}}$.

C'est vrai au rang $i = 0$. En supposant les assertions vraies au rang $i - 1$, on remarque que $g_i \equiv g_{i-1} \pmod{p^{2^{i-1}}}$. Donc $\varphi'(g_i) \equiv \varphi'(g_{i-1}) \pmod{p^{2^{i-1}}}$, puisque φ' est un polynôme de $R[X]$. On calcule

$$1 - \varphi'(g_i)s_i \equiv (1 - \varphi'(g_i)s_{i-1})^2 \pmod{p^{2^i}}.$$

Comme la parenthèse est $\equiv 1 - \varphi'(g_{i-1})s_{i-1} \equiv 0 \pmod{p^{2^{i-1}}}$ par hypothèse, son carré est bien $\equiv 0 \pmod{p^{2^i}}$. (On retrouve l'algorithme de Newton pour l'inverse.)

Pour la deuxième assertion, on a besoin du lemme suivant :

Lemme 1. *Soit $\varphi \in R[X]$ et $g \in R$, on a le développement de Taylor*

$$\varphi(X + g) = \varphi(g) + \varphi'(g)X + \psi(X)X^2,$$

avec $\psi \in R[X]$.

(Il est important que ψ soit à coefficients dans R : on n'introduit pas de dénominateurs !)

Preuve. Si $A \in R[X]$, on a

$$A(X) = A(0) + A'(0)X + \psi(X)X^2,$$

avec $\psi \in R[X]$, et on applique cette identité à $A(X) = \varphi(X + g) \in R[X]$. □

On applique donc le développement de Taylor

$$\varphi(X + g_{i-1}) = \varphi(g_{i-1}) + \varphi'(g_{i-1})X + \psi(X)X^2,$$

avec $X := g_i - g_{i-1} \equiv -\varphi(g_{i-1})s_{i-1} \pmod{p^{2^i}}$, qui vérifie $X^2 \equiv 0 \pmod{p^{2^i}}$. On obtient

$$\varphi(g_i) \equiv \varphi(g_{i-1})\,(1 - \varphi'(g_{i-1})s_{i-1}) \pmod{p^{2^i}}.$$

Par hypothèse de récurrence, chacun des deux termes est $\equiv 0 \pmod{p^{2^{i-1}}}$, donc $\varphi(g_i) \equiv 0 \pmod{p^{2^i}}$.