Université de Bordeaux — **N1MA8W02 – Master**
Mathématiques — Année 2017–2018

Devoir Surveillé, 7 mars 2018
Corrigé.

Exercice 1 – *[Une stratégie alternative pour la division euclidienne]*
Soit K un corps. pour tout polynôme $F \in K[x]$, et tout entier k, on note

$$rev_k(F)(x) = x^k F\left(\frac{1}{x}\right).$$

1) *Montrer que si $k \geqslant \deg F$, alors $rev_k(F) \in K[x]$ et que si $k = \deg F$, alors $rev_k(rev_k(F)) = F$.*

On écrit

$$F = a_m x^m + \cdots + a_1 x + a_0.$$

Alors

$$\mathrm{rev}_k(F) = a_m x^{k-m} + \cdots + a_1 x^{k-1} + a_0^k.$$

Dans cette expression, les exposants de x appartiennent à $[[k-m, k]]$. Comme $k \geqslant m$, ces exposants sont des entiers positifs ou nuls donc $\mathrm{rev}_k(F) \in K[x]$.

$$\mathrm{rev}_k(\mathrm{rev}_k(F))(x) = \mathrm{rev}_k\left(x^k F\left(x^{-1}\right)\right) = x^k x^{-k} F(x) = F(x).$$

Ce résultat est vérifié pour tout k.

On remarque que si $k = \deg F$,

$$\mathrm{rev}_k(F) = a_k + a_{k-1}x + \cdots + a_1 x^{k-1} + a_0 x^k.$$

On "renverse" le polynôme F.

2) *On note* `Kx` *l'anneau $K[x]$. Écrire sur votre fichier sage une fonction* `rev(Kx,k,F)` *qui rend le polynôme $rev_k(F)$ si $k \geqslant \deg F$ et "Erreur" sinon. Attention de bien rendre un polynôme. On rappelle que si* `A` *est un polynôme vu par sage comme un élément de $K(x)$, alors* `Kx(A)` *sera bien considéré comme un élément de $K[x]$.*

3) *Soient F et G deux polynômes non nuls de $K[x]$ de degrés respectifs m et n tels que $m \geqslant n$. Soient Q et R le quotient et le reste de la division euclidienne de F par G. Montrer que*

$$rev_m(F) = rev_{m-n}(Q)rev_n(G) + x^{m-n+1}\mathrm{rev}_{n-1}(R).$$

Les polynômes Q et R sont tels que $F = GQ + R$. On obtient donc les égalités suivantes.

$$\begin{aligned}
\mathrm{rev}_m(F) &= x^m F(1/x) \\
&= x^m G(1/x)Q(1/x) + x^m R(1/x) \\
&= (x^n G(1/x))(x^{m-n}Q(1/x)) + x^{m-n+1}x^{n-1}R(1/x) \\
&= \mathrm{rev}_n(G)\mathrm{rev}_{m-n}(Q) + x^{m-n+1}\mathrm{rev}_{n-1}(R)
\end{aligned}$$

De plus, comme $\deg Q = m-n$, $\deg F = m$, $\deg G = n$ et $\deg R \leqslant n-1$, les éléments $\mathrm{rev}_{m-n}(Q)$, $\mathrm{rev}_m(F)$, $\mathrm{rev}_n(G)$ et $\mathrm{rev}_{n-1}(R)$ appartiennent à $K[x]$.

4) *Montrer que la classe de $\mathrm{rev}_n(G)$ dans $K[x]/(x^{m-n+1})$ est inversible.*

Écrivons $G = \sum_{i=0}^{n} b_i x^i$. Alors $\mathrm{rev}_n(G) = \sum_{i=0}^{n} b_{n-i} x^i$. Le coefficient constant de ce polynôme est $b_n \neq 0$ puisque $n = \deg G$. On en déduit que $\mathrm{rev}_n(G)$ est premier à x, donc à x^{m-n+1}, ce qui montre que $\mathrm{rev}_n(G)$ est inversible dans $K[x]/(x^{m-n+1})$.

5) *On rappelle que si A et B sont deux polynômes, la commande* `xgcd(A,B)` *rend (d,u,v) où $d = \mathrm{pgcd}(A,B)$ et $Au+Bv = d$. Écrire sur votre fichier sage une fonction* `Inverse1(A,B)` *qui utilise* `xgcd` *pour rendre un polynôme A^* tel que $AA^* \equiv 1 \mod B$ si A est inversible modulo B et qui rend "Erreur" sinon.*

6) *On considère l'algorithme suivant.*

`Division euclidienne(F,G)`

Entrées : $F, G \in K[x]$ tels que $G \neq 0$, $n \in \mathbb{N} \setminus \{0\}$.

Sorties : le quotient et le reste de la division euclidienne de F par G.

Si $\deg F < \deg G$, sortir $(0, F)$.
$(m,n) = (\deg F, \deg G)$
$G_1 = rev_n(G)^{-1} \mod x^{m-n+1}$
$Q_1 = rev_m(F)G_1 \mod x^{m-n+1}$
$Q = rev_{m-n}(Q_1)$
Sortir $(Q, F-GQ)$

Exécuter cet algorithme "à la main" sur les polynômes de $\mathbb{F}_{17}[x]$ suivants. $F = 5x^5+4x^4+3x^3+2x^2+x$ et $G = x^2+2x+3$. Il n'est pas demandé ici d'implémenter l'algorithme sur machine, mais plutôt de l'exécuter pas à pas, en s'aidant de sage. On indiquera les commandes utilisées et les résultats intermédiaires obtenus. On commence par définir $\mathbb{F}_{17}[x]$.

```
kx.<x>=PolynomialRing(GF(17))
```

Ici, $m = 5$ et $n = 3$. Il est clair que $\mathrm{rev}_2(G) = 3x^2+2x+1$. Soit G_0 ce polynôme. `G1=Inverse1(G0,x**4)` donne $G_1 = 4x^3 + x^2 + 15x + 1$. Ensuite, on calcule `Q1=(rev(kx,5,F)*G1) % x**4`, ce qui donne $Q_1 = 3x^3 + 11x + 5$, donc $Q = \mathrm{rev}_3(Q_1) = 5x^3 + 11x^2 + 3$. La commande `R=F-GQ` donne alors $R = 12x + 8$ qui est bien de degré strictement inférieur à 2.

7) *Démontrer que cet algorithme donne bien le resultat annoncé.* On sait que

$$\mathrm{rev}_m(F) = \mathrm{rev}_{m-n}(Q)\mathrm{rev}_n(G) + x^{m-n+1}\mathrm{rev}_{n-1}(R).$$

Comme de plus, chacun des termes de cette expression est un polynôme, on en déduit que

$$\mathrm{rev}_m(F) \equiv \mathrm{rev}_{m-n}(Q)\mathrm{rev}_n(G) \mod x^{m-n+1}. \tag{1}$$

Soit G_1 un polynôme tel que $G_1\mathrm{rev}_n(G) \equiv 1 \mod x^{m-n+1}$, la multiplication par G_1 dans l'équation (1) donne

$$G_1\mathrm{rev}_m(F) = \mathrm{rev}_{m-n}(Q).$$

Or si $Q_1 = \mathrm{rev}_{m-n}(Q)$, alors d'après la question 1), $\mathrm{rev}_{m-n}(Q_1) = Q$. Une fois le quotient Q calculé, il est clair que $R = F - GQ$.

8) *Répondre à l'une des deux questions suivantes.*

a) Quelle est la complexité algébrique du calcul de l'inverse modulaire G_1 en appliquant l'algorithme d'Euclide étendu vu en cours ?

Comme $\deg G = m$, la complexité algébrique de l'algorithme d'Euclide étendu appliqué à G et x^{m-n+1} est en $O((m+1)(m-n+1))$.

b) Quelle est la complexité algébrique de la division euclidienne classique de F par G ?

Comme $\deg F = m$ et $\deg G = n$, la complexité algébrique de la division de F par G est en $O((m+1)(m-n+1))$.

La réponse est la même pour les deux questions. La stratégie proposée ne semble donc pas apporter d'amélioration. Heureusement, l'exercice 2 propose un algorithme plus rapide pour calculer G_1.

Exercice 2 – [Inversion rapide modulo x^n]

Soit K un corps. Soient $F \in K[x]$ et n un entier naturel non nul. On suppose que $F(0) = 1$. On sait que dans ce cas, la classe de F dans $K[x]/(x^n)$ est inversible. Cet exercice porte sur un algorithme rapide pour calculer son inverse.

Soit (A_i) la suite définie de la manière suivante.

$$A_0 = 1 \quad , \quad A_{i+1} = 2A_i - FA_i^2 \quad \forall i \geqslant 0.$$

1) *Montrer que $FA_i \equiv 1 \mod x^{2^i}$. pour tout $i \geqslant 0$.*

Procédons par récurrence.

Pour $i = 0$, $FA_0 = F$ et $x^{2^0} = x$. Le fait que $F(0) = 1$ montre que $F \equiv 1 \mod x$. La congruence est donc vérifiée au rang $i = 0$.

On suppose que $FA_i \equiv 1 \mod x^{2^i}$.

$$\begin{aligned} FA_{i+1} - 1 &= 2A_iF - F^2A_i^2 - 1 \\ &= -(A_iF - 1)^2. \end{aligned}$$

L'hypothèse de récurrence montre que x^{2^i} divise $A_iF - 1$, donc $x^{2^{i+1}}$ divise $(A_iF - 1)^2 = 1 - FA_{i+1}$. On a donc bien montré la congruence au rang $i+1$.

2) *En déduire un algorithme pour calculer l'inverse de F modulo x^n et écrire la fonction correspondante* `Inverse2(n,F)` *sur votre fichier sage.*

3) *On suppose que la complexité algébrique de la multiplication de deux polynômes de $K[x]$ de degrés inférieurs à un entier N est en $O(N \log N)$ opérations dans K. On rappelle que prendre le reste de la division d'un polynôme par x^N revient à tronquer le polynôme : on ne prend pas cette opération en compte dans le calcul de la complexité algébrique.*

Soient $C(n)$ la complexité algébrique de `Inverse2(n,F)` *et $r = \lceil \log n \rceil$ le plus petit entier supérieur ou égal à $\log n$. Montrer que $C(n)$ est en $O(r2^r)$. En déduire que $C(n)$ est en $O(n \log n)$.*

On rappelle la notation $\log = \log_2$.

Dans cet algorithme, les opérations les plus couteuses sont les mutliplications du calcul de chaque A_i. Chacune de ces multiplications se font modulo x^{2^i}, donc leur complexité algébrique est en $O(2^i \log(2^i)) = O(i2^i)$. Soit

Il existe une constante M telle que $C(n) \leqslant M \sum_{i=1}^{r} i2^i$. Or :

$$\sum_{i=1}^{r} i2^i \leqslant r \sum_{i=1}^{r} 2^i \leqslant r(2^{r+1} - 1) \leqslant r2^{r+1}$$

en appliquant un résultat bien connu sur les sommes géométriques. On en déduit que $C(n)$ est en $O(r2^r)$. Précisons l'ordre de grandeur de n en fonction de celle de r.

$$r - 1 < \log n \leqslant r$$

donc $r < \log n + 1$ est en $O(\log n)$ et $2^{r-1} < n$ donc $2^r < 2n$ donc 2^r est en $O(n)$. On obtient bien que $r2^r$ est en $O(n \log n)$.

Exercice 3 – [Division euclidienne rapide]

Écrire sur votre fichier sage une fonction `Division(F,G)` *qui en utilisant certaines fonctions des questions précédentes effectue la division euclidienne F par G avec une complexité algébrique en $O(N \log N)$, où F et G sont des polynômes de $K[x]$ de degrés inférieurs à N et tels que $G \neq 0$ (on suppose encore que la multiplication de tels polynômes est en $O(N \log N)$).*