Université Bordeaux 1 **N1MA8W02 – Master**
Mathématiques Année 2015–2016

Devoir Surveillé, 2 mars 2016

Durée 1h30.

À la fin de l'épreuve, votre fichier "`votre_nom.sage`" est à envoyer à l'adresse : `arnaud.jehanne@u-bordeaux.fr`

Exercice 1 – [Interpolation d'Hermite]

1) On cherche à trouver un polynôme P de $\mathbb{Q}[x]$ de degré inférieur ou égal à 5 tel que $P(1) = 0$, $P'(1) = -3$, $P''(1) = 2$, $P'''(1) = 42$, $P(2) = 11$ et $P'(2) = 45$.

a) Sur votre copie, traduire ces contraintes en termes de problème de restes chinois.

b) Résoudre le problème en utilisant sage, par exemple la commande `crt`. Sur votre copie, écrire la commande utilisée et le résultat obtenu.

2) Pour tout polynôme P et tout entier naturel i, on note $P^{(i)}$ la dérivée i-ème de P avec la convention : $P^{(0)} = P$.

a) Écrire sur votre fichier ".sage" une fonction `Hermite1` qui prend en entrée une liste $l = [a, b]$, où $a \in \mathbb{Q}$ et où b est elle même une liste $[b_0, \ldots, b_r]$ d'éléments de $\mathbb{Q}$, et qui en sortie rend le polynôme P de degré minimal tel que $P^{(i)}(a) = b_i$ pour tout $i \in [[0, r]]$.

Pour le calcul de $k!$, on peut utiliser la commande `factorial(k)`.

b) Écrire sur votre fichier ".sage" une fonction `Hermite` qui en entrée prend une liste $l = [[a_0, b_0], \ldots, [a_s, b_s]]$ où les a_i sont des éléments de $\mathbb{Q}$ et les b_i des listes $[b_{i0}, \ldots, b_{ir_i}]$ d'éléments de $\mathbb{Q}$ et qui en sortie rend le polynôme de degré minimal P tel que pour tout $i \in [[0, s]]$ et tout $j \in [[0, r_i]]$,

$$P^{(j)}(a_i) = b_{ij}.$$

Cette fonction utilisera votre fonction `Hermite1` de la question 2.a et la fonction `crt` de sage.

Exercice 2 – [Convolution rapide]

Cet exercice est à rédiger sur votre copie.

Soit n une puissance de 2. Soit K un corps de caractéristique différente de 2 qui contient une racine primitive n-ème de l'unité notée ω. Soient f et g deux polynômes à coefficients dans K de degrés strictement inférieurs à n. On note $f *_n g$ le reste de la division euclidienne de fg par $x^n - 1$. On considère l'algorithme suivant.

Algorithme 1. Convolution rapide

Entrées: $\{\omega, \omega^2, \ldots, w^{n/2-1}\}$, $f, g \in K[x]$, $\deg f < n$, $\deg g < n$
Sorties: $f *_n g$

1: Si $n = 1$, alors sortir fg
2: Calculer les restes respectifs f_0 et g_0 de la division de f et g par $x^{n/2} - 1$, ainsi que les restes respectifs f_1 et g_1 de la division de f et g par $x^{n/2} + 1$
3: Appeler récursivement l'algorithme pour calculer $h_0(x) = f_0(x) *_{n/2} g_0(x)$ et $h_1(\omega x) = f_1(\omega x) *_{n/2} g_1(\omega x)$
4: Sortir $(h_0 + h_1 + x^{n/2}(h_0 - h_1))/2$

1) Exécuter à la main cet algorithme avec les données $n = 4$, $f = 1 + x^3$ et $g = 1 + x + x^2$, le corps de base étant $K = \mathbb{C}$, puis vérifier le résultat trouvé.

2) Montrer que $2fg \equiv (x^{n/2} + 1)f_0g_0 - (x^{n/2} - 1)f_1g_1 \mod x^n - 1$.

3) Montrer que $h_1(x) \equiv f_1(x)g_1(x) \mod x^{n/2} + 1$.

4) Montrer que l'algorithme "Convolution rapide" calcule bien $f *_n g$.

5) Calculer la complexité algébrique de cet algorithme.