Arithmétique Algorithmique

http://www.math.univ-lyon1.fr/~roblot/ens.html

Partie I

Introduction

Références

- J. von zur Gathen, J. Gerhard, *Modern Computer Algebra*, Cambridge University Press, 2003
- H. Cohen, *A Course in Computational Algebraic Number Theory*, GTM **38**, Springer-Verlag, 1993
- H. Riesel, *Prime Numbers and Computer Methods for Factorization*, Progress in Mathematics, Birkhaüser, 1985
- R. Crandall, C. Pomerance, *Primer Numbers, A Computational Perspective*, Springer, 2001

Complexité

Notation en "grand Oh". Pour f et g deux fonctions de E dans $\mathbb{R}$ avec $g > 0$. On note $f \in O(g)$ s'il existe $C > 0$ (constante) tel que

$$|f(x)| \leq C\, g(x) \quad \text{pour tout } x \in E.$$

On note $f \in O_{c_1,\ldots,c_s}(g)$ si la constante C dépend de paramètres $c_1, \ldots, c_s$.

Exemples.

$$f \in O(1) \iff f \text{ est bornée},$$
$$f \in O(|f|),$$
$$x^a + \log(x)^b \in O(x^a) \text{ pour tout } a, b > 0 \ (x > 0),$$
$$f \in O(g) \not\Rightarrow e^f \in O(e^g)\,.$$

Autre notation. $f = O(g)$, mais attention

$$f = O(g) \quad \text{et} \quad h = O(g) \not\Rightarrow f = h, \text{ ni même } f = O(h).$$

Complexité

Définition. Pour $N \geq 1$, $\beta > 0$ et $0 \leq \alpha \leq 1$, on pose

$$L(\alpha, \beta; N) := \exp\left(\beta (\log N)^{\alpha} (\log \log N)^{1-\alpha}\right)$$

Taille des données. La taille d'un entier N est $\in O(\log N)$
La taille d'un polynôme de degré d avec des coefficients bornés par B est $\in O(d \cdot \log B) = O_B(d)$

Classes de complexité. Avec un entier N en entrée

- Complexité exponentielle : $\alpha = 1$ et $L(1, \beta; N) = e^{\beta \log N} = N^{\beta}$
- Complexité polynômiale : $\alpha = 0$ et $L(0, \beta; N) = (\log N)^{\beta}$
- Complexité sous-exponentielle : $0 < \alpha < 1$

Complexité probabiliste. L'algorithme (**qui rend toujours un résultat exact**) utilise des nombres aléatoires et il est possible qu'il ne termine jamais, même si on peut estimer la complexité probable.

Complexité