Université Paris 7
IUP 2
Analyse de Performance et Simulation
Année 2003-2004

**TD no 6**
**Algorithmes de Brent et Floyd**
**Calcul modulaire**
**Théorème des restes chinois**

Pour toute question pendant le TD, commencez par lire le résumé de cours à la fin de la feuille.

**Exercice 1 [ Algorithmes de Brent et Floyd ]**

1 A quoi servent ces deux algorithmes ?

2 Combien de valeurs est-il nécessaire de stocker en mémoire quand on utilise ces deux algorithmes ?

3 Ecrire en pseudo-code l'algorithme de Brent.

**Exercice 2 [ Calcul modulaire ]**

Trouvez s'ils existent le $1^{er}$ et le $10^{e}$ $n$ tels que :

a) $3^n + 7^n = 0 \bmod 13$

b) $3^n + (-4)^n = 0 \bmod 13$

c) $n^2 + 1 = 4 \bmod 7$

d) $3^n = n7^n \bmod 13$

**Exercice 3 [ Théorème des restes chinois (TRC) ]**

a) Résoudre le système suivant en utilisant le théorème 2 :

$$x \in \mathbb{Z}, \begin{cases} x &= 2 \bmod 7 \\ x &= 3 \bmod 8 \end{cases}$$

b) Calculez $x = 43 * 12$ en utilisant le théorème 4 avec la décomposition

$$n_1 = 5, n_2 = 9, n_3 = 13.$$

Pour éviter de trop gros calculs, on vous donne : $45 * 11 = 495$ et $38 * 13 = 494$.

**Exercice 4 [ Applications du TRC ]**

Résoudre les systèmes suivants :

a)

$$x, x_i \in \mathbb{Z}, \begin{cases} x &= x_1 \bmod 4 \\ x &= x_2 \bmod 15 \\ x &= x_3 \bmod 7 \end{cases}$$

b)

$$n \in \mathbb{Z}, \begin{cases} 4^n &= 7 \bmod 9 \\ 2^n &= 3 \bmod 11 \end{cases}$$

**Résumé de cours**

## Arithmétique modulaire

**Théorème 1 (Petit théorème de Fermat)** *Soit $p$ un nombre premier, et $x$ un entier vérifiant $x \neq 0 \bmod p$, alors $x^{p-1} = 1 \bmod p$. En levant la seconde hypothèse sur $p$, on obtient la forme équivalente $x^p = x \bmod p$.*

## Le théorème des restes chinois (TRC)

Le TRC est un outil fondamental pour ce que l'on appelle le calcul modulaire. Si l'on veut faire des calculs avec de grands entiers il existe plusieurs méthodes dont les deux les plus utilisées sont la numération positionnelle et le calcul modulaire. Ce dernier consiste à effectuer de nombreux petits calculs modulo des nombres premiers entre eux (souvent premiers tout simplement) et à recomposer le résultat en utilisant le TRC. Par exemple, si on veut multiplier deux entiers positifs, $x$ et $y$, on choisit $n$ nombres premiers distincts $p_i$ tels que $\prod p_i > xy$, et on effectue la réduction de $x$ et $y$ modulo chaque $p_i$, ce qui donne les nombres $x_i$ et $y_i$. On multiplie alors, dans $\mathbb{Z}/p_i\mathbb{Z}$, $x_i$ par $y_i$ ce qui donne $z_i$ ; le TRC permet alors de reconstituer $z = xy$ à partir des $z_i$ ainsi calculés. Mais ce n'est là qu'une application particulière du TRC.

Le TRC existe sous plusieurs formes.

**Théorème 2** *[TRC pour deux éléments.] Soient $n$ et $m$ deux entiers premiers entre eux. Soient $y$ et $z$ deux entiers quelconques ; alors le système*

$$\begin{aligned} x &= y \bmod n \\ x &= z \bmod m \end{aligned}$$

*possède une solution unique modulo $mn$. Cette solution est donnée par la formule*

$$(zun + yvm) \bmod mn$$

*avec $u$ et $v$ le coefficients de Bezout respectifs de $n$ et $m$.*

**Théorème 3 (Théorème de Bezout)** *Soient $n$ et $m$ deux entiers. $n$ et $m$ sont premiers entre eux si et seulement si il existe deux entiers $u$ et $v$ tels que*

$$un + vm = 1$$

.

**Théorème 4** *[TRC pour $r$ éléments.] Soient $(n_i)$ $r$ entiers premiers entre eux. Soient $(x_i)$ $r$ entiers quelconques ; alors le système*

$$x = x_i \bmod n_i$$

*possède une solution unique modulo $\prod n_i$. Cette solution est trouvée en calculant la suite*

$$\begin{aligned} y_1 &= x_1 \bmod n_1 \\ y_2 &= N_2(C_2(x_2 - y_1) \bmod n_2) + y1 \\ y_3 &= N_3(C_3(x_3 - y_2) \bmod n_3) + y2 \\ &\cdot \\ &\cdot \\ &\cdot \\ y_r &= N_r(C_r(x_r - y_{r-1}) \bmod n_r) + y_{r-1} \end{aligned}$$

*dans laquelle $N_i = \prod_{j<i} n_j$, les coefficients $C_i$ vérifiant $C_i N_i = 1 \bmod n_i$, étant obtenus en appliquant l'algorithme d'Euclide étendu à $N_i$ et $n_i$. Dans ces conditions, le nombre $y_r$ obtenu comme le dernier de la suite, est l'unique solution, dans l'intervalle $[0..\prod n_i[$, du système de congruences.*

**Importance du TRC pour l'arithmétique modulaire.** En raison de la correspondance entre tout nombre de l'intervalle $[0, \prod n_i[$ et les $r$-uplets de $[0, n_1[\times ... \times [0, n_r[$, on dispose d'une arithmétique sur l'intervalle $[0, \prod n_i[$ définie par les correspondances :

$$0 \simeq (0, ..., 0)$$

$$1 \simeq (1, ..., 1)$$

$$x \simeq (x \bmod n_1, ..., (x \bmod n_r)$$

$$(x + y) \bmod \prod n_i \simeq ((x + y) \bmod n_1, ..., ((x + y) \bmod n_r)$$

$$(x - y) \bmod \prod n_i \simeq ((x - y) \bmod n_1, ..., ((x - y) \bmod n_r)$$

$$(xy) \bmod \prod n_i \simeq ((xy) \bmod n_1, ..., ((x - y) \bmod n_r)$$