Introduction à la Cryptologie

Chapitre 4 : Arithmétique modulaire

Michael Eisermann (Institut Fourier, UJF Grenoble)

Année 2008-2009
IF / IMAG, Master 1, S1-S2

document mis à jour le 7 juillet 2009

www-fourier.ujf-grenoble.fr/~eiserm/cours#crypto

Objectifs de ce chapitre

Développement mathématique :

- Comprendre le calcul dans $\mathbb{Z}$ modulo un entier m
- Construire l'anneau quotient $\mathbb{Z}/m$ des entiers modulo m

Développement algorithmique :

- Développer des algorithmes efficaces pour le calcul dans $\mathbb{Z}/m$
- Puissance modulaire rapide (« puissance dichotomique »)

Sommaire

Entiers pairs et impairs

Les nombres entiers se répartissent en entiers pairs et entiers impairs :

$$\bar{0} := \{a \in \mathbb{Z} \mid a \operatorname{rem} 2 = 0\}$$
$$\bar{1} := \{a \in \mathbb{Z} \mid a \operatorname{rem} 2 = 1\}$$

Les tables suivantes résument les règles comme « pair plus impair = impair » ou « pair fois impair = pair » :

$+$	$\bar{0}$	$\bar{1}$
$\bar{0}$	$\bar{0}$	$\bar{1}$
$\bar{1}$	$\bar{1}$	$\bar{0}$

$\cdot$	$\bar{0}$	$\bar{1}$
$\bar{0}$	$\bar{0}$	$\bar{0}$
$\bar{1}$	$\bar{0}$	$\bar{1}$

Entiers modulo 3

Généralisons cette idée des entiers pairs et entiers impairs, et regardons maintenant le reste modulo 3 au lieu de 2.

Ici les nombres entiers se répartissent en trois classes :

$$\bar{0} := \{a \in \mathbb{Z} \mid a \operatorname{rem} 3 = 0\}$$
$$\bar{1} := \{a \in \mathbb{Z} \mid a \operatorname{rem} 3 = 1\}$$
$$\bar{2} := \{a \in \mathbb{Z} \mid a \operatorname{rem} 3 = 2\}$$

Les tables suivantes résument les règles de calculs que l'on observe sur les restes modulo 3 :

$+$	$\bar{0}$	$\bar{1}$	$\bar{2}$
$\bar{0}$	$\bar{0}$	$\bar{1}$	$\bar{2}$
$\bar{1}$	$\bar{1}$	$\bar{2}$	$\bar{0}$
$\bar{2}$	$\bar{2}$	$\bar{0}$	$\bar{1}$

$\cdot$	$\bar{0}$	$\bar{1}$	$\bar{2}$
$\bar{0}$	$\bar{0}$	$\bar{0}$	$\bar{0}$
$\bar{1}$	$\bar{0}$	$\bar{1}$	$\bar{2}$
$\bar{2}$	$\bar{0}$	$\bar{2}$	$\bar{1}$

L'objectif de ce chapitre et de généraliser cette construction de $m = 2$ et $m = 3$ à tout $m \in \mathbb{Z}$ puis d'analyser assez finement la structure qui en résulte.

Processeurs binaires

Tout nombre entier a s'écrit de manière unique en base B comme

$$a = \langle a_n, a_{n-1}, \ldots, a_1, a_0 \rangle_B := a_n B^n + a_{n-1} B^{n-1} + \cdots + a_1 B + a_0.$$

où $a_n, a_{n-1}, \ldots, a_1, a_0 \in \mathbb{Z}$ vérifient $0 \leq a_k < B$ pour tout k.

Calculer modulo B signifie que l'on ne retient que le dernier « chiffre » a_0.

Calculer modulo B^k signifie que l'on ne retient que les k derniers chiffres.

Exemple

Un processeur à 64 bits fournit des « petits entiers » allant de 0 à $2^{64} - 1$.

Or, l'addition et la multiplication fournies ne sont pas celles des entiers !

Si le résultat dépasse la plage $\{0, \ldots, 2^{64} - 1\}$, le processeur ne retient du résultat que les 64 bits bas et supprime d'éventuels bits hauts.

Ceci revient à réduire systématiquement tous les calculs modulo 2^{64}.

Conclusion

La structure mathématique ainsi implémentée n'est pas l'anneau $(\mathbb{Z}, +, \cdot)$ des entiers mais l'anneau quotient $(\mathbb{Z}/2^{64}, +, \cdot)$ des entiers modulo 2^{64}.

La congruence modulo m

Définition (congruence modulo m)

Soient $a, b, m \in \mathbb{Z}$. Si $m \mid a - b$, on dit que a et b sont ***congrus modulo*** m, ou aussi que a est ***congru à*** b ***modulo*** m, noté $a \equiv b \mod m$ ou $a \equiv b \ (m)$.

Exemples

- $a \equiv b \ (0)$ si et seulement si $a = b$.
- $a \equiv b \ (1)$ pour tout $a, b \in \mathbb{Z}$.
- $a \equiv b \ (2)$ si et seulement a et b ont la même parité.

La congruence modulo m est une relation d'équivalence

Proposition (relation d'équivalence)

Pour tout $a, b, c \in \mathbb{Z}$ on a

- $a \equiv a \ (m)$ *(réflexivité)*
- $a \equiv b$ *implique* $b \equiv a$ *(symétrie)*
- $a \equiv b$ *et* $b \equiv c$ *impliquent* $a \equiv c$ *(transitivité)*

Proposition (représentant canonique)

- *Si* $a = qm + r$ *alors* $a \equiv r \ (m)$.
- *En particulier* $r = a \operatorname{rem} m$ *vérifie* $a \equiv r \ (m)$.
- *Si* $m > 0$ *et* $0 \leq r < m$, $0 \leq r' < m$, *alors* $r \equiv r' \ (m)$ *implique* $r = r'$.

Démonstration. Explicitons le dernier point, le reste étant clair.
On peut supposer que $r \geq r'$. Ainsi on obtient $0 \leq r - r' < m$.
Or, la congruence $r \equiv r' \ (m)$ veut dire que $m \mid r - r'$.
Ceci n'est possible que pour $r - r' = 0$. □

Autrement dit, pour $m > 0$ et tout $a \in \mathbb{Z}$ le reste $r = a \operatorname{rem} m$ est l'unique représentant de la classe de congruence de a modulo m vérifiant $0 \leq r < m$.

Calcul modulo m : compatibilité

Proposition (compatibilité avec les opérations)

Si $a \equiv a' \ (m)$ *et* $b \equiv b' \ (m)$ *alors*

- $a + b \equiv a' + b' \ (m)$.
- $a \cdot b \equiv a' \cdot b' \ (m)$.
- $a^n \equiv a'^n \ (m)$.

Démonstration. Par hypothèse on a $a - a' = ms$ et $b - b' = mt$ où $s, t \in \mathbb{Z}$.

- Pour montrer $a + b \equiv a' + b' \ (m)$ on vérifie que $(a + b) - (a' + b') = (a - a') + (b - b') = ms + mt = m(s + t) \in m\mathbb{Z}$.
- Pour montrer $a \cdot b \equiv a' \cdot b' \ (m)$ on vérifie que $a \cdot b - a' \cdot b' = a(b - b') + b'(a - a') = amt + b'ms = m(at + b's) \in m\mathbb{Z}$.
- Par récurrence $a^n = \underbrace{a \cdots a}_{n \text{ fois}} \equiv \underbrace{a' \cdots a'}_{n \text{ fois}} = a'^n \ (m)$. □

Calcul modulo m : exemples

Exemples

Si $a = \langle a_n, a_{n-1}, \ldots, a_1, a_0 \rangle_{10} = a_n 10^n + a_{n-1} 10^{n-1} + \cdots + a_1 10 + a_0$, alors

- $a \equiv a_0 \ (10)$ car $10 \equiv 0 \ (10)$,
- $a \equiv \sum_{k=0}^{n} a_k \ (9)$ car $10 \equiv 1 \ (9)$,
- $a \equiv \sum_{k=0}^{n} (-1)^k a_k \ (11)$ car $10 \equiv -1 \ (11)$.

Ceci généralise les critères bien connus pour tester la divisibilité par 10, 9, 11.

Exercice

Calculer (de tête) la dernière décimale de 3265879^{27247}.

Solution. Calculant modulo 10 on trouve

$$3265879^{27247} \equiv 9^{27247} \equiv (-1)^{27247} \equiv -1 \equiv 9.$$

☺

Relations d'équivalence

Rappel. Une ***relation*** sur un ensemble X est une partie $R \subset X \times X$.
Au lieu de $(x, y) \in R$ on écrit xRy et on dit que x ***est en relation avec*** y.

Définition

Une ***relation d'équivalence*** sur X est une relation $R \subset X \times X$ qui soit

réflexive, $\forall x \in X : xRx$,
symétrique, $\forall (x, y) \in X \times X : xRy \Longrightarrow yRx$,
transitive, $\forall (x, y, z) \in X \times X \times X : xRy \wedge yRz \Longrightarrow xRz$.

Exemple

Pour $m \in \mathbb{Z}$ on a la relation R_m de congruence modulo m sur $\mathbb{Z}$, définie par $xR_m y$ si et seulement si $x \equiv y \ (m)$. C'est une relation d'équivalence.

Exemple

Toute application $f : X \to Y$ définit sur X une relation d'équivalence $R_f := \{(x, x') \in X \times X \mid f(x) = f(x')\}$.

Classes d'équivalence et quotient

Soit X un ensemble et $R \subset X \times X$ une relation d'équivalence.

Définition (classes d'équivalence et quotient)

La ***classe d'équivalence*** d'un élément $x \in X$, toujours par rapport à R, est l'ensemble $\mathfrak{cl}(x) := \{x' \in X \mid x'Rx\}$ formé des éléments équivalents à x.

L'ensemble des classes d'équivalences est noté par $X/R := \{\mathfrak{cl}(x) \mid x \in X\}$ et appelé ***l'ensemble quotient*** de X modulo R.

On définit ***l'application quotient*** $\pi : X \to X/R$ par $\pi(x) := \mathfrak{cl}(x)$.

Si $C \in X/R$ est une classe d'équivalence, alors tout élément $x \in C$ est appelé un ***représentant*** de la classe C.

Observations

Chaque classe d'équivalence $\mathfrak{cl}(x)$ est non vide car $x \in \mathfrak{cl}(x)$ par réflexivité.

Si $y \in \mathfrak{cl}(x)$ alors $\mathfrak{cl}(y) \subset \mathfrak{cl}(x)$ par transitivité, puis $\mathfrak{cl}(y) = \mathfrak{cl}(x)$ par symétrie.

Ainsi deux classes d'équivalences sont soit égales soit disjointes :
si $z \in \mathfrak{cl}(x) \cap \mathfrak{cl}(y)$, alors $\mathfrak{cl}(z) = \mathfrak{cl}(x)$ et $\mathfrak{cl}(z) = \mathfrak{cl}(y)$, donc $\mathfrak{cl}(x) = \mathfrak{cl}(y)$.

Les classes d'équivalence forment une partition

Définition

Une ***partition*** d'un ensemble X est une famille $P \subset \mathscr{P}(X)$ de sous-ensembles de X telle que

- Toute sous-ensemble $C \in P$ est non vide.
- Deux sous-ensembles $C, C' \in P$ sont soit égaux soit disjoints.
- Chaque élément $x \in X$ appartient à un (unique) sous-ensemble $C \in P$.

Proposition

Les éléments du quotient X/R forment une partition de X :
tout élément $x \in X$ appartient à exactement une classe $C \in X/R$.

Réciproquement, toute partition $P \subset \mathscr{P}(X)$ définit une relation d'équivalence sur X par $R = \{(x, x') \in X \times X \mid \exists C \in P : x \in C \wedge x' \in C\}$.

L'ensemble quotient $\mathbb{Z}/_m$ des entiers modulo m

Définition

$\mathbb{Z}/_m = \mathbb{Z}/m\mathbb{Z}$ est l'ensemble quotient de $\mathbb{Z}$ par la relation de congruence R_m.
On note $\pi_m \colon \mathbb{Z} \to \mathbb{Z}/_m$ l'application quotient définie par $\pi_m(x) := \mathfrak{cl}(x)$.

On a la propriété fondamentale : $\mathfrak{cl}(x) = \mathfrak{cl}(y)$ si et seulement si $x \equiv y \ (m)$.

Slogan

La congruence « $\equiv$ » sur $\mathbb{Z}$ devient l'égalité « $=$ » dans le quotient $\mathbb{Z}/_m$.

C'est un petit pas pour une définition, mais un bond de géant conceptuel.

Notation

On écrit π au lieu de π_m si l'entier m est clair par le contexte.
La classe de $x \in \mathbb{Z}$ dans $\mathbb{Z}/_m$ est notée $\mathfrak{cl}(x) = \pi_m(x) = \pi(x) = \overline{x}$.

Classes de congruence modulo m

Exemples

- Soit $m = 0$. Ici $a \equiv b \ (0)$ ssi $a = b$, donc pour tout $a \in \mathbb{Z}$ on a $\mathfrak{cl}(a) = \{a\}$. Ainsi $\mathbb{Z}/0\mathbb{Z} = \{\{a\} \mid a \in \mathbb{Z}\}$ et $\pi_0 \colon \mathbb{Z} \to \mathbb{Z}/0\mathbb{Z}$, $n \mapsto \{n\}$ est une bijection.
- Soit $m = 1$. Ici $a \equiv b \ (1)$ pour tout $a, b \in \mathbb{Z}$, donc $\mathfrak{cl}(a) = \mathbb{Z}$. Ainsi $\mathbb{Z}/1\mathbb{Z} = \{\mathbb{Z}\} = \{\mathfrak{cl}(0)\}$.
- Soit $m = 2$. Ici $a \equiv b \ (2)$ ssi a et b ont la même parité. Il y a donc deux classes : $\bar{0} = \{\text{entiers pairs}\}$, $\bar{1} = \{\text{entiers impairs}\}$. Ainsi $\mathbb{Z}/2\mathbb{Z} = \{\bar{0}, \bar{1}\}$.

Nous avons déjà constaté le cas général :

Proposition

Pour $m \geq 1$ et tout $a \in \mathbb{Z}$ le reste $r = a \operatorname{rem} m$ est l'unique représentant de la classe de congruence de a modulo m vérifiant $0 \leq r < m$.

Corollaire

Pour $m \geq 1$ l'ensemble quotient $\mathbb{Z}/_m = \{\bar{0}, \bar{1}, \ldots, \overline{m-1}\}$ est de cardinal m.

Pour $m = 0$, par contre, l'ensemble quotient $\mathbb{Z}/_0 \cong \mathbb{Z}$ est infini.

Addition et multiplication passent au quotient

Proposition

La compatibilité de la congruence avec l'addition et la multiplication dans $\mathbb{Z}$ permet de définir une addition et une multiplication sur le quotient $\mathbb{Z}/_m$ par

$$+ \colon \mathbb{Z}/_m \times \mathbb{Z}/_m \to \mathbb{Z}/_m, \qquad \mathfrak{cl}(a) + \mathfrak{cl}(b) := \mathfrak{cl}(a+b)$$
$$\cdot \colon \mathbb{Z}/_m \times \mathbb{Z}/_m \to \mathbb{Z}/_m, \qquad \mathfrak{cl}(a) \cdot \mathfrak{cl}(b) := \mathfrak{cl}(a \cdot b)$$

Démonstration. Il faut assurer que ces opérations sont bien définis, c'est-à-dire le résultat ne dépend pas du choix des représentants a, b.

Si $\mathfrak{cl}(a) = \mathfrak{cl}(a')$ et $\mathfrak{cl}(b) = \mathfrak{cl}(b')$, alors $a \equiv a' \ (m)$ et $b \equiv b' \ (m)$ et

- $a + b \equiv a' + b' \ (m)$, donc $\mathfrak{cl}(a+b) = \mathfrak{cl}(a'+b')$,
- $a \cdot b \equiv a' \cdot b' \ (m)$, donc $\mathfrak{cl}(a \cdot b) = \mathfrak{cl}(a' \cdot b')$. □

L'anneau $\mathbb{Z}/m$ des entiers modulo m

La construction du quotient $\pi\colon \mathbb{Z} \to \mathbb{Z}/m$ se résume dans le résultat suivant :

Théorème

Sur l'ensemble quotient $\mathbb{Z}/m$ il existe une unique addition $+$ et une unique multiplication $\cdot$ telles que $\pi(a+b) = \pi(a) + \pi(b)$ et $\pi(a \cdot b) = \pi(a) \cdot \pi(b)$.

Le triplet $(\mathbb{Z}/m, +, \cdot)$ ainsi construit est un anneau :

(A1 : associativité)	$\forall a, b, c :$	$(a+b)+c = a+(b+c)$
(A2 : commutativité)	$\forall a, b :$	$a+b = b+a$
(A3 : élément neutre)	$\exists 0\ \forall a :$	$0+a = a$
(A4 : élément opposé)	$\forall a\ \exists b :$	$a+b = 0$
(M1 : associativité)	$\forall a, b, c :$	$(a \cdot b) \cdot c = a \cdot (b \cdot c)$
(M2 : commutativité)	$\forall a, b :$	$a \cdot b = b \cdot a$
(M3 : élément neutre)	$\exists 1 \neq 0\ \forall a :$	$1 \cdot a = a$
(D : distributivité)	$\forall a, b, c :$	$a \cdot (b+c) = (a \cdot b) + (a \cdot c)$

Démonstration. Exercice ! □

Implémentation des opérations modulo m

Slogan

Calculer dans le quotient $\mathbb{Z}/m$, c'est calculer dans $\mathbb{Z}$ modulo m,
c'est donc la réduction au reste de la division euclidienne par m.

Algorithme 4.1 addition modulo m

Entrée: trois entiers a, b, m tels que $0 \leq a < m$ et $0 \leq b < m$
Sortie: l'entier c vérifiant $0 \leq c < m$ tel que $c \equiv a + b \ (m)$.

```
c ← a + b                          // on effectue l'addition dans ℤ
si c ≥ m alors c ← c − m            // on évite ainsi la division euclidienne
retourner c
```

Algorithme 4.2 multiplication modulo m

Entrée: trois entiers a, b, m tels que $0 \leq a < m$ et $0 \leq b < m$
Sortie: l'entier c vérifiant $0 \leq c < m$ tel que $c \equiv a \cdot b \ (m)$.

```
retourner (a · b) rem m            // réduire le produit modulo m
```

Exercice

Écrire les tables d'addition et de multiplication de $\mathbb{Z}/4$ puis de $\mathbb{Z}/5$.

Puissance modulaire : premières tentatives

Objectif : Calculer $a^n \operatorname{rem} m$ de manière efficace.

Algorithme 4.3 puissance modulaire (très inefficace)

Entrée: trois entiers a, n, m tels que $0 \leq a < m$ et $n \geq 0$
Sortie: l'entier p vérifiant $0 \leq p < m$ tel que $p \equiv a^n \ (m)$

```
p ← 1
tant que n > 0 faire
    p ← p · a                      // le produit est calculé dans ℤ
    n ← n − 1
fin tant que
retourner p rem m                  // réduction modulo m à la fin
```

Algorithme 4.4 puissance modulaire (moins inefficace)

Entrée: trois entiers a, n, m tels que $0 \leq a < m$ et $n \geq 0$
Sortie: l'entier p vérifiant $0 \leq p < m$ tel que $p \equiv a^n \ (m)$

```
p ← 1
tant que n > 0 faire
    p ← (p · a) rem m              // réduction modulo m chaque fois
    n ← n − 1
fin tant que
retourner p                        // on sait déjà que 0 ≤ p < m
```


Puissance modulaire rapide : l'idée

Il est facile de calculer les puissances $a^1, a^2, a^4, a^8, a^{16}, \ldots$:
on pose $a_0 := a$ puis on calcule $a_{i+1} := a_i \cdot a_i$ par récurrence.

Ainsi on calcule $(a_0, a_1, a_2, \ldots, a_{\ell-1})$ avec $\ell - 1$ multiplications seulement !

Quant à une puissance a^n pour $n \geq 1$ quelconque on écrit n en base 2 :

$$n = \sum_{i=0}^{\ell-1} n_i 2^i \quad \text{où} \quad n_i \in \{0, 1\}.$$

Nous pouvons supposer que $n_{\ell-1} = 1$, donc $2^{\ell-1} \leq n < 2^\ell$.
En supprimant les termes nuls on obtient $n = \sum_{i \in I} 2^i$, d'où

$$a^n = a^{\sum_{i \in I} 2^i} = \prod_{i \in I} a^{2^i} = \prod_{i \in I} a_i.$$

Ainsi on calcule a^n avec $\ell - 1 + |I|$ multiplications seulement !

À noter que $\ell = \operatorname{len}(n)$ est la longueur de n est base 2,
et $|I|$ est le nombre de chiffres 1 dans ce développement.

Au total on calcule a^n avec moins de $2 \operatorname{len}(n)$ multiplications.

Puissance modulaire rapide : l'algorithme

Algorithme 4.5 puissance modulaire rapide

Entrée: trois entiers a, n, m tels que $0 \leq a < m$ et $n \geq 0$
Sortie: l'entier p vérifiant $0 \leq p < m$ tel que $p \equiv a^n \ (m)$

```
p ← 1                                   // invariant pa^n ≡ a^n modulo m
tant que n > 0 faire
   tant que n est pair faire
      a ← (a · a) rem m,  n ← n/2       // pa^n reste invariant modulo m
   fin tant que
   p ← (p · a) rem m,  n ← n − 1        // pa^n reste invariant modulo m
fin tant que
retourner p                             // ici n = 0 donc p = pa^n
```

Théorème

L'algorithme de puissance modulaire rapide explicité ci-dessus est correct :

- *Il se termine après moins de* $2\,\mathrm{len}(n)$ *itérations.*
- *Il renvoie la valeur* $p = a^n \bmod m$ *comme spécifiée.*

Démonstration. Après au plus deux multiplications on divise n par deux.
Chacune des multiplications préserve la valeur de pa^n modulo m.
À la fin on a $n = 0$, donc $p = pa^n$ est la valeur cherchée. □

Puissance modulaire rapide : exemple

Exercice

Calculer les trois dernières décimales de 23^{1030}.

Solution. Calculons d'abord $a_k = 23^{2^k} \bmod 1000$:

k	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}
a_k	023	529	841	281	961	521	441	481	361	321	041

En écrivant l'exposant 1030 en binaire on trouve $1030 = 2^{10} + 2^2 + 2^1$ puis

$$23^{1030} = 23^{2^{10}+2^2+2^1} = 23^{2^{10}} \cdot 23^{2^2} \cdot 23^{2^1} \equiv 41 \cdot 841 \cdot 529 \equiv 41 \cdot 889 \equiv 449.$$

De manière équivalente, traçons les étapes de l'algorithme :

n	1030	515	514	257	256	128	64	32	...
a	23	529	529	841	841	281	961	521	...
p	1	1	529	529	889	889	889	889	...

n	...	64	32	16	8	4	2	1	0
a	...	961	521	441	481	361	321	041	041
p	...	889	889	889	889	889	889	889	449

☺