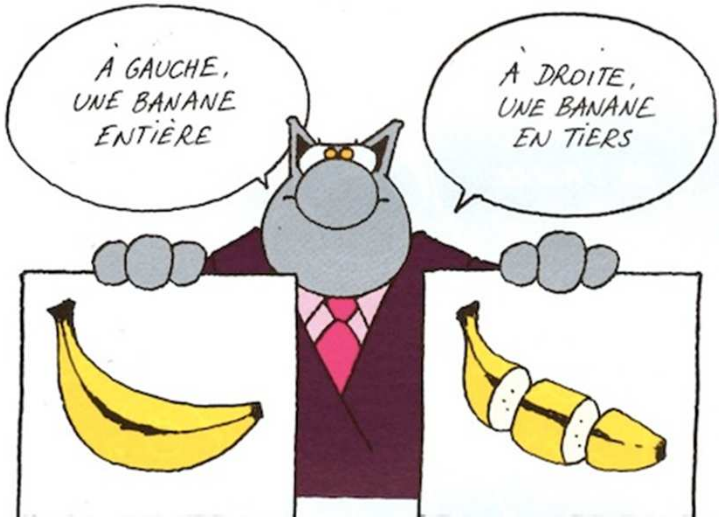
# INF1130

# Mathématiques pour l'informatique

**Zied Zaier, PhD**
Département d’informatique
Université du Québec à Montréal




Cours 6

# NOMBRES ENTIERS ET DIVISION

**CE DOCUMENT EST INSPIRÉ DES TRAVAUX DES PROFESSEURS KENNETH H. ROSEN ET TIMOTHY WALSH.**

# Nombres entiers et division

- Vous savez déjà ce que sont des entiers et la division ….
- **MAIS:** Certaines notations, terminologies, et théorêmes associés à ces concepts peuvent être utiles.
- Ces concepts sont la base de la *théorie des nombres*.
  - Important dans plusieurs algorithmes modernes (fonctions de hashage, cryptographie, signatures digitales).

# Notions de Diviseur , Facteur, Multiple

- Posons $a,b \in \mathbf{Z}$ avec $a \neq 0$.
- **Déf.:** $a|b \equiv$ "*a divise b*" $:\equiv (\exists c \in \mathbf{Z}: b = ac)$
  "Il existe un entier *c* tel que *c* multiplié par *a* égal *b*."
  – Exemple: $3|12 \Leftrightarrow$ **Vrai**, mais $3|7 \Leftrightarrow$ **Faux**.
- SSI *a* **divise** *b*, alors *a* est un *facteur* ou un *diviseur* de *b*, et *b* est un *multiple* de *a*.
- Ex.: "*b* est pair" $:\equiv 2|b$.

# Faits sur la notion de Diviseur

- **Théorème:** $\forall a,b,c \in \mathbf{Z}$:
  1. $a|0$
  2. $(a|b \wedge a|c) \rightarrow a \mid (b + c)$
  3. $a|b \rightarrow a|bc$
  4. $(a|b \wedge b|c) \rightarrow a|c$
- **Preuve** de (2): $a|b$ veut dire que qu'il existe un $s$ tel que $b=as$, et $a|c$ veut dire qu'il existe un $t$ tel que $c=at$, et $b+c = as+at = a(s+t)$, et qu'ainsi $a|(b+c)$.■

# Version détaillée de la preuve.

- Démontrez que

$$\forall a,b,c \in \mathbf{Z}: (a|b \wedge a|c) \rightarrow a \mid (b + c).$$

- Sachant que $a$, $b$, $c$ des entiers tels que $a|b$ et $a|c$, démontrez que $a \mid (b + c)$.
- Par définition de $|$, et sachant que $\exists s$: $b=as$, et $\exists t$: $c=at$. Sachant que $s$, $t$, sont aussi des entiers.
- Alors $b+c = as + at = a(s+t)$, donc $\exists u$: $b+c=au$, où $u=s+t$. Par conséquent $a|(b+c)$.

# Nombres premiers

- Un nombre $p>1$ est premier SSI il n'est pas le produit de deux nombres entiers plus grand que 1:

  $$p>1 \wedge \neg\exists a,b \in \mathbf{N}\text{:}\ a>1,\ b>1,\ ab=p.$$

- Les seuls facteurs positifs d'un nombre premier $p$ sont 1 et $p$. Ex: 2,3,5,7,11,13...
- Les nombres entiers non premiers plus grand que 1 sont composés, puisqu'ils sont déduits, composés, de la multiplication d'au moins deux entiers plus grands que 1.

# Théorême fondamental de l'Arithmétique

**Sa " Factorisation en nombres premiers"**

- Chaque entier positif a une représentation *unique* de produits d'une série non décroissante de 0 ou plus nombres premiers (facteurs premiers).
  - Exemples:
    - $4 = 2\cdot2$
    - $2000 = 2\cdot2\cdot2\cdot2\cdot5\cdot5\cdot5$; $2001 = 3\cdot23\cdot29$; $2002 = 2\cdot7\cdot11\cdot13$;
    - $1024 = 2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2 = 2^{10}$

# Application des nombres premiers

- Quand vous accédé à un site web sécurisé, le fureteur et le serveur web échange des données cryptées utilisant peut être un mode d'encryption RSA.
- Le cryptage à clé-public implique l'échange de la clé public contenant le produit *pq* de deux grands nombres premiers aléatoires *p* et *q* (une clé privée) laquelle doit être tenue secrète par un tier donné.
- Donc, la sécurité des transactions sur le site web dépend du fait que tous les algorithmes de décompositions connus sont indéchiffrables
  - **Note:** Il existe malgré tout un algorithme de décomposition déchiffrable, alors le RSA n'est pas sécure.

# L'Algorithme de division (théorême)

- C'est en fait un *théorême*, pas un algorithme…
- **Théorême:** Pour un *dividende entier a* et un *diviseur* $d \neq 0$, il existe un *quotient q* et un *reste* $r \in \mathbf{N}$ entier unique, tels que

  $$a = dq + r \text{ et } 0 \leq r < |d|.$$

- Formellement, le théorême est:

  $$\forall a,d \in \mathbf{Z},\ d \neq 0: \exists\ q,r \in \mathbf{Z}: 0 \leq r < |d|,\ a = dq + r.$$

- $q$ et $r$ sont déduits par: $q = \lfloor a/d \rfloor$, $r = a - qd$.

# Plus grand commun diviseur

- Le *plus grand commun diviseur pgcd*(*a*,*b*) des entiers *a*,*b* (!= 0) est le plus grand (plus positif) entier *d* qui est un diviseur de *a* et *b*.

$$d = pgcd(a,b) = \max(d\colon d|a \wedge d|b) \Leftrightarrow$$
$$d|a \wedge d|b \wedge \forall e \in \mathbf{Z}, (e|a \wedge e|b) \rightarrow d \geq e$$

- **Exemple:** pgcd(24,36)=?
  Diviseurs communs positifs: 1,2,3,4,6,12.
  *pgcd*(24,36) = 12.

# PGCD (raccourci)

- Si une décomposition en facteurs premiers est
  $a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}$ et $b = p_1^{b_1} p_2^{b_2}, \dots p_n^{b_n}$
  alors le PGCD est donné par:

$$\boxed{p\,\mathrm{gcd}(a,b) = p_1^{\min(a_1,b_1)} p_2^{\min(a_2,b_2)} \dots p_n^{\min(a_n,b_n)}.}$$

- Exemples:
  - $a$=84=2·2·3·7 = $2^2 \cdot 3^1 \cdot 7^1$
  - $b$=96=2·2·2·2·2·3 = $2^5 \cdot 3^1 \cdot 7^0$
  - *pgcd*(84,96) = $2^2 \cdot 3^1 \cdot 7^0$ = 2·2·3 = 12.

# Premier relatif ou copremier

- Les entiers *a* et *b* sont premiers *relatifs* **ssi** le *pgcd*(a,b) = 1.
  - **Exemple:** 21 et 10 ne sont pas premiers, mais sont *copremiers*. $21=3\cdot 7$ et $10=2\cdot 5$, donc ils n'ont pas de facteurs communs

    > 1, donc leur *pgcd* = 1.
- Un ensemble d'entiers $\{a_1,a_2,\ldots\}$ est *copremier* paire à paire si toutes les paires $(a_i, a_j)$, pour $i\neq j$, sont copremiers.

# Plus petit commun multiple

- *ppcm*(*a*,*b*) des entiers positifs *a*, *b*, est le plus petit entier positif qui est un multiple de *a* et *b*. *E.g.* *ppcm*(6,10)=30= $2^1 \cdot 3^1 \cdot 5^1$

  $m = ppcm(a,b) = \min(m: a|m \wedge b|m) \Leftrightarrow$
  $a|m \wedge b|m \wedge \forall n \in \mathbf{Z}: (a|n \wedge b|n) \rightarrow (m \leq n)$

- Si la décomposition en facteurs premiers est

  $a = p_1^{a_1} p_2^{a_2} \ldots p_n^{a_n}$ et $\qquad b = p_1^{b_1} p_2^{b_2} \ldots p_n^{b_n}$

- Le *ppcm* est donné par:

$$\boxed{\mathrm{ppcm}(a,b) = p_1^{\max(a_1,b_1)} p_2^{\max(a_2,b_2)} \ldots p_n^{\max(a_n,b_n)}.}$$

# L'opérateur modulo **mod**

- Un opérateur de *reste* de la division entière.
- Posons $a,d \in \mathbf{Z}$ avec $d>1$. Alors $a$ **mod** $d$ donne le reste $r$ de l'"algorithme" de la division entière avec un dividende $a$ et un diviseur $d$; *i.e.* le reste quand $a$ est divisé par $d$.
  - Utilisation de la division longue.
- Le modulo ($a$ **mod** $d$) peut être calculé par:

$$a - d \cdot \lfloor a/d \rfloor.$$

- En langages C/C++/Java, "%" => mod.

# Congruence modulaire

- Posons $a,b \in \mathbf{Z}$, $m \in \mathbf{Z}^+$.

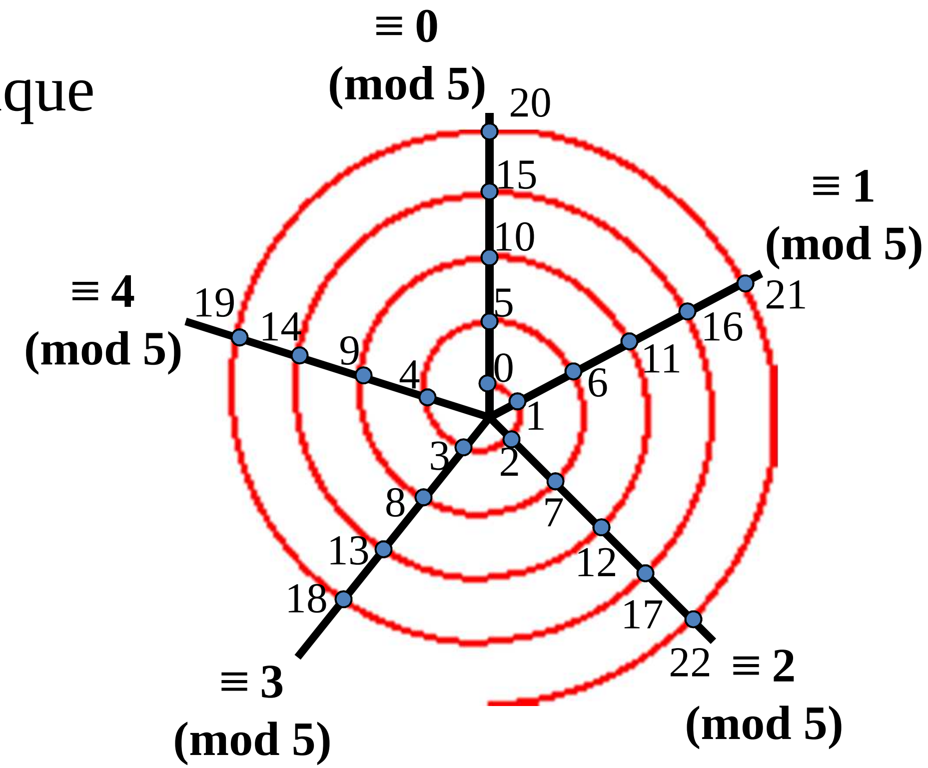
  Où $\mathbf{Z}^+ = \{n \in \mathbf{Z} \mid n > 0\} = \mathbf{N} - \{0\}$ (entiers +).

- Alors *a est congruent à b modulo m*, écrit: "$a \equiv b \pmod{m}$", **ssi** $m \mid a - b$.
  - Note: Le symbole de congruence "≡" à aussi été utilisé avec un autre sens "est défini par".

- Aussi équivalent à: $(a - b) \bmod m = 0$.

# Visualisation en spirale du **mod**

Exemple:
Arithmétique
modulo-5



# Théorême de congruence utiles

- **Théorême:** Posons $a,b \in \mathbf{Z}$, $m \in \mathbf{Z}^{+}$. alors:

  $a \equiv b \pmod{m} \Leftrightarrow \exists k \in \mathbf{Z}\ a = b + km$.

- **Théorême:** Posons $a,b,c,d \in \mathbf{Z}$, $m \in \mathbf{Z}^{+}$. alors si $a \equiv b \pmod{m}$ et $c \equiv d \pmod{m}$,

  alors:

  - $a+c \equiv b+d \pmod{m}$, et
  - $ac \equiv bd \pmod{m}$

# Théorême de congruence utiles

- **Démonstration:** Sachant que $a \equiv b \pmod{m}$ et $c \equiv d \pmod{m}$, il existe des entiers $s$ et $t$ tels que

$$b = a + sm \text{ et } d = c + tm$$

donc:

$$b + d = (a + sm) + (c + tm) = (a + c) + m(s + t)$$

ainsi:

$$a + c \equiv b + d \pmod{m}$$

# Théorême de congruence utiles

- **Démonstration:** Sachant que $a \equiv b \pmod{m}$ et $c \equiv d \pmod{m}$, il existe des entiers $s$ et $t$ tel que

  $b = a + sm$ et $d = c + tm$

  *donc:*

$$bd = (a + sm)(c + tm) = ac + m(at + cs + stm)$$

  *ainsi:*

$$ac \equiv bd \pmod{m}$$

# Slides supplémentaires

Facultatif

# Exemple: Encryption de message

1. Convertion du message en majuscule.
2. Chaque lettre est associée à un nombre de 1 à 26.
3. Appliquer une fonction modulaire (mod) inversible à chaque nombre.
4. Convertion arrière des codes numériques en lettres.

# Lettre ←→ Nombre
# Table de Conversion

| A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |

| N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |

# Exemple de message encrypté

- Avec une fonction d'encryption:

$$f\,(a) = (3a + 9)\ \mathbf{mod}\ 26$$

- Message à encrypter

1. MATHEMATIQUES
2. 13,1,20,8,5,13,1,20,9,17,21,4,19
3. 22,12,17,7,24,22,12,17,10,8,20,21,14
4. $\boxed{\text{VLQGXVLQJHTUN}}$

**Message encrypté**

# Exemple de message encrypté

- La décryption est similaire mais en utilisant la forme inverse de la fonction d’encryption.
- L’inverse de

$$f(a) = (3a + 9) \ \mathbf{mod}\ 26$$

est donné par:

$$g(a) = 9\,(a - 9) \ \mathbf{mod}\ 26 = (9a - 3) \ \mathbf{mod}\ 26$$

# Inverse Modulaire

- Avec une fonction d'encryption de forme générale: $f(a) = (sa + t)$ **mod** 26
- Si nous avons une fonction d'encryption:

$$f(a) = (3a + 9) \textbf{ mod } 26$$

- Nous exprimons la fonction inverse par:

$$g(a) = (9a - 3) \textbf{ mod } 26$$

- Vérifions que: $g(f(a)) \equiv g(3a+9) \pmod{26}$

  $\equiv 9(3a+9)-3 \pmod{26} \equiv 27a+81-3 \pmod{26}$

  $\equiv 27a+78 \pmod{26} \equiv a \pmod{26}$.
- Avec $a$ dans l'intervalle [0,25] nous avons

$g(f(a)) = a$ donc $g$ et $f$ sont des inverses.

# Inverse Modulaire

- Comment trouver une forme inverse de façon analytique? Par exemple: $f(a) = 3a \textbf{ mod } 26$
- Cherchons une constante $x$ et un inverse de la forme: $g(a) = xa$
- Alors la condition $g(f(a)) \equiv a \pmod{26}$ donne:

$$g(f(a)) \equiv x \cdot 3a \pmod{26} \equiv a \pmod{26}$$

- Si nous avons une sol'n pour $a$=1, ça fonctionne aussi pour toutes les autres valeurs de $x$. Donc avec $a$=1 nous avons:

$$3x \equiv 1 \pmod{26}$$

I.e. nous cherchons un *inverse* à **3 modulo 26**.

# Inverse Modulaire

- DÉFINITION: L’ ***inverse*** de $e$ modulo $N$ est le nombre $d$ entre 1 et $N$-1 tel que

$$de \equiv 1 \pmod{N}$$

- ***Si*** ce nombre existe: Quel est l’inverse de **3 modulo 26**?
  - 9 puisque $9 \cdot 3 = 27 \equiv 1 \pmod{26}$.
- L’inverse de 4 mod 8 ??
  - Aucun puisque $4x$ mod 8 est 0
- $\boldsymbol{e}$ a un inverse **modulo *N*** SSI $\boldsymbol{e}$ et $\boldsymbol{N}$ sont premiers relatifs.

# Inverse Modulaire

- Si $a$ et $b$ sont des entiers positifs, le pgcd de $a$ et $b$ peut être exprimé comme une combinaison linéaire de $a$ et $b$. I.e., des entiers $s,t$ pour lesquels:

$$\mathbf{pgcd}(a,b) = sa + tb$$

# Exemple: Inverse Modulaire

$5 \cdot 14 - 3 \cdot 23 = 1$ découle de:

- $\gcd(14,23) = 1$
  - Un nombre qui divise 14 et 23 divise aussi 1
- **L'inverse de 14 modulo 23 est 5**
  - $5 \cdot 14 = 1 + 3 \cdot 23$
  - $5 \cdot 14 \equiv 1 \pmod{23}$
- "Un" inverse de 23 modulo 14 est -3
  - $-3 \cdot 23 = 1 - 5 \cdot 14$
  - $-3 \cdot 23 \equiv 1 \pmod{14}$
  - $11 \cdot 23 \equiv 1 \pmod{14}$
  - "L'" inverse est 11

# Inverse Modulaire

- Si un inverse $d$ existe pour $e$ modulo $N$, nous avons $de \equiv 1 \pmod{N}$ tel que pour un entier $k$, $de = 1 + kN$, donc $1 = de - kN$.
- Cette équation implique que pour n'importe quels nombres entiers divisent $e$ et $N$ doivent diviser 1, donc doivent être 1, donc $e, N$ sont premier relatifs.

# Inverse Modulaire

- Supposons que $e,N$ sont premiers relatifs. Nous pouvons écrire:

  $\mathbf{1 = \mathit{se} + \mathit{tN}}$. Avec $\mathbf{\mathit{se} = 1\text{-}\mathit{tN}}$.
- En appliquant mod $N$ de chaque côté donne: $\mathbf{\mathit{se} \equiv 1 \ (mod\ \mathit{N})}$ .
- Donc $s$ peut être considéré comme un inverse de $e$ sauf qu'il peut être hors de l'intervalle donc fixons $\mathbf{\mathit{d} = \mathit{s} \bmod \mathit{N}}$.

# Algorithme d'Euclide étendu (suite sec. 4)

- L'algorithme d'Euclide étendu permet de déduire des inverses explicites à partir d'entiers *s* et *t*.
- Cet algo. est une version améliorée de l'algo. D'Euclide régulier sauf qu'il conserve le quotient $q = x/y$ ainsi que déjà le reste $r = x$ **mod** $y$.
- Ce qui permet d'écrire le pgcd(a,b) sous forme d'une combinaison linéaire de *a* et *b*.

# Algorithme d'Euclide étendu
# Exemples

pgcd(33,77)

| Étape | $x = qy + r$ | $x$ | $y$ | pgcd = $ax+by$ |
|---|---|---|---|---|
| 0 | - | 33 | 77 | |
| 1 | 33=0·77+33 | 77 | 33 | 11= 77 - 2·(33-0·77)<br>= **-2**·33 + **1**·77 |
| 2 | 77=2·33+11 | 33 | 11 | 11 = 77 - 2·33 |
| 3 | 33=3·11+0 | 11 | 0 | Solutionner pour $r$ |

Donc $s$ = -2 et $t$ = 1

# Algorithme d'Euclide étendu
## Exemples

inverse de 244 modulo 117

pgcd(244,117):

| Étape | $x = qy + r$ | $x$ | $y$ | pgcd = $ax+by$ |
|---|---|---|---|---|
| 0 | - | 244 | 117 | |
| 1 | 244=2·117+10 | 117 | 10 | 1= 3·117-35·(244- 2·117)<br>= **-35**·244+**73**·117 |
| 2 | 117=11·10+7 | 10 | 7 | 1=-2·10+3·(117-11·10)<br>= 3·117-35·10 |
| 3 | 10=7+3 | 7 | 3 | 1=7-2·(10-7)<br>= -2·10+3·7 |
| 4 | 7=2·3+1 | 3 | 1 | 1=7-2·3 |
| 5 | 3=3·1+0 | 1 | 0 | Solutionner pou r. |