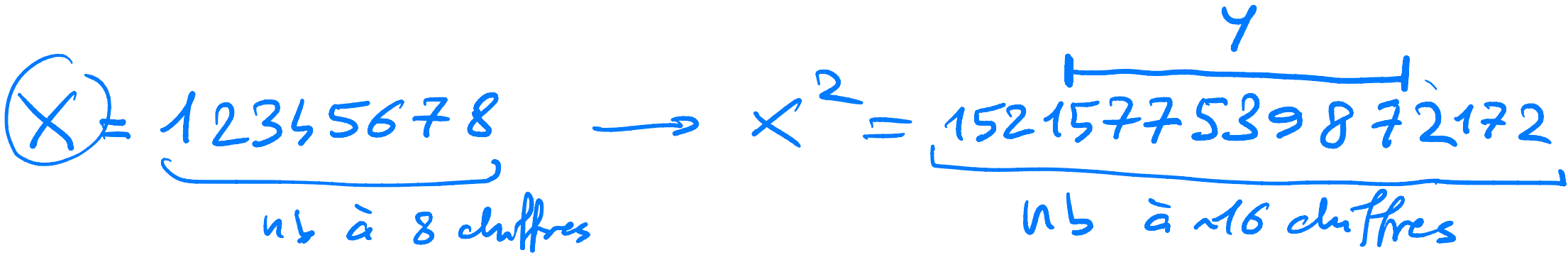
Génération de nombres aléatoires (→ recherche de nombres premiers)

Cours Turing – Semaine 9

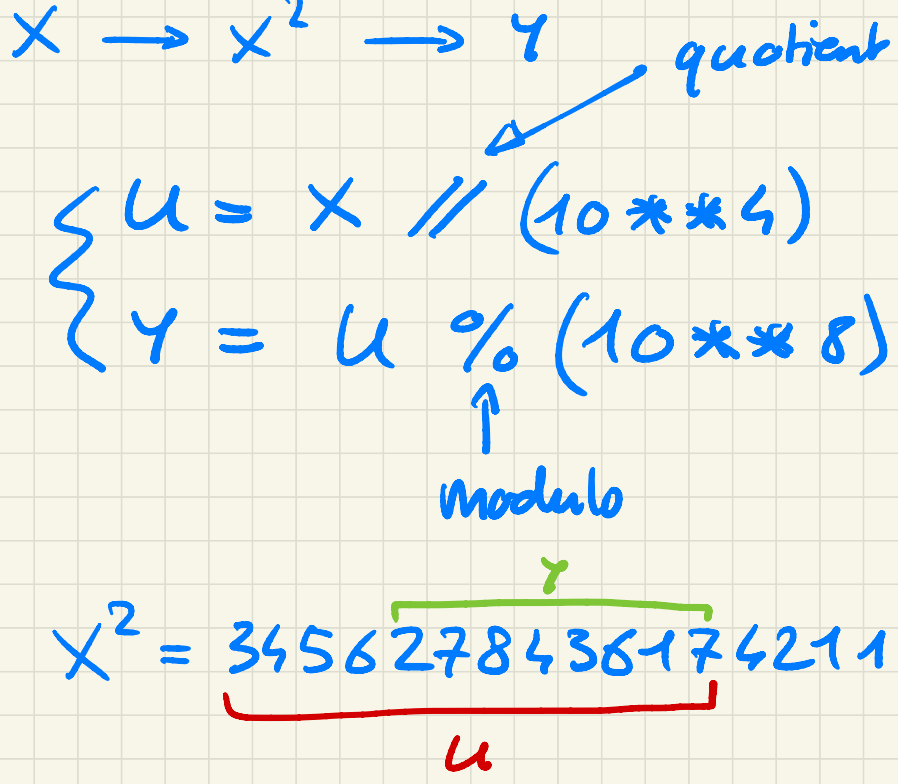
Génération de nombres aléatoires
Premier essai: la méthode des “carrés tronqués”



X = 57 → x^2 = 3249

Y = 24 → y^2 = 0576

X = 60 → x^2 = 3600


$x \longrightarrow x^2 \longrightarrow y$
quotient
$u = x \,//\, (10**4)$
$y = u \,\%\, (10**8)$
modulo
$x^2 = 345627843617\,4211$
y
u

Deuxième essai:
les “générateurs à congruence linéaire”

paramètres a, b et m

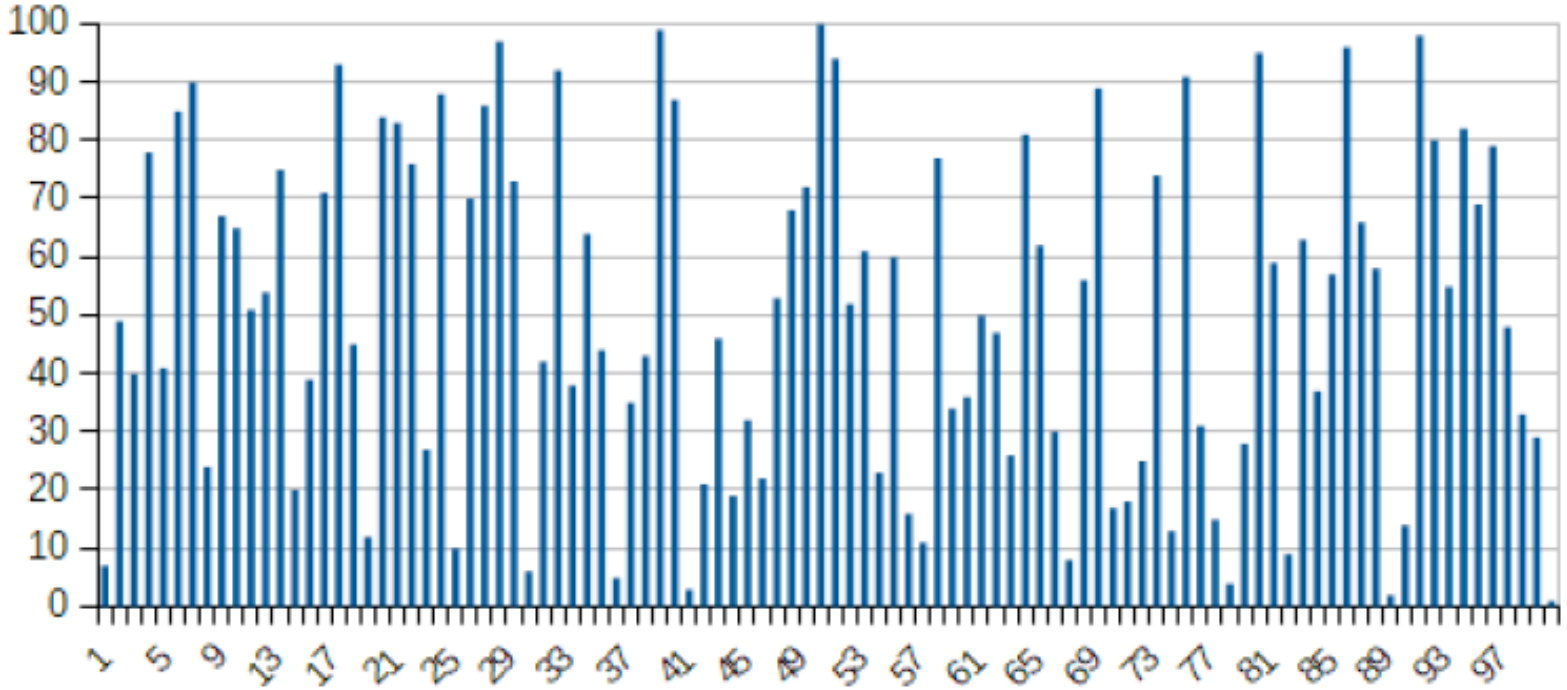
$$x \longmapsto (ax+b) \pmod m = y$$

$:=$ reste de la division euclidienne de $ax+b$ par m $\quad \in \{0..m-1\}$

Ex: $101 \pmod 7 = 3$

car $101 = 7 \cdot 14 + 3$

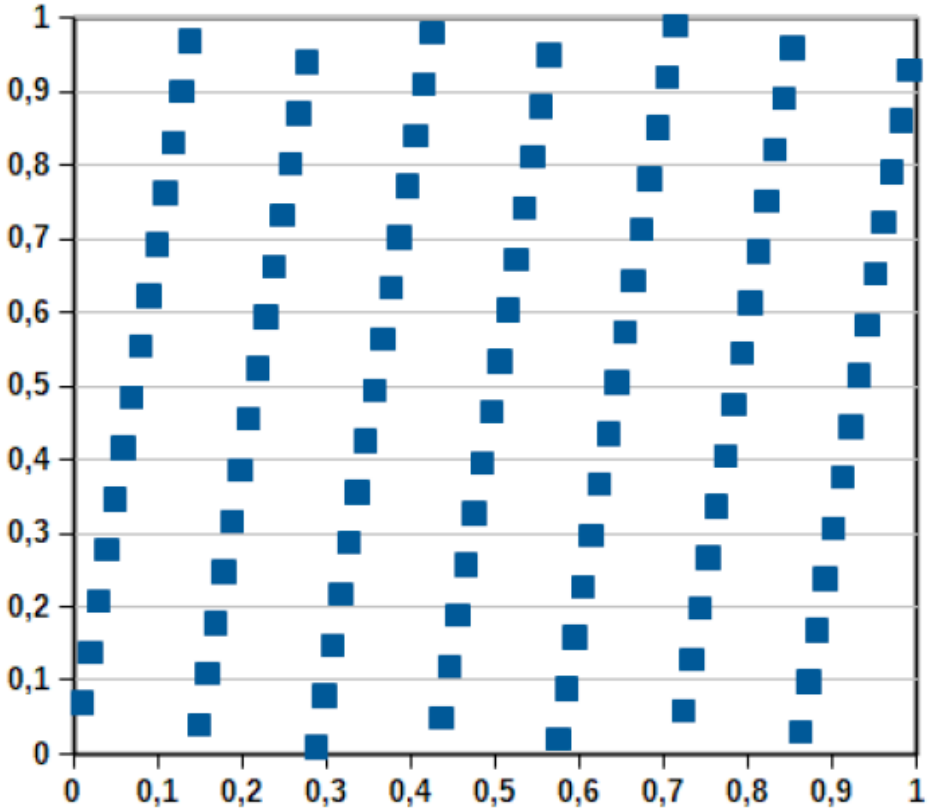
Générateurs à congruence linéaire



$m = 101$, $a = 7$, $b = 0$, $x = 1$ (départ)

Générateurs à congruence linéaire

$$\frac{(a \cdot x + b)(mod\ m)}{m}$$



$$\frac{x}{m}$$

Conversions:

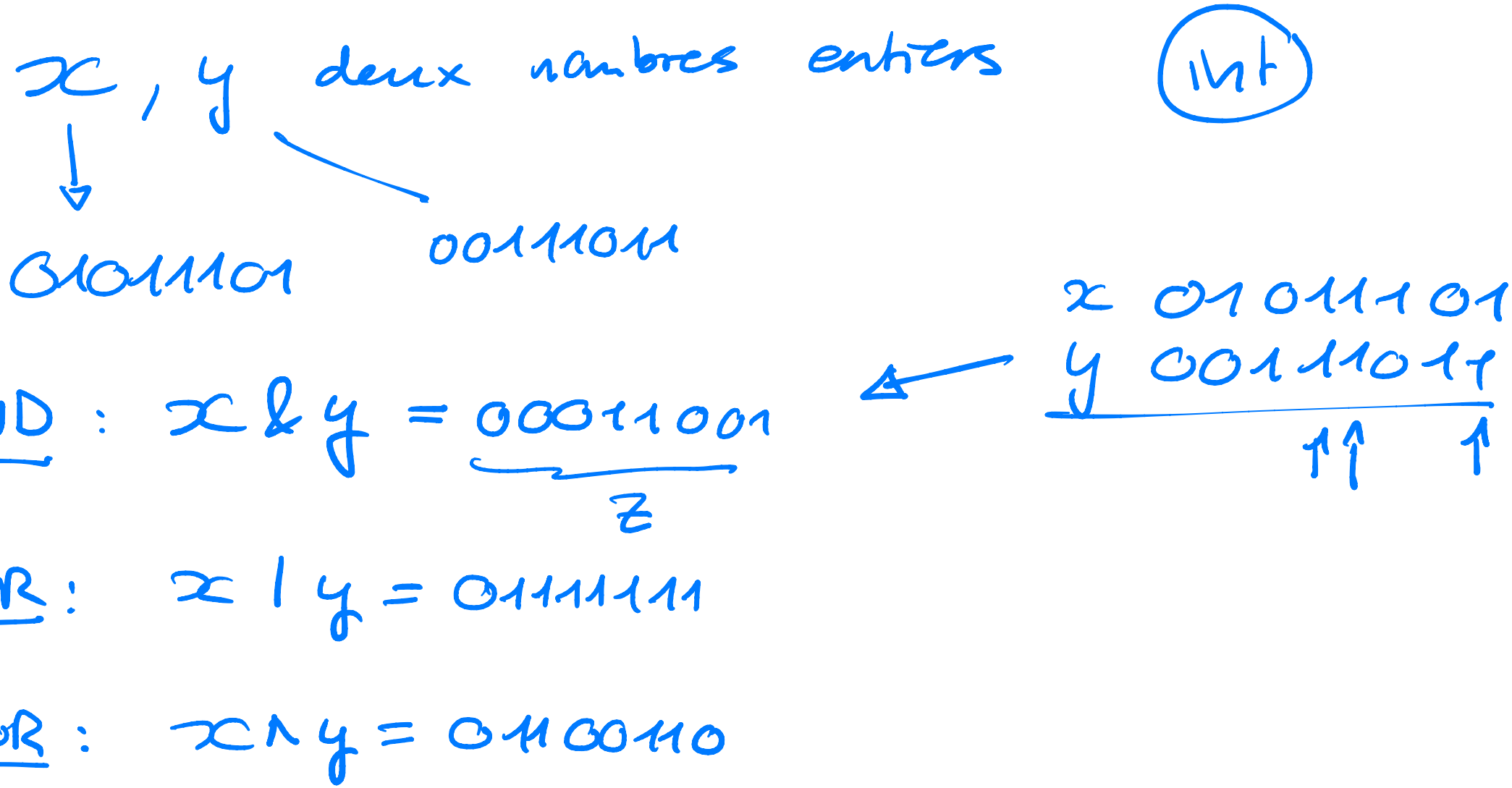
x (int) $= 37$ $\qquad 37 = 32 + 4 + 1$

$$\Rightarrow \begin{cases} \text{bin}(x) = \text{"0b100101"} \\ \text{hex}(x) = \text{"0x25"} \end{cases}$$

format str !

$32 = 2 \cdot 16 + 5 \cdot 1$

Opérations binaires sur des nombres entiers



Algorithme Xorshift

Préliminaire :

- $x << 3$
  = mult. par 8

$x = 1101_0$

↓

$x << 3 = 1101_0000$

- $x >> 3$
  = division euclidienne par 8
  (résultat = quotient)

$x >> 3 = 11$ ~~0_{10}~~

Xorshift : (sur 32 bits)

$x \longrightarrow x \wedge (x \ll 13) = y$

$y \longrightarrow y \wedge (y \gg 17) = z$

$z \longrightarrow z \wedge (z \ll 5) = x'$

⚠ à chaque étape, on ne doit considérer que les 32 premiers du nombre

x représenté sur plus que 32 bits

→ x & 0b11111111111111111111111111111111

ou x & 0xFFFFFFFF

alternativement :

x % (2**32)

Autres méthodes

- Mersenne twister
- Théorie du chaos
- Bruit atmosphérique
- Polarisation des photons

random.org