

PF1 — Principes de Fonctionnement des machines binaires

Jean.Baptiste.Yunes@univ-paris-diderot.fr

Version 1.3

numération et arithmétique

numération et arithmétique en machine

codage (texte, images)

compression, cryptographie, contrôle
d'erreur

calcul propositionnel

circuits numériques

(3 séances) numération + numération
en machine

(3 séances) numérisation + codage /
compression

(2 séances) calcul propositionnel

(2 séances) circuits combinatoires

Numération

représentation des nombres

cardinaux : un, deux, trois, quatre...

ordinaux : premier, deuxième, troisième...

le système des **objets témoins**
compter avec des cailloux
un caillou $\equiv$ un objet de la collection
le représentant est plus petit que la
collection...



le système des **objets témoins**

exemples :

compter les jours de prison

le chapelet (égrener des prières)

les « coups » de cloches de l'église



repose sur la construction d'une bijection
pas pratique pour les grands nombres



comme pour les « jours de prison », on regroupe par paquets, paquets de paquets, etc

la taille d'un paquet est habituellement uniforme mais pas nécessairement

cela définit la base

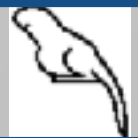


numération égyptienne

3000 ans av. J.C.

additive, on ajoute les éléments du mot
représentant le nombre

beaucoup...

1	10	100	1.000	10.000	100.000	1.000.000
						
bâton	os	corde	lotus	doigt	têtard	Heh



numération chinoise à bâtons

I^{er} siècle av. J.C.

18 symboles

pas de 0 (espace)

<u>Chiffres des unités ou chiffres des centaines :</u>									
I	II	III	IIII	IIIII	⊥	⊥⊥	⊥⊥⊥	⊥⊥⊥⊥	
1	2	3	4	5	6	7	8	9	
<u>Chiffres des dizaines ou chiffres des milliers :</u>									
—	=	≡	≡≡	≡≡≡	⊥	⊥⊥	⊥⊥⊥	⊥⊥⊥⊥	
1	2	3	4	5	6	7	8	9	
<u>Exemples:</u>									
1997:	—	⊥⊥⊥⊥	≡≡≡	⊥⊥					
804:						⊥⊥⊥		IIII	

4566 :

≡≡≡ II III ⊥ ⊥

numération romaine

antiquité romaine 800 av. J.C.

additive et partiellement soustractive : un chiffre de valeur plus petite que celui placé à sa droite doit être retranché à celui-ci

arithmétique difficile

1	5	10	50	100	500	1000
I	V	X	L	C	D	M



numération positionnelle

une grande invention...qui nécessite le 0

la position du chiffre dans le mot modifie sa contribution (paquets de paquets, paquets,...)

dans « 2 » le '2' représente deux unités, soit 2×1
ou encore 2×10^0

dans « 5000 » le '5' représente cinq milliers, soit 5×10^3 où 3 est la position à partir de la droite...

pour une base $b > 0$, il y a b **symboles**
utilisés comme **chiffres** $0, 1, 2, \dots, b-1$

attention 0 (par exemple) désigne soit un nombre
soit un chiffre (qui n'a pas de valeur en soi)

1	2	3	4	5	6	7	8	9	0	chiffres indiens (vers le X ^{ème} siècle)
1	2	3	4	5	6	7	8	9	0	chiffres arabes (vers le XIII ^{ème} siècle)
1	2	3	4	5	6	7	8	9	0	chiffres gothiques (XIV ^{ème} siècle)
1	2	3	4	5	6	7	8	9	0	chiffres modernes (après le XV ^{ème} siècle)
1	2	3	4	5	6	7	8	9	0	chiffres modernes dactylographiés

en notation positionnelle de gauche à droite pour la base $b > 0$

le **mot** $a_{n-1}a_{n-2}\dots a_1a_0$ représente le **nombre**

$$\sum_{i=0}^{n-1} a_i b^i$$

10^3 10^2 10^1 10^0

$$2849 = 2 \times 1000 + 8 \times 100 + 4 \times 10 + 9 \times 1$$

Binaire : base 2

Egypt XXIV av. JC, Chine IX av. JC, Inde II av. JC, Polynésie XIV ap. JC, ordinateurs

il y a 10 types de personnes, ceux qui connaissent la base 2 et les autres (folklore informatique)

Octal : base 8

Mexique (on compte avec les interstices entre les doigts), informatique

0777

Décimal : base 10

On en retrouve la trace un peu partout dans le monde (les humains ont habituellement dix doigts)

Systeme international d'unités - 1889

inspiré du système métrique (révolution française)

Duodécimal : base 12

Népal, monnaies, mois, quantités
commerciales

une douzaine d'œufs

douze deniers dans un sou

douze mois dans l'année

Hexadécimal : base 16

informatique

0xDEADBEEF

Sexagésimal : base 60

Babyloniens II av. JC, Sumériens III av. JC, angles, minutes

1°45'

Babylonian digits

 1	 11	 21	 31	 41	 51
 2	 12	 22	 32	 42	 52
 3	 13	 23	 33	 43	 53
 4	 14	 24	 34	 44	 54
 5	 15	 25	 35	 45	 55
 6	 16	 26	 36	 46	 56
 7	 17	 27	 37	 47	 57
 8	 18	 28	 38	 48	 58
 9	 19	 29	 39	 49	 59
 10	 20	 30	 40	 50	

Bases hybrides

1 mois et 15 jours ?

17:28:30

Base courantes en informatique : 2, 8, 16

Chiffres couramment utilisés en informatique:

0, 1 (base 2)

0, 1, 2, 3, 4, 5, 6, 7 (base 8)

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
(base 16)

On confond souvent chiffre et nombre

un **chiffre** est une lettre, un symbole

ne vaut rien en tant que tel

un **nombre** représente une quantité, un nombre
s'écrit avec des chiffres, est un mot sur
l'alphabet des chiffres

La lettre/chiffre 0

Le nombre 0

On décore parfois le mot de la base utilisée pour l'écrire (en général la base est elle-même représentée en base 10)

$$3876 = (3876)_{10}$$

$$(10111011)_2 = (187)_{10} = 187$$

le **mot** *un-zéro-un-un-un-zéro-un-un*
représente le même **nombre** que le mot
un-huit-sept

Numérations exotiques

Bibi-binaire

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
....															
• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •	• • • •
0	1	J	C	Γ	Π	ℵ	Λ	ℓ	℥	U	7	⌋	L	✓	И
HO	HA	HE	HI	BO	BA	BE	BI	KO	KA	KE	KI	DO	DA	DE	DI
0	1	J	C	Γ	Π	ℵ	Λ	ℓ	℥	U	7	⌋	L	✓	И
o	a	e	i	bo	ba	be	bi	co	ca	ce	ci	do	da	de	di

Shadok

GA, BU, ZO, MEU

Conversion entre bases

Comment $(1011101)_2$ s'écrit-il en base 10 ?

Comment $(12987)_{10}$ s'écrit-il en base 3 ?

Conversion d'un décimal en base b ?

méthode empirique : approximations
successives/encadrements successifs

$$2^7=128 \leq \mathbf{167} < 2^8=256 \quad 167 - 128 = \mathbf{39}$$

$$2^5=32 \leq \mathbf{39} < 2^6=64 \quad 39 - 32 = \mathbf{7}$$

$$2^2=4 \leq \mathbf{7} < 2^3=8 \quad 7 - 4 = \mathbf{3}$$

$$2^1=2 \leq \mathbf{3} < 2^2=4 \quad 3 - 2 = \mathbf{1}$$

$$2^0=1 \leq \mathbf{1} < 2^1=2 \quad 1 - 1 = \mathbf{0}$$

$$167 = (10100111)_2$$

Conversion d'un nombre décimal en
base b ?

par un **algorithme**

c'est-à-dire une méthode **systematique**,
documentée, **non ambiguë** et qui
conduit au résultat attendu

ex.: la méthode des divisions successives

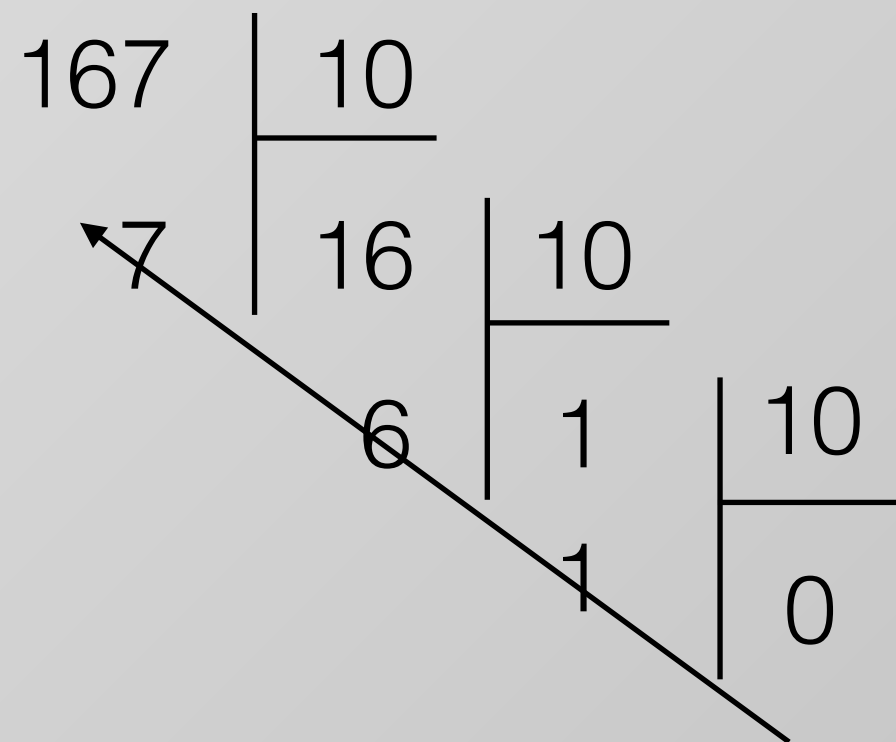
si on divise n par 10, le reste de la division est le chiffre des unités ($n=10q+r$ avec $0 \leq r < 10$)

si on divise le quotient de la division précédente par 10, le reste est le chiffre des dizaines

si on divise le quotient de la division précédente par 10, le reste est le chiffre des centaines

...

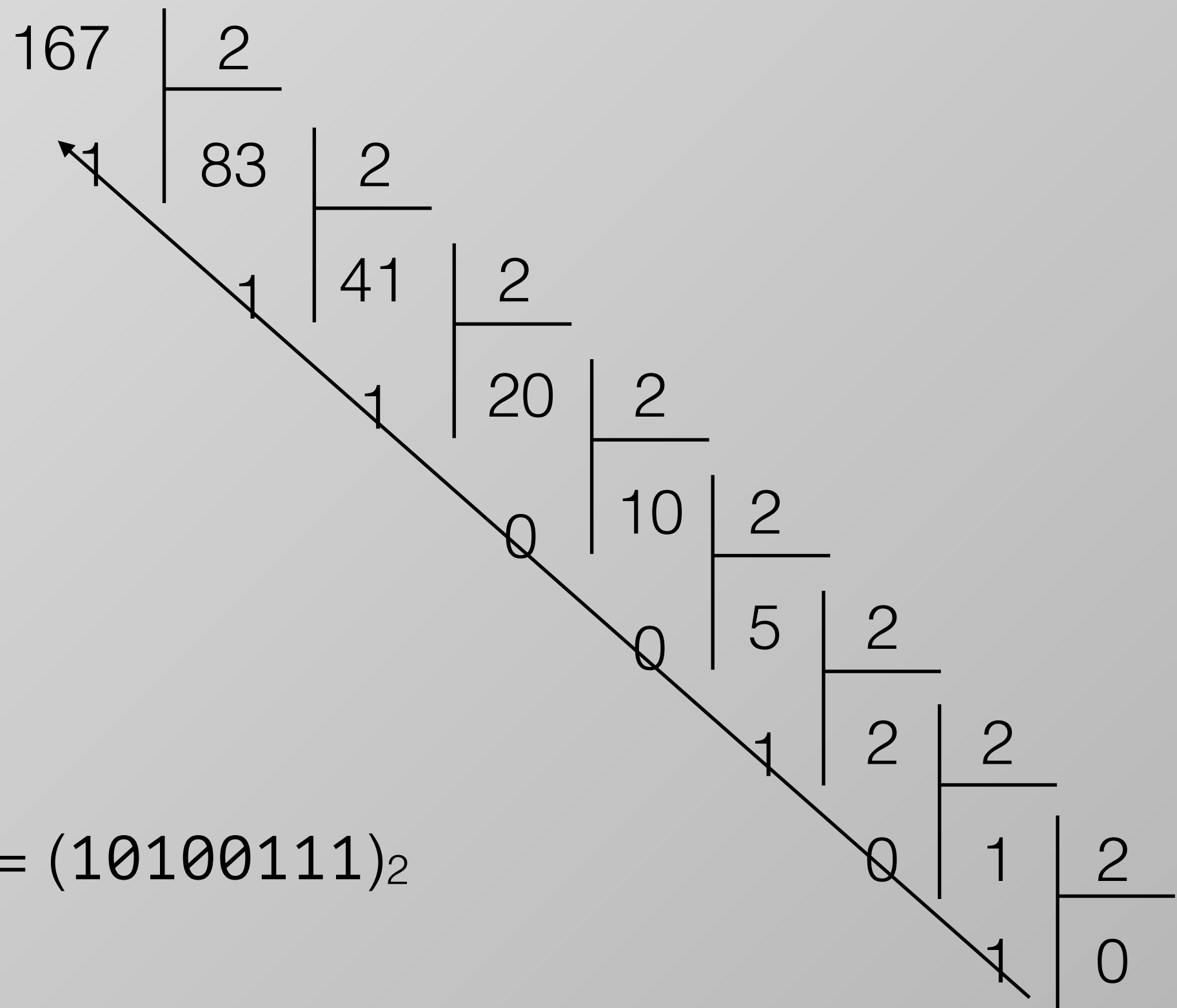
jusqu'à ce que le quotient soit nul



$$167 = (167)_{10}$$

Ce procédé fonctionne pour convertir
vers n'importe quelle base

il suffit de faire les divisions successives
appropriées



$$167 = (10100111)_2$$

Convertir un nombre en base b en
décimal ?

on peut encore utiliser des divisions

petit souci, il faut faire de l'arithmétique
en base b !

$(1363)_7$? Attention $10 = (13)_7$

nécessite la maîtrise des tables...

$$\begin{array}{r|l}
 (1363)_7 & (13)_7 \\
 (5)_7 & (104)_7 \mid (13)_7 \\
 (3)_7 & (5)_7
 \end{array}$$

$(535)_{10} = (1363)_7$

+	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	10
2	2	3	4	5	6	10	11
3	3	4	5	6	10	11	12
4	4	5	6	10	11	12	13
5	5	6	10	11	12	13	14
6	6	10	11	12	13	14	15

x	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	11	13	15
3	0	3	6	12	15	21	24
4	0	4	11	15	22	26	33
5	0	5	13	21	26	34	42
6	0	6	15	24	33	42	51

Convertir un nombre en base b en
décimal ?

il suffit de « recomposer »

$$(1363)_7 ?$$

$$1 \times 7^3 + 3 \times 7^2 + 6 \times 7^1 + 3 \times 7^0 =$$

$$1 \times 343 + 3 \times 49 + 6 \times 7 + 3 \times 1 =$$

$$(535)_{10}$$

On peut utiliser avantageusement une autre décomposition de $(1363)_7$ celle dite de Horner (William Georges Horner 1819-1845)

$$(1363)_7 = (((\mathbf{1} \times 7) + \mathbf{3}) \times 7 + \mathbf{6}) \times 7 + \mathbf{3}$$

7, 10, 70, 76, 532, 535

3 multiplications, 3 additions

méthode précédente : 7 multiplications, 3 additions



on a donc au moins deux algorithmes/
méthodes pour obtenir le même résultat

leur fonctionnement n'est pas le même

des algorithmes produisant les mêmes
résultats peuvent être comparés

attention: cela dépend du critère de
comparaison

Relations entre bases

le cas des bases « informatiques » 2
(binaire), 8 (octale), 16 (hexadécimale)

$$8 = 2^3$$

$$16 = 2^4$$

qu'en faire ?

Retour sur le cas de la base 10

Le nombre $(749426732)_{10}$

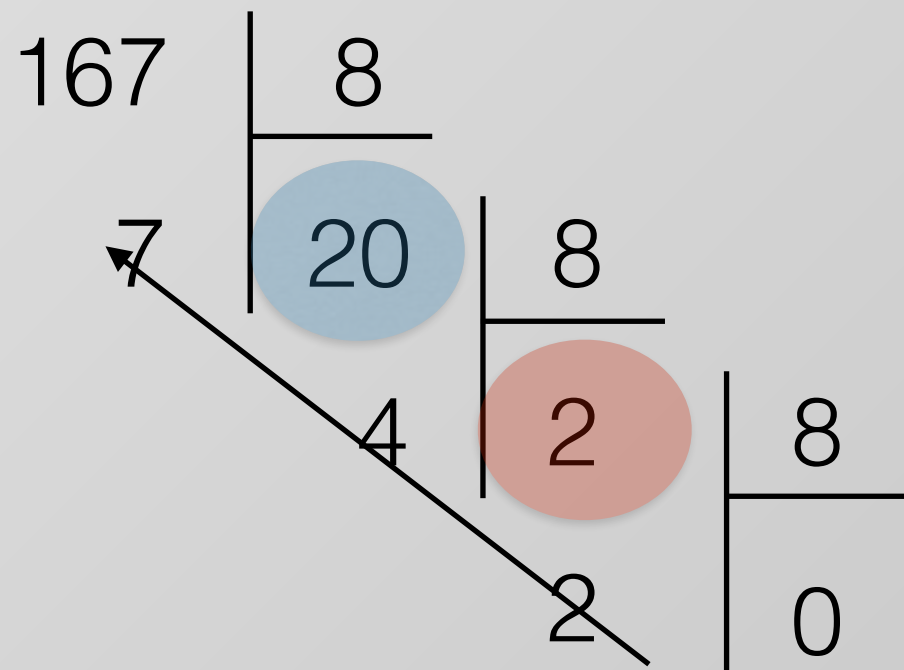
s'écrit $(\underline{07}\mathbf{49}\underline{42}\mathbf{67}\underline{32})_{100}$

en effet $100 = 10^2$

on groupe les chiffres « deux par deux »

en base 100 on a besoin de 100 chiffres, on les note (par exemple) de 00 à 99.

$$167 = (247)_8$$

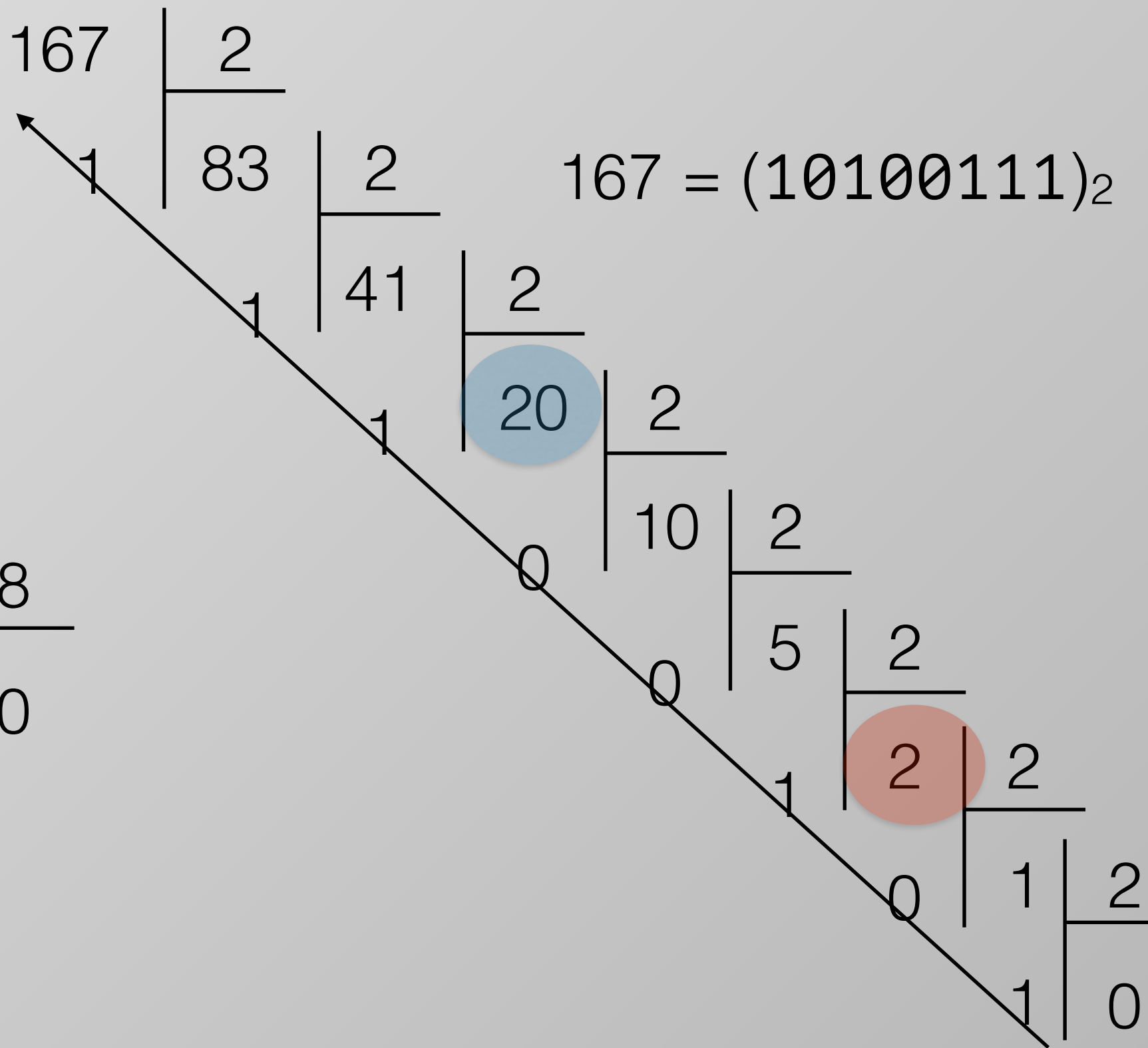


$$(2)_8 = (010)_2$$

$$(4)_8 = (100)_2$$

$$(7)_8 = (111)_2$$

$$167 = (247)_8 = (\underline{010} \underline{100} \underline{111})_2$$



$$167 = (10100111)_2$$

pour passer de la base 2 à la base 8 on
replie par paquets de trois chiffres
binaires.

pour passer de la base 8 à la base 2 on
déplie en sens inverse

Base 2, simple mais écriture longue des nombres

Base 16, plus « compliquée » à manipuler mais écriture plus compacte

Jouons avec les bases

base positive, toujours ?

chiffres positifs, toujours ?

base entière, toujours ?

...

Bases exotiques : base -2

0		0×-2^0	$(0)_{-2}$
1		1×-2^0	$(1)_{-2}$
2		$1 \times -2^2 + 1 \times -2^1 + 0 \times -2^0$	$(110)_{-2}$
3		$1 \times -2^2 + 1 \times -2^1 + 1 \times -2^0$	$(111)_{-2}$
4		$1 \times -2^2 + 0 \times -2^1 + 0 \times -2^0$	$(100)_{-2}$
5		$1 \times -2^2 + 0 \times -2^1 + 1 \times -2^0$	$(101)_{-2}$
6	$1 \times -2^4 + 1 \times -2^3 + 0 \times -2^2 + 1 \times -2^1 + 0 \times -2^0$		$(11010)_{-2}$
7	$1 \times -2^4 + 1 \times -2^3 + 0 \times -2^2 + 1 \times -2^1 + 1 \times -2^0$		$(11011)_{-2}$
8	$1 \times -2^4 + 1 \times -2^3 + 0 \times -2^2 + 0 \times -2^1 + 0 \times -2^0$		$(11000)_{-2}$

Bases exotiques : base négatrinaire

Base « ordinaire » mais avec des chiffres négatifs

La base négatrinaire ou **ternaire balancée**

Base 3, chiffres -1, 0 et 1 que l'on note 1, 0, 1

C'est une base d'Avižienis

certaines autorisent l'addition
sans propagation de retenue !?



En ternaire balancée

Entier

-7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7

TB

111 110 111 11 10 11 1 0 1 11 10 11 111 110 111



Bases exotiques : Base de « Fibonacci »

prenons les termes de la suite de
Fibonacci

$$F_0 = 1, F_1 = 2, F_{n+2} = F_{n+1} + F_n \text{ pour } n \geq 0.$$

donc 1, 2, 3, 5, 8, 13, 21, etc

prenons les mots de 6 bits

0	1	2	3	4	5	6	7	8	9	10	11	12	13
000000	000001	000010	000100	000101	001000	001001	001010	010000	010001	010010	010100	010101	100000

Base de « Fibonacci »

problème : pas de représentation unique

$$10 = 8+2 \text{ ou } 5+3+2$$

$$16 = 13+3 \text{ ou } 8+5+3 \text{ ou } 13+2+1 \text{ ou } 8+5+2+1$$

Base de **Zeckendorf**

représentation standard sans **11** dans
l'écriture



Bases exotiques : Base factorielle (factoradique)

Tout nombre n peut s'écrire

$$n = \dots c_5 \cdot 5! + c_4 \cdot 4! + c_3 \cdot 3! + c_2 \cdot 2! + c_1 \cdot 1!$$

où c_i est un nombre compris entre 0 et i

ex : 521 s'écrit 41221 c'est-à-dire

$$4 \times 120 + 1 \times 24 + 2 \times 6 + 2 \times 2 + 1 \times 1$$

Base factorielle

inconvenient : nombre arbitraire de *chiffres*...

avantage : les rationnels ont une écriture finie!

Le nombre e qui s'écrit $1/1! + 1/2! + 1/3! + 1/4! + \dots$
s'écrit $0,11111\dots$ en base factorielle :-)

e a une écriture périodique! (mais c'est un transcendant)

Bases exotiques : Base « fractions continues »

On sait que tout nombre réel peut s'écrire

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$$

avec a_0 dans **Z** et $a_{i>0}$ dans **N⁺**



John Wallis - source Wikipédia

Bases exotiques : Base « phinaire », base d'or

$$\varphi = (1 + \sqrt{5}) / 2$$

Tout nombre réel non négatif se représente uniquement avec les chiffres 0 et 1, sans jamais utiliser 11 (comme dans Zeckendorf)

Base « phinaire », base d'or

1 s'écrit 1, $1 = \varphi^0$

2 s'écrit 10,01, $2 = \varphi^1 + \varphi^{-2}$

3 s'écrit 100,01, $3 = \varphi^2 + \varphi^{-2}$

4 s'écrit 101,01, $4 = \varphi^2 + \varphi^0 + \varphi^{-2}$

5 s'écrit 1000,1001, $5 = \varphi^3 + \varphi^{-1} + \varphi^{-4}$

difficile d'expliquer ici comment fonctionne la conversion...

on peut vérifier sachant que $\varphi^2 = 1 + \varphi$ et $1/\varphi = \varphi - 1$