



Logică computațională

Curs 2

Lector dr. Mihiș Andreea-Diana

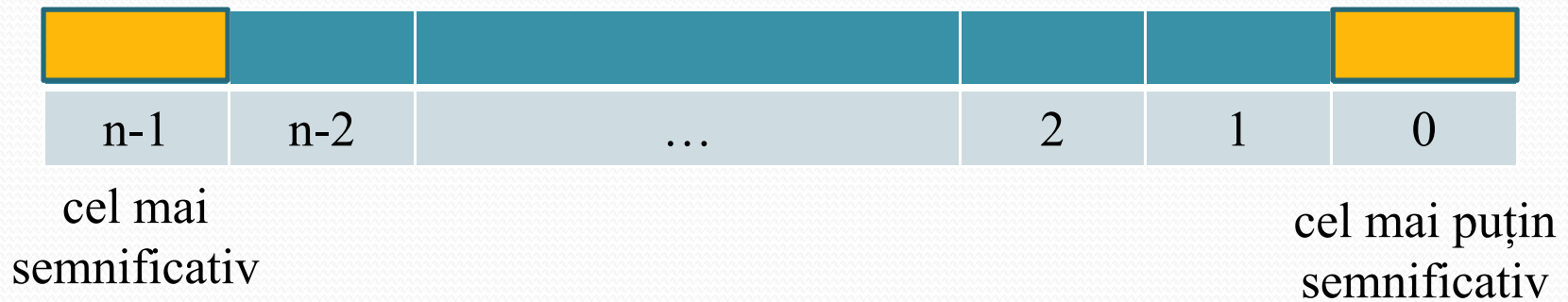
...

...0101010000110101010001110010010010001010011100001010101000100010010010
00100101000100010010010001000001110001010010²01010100011100100111010010100
1000100100101001010001100011010101010001...



Reprezentarea binară a nr.

- într-o locație de memorie – k octeți = n biți (8, 16, 32, 64)



Reprezentarea nr. întregi fără semn

$$x_{(10)} \rightarrow y_{(2)}$$



Intervale de reprezentare

0	0	...	0	0	0	= 0
n-1	n-2	...	2	1	0	

1	1	...	1	1	1	$= 2^n - 1$
n-1	n-2	...	2	1	0	

$n = 8$ $[0 , 255]$

$n = 16$ $[0 , 65535]$

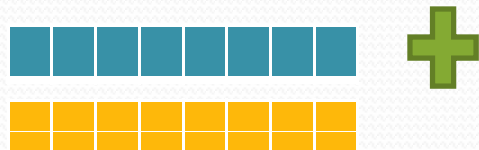
$n = 32$ $[0 , 4\,294\,967\,295]$


$n = 64$ $[0 , 18\,446\,824\,753\,389\,551\,615]$



Aritmetica nr. întregi fără semn

- adunarea



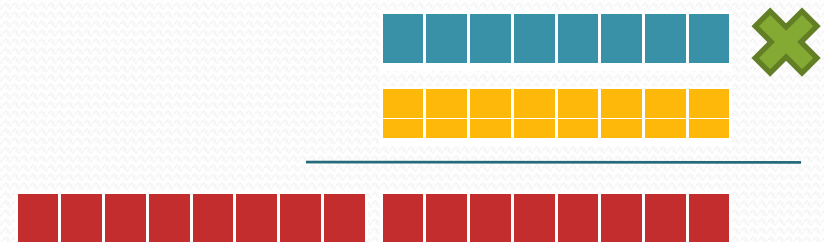
 nu se păstrează
în rezultat

- scăderea

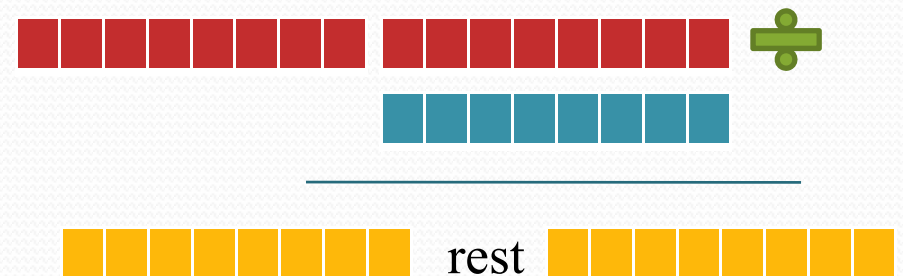


 nu se păstrează
în rezultat

- înmulțirea



- împărțirea



Algoritmul de înmulțire a întregilor fără semn

DATE de înmulțitul **M** și înmulțitorul **Q**

CA ← 0

PENTRU $i \leftarrow 1, n$ EXECUTĂ

DACĂ $Q_0 = 1$ ATUNCI

CA ← **A** + **M**

SF. DACĂ

CAQ se deplasează spre dreapta cu 1 poziție

SF. PENTRU

REZULTATE **AQ**

M	C	A	Q

Algoritmul de împărțire a întregilor fără semn

DATE deîmpărțitul **AQ** și împărțitorul **M**

PENTRU $i \leftarrow 1, n$ EXECUTĂ

CAQ se deplasează spre stânga cu 1 poziție

DACĂ **CA** \geq **M** ATUNCI

$Q_0 \leftarrow 1$

CA \leftarrow **CA** - **M**

ALTFEL

$Q_0 \leftarrow 0$

SF. DACĂ

SF. PENTRU

REZULTATE câtul **Q** și restul **A**

M	C	A	Q

Coduri de reprezentare a întregilor cu semn

- scopul: - simplificarea operațiilor (-)
- convenție întreagă (supraunitară)

$n \in \{8, 16, 32, 64\}$



bitul de semn

0 + 1 -

Codul direct

$$x \in \mathbb{Z}, |x| < 2^{n-1}$$

$$[x]_{\text{dir}} = \begin{cases} x, & \text{dacă } x \geq 0 \\ 2^{n-1} + |x|, & \text{dacă } x \leq 0 \end{cases}$$

dezavantaj: $[+0]_{\text{dir}}: |0|0\dots 0|$ și $[-0]_{\text{dir}}: |1|0\dots 0|$



Codul invers

$$x \in \mathbb{Z}, |x| < 2^{n-1}$$

$$[x]_{\text{inv}} = \begin{cases} x, & \text{dacă } x \geq 0 \\ 2^{n-1} - |x|, & \text{dacă } x \leq 0 \end{cases}$$

dezavantaj: $[+0]_{\text{inv}}: |0|0...0|$ și $[-0]_{\text{inv}}: |1|1...1|$



Codul complementar

$$x \in \mathbb{Z}, |x| < 2^{n-1}$$

$$[x]_{\text{compl}} = \begin{cases} x, & \text{dacă } x \geq 0 \\ 2^n - |x|, & \text{dacă } x < 0 \end{cases}$$

Obs.: dacă $x \leq 0$, $[x]_{\text{compl}} = [x]_{\text{inv}} + 1$

dacă $x \geq 0$, $[x]_{\text{compl}} = [x]_{\text{inv}} = [x]_{\text{dir}}$

avantaj: $[+0]_{\text{compl}}: |0|0...0|$

nu e nr.: $|1|0...0|$



Intervale de reprezentare

$n=8$ $[-127, 127]$

$n=16$ $[-32767, 32767]$

$n=32$ $[-2\,147\,483\,647, 2\,147\,483\,647]$

$n=64$ $[-9\,223\,412\,376\,694\,775\,807, +9\,223\,412\,376\,694\,775\,807]$



Operații în cod complementar: \oplus

$$\forall a, b \in [0, 2^n), a \oplus b = \begin{cases} a + b & , \text{dacă } a+b < 2^n \\ a + b - 2^n & , \text{dacă } a+b \geq 2^n \end{cases}$$

Reguli: dacă a și b au același **semn \neq semnul** $a \oplus b$ – **depășire**
 t_{n-1} se pierde (nu se păstrează în rezultat)

$$[x+y]_{\text{compl}} = [x]_{\text{compl}} \oplus [y]_{\text{compl}} \quad \forall x, y \in \mathbb{Z}, \text{ a. î. } |x|, |y|, |x+y| < 2^{n-1}$$
$$[x-y]_{\text{compl}} = [x]_{\text{compl}} \oplus [-y]_{\text{compl}}$$

Convenția subunitară

$n \in \{8, 16, 32, 64\}$



bitul de semn

$0+$ $1-$

Coduri

$x \in \mathbb{R}, |x| < 1$ cu max. $n-1$ cifre după “,”

$$[x]_{\text{dir}} = \begin{cases} x, & \text{dacă } x \geq 0 \\ 1 + |x|, & \text{dacă } x \leq 0 \end{cases}$$

$$[x]_{\text{inv}} = \begin{cases} x, & \text{dacă } x \geq 0 \\ 2 - 2^{-n+1} - |x|, & \text{dacă } x \leq 0 \end{cases}$$

$$[x]_{\text{compl}} = \begin{cases} x, & \text{dacă } x \geq 0 \\ 2 - |x|, & \text{dacă } x \leq 0 \end{cases}$$

Obs.: dacă $x \leq 0$, $[x]_{\text{compl}} = [x]_{\text{inv}} + 2^{-n+1}$

dacă $x \geq 0$, $[x]_{\text{compl}} = [x]_{\text{inv}} = [x]_{\text{dir}}$

Operații

$$\forall a, b \in [0, 1), a \oplus b = \begin{cases} a + b & , \text{dacă } a+b < 2 \\ a + b - 2 & , \text{dacă } a+b \geq 2 \end{cases}$$

Reguli: dacă a și b au același **semn** \neq **semnul** $a \oplus b$ – **depășire**
 t_{n-1} se pierde (nu se păstrează în rezultat)

$$\begin{aligned} [x+y]_{\text{compl}} &= [x]_{\text{compl}} \oplus [y]_{\text{compl}} \\ \forall x, y \in \mathbb{R}, \text{ a. î. } |x|, |y|, |x+y| < 1 \text{ cu max. } n-1 \text{ cifre după „,”} \\ [x-y]_{\text{compl}} &= [x]_{\text{compl}} \oplus [-y]_{\text{compl}} \end{aligned}$$

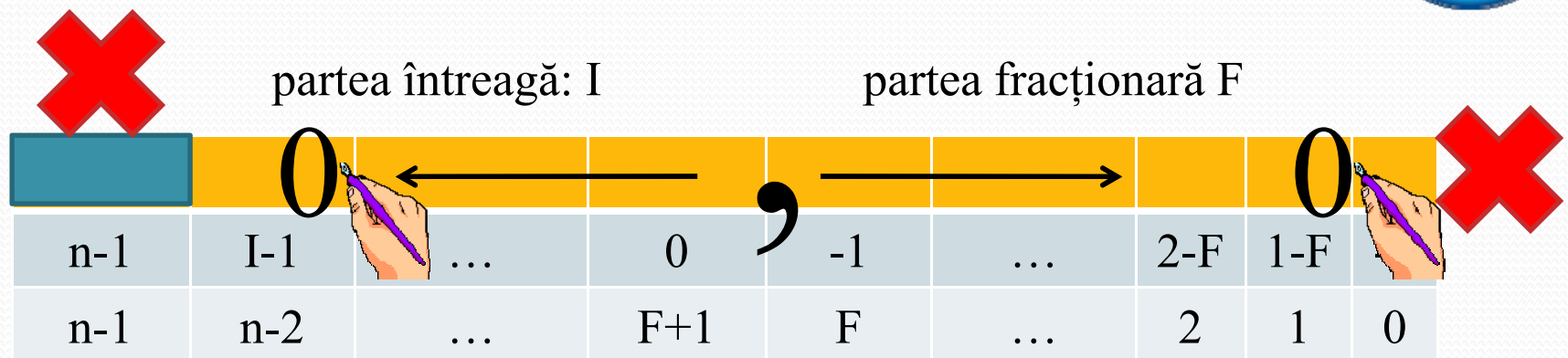


Reprezentări ale nr. reale

- se aproximează la nr. raționale
- pe k octeți (biți: 8, 16, 32 – cuvânt, 64 – dublu cuvânt)

Reprezentarea în virgulă fixă

- n biți
- $-2^I + 2^{-F} \leq |x| \leq 2^I - 2^{-F}$
- Dezavantaj: pierderea cifrelor cele mai semnificative



bitul de semn

$0 +$ $1 -$

Reprezentarea în virgulă mobilă (flotantă)

- precizie mai mare (pt. nr. f. mari / f. mici)
- la depășire se pierde cifrele cel mai puțin semnificative
- $\forall x \in \mathbb{R}, x = \pm 0, m * b^e$
 - m - mantisa numărului
 - b - bază de numerație
 - e - exponent
- ! b=2



Mantisă subunitară

- Def 1: Un număr real x se scrie cu ***mantisă subunitară*** și exponent al unei baze b , dacă $x = \pm 0, m * b^e$
- Def 2: Un număr real x , $x \neq 0$, se scrie cu ***mantisa subunitară normalizată***, dacă x este scris cu mantisă subunitară și exponent al bazei b și dacă are loc: $\frac{1}{b} \leq m < 1$.

Ex : $0,12345678 * 10^4$ - este scris normalizat

$0,004371 * 10^{-4}$ - nu este scris normalizat

Mantisa "supraunitară"

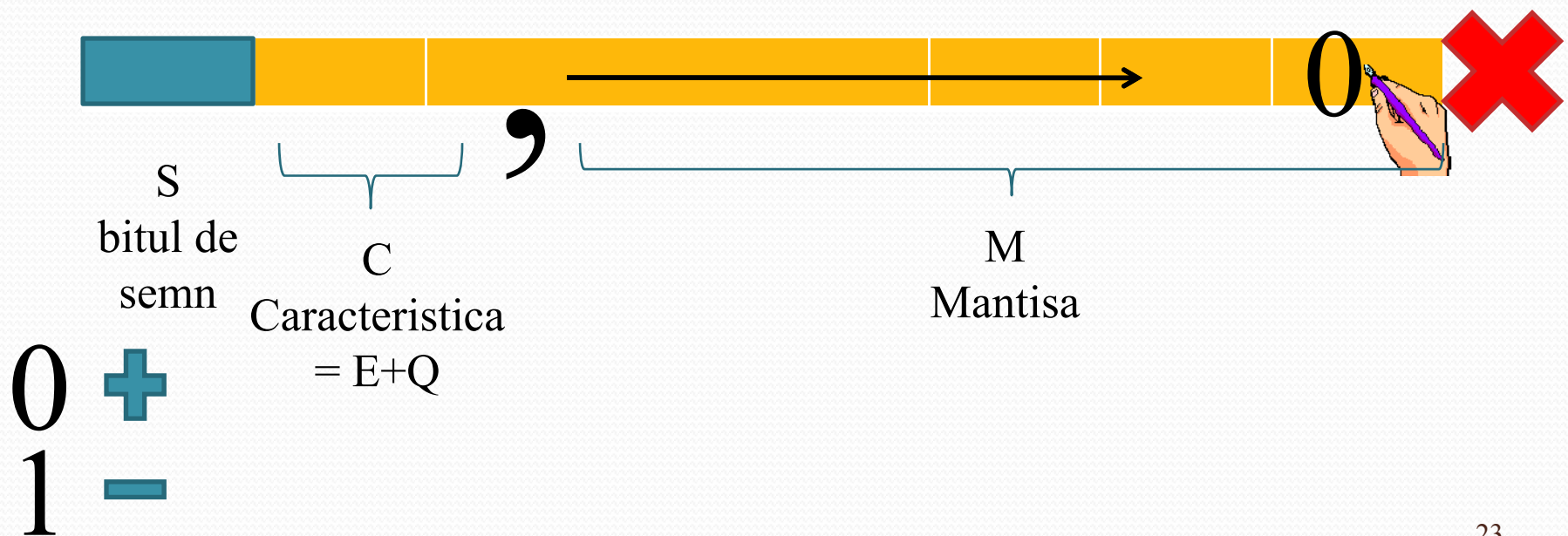
- Def 3: Un număr real x , $x \neq 0$, este scris cu ***mantisa între 1 și 2***, dacă x se scrie în baza 2 sub forma : $x = \pm 1, m * 2^e$
- Def 4: Un număr real x este reprezentat în calculator în ***virgulă mobilă*** dacă pentru reprezentarea internă se utilizează scrierea lui x în baza 2 cu exponent și cu ***mantisă subunitară*** sau cu ***mantisă între 1 și 2***.

Reprezentarea în virgulă mobilă

$n \in \{32, 64\}$ IEEE P754 Simplă precizie / Dublă precizie

C pe 8/11 biți; M pe 23/52

Q – deplasament $\in \{127, 1023\}$



Valori speciale

Valoare	S (semn)	C (caracteristica)	M (mantisa)
0₊	0	0...0	0...0
0₋	1	0...0	0...0
-inf	1	1...1	0...0
+inf	0	1...1	0...0
NaN (not a number)	1 sau 0	1...1	valoare nenulă

Intervale de reprezentare

Precizie	Binar Valoare absolută	Zecimal Valoare absolută
Simplă	minim = 2^{-126}	minim $\approx 10^{-38}$
	maxim = $(2-2^{-23}) \cdot 2^{127}$	maxim $\approx 10^{38}$
Dublă	minim = 2^{-1022}	minim $\approx 10^{-308}$
	maxim = $(2-2^{-52}) \cdot 2^{1023}$	maxim $\approx 10^{308}$