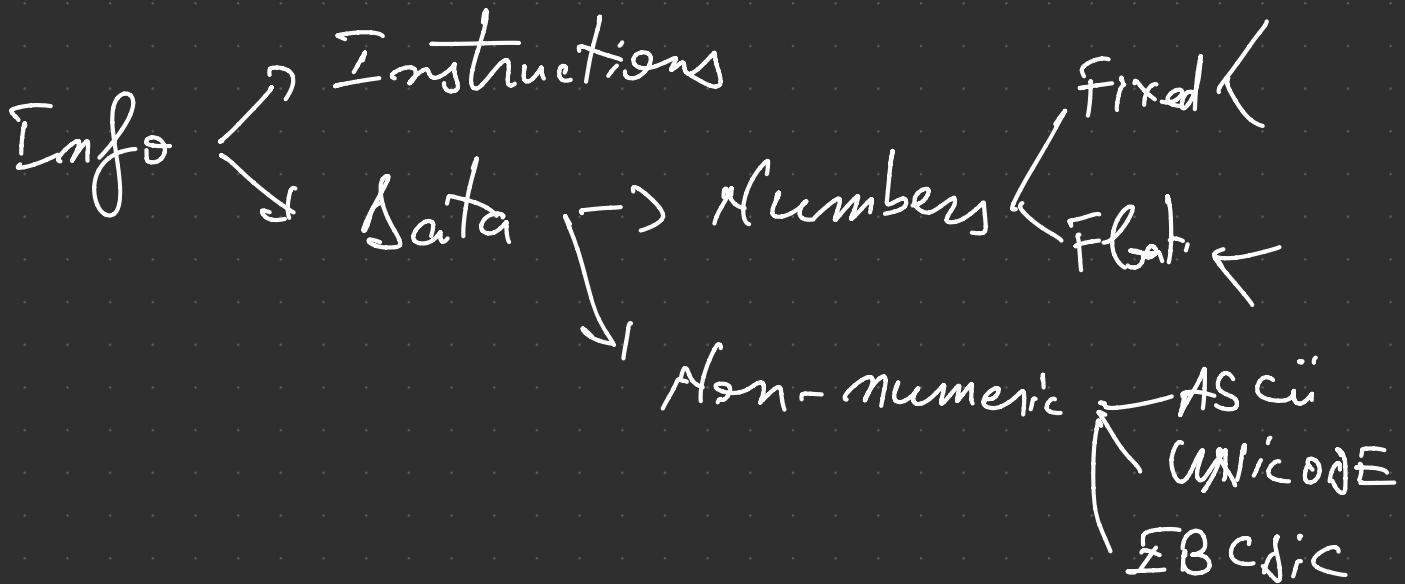


# Recapitulare

## 1. Clasificarea Informației

bit → binary digit



bit → byte (8 biti)  
↓ cuvinte (mult. de 8)

UNICODE → local  
↓ → international

UTF-8 / 16 / 32

Nr. virgulă fixă

- întregi
- fractionare

$$0 \rightarrow 2^{-1} \quad (\text{uint } 32)$$

$$2^{31} \rightarrow 2^{31-1} \quad (\text{int } 32)$$


---

Nr. virgulă mobilă

Numitor  $\rightarrow$

~~2~~  $\times$

$$3.14 + 1e20 - 1e20 = 0$$

$$3.14 + (1e20 - 1e20) = 3.14 \quad \text{000}$$

$\rightarrow$  ~~Asociativitate~~

$$3.14 + 1 \cdot 10^{20} - 10^{20} \left( 3.14 \cdot 10^{-20} + 1 \right)$$


---

Reprezentarea nr. cu virg. fixă

$X = x_{n-1} x_{n-2} \dots x_0 + x_{-1} x_{-2} \dots x_{-m}$

$\underbrace{\phantom{x_{n-1} x_{n-2} \dots x_0}}_{\text{Intreg}}$        $\underbrace{\phantom{x_{-1} x_{-2} \dots x_{-m}}}_{\text{Fractionar}}$

MSB  $\rightarrow x_{n-1}$

LSB  $\rightarrow x_{-m}$

$$X = \sum_{j=-m}^{n-1} x_j \cdot 2^j$$

N<sup>j</sup> → pondera  
 cifre i j

---

$$103 \stackrel{=}{(n-1)} \begin{matrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \end{matrix}$$

↓  
 ponderare

---

P. Întreagă  $X = \sum_{i=0}^{n-1} x_i \cdot 2^i$

P. Fractionară  $X = \sum_{i=0}^{n-1} x_i \cdot 2^{i-n}$

---

Valoarea unui nr. fract. de mg.  
 fixă se obține împărțind val. întreagă  
 a aceleasi configurații la 2 la puterea  
 nr. de poz. binare a părții fractionare.

$$\frac{103}{128} = .1100111 \quad (2)$$

for ~ bit de senn O

neg.  $\rightarrow$  1

$$x = x_{n-1} \ x_{n-2} \ \dots \ x_1 \ x_0$$

↓  
 semm  
 magnitudine

$$x - y = x + (-y) \quad (\text{the func.})$$

in sum

$$\text{MAX}_{FRA} = \underbrace{2 \cdot 1 \dots 1}_{= 1 - 2^{-m+1}} - \frac{2^{m-1} - 1}{2^{m-1}}$$

$$2^{m-1} - 1 = 10^P \quad \begin{cases} P = \log_{10}(2^{m-1} - 1) \\ P \leq \log_{10} 2^m \approx 0,3(n-1) \end{cases}$$

- Complexity
  - moderate
  - disadvantage
- Hardware
  - favorable immutability
  - acumulate DDAR  
PDT + PDT

# Complement de 1

$$\text{pot}_{(\text{sm})} = \text{pot}(c_1)$$

- nu este repr. pozitională
- este nevoie de un End Around Carry

Precizie  $\rightarrow$  SM

- dezavantajă  $\rightarrow$  Adunare în EAC
- Nr Neg.  $\rightarrow$  negarea celor positive  
 $\sim$  Nr

# Complement de 2

MSB codifică semnul

$$N_2 C_2 = N_2 C_1 + 1$$

→ ignoră EAC la adunare

→ conversia SM  $\leftrightarrow$  C2

- bit semn neschimbăt
- negare la toti bitii
- adunare 1 (ignoră EAC dacă ajunge la MSB)

$$-4 = 10100_{SM}$$

$$= 11100_{C_2}$$

→ adunare / scădere mai simplă ca SM/C1

→ înmulțire mai grea

Avantajele aritmeticii în C<sub>2</sub>

- > adunare / scădere
- > indiferent de semn  $x - y = x + (-y)$
- > bitul de semn este folosit la fel ca cel de magnitudine
- > se ignorează ~~Este~~ rezultat

Compoziție → cele positive sunt identice

între SM și C<sub>2</sub>

$$\begin{array}{rcl} -1 & = & -11 \\ -2 & = & -10 \\ \dots & & \dots \end{array} \quad (\text{pe } 5 \text{ biți})$$

Anomalie  $-16 = 1.0000$

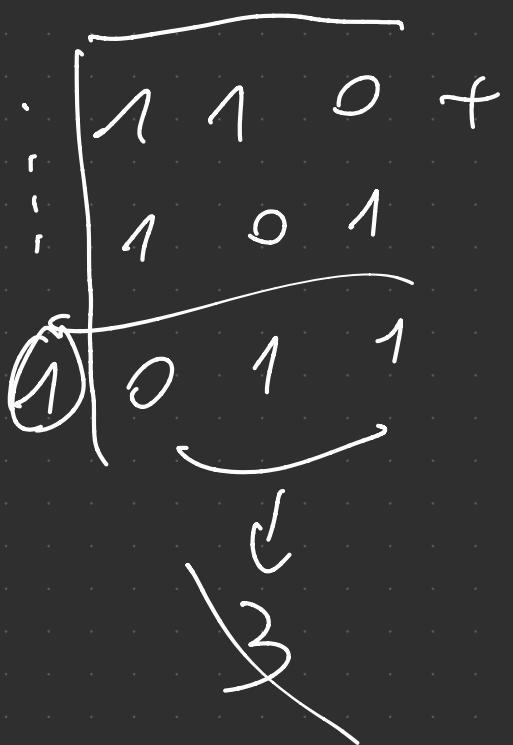
$$\begin{array}{rcl} -8 & \text{și} & 8 \\ & & \text{unsigned} \end{array} \quad \begin{array}{c} -2 \\ n-1 \end{array}$$

C → aceeași config. binară

# Overflow

→ la op. lora semn

→ transport de MSB



## Interpretarea Robertson

→ facilitățile înmulțirea op. în  $C_2$   
~ val. unic nr. neg. în  $C_2$  este egal  
cu val. nr. poz. obț.

# Reprezentarea decimală

BCD 8421

Exces de 3

Dariclu-cinci

tetradă = nibble = 4 biți consec.

$$297_{10} = 0010_2 + 1001_2 - 0111_2 \quad \text{BCD}$$

pondere  $10^i \times 2^j$   $i \rightarrow \text{pot. cif}$   
 $j \in \{0, 3\}$

|     |   |     |
|-----|---|-----|
| 0.2 | 2 | 0   |
| 0.4 | 2 | 0   |
| 0.8 | 2 | 0   |
| 1.6 | 2 | 1   |
| 1.2 | 2 | ... |

$j \rightarrow \text{pot. bit în tetradă}$

$$j \in \{0, 3\}$$

$$\begin{array}{r|l} 2 & 9 \\ 00\ 10 & 1001 \end{array}$$

$$\begin{array}{r|l} 10 \cdot 2^0 & 2 \\ 10 \cdot 2^1 & 3 \\ 10 \cdot 2^2 & 2 \\ 10 \cdot 2^3 & 2 \end{array}$$

$\rightarrow$  reprezentare  
pozițională

