

# Codifica dell'informazione numerica



Nicola Basilico, [nicola.basilico@unimi.it](mailto:nicola.basilico@unimi.it)  
Gabriella Trucco, [gabriella.trucco@unimi.it](mailto:gabriella.trucco@unimi.it)  
Matteo Re, [matteo.re@unimi.it](mailto:matteo.re@unimi.it)

*Architetture degli Elaboratori I, Laboratorio - Corso di Laurea in Informatica, A.A. 2019-2020*

## Informazioni (edizione 2019-2020)

### Laboratorio di Architettura degli Elaboratori I

- Turno A (A-C): **Nicola Basilico**, ricevimento su appuntamento: `nicola.basilico@unimi.it`
- Turno B (D-N): **Gabriella Trucco**, ricevimento su appuntamento: `gabriella.trucco@unimi.it`
- Turno C (O-Z): **Matteo Re**, ricevimento su appuntamento: `matteo.re@unimi.it`
- Sito del corso: <http://teaching.basilico.di.unimi.it/doku.php/pub/arch1-lab-2019-2020>

# Rappresentazione dei numeri: notazione posizionale

- Base  $B$ : numero di simboli usati per rappresentare i numeri nel sistema posizionale.
  - $B = 10$ , simboli  $\{0, 1, \dots, 9\}$
  - $B = 16$ , simboli  $\{0, 1, \dots, 9, A, B, C, D, E, F\}$
- Notazione posizionale:
  - ogni simbolo ha una posizione, descritta con un intero  $i$
  - ad ogni posizione  $i$  si associa un peso  $p_i$
  - al simbolo in posizione  $i$  viene associato il valore dato da:

$$\left[ \text{VALORE DEL SIMBOLO IN POSIZIONE } i \right] \times p_i$$

- di solito  $p_i = B^i$
- più è alto il peso associato a un simbolo, più **significativo** è quel simbolo

## Rappresentazione dei numeri: notazione posizionale

Esempio:  $(147)_{10}$

cifre	1	4	7
posizioni $i$	2	1	0
pesi $p_i$	$10^2$	$10^1$	$10^0$

$$(147)_{10} = 1 \cdot 10^2 + 4 \cdot 10^1 + 7 \cdot 10^0$$

In generale, dato un  $N$  intero non negativo scritto con  $n$  cifre come  $(c_{n-1}c_{n-2} \dots c_0)_B$ , il valore rappresentato è:

$$\sum_{i=0}^{n-1} c_i \cdot B^i = c_0 \cdot B^0 + c_1 \cdot B^1 + \dots + c_{n-2} \cdot B^{n-2} + c_{n-1} \cdot B^{n-1}$$

## Da base $B$ a base decimale

- PROBLEMA: Scrivere in base  $B = 10$  un numero dato in base  $B \neq 10$  ( $B = 2$  o  $B = 16$ )
- SOLUZIONE: applicare il metodo **polinomiale**

Esempio 1:

$$(1010)_2 = (10)_{10}$$

Esempio 2:

$$(3AC)_{16} = (940)_{10}$$

## Da base decimale a base $B$

- PROBLEMA: Scrivere in base  $B \neq 10$  ( $B = 2$  o  $B = 16$ ) un numero dato in base  $B = 10$
- SOLUZIONE: applicare il metodo **iterativo** (divisioni)

### Procedimento

Abbiamo un numero  $(N)_{10}$  da convertire nella base  $B$ :

1. dividere  $N$  per  $B$  (con una divisione intera);
2. il resto della divisione diventa la prima cifra meno significativa che resta da calcolare del numero in base  $B$ ;
3. se il quoziente è 0 abbiamo finito;
4. se il quoziente è diverso da zero si torna al passo 1 considerando il quoziente come dividendo;

## Da base decimale a base $B$

Esempio 1:  $(13)_{10} = (?)_2$

$13 : 2 = 6$	resto = 1	LSD	$(13)_{10} = (1101)_2$
$6 : 2 = 3$	resto = 0		
$3 : 2 = 1$	resto = 1		
$1 : 2 = 0$	resto = 1	MSD	

Esempio 2:  $(4021)_{10} = (?)_{16}$

## Da base $B = 2$ a base $B = 16$ e *vice versa*

- PROBLEMA: Scrivere in base  $B = 2$  ( $B = 16$ ) un numero dato in base  $B = 16$  ( $B = 2$ )
- SOLUZIONE: **raggruppamento** dei simboli e **lookup table**

Esempio 1:  $(111001)_2 = (?)_{16}$

$16 = 2^4$ , raggruppo i bit partendo da destra a gruppi di 4:

0011 | 1001

Con 4 bit ho  $2^4$  valori, uno per ogni simbolo della base  $B = 16$ .

0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Ispezionando la tabella:

$$(0011 | 1001)_2 = (39)_{16}$$



Da base  $B = 2$  a base  $B = 16$  e *vice versa*

Esempio 2:  $(EA0)_{16} = (?)_2$

$E \mid A \mid 0$

0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

$(E \mid A \mid 0)_{16} = (1110 \ 1010 \ 0000)_2$

## Somma di interi non negativi

Come la somma in decimale, ricordare che quando sommiamo singole cifre binarie:

$0 + 0 = 0$	riporto = 0
$0 + 1 = 1$	riporto = 0
$1 + 0 = 1$	riporto = 0
$1 + 1 = 0$	riporto = 1
$1 + 1 + 1 = 1$	riporto = 1

Esempio 1:  $(1101)_2 + (111)_2$

Esempio 2:  $(111010)_2 + (1101110010)_2$

## Interi: complemento a 2

- Rappresentazione in **complemento a 2** (C2)
  - **Dati  $n$  bit**, un numero positivo  $N$  è rappresentato in modo standard (come abbiamo visto per i non negativi)
  - $-N$ , invece si rappresenta come  $2^n - N$
- Metodo operativo per rappresentare  $-N$ :
  - Rappresentare il modulo  $N$  in modo standard
  - Complementare a 1 tutti i bit ( $1 \leftarrow 0, 0 \leftarrow 1$ )
  - Sommare 1

## Interi: complemento a 2

- In complemento a 2, con  $n$  bit, possiamo rappresentare gli interi nell'intervallo:  $[-2^{n-1}; 2^{n-1} - 1]$
- Esempio: con 3 bit rappresentiamo i numeri in ...  $[-4; 3]$

$N_{(10)}$	$N_{(C2)}$	$N_{(10)}$	$N_{(C2)}$
-4	100	0	000
-3	101	+1	001
-2	110	+2	010
-1	111	+3	011

- Il primo bit indica ancora il segno
- Lo zero ha una sola codifica

## Interi: complemento a 2

Esempio 1:  $(-18)_{10} = (?)_{C2}$

Esempio 2:  $(9)_{10} = (?)_{C2}$

Esempio 3:  $(-6)_{10} = (1010)_{C2}$ , notazione *modulo e segno*?

## Somma di interi

- Rappresentare i numeri in C2
- Effettuare la somma in modo standard
- Non considerare l'eventuale riporto oltre il bit di segno

Esempio 1:  $(60)_{10} - (54)_{10} = (60)_{10} + (-54)_{10}$

## Somma di interi: overflow

- Sommiamo due interi in C2 rappresentati con  $n$  bit, quindi appartenenti a  $[-2^{n-1}; 2^{n-1} - 1]$
- Può succedere che il risultato cada al di fuori dell'intervallo
- Overflow:  $n$  bit in C2 bastano per rappresentare gli operandi, ma non per rappresentare il risultato
- Come riconoscerlo?
  - può succedere solo quando si sommano due operandi dello stesso segno: **se il segno del risultato è diverso da quello degli operandi** è avvenuto un overflow
  - **gli ultimi due riporti sono diversi** tra loro (01 o 10)

## Somma di interi: esempio di overflow

Esempio:  $(100)_{C2} + (101)_{C2}$

riporto	1	0		
	1	0	0	+
	1	0	1	=
<hr/>				
somma	(1)	0	0	1

- Sto sommando su **3** bit  $-4$  e  $-3$ , il risultato sarebbe:  
 $-7 < -2^{3-1}$ , non rappresentabile in C2 su 3 bit.



## Somma di interi: Esempi

Esempio 1:  $-(1101)_2 - (111)_2$

Esempio 2:  $-(1101)_2 - (111)_2$  Stesso esempio di prima, ma con l'aggiunta di un bit: anzich   $\frac{1}{2}$  lavorare su 5 bit, ne considero 6. Come cambia il risultato?

## Interi: complemento a 2

Come passare da C2 a base 10?

- Procedimento inverso:
  - Sottrarre 1
  - Complementare a 1
  - Convertire da binario a decimale e aggiungere il segno
- Metodo facilitato di verifica:
  - convertire in decimale con l'algoritmo standard assegnando al bit più significativo peso negativo
  - Esempio:  $(10100)_{C2} = -1 \times 2^4 + 1 \times 2^2 = (-12)_{10}$