Modulo: Approfondimenti sui Sistemi Aritmetici di un computer: tipo reale [P2_03]

Unità didattica: Sistema Aritmetico Reale Floating-Point [1-AT]

Titolo: Rappresentazione in memoria dei numeri reali

Argomenti trattati:

- ✓ Notazione scientifica dei numeri reali (segno, mantissa, esponente)
- ✓ Rappresentazione binaria della mantissa a bit implicito
- ✓ Sistema Aritmetico Floating Point IEEE Standard 754
- ✓ Oggetti del S.A. Standard e loro caratterizzazione

Prerequisiti richiesti: aritmetica binaria, sistema arit-metico intero

Notazione scientifica

La notazione più compatta per rappresentare i **numeri reali** è quella **scientifica** in cui si rappresentano solo le ***cifre significative***.

Esempi

300000000

0.00000100

0.00450010

zeri significativi

1 cifra significativa

zeri non significativi

5 cifre significative

Nella **notazione scientifica** un numero reale x si rappresenta tramite una mantissa m, contenente le cifre significative, ed un esponente p, indicazione degli zeri significativi, tali che $x = m \cdot \beta^p$ dove β è la **base** del sistema di numerazione.

Esempi:			notazione scientifica
300000000	$= 3 \cdot 10^{8}$	=	3e+8
0.00000100	$= 1 \cdot 10^{-6}$	=	1e-6
0.00450010	$= 4.5001 \cdot 10^{-3}$	=	4.5001e-3

Per individuare univocamente la notazione scientifica di un numero, si opera la **normalizzazione** della mantissa da cui discende il valore dell'esponente:

Esempio: **2** schemi di normalizzazione (base $\beta=10$)

356000000	=	$3.5600 \cdot 10^{+8}$	=	$0.35600 \cdot 10^{+9}$
0.00000123	=	$1.2300 \cdot 10^{-6}$	=	$0.12300 \cdot 10^{-5}$
0.00450010	=	$4.5001 \cdot 10^{-3}$	=	$0.45001 \cdot 10^{-2}$

...oppure...

1

mantissa m tale che

$$1 \le m < \beta=10$$

la cifra delle unità può essere 1,2,3,4,5,6,7,8,9 ma <u>non</u> 0

2

mantissa m tale che

$$0.1=\beta^{-1} \le m < 1$$

la cifra dei decimi può essere 1,2,3,4,5,6,7,8,9 ma <u>non</u> 0

Esempio: base $\beta=2$

bit esplicito

110110000	=	$1.1011 \cdot 2^{+1000}$	=	$0.11011 \cdot 2^{+1001}$
0.000001011	=	$1.0110 \cdot 2^{-0110}$	=	$0.10110 \cdot 2^{-0101}$
0.0010111	=	$1.0111 \cdot 2^{-0011}$	=	$0.10111 \cdot 2^{-0010}$

1. mantissa m tale che $1 \le m < \beta=2$

2. mantissa m tale che $1/2=\beta^{-1} \le m < 1$

la prima cifra della mantissa può essere solo 1

La prima cifra (significativa) della mantissa è sempre 1

(allora può **non** essere memorizzata!)

rappresentazione a **bit implicito**

bit implicito

110110000	=	$\boxed{1.}1011 \cdot 2^{+1000}$	=	$\boxed{0.1}1011 \cdot 2^{+1001}$
0.000001011	=	$\boxed{1.}0110 \cdot 2^{-0110}$	=	$\boxed{0.1}0110 \cdot 2^{-0101}$
0.0010111	=	$\boxed{1.}0111 \cdot 2^{-0011}$	=	$\boxed{0.1}0111 \cdot 2^{-0010}$

1. mantissa m tale che $1 \le m < \beta$

2. mantissa m tale che $\beta^{-1} \le m < 1$

SISTEMA ARITMETICO REALE FLOATING-POINT

Sistema Aritmetico Floating Point
$F(\beta, t, E_{min}, E_{max})$

Denota l'insieme dei numeri reali rappresentati in un computer e le operazioni definite su di essi.

Parametri del sistema aritmetico $F(\beta, t, E_{min}, E_{max})$:

- β = base del sistema di numerazione
- t = numero di cifre$_\beta$ per la mantissa
- $E_{min} < E_{max}$ = limitazioni per il campo esponente

Sistema Aritmetico Floating Point IEEE Standard 754 (1985)

$\beta = 2$

Formato	Tipo	num. bit	ling. C
Basic	*single*	**32** (4 byte)	*float*
	double	**64** (8 byte)	*double*
Extended	*double*	**80** (10 byte)	*long double*

opzionale

(prof. M. Rizzardi)

Nel ***S.A. IEEE Std.*** $\mathbf{F(2, t, E_{min}, E_{max})}$ un numero (normalizzato) del **formato Basic** è rappresentato in memoria come:

s	e	m
segno	esponente	mantissa

dove

- **s** denota il ***segno della mantissa***;
- l'***esponente*** **e** è rappresentato come "*intero biased*";
- la ***mantissa*** **m** è rappresentata con **s** *per segno e modulo* su **t** bit a *bit implicito* ℓ† (**precisione = t+1**) ed è generata con lo schema del *round to nearest*;

ed il suo **valore** è pertanto $\quad x = (-1)^{\mathbf{s}}[\ell.\mathbf{m}] \times 2^{\mathbf{e}-\mathrm{Bias}}$

---

† il primo bit ℓ non è rappresentato esplicitamente, ma assume un valore convenzionale: pertanto è come se la mantissa fosse rappresentata su $\mathbf{t}+1$ bit.

Perché i campi si susseguono in questo modo?

È così assicurato l'ordinamento dei numeri!!!

(considerando tutti i bit insieme come se fosse un intero)

bit +signif. ← bit –signif.

s	e	m

... a parità di segno ...
l'ordine di grandezza predomina

I negativi precedono i positivi

s	e	m

IEEE Standard 754

Tipo	numero bit esponente	Bias	t = numero bit mantissa
single	8 e∈{0..255}	127	23
double	11 e∈{0..2047}	1023	52
double EXTEND	15 e∈{0..32767}	16383	64

opzionale

E_{min} E_{max}

senza bit implicito

Il *S.A. IEEE Standard 754* stabilisce:

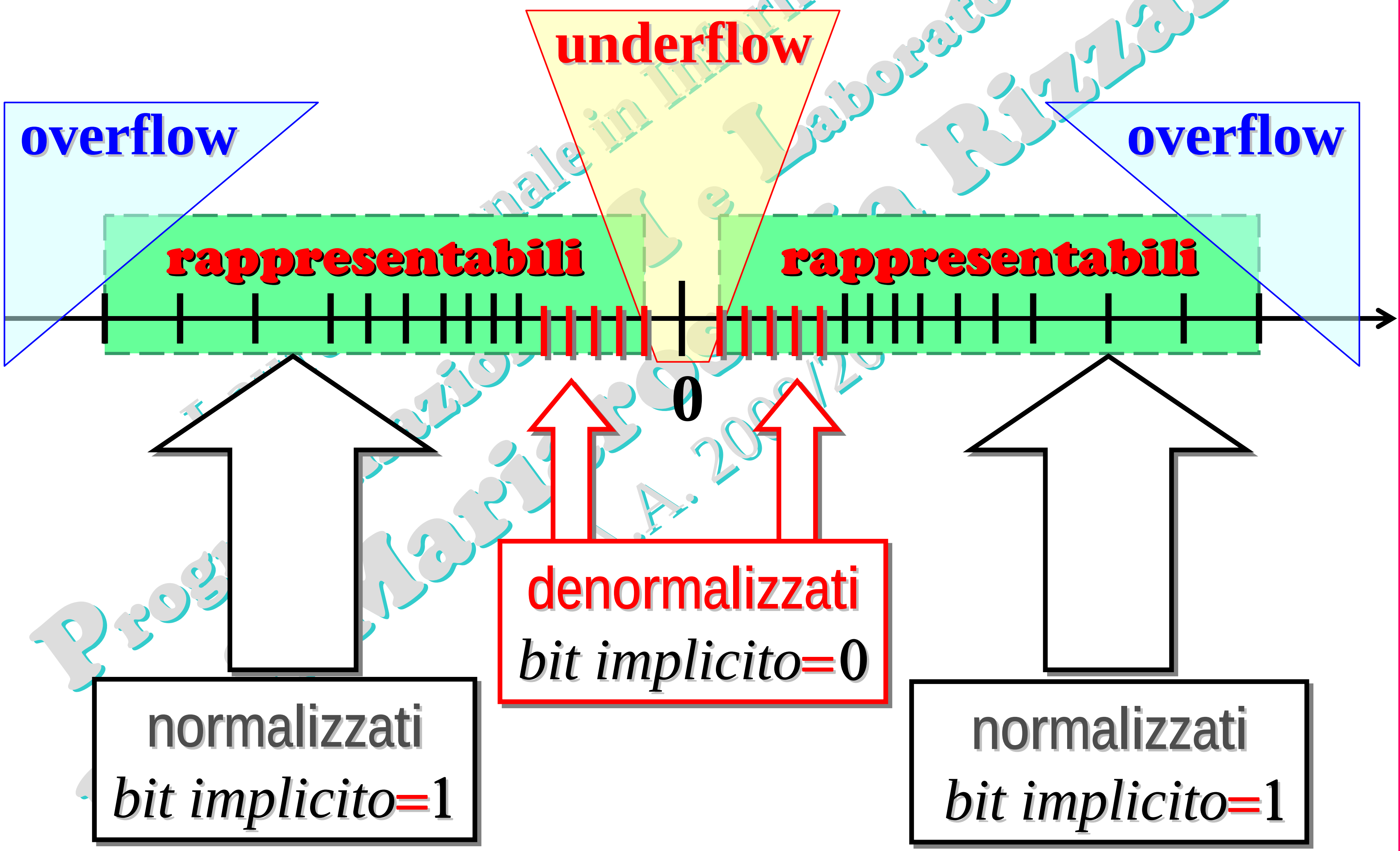
- gli oggetti del ***Sistema Aritmetico Floating-Point***;
- le operazioni definite su ciascun oggetto.

valore bit implicito

Oggetto	Caratterizzazione: esponente **e**	mantissa **m**	ℓ
Numeri normalizzati valore $= (-1)^{s}[\ell.\mathbf{m}]\times 2^{\mathbf{e}-\mathrm{Bias}}$	$E_{min} < \mathbf{e} < E_{max}$	$\mathbf{m} \geq 0$	1
Infinito con segno	$\mathbf{e} = E_{max}$	$\mathbf{m} = 0$	-
NaN (Not A Number)	$\mathbf{e} = E_{max}$	$\mathbf{m} \neq 0$	-
Zero con segno	$\mathbf{e} = E_{min}$	$\mathbf{m} = 0$	-
Numeri denormalizzati valore $= (-1)^{s}[\ell.\mathbf{m}]\times 2^{\mathbf{e}-\mathrm{Bias}+1}$	$\mathbf{e} = E_{min}$	$\mathbf{m} \neq 0$	0

I numeri Floating Point non sono uniformemente distribuiti sull'asse reale.

Esempio: numeri del **Formato Basic**



Esempio

$F(\beta=2,\ t=2,\ E_{min}=0,\ E_{max}=3)$

2 bit per la mantissa

n = 2 bit per l'esponente (**Bias** = 1)

s	e		valore espon.	m		ℓ	valore numero
0	0	0	-	0	0	0	+ zero
0	0	0	0	0	1	0	$0.01\ 2^0=+1/4$
0	0	0	0	1	0	0	$0.10\ 2^0=+1/2$
0	0	0	0	1	1	0	$0.11\ 2^0=+3/4$
0	0	1	0	0	0	1	$1.00\ 2^0=+1$
0	0	1	0	0	1	1	$1.01\ 2^0=+5/4$
0	0	1	0	1	0	1	$1.10\ 2^0=+3/2$
0	0	1	0	1	1	1	$1.11\ 2^0=+7/4$
					...		

$\text{valore}=(-1)^s[\ell.m]\times 2^{E_{min}-Bias+1}$

denormalizzati

realmin

$\text{valore}=(-1)^s[\ell.m]\times 2^{e-Bias}$

normalizzati

...

s	e		valore espon.	m		ℓ	valore numero
0	1	0	+1	0	0	1	$1.00\ 2^{+1}=+2$
0	1	0	+1	0	1	1	$1.01\ 2^{+1}=+5/2$
0	1	0	+1	1	0	1	$1.10\ 2^{+1}=+3$
0	1	0	+1	1	1	1	$1.11\ 2^{+1}=+7/2$
0	1	1		0	0		Inf
0	1	1		0	1		NaN
0	1	1		1	0		NaN
0	1	1		1	1		NaN
1	0	0	-	0	0	0	- zero
1	0	0	0	0	1		
1	...	...		...	...		

normalizzati

$$\text{valore}=(-1)^s[\ell.m]\times 2^{e-\text{Bias}}$$

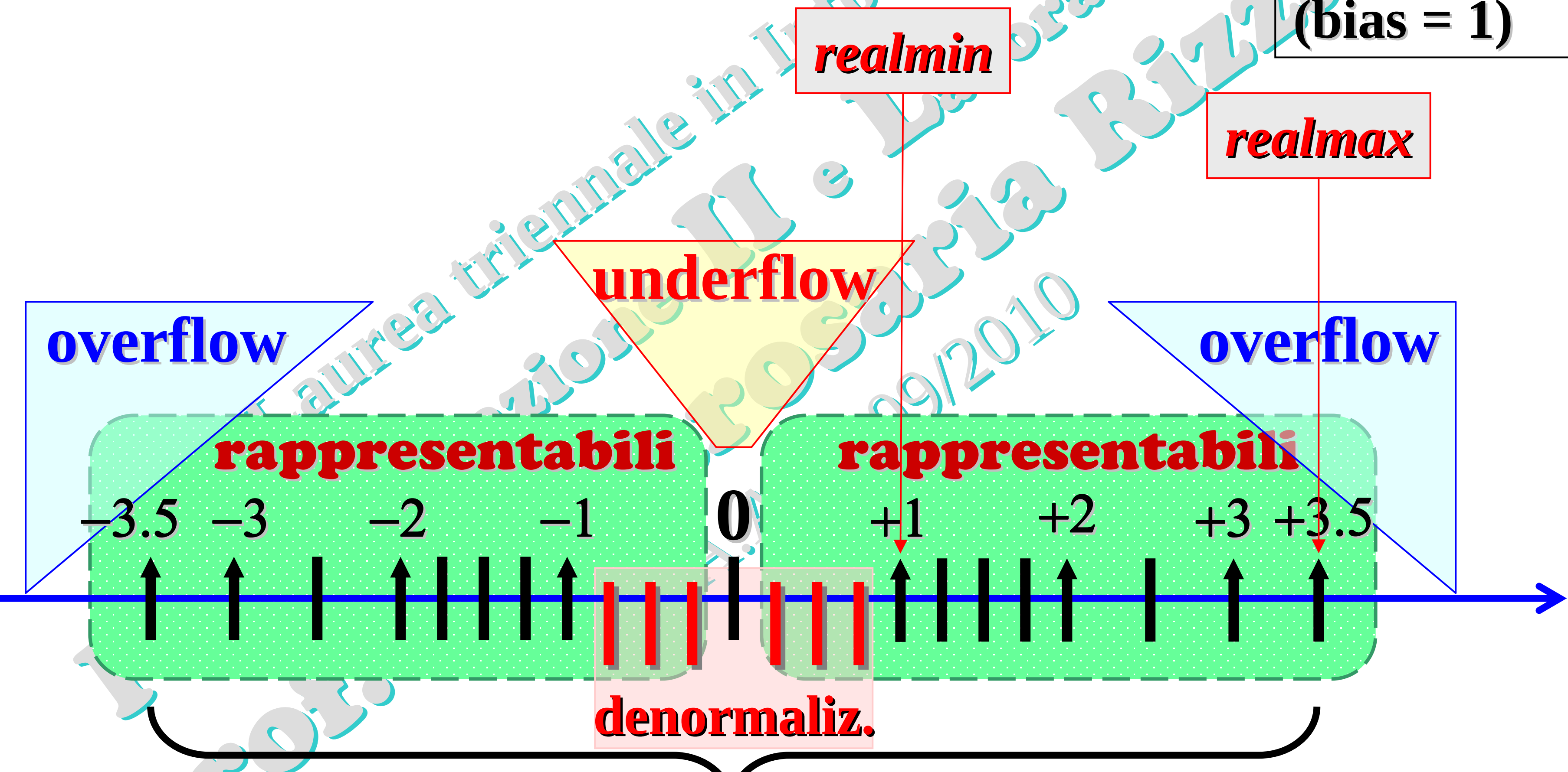
realmax

NEGATIVI

$$F(\beta=2,\ t=2,\ E_{min}=0,\ E_{max}=3)$$

2 bit per la mantissa

2 bit per l'esponente (bias = 1)



$$F(\beta=2,\ t=2,\ E_{min}=0,\ E_{max}=3)$$