SAPIENZA
Università di Roma
Dipartimento di Informatica

Rappresentazione dei numeri razionali

Prof. Daniele Gorla

Numeri razionali in virgola fissa

Sempre un sistema posizionale in base b (≥ 2).
Le prime m cifre rappresentano la parte intera, le successive n la parte frazionaria.

$$c_{m-1}\ldots c_1 c_0 , c_{-1} c_{-2} \ldots c_{-n} = \sum_{i=0}^{m-1} c_i b^i + \sum_{i=-1}^{-n} c_i b^i = \sum_{i=0}^{m-1} c_i b^i + \sum_{i=1}^{n} \frac{c_{-i}}{b^i}$$

con $c_i \in \{ 0 , \ldots , b\text{-}1 \}$.

Quindi, un numero razionale N è una coppia

$$< Ni \, , Nf >$$

formata da una parte intera (Ni) e una frazionaria (Nf)

Cambiamento di base

Trasformare $< Ni \, , Nf >_a$ in $< Ni' , \; Nf' >_b$

- Per la parte intera, segue il procedimento di cambiamento di base per numeri naturali

- Per la parte frazionaria, procediamo in maniera simile:
  - se la base di arrivo è 10, usa il *metodo polinomiale*
  - se la base di partenza è 10, usa un metodo di *moltiplicazioni iterate* (vedi dopo)
  - altrimenti, effettua due conversioni:
    - una da base a a base 10 (metodo polinomiale)
    - l'altra da base 10 a base b (moltiplicazioni iterate)

Metodo polinomiale (da base b a base 10)

$$c_{m-1}\ldots c_1 c_0 , c_{-1} c_{-2} \ldots c_{-n} = \sum_{i=0}^{m-1} c_i b^i + \sum_{i=1}^{n} \frac{c_{-i}}{b^i}$$

Esempio: convertire $1011{,}011_2$ in base 10

$$1011{,}011_2 = \left(1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 + \frac{0}{2^1} + \frac{1}{2^2} + \frac{1}{2^3}\right)_{10}$$

$$= \left(8 + 2 + 1 + \frac{1}{4} + \frac{1}{8}\right)_{10} = \left(11 + \frac{2+1}{8}\right)_{10} = \left(11 + \frac{3}{8}\right)_{10} = 11{,}375_{10}$$

Conversione della parte frazionaria (da base 10 a base *b*)

Diciamo di voler convertire un numero frazionario puro

$$F = 0,c_{-1}\, c_{-2} \dots c_{-n}$$

Sappiamo che F rappresenta il numero

$$\sum_{i=1}^{n} \frac{c_{-i}}{b^i} = \frac{c_{-1}}{b} + \frac{c_{-2}}{b^2} + \frac{c_{-3}}{b^3} + \dots + \frac{c_{-(n-1)}}{b^{n-1}} + \frac{c_{-n}}{b^n}$$

Se moltiplichiamo F per b otteniamo

$$b \cdot F = c_{-1} + \frac{c_{-2}}{b} + \frac{c_{-3}}{b^2} + \dots + \frac{c_{-(n-1)}}{b^{n-2}} + \frac{c_{-n}}{b^{n-1}}$$

cioè un numero della forma $c_{-1}\, , c_{-2} \dots c_{-n}$

Quindi, $b{\cdot}F$ è un numero la cui parte intera è la prima cifra frazionaria di F e la cui parte frazionaria è formata dalle restanti cifre frazionarie di F.

Conversione della parte frazionaria (da base 10 a base *b*)

A questo punto, iteriamo sul numero frazionario puro

$$F^{(2)} = 0,c_{-2}\, c_{-3} \dots c_{-n}$$

Se moltiplichiamo $F^{(2)}$ per b otteniamo

$$b \cdot F^{(2)} = c_{-2} + \frac{c_{-3}}{b} + \frac{c_{-4}}{b^2} + \dots + \frac{c_{-(n-1)}}{b^{n-3}} + \frac{c_{-n}}{b^{n-2}}$$

cioè un numero della forma $c_{-2}\, , c_{-3} \dots c_{-n}$

Itera questo procedimento finché:

- $F^{(k)} = 0$, per qualche k *(N.B.: diversamente dal metodo di divisioni iterate, questo non sempre avviene)*
- oppure ottieni una parte periodica (che si ripete all'infinito)
- oppure hai raggiunto il massimo numero di cifre disponibili per la rappresentazione della parte frazionaria in base b

Esempio:

Convertire $17{,}416_{10}$ in base 2 con 8 bit sia per P.I. che per P.F.

1. Converti parte intera (*divisioni iterate*):

   17:2 = 8 resto 1   8:2 = 4 resto 0   4:2 = 2 resto 0
   2:2 = 1 resto 0   1:2 = 0 resto 1

   Quindi, $17_{10} = 10001_2$

2. Converti parte frazionaria (*moltiplicazioni iterate*):

$0{,}416 \times 2 = 0{,}832$	da cui	P.I. = 0	P.F. = 0,832
$0{,}832 \times 2 = 1{,}664$	da cui	P.I. = 1	P.F. = 0,664
$0{,}664 \times 2 = 1{,}328$	da cui	P.I. = 1	P.F. = 0,328
$0{,}328 \times 2 = 0{,}656$	da cui	P.I. = 0	P.F. = 0,656
$0{,}656 \times 2 = 1{,}312$	da cui	P.I. = 1	P.F. = 0,312
$0{,}312 \times 2 = 0{,}624$	da cui	P.I. = 0	P.F. = 0,624
$0{,}624 \times 2 = 1{,}248$	da cui	P.I. = 1	P.F. = 0,248
$0{,}248 \times 2 = 0{,}496$	da cui	P.I. = 0	P.F. = 0,496

Perciò $0{,}416_{10} = 0{,}01101010_2$

Quindi, $17{,}416_{10} = 00010001{,}01101010_2$

Attenzione:

il numero così ottenuto è un'approssimazione (inferiore) del numero di partenza (ci siamo fermati quando la parte frazionaria non era ancora 0)

Infatti:

$$00010001{,}01101010_2 = 2^4 + 1 + \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^5} + \frac{1}{2^7}$$

$$= 17 + \frac{2^5 + 2^4 + 2^2 + 1}{2^7} = 17 + \frac{32+16+4+1}{128} = 17 + \frac{53}{128}$$

$$= 17{,}4140625 < 17{,}416$$

Esempio (con periodicità):

Convertire $120{,}03_{10}$ in base 5

1. Converti parte intera (*divisioni iterate*):

   120:5 = 24 resto 0 24:5 = 4 resto 4 4:5 = 0 resto 4

   Quindi, $120_{10} = 440_5$

2. Converti parte frazionaria (*moltiplicazioni iterate*):

   $0{,}03 \times 5 = 0{,}15$ da cui P.I. = 0 P.F. = 0,15
   $0{,}15 \times 5 = 0{,}75$ da cui P.I. = 0 P.F. = 0,75
   $0{,}75 \times 5 = 3{,}75$ da cui P.I. = 3 P.F. = 0,75
   $0{,}75 \times 5 = 3{,}75$ da cui P.I. = 3 P.F. = 0,75
   …

   Perciò $0{,}03_{10} = 0{,}00333\ldots_5$

   Quindi, $120{,}03_{10} = 440{,}00\overline{3}_5$

Conversione opposta (con periodicità):

Convertire $0{,}0\overline{3}$ da base 5 a base 10.

Applichiamo il metodo polinomiale:

$$0{,}0\overline{3}_5 = \left(\frac{0}{5^1} + \frac{3}{5^2} + \frac{3}{5^3} + \ldots\right)_{10} = \sum_{i>1} \frac{3}{5^i} = 3 \sum_{i>1} \frac{1}{5^i}$$

$$= 3\left(\sum_{i>0} \frac{1}{5^i} - \frac{1}{5}\right) = 3\left(\frac{1}{4} - \frac{1}{5}\right) = \frac{3}{20} = 0{,}15_{10}$$

dove abbiamo usato la serie geometrica: $\sum_{i>0} \frac{1}{c^i} = \frac{1}{c-1}$
(per $c > 1$)

Problemi della rappresentazione in virgola fissa

L'intervallo dei reali rappresentabile è piccolo e con approssimazioni grossolane

Esempio: avendo a disposizione 32 bit e assegnandone 20 per la P.I. (in Ca2) e 12 per la P.F. si ha

- P.I. $\in \{-2^{19}+1, \ldots, 2^{19}-1\} = \{-524.287, \ldots, 524.287\}$
- la P.F si hanno a disposizione al più 4 cifre frazionarie in base 10 (infatti $2^{-12} = \frac{1}{4096} \approx 0{,}00025$)

Ovviamente, si può ridurre la P.I. a favore della P.F., per aumentare la precisione (di poco però), a scapito dell'ampiezza dell'intervallo

In ogni caso, ***non è una rappresentazione adeguata per calcoli scientifici reali!!***

Rappresentazione in virgola mobile

Un razionale r è rappresentato dalla terna

$$< s, m, e >$$

Gli elementi della terna sono chiamati rispettivamente

- *bit di segno* (s=0 per numero positivo, s=1 per numero negativo)
- *mantissa*, un numero razionale m in virgola fissa espresso in base b
- *esponente*, un intero e espresso in Complemento alla base b.

La terna $< s, m, e >$ rappresenta il numero

$$(-1)^s \cdot m \cdot b^e$$

Questa rappresentazione si ispira alla famosa notazione scientifica per cui scriviamo

-5×10^3 invece di -5000 o 4×10^{-2} invece di 0,04

Forma Normalizzata

Per garantire l'unicità della rappresentazione di un numero, si adotta una *forma normalizzata* in cui la mantissa è tale che:

- la sua parte intera è nulla
- la sua parte frazionaria inizia con una cifra non nulla

Es: 0,1011

D'ora in poi useremo sempre questa convenzione, per cui la terna $< s\ ,\ m\ ,\ e >$ sarà tale che m è semplicemente un naturale in base b e il numero rappresentato da tale terna è

$$(-1)^s \cdot 0,m \cdot b^e$$

OSS.: l'unico numero che non può rispettare la forma normalizzata è lo zero, che verrà codificato come < 0 , 0…0 , 0...0 >

Cambiamento di base in virgola mobile

SAPIENZA Università di Roma Dipartimento di Informatica

Trasformare N_a: $< s\ ,\ m_a\ ,\ e_a >$ in N'_b: $< s\ ,\ m_b\ ,\ e_b >$

1. trasforma la tripla $< s\ ,\ m_a\ ,\ e_a >$ in un numero in virgola fissa (senza segno):
$$0,m_a \times a^{e_a} = (h,k)_a$$
2. applica il procedimento di conversione da base a a base b per il numero in virgola fissa $(h,k)_a$ ottenendo $(p,q)_b$

3. Converti $(p,q)_b$ dalla forma in virgola fissa a quella in virgola mobile (normalizzata), ottenendo così $< s\ ,\ m_b\ ,\ e_b >$

Esempio

SAPIENZA Università di Roma Dipartimento di Informatica

Convertire in base 2 il numero $< 0\ ,\ 9375\ ,\ -1 >_{10}$ assumendo di avere 1 bit per il segno, 8 per la mantissa e 4 per l'esponente.

1. Il numero originale è $0,9375 \times 10^{-1}$.
   Lo trasformo in virgola fissa: $0,09375_{10}$

2. Applico il metodo delle moltiplicazioni iterate:
   $0,09375 \times 2 = 0,1875$ $\quad 0,1875 \times 2 = 0,375$ $\quad 0,375 \times 2 = 0,75$
   $0,75 \times 2 = 1,5$ $\quad 0,5 \times 2 = 1,0$
   ottenendo quindi $0,09375_{10} = 0,00011_2$

3. Trasformo tale numero in virgola mobile normalizzata: $0,11 \times 2^{-3}$

4. la rappresentazione cercata, in forma di tripla, è:
$$< 0\ ,\ 11000000\ ,\ 1101 >_2$$

Range dei numeri in virgola mobile

SAPIENZA Università di Roma Dipartimento di Informatica

Supponiamo di avere M bit di mantissa e E di esponente

numeri negativi: La mantissa va da $-0,\underbrace{11\ldots1}_{M}$ a $-0,1\underbrace{0\ldots0}_{M-1}$

numeri positivi: La mantissa va da $+0,1\underbrace{0\ldots0}_{M-1}$ a $+0,\underbrace{11\ldots1}_{M}$

L'esponente, in Ca2, va da $-2^{E-1}+1$ a $+2^{E-1}-1$

Quindi, i numeri positivi sono tra $0,1 \times 2^{-2^{E-1}+1}$ e $0,1.....1 \times 2^{2^{E-1}-1}$
i numeri negativi sono tra $-0,1......1 \times 2^{2^{E-1}-1}$ e $-0,1 \times 2^{-2^{E-1}+1}$

Relazione fra numero di bit di *M* ed *E* (a parità di *M*+*E*)

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

E=3 bit e *M*=4 bit (in base 2)

E \ M	1000	1001	...	1111
101 (=-3)	0,0625	0,0703125	...	0,1171875
110 (=-2)	0,125	0,140625	...	0,234375
111 (=-1)	0,25	0,28125	...	0,46875
000 (=0)	0,5	0,5625	...	0,9375
001 (=1)	1	1,125	...	1,875
010 (=2)	2	2,25	...	3,75
011 (=3)	4	4,5	...	7,5

E=2 bit e *M*=5 bit

E \ M	10000	10001	10010	10011	10100	...	11111
11 (=-1)	0,25	0,26525	0,28125	0,296875	0,3155	...	0,484375
00 (=0)	0,5	0,53125	0,5625	0,59375	0,625	...	0,96875
01 (=1)	1	1,0625	1,125	1,1875	1,25	...	1,9375

Precisione e Ampiezza

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- ***precisione** della rappresentazione:* distanza tra due numeri adiacenti
- ***ampiezza** della rappresentazione:* caratterizzata dal valore assoluto del più grande/piccolo numero rappresentabile

IDEALE: rappresentazione ampia e precisa

- Maggior **precisione** → più bit alla mantissa
- Maggior **ampiezza** → più bit all'esponente

Occorre dunque trovare un compromesso!

I quattro formati standard IEEE 754 - 1985

	Precisione			
	Singola	**Singola estesa**	**Doppia**	**Doppia estesa**
bit di segno	1	1	1	1
bit della mantissa	23	≥ 31	52	≥ 63
bit dell esponente	8	≥ 11	11	≥ 15
bit del formato	32	≥ 43	64	≥ 79