laboratorio di architettura degli elaboratori

rappresentazione dei numeri

Daniele Radicioni

$ whoami

- Daniele Radicioni
- ufficio al terzo piano
- http://www.di.unito.it/~radicion/
- e-mail: daniele.radicioni@unito.it
- ricevimento: tendenzialmente al lunedì pomeriggio 14-16, su appuntamento
- NO consulenza via mail!!!
- materiale del corso disponibile su I-learn

ripasso sulla rappresentazione

Number Systems

Base (b)	Number system	Alphabet
2	Binary system	0,1
8	Octal system	0,1,2,3,4,5,6,7
10	Decimal system	0,1,2,3,4,5,6,7,8,9
16	Hexadecimal system	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Number Systems

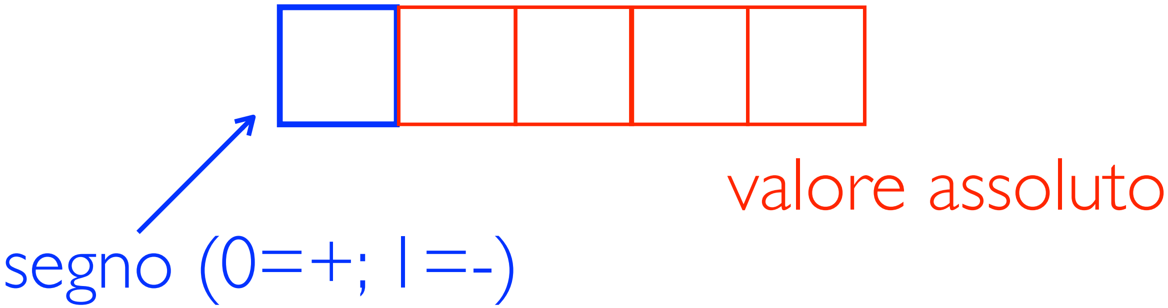
- positional number systems represent numbers as a sequence of digits z_i, such as

$$z_n\ z_{n-1}\ \ldots\ z_1\ z_0\ .\ z_{-1}\ z_{-2}\ \ldots\ z_{-m}\ ; \qquad \text{e.g., } 1234.56$$

- the value X_b of the number is the sum of all single values of the positions $z_i b^i$:

$$X_b = z_n b^n + z_{n-1} b^{n-1} + \cdots + z_1 b^1 + z_0 b^0 + z_{-1} b^{-1} + \cdots + z_{-m} b^{-m} = \sum_{i=-m}^{n} z_i b^i$$

rappresentazione modulo e segno



Decimale	Binario
11 =	0 1011
-11 =	1 1011

- Rappresentazione su N cifre binarie:
  - una cifra binaria (per convenzione quella più a sinistra) per codificare il segno
  - le rimanenti N-1 cifre rappresentano la codifica in forma binaria fissa del valore assoluto del numero (che per definizione è un numero naturale)

rappresentazione modulo e segno

- intervallo di numeri rappresentabili su N bit

$$[-(2^{N-1} - 1), +(2^{N-1} - 1)]$$

- svantaggi
  - Lo zero può essere rappresentato in due modi diversi (00…00, ossia +0, e 10…00, ossia –0)
  - L'operazione di somma tra due numeri non può essere realizzata applicando l'algoritmo di somma sulla rappresentazione dei moduli

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numero corrisponderebbero se avessimo usato la codifica modulo e segno?

0	0000	
1	0001	
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	
7	0111	
8	1000	
9	1001	
10	1010	
11	1011	
12	1100	
13	1101	
14	1110	
15	1111	

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numero corrisponderebbero se avessimo usato la codifica modulo e segno?

0	0000	+0
1	0001	+1
2	0010	+2
3	0011	+3
4	0100	+4
5	0101	+5
6	0110	+6
7	0111	+7
8	1000	-0
9	1001	-1
10	1010	-2
11	1011	-3
12	1100	-4
13	1101	-5
14	1110	-6
15	1111	-7

Rappresentazione in complemento a uno

- Rappresentazione complemento a 1 su N cifre binarie:
  - Numero positivo $+k$: rappresentazione come modulo e segno (N-1 bit per il valore k e il *MSB* è messo a 0)
  - Numero negativo $-k$: codifico in binario puro $(2^N – 1) - k$
  - $(2^N – 1)$ su N bit è una sequenza di N cifre 1

Decimale	C1	Binario puro
+11	01011	11
-11	10100	20 (2^5-1-11)

Rappresentazione in complemento a uno

- La rappresentazione in C1 di un numero negativo -k si può anche ottenere applicando l'operatore di complemento a 1 al modulo e segno di +k:

- Operatore di complemento a 1: complemento a 1 bit a bit, cioè sostituire 0 con 1 e viceversa 1 con 0

Decimale	binario
+11	01011
-11	10100

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numero corrisponderebbero se avessimo usato la codifica Complemento a 1?

0	0000	
1	0001	
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	
7	0111	
8	1000	
9	1001	
10	1010	
11	1011	
12	1100	
13	1101	
14	1110	
15	1111	

di.unito.it
DIPARTIMENTO DI INFORMATICA

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numeri corrisponderebbero se avessimo usato la codifica Complemento a 1?

0	0000	
1	0001	+1
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	
7	0111	
8	1000	
9	1001	
10	1010	
11	1011	-4
12	1100	
13	1101	
14	1110	
15	1111	

positivi: modulo & segno

1011 segno negativo
1011 complemento:
0100 = 4
quindi -4

di.unito.it
Dipartimento di Informatica

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numero corrisponderebbero se avessimo usato la codifica Complemento a 1?

0	0000	+0
1	0001	+1
2	0010	+2
3	0011	+3
4	0100	+4
5	0101	+5
6	0110	+6
7	0111	+7
8	1000	-7
9	1001	-6
10	1010	-5
11	1011	-4
12	1100	-3
13	1101	-2
14	1110	-1
15	1111	-0

Rappresentazione in complemento a uno

- Caratteristiche molto simili alla rappresentazione modulo e segno, ossia:
  - Semplicità della rappresentazione
  - Stesso intervallo dei numeri rappresentabili

$$[\, -(2^{N-1}-1), +(2^{N-1}-1)\,]$$

  - Due rappresentazioni possibili per lo zero ("000...0" per +0 e "111...1" per –0)

Rappresentazione in complemento a due

- Rappresentazione complemento a 2 su N cifre binarie:
  - Numero positivo +*k*: rappresentazione come modulo e segno (N-1 bit per il valore k e il MSB è messo a 0)
  - Numero negativo -*k*: codifico in binario puro $(2^N) - k$

Decimale	C2	Binario puro
+11	01011	11
-11	10101	21 (2^5-11)

di.unito.it
Dipartimento di Informatica

Rappresentazione in complemento a due

- La rappresentazione in C2 di un numero negativo -k si può anche ottenere applicando l'operatore di complemento a 2 al modulo e segno di +k:
  - Operatore di complemento a 2: complemento a 1 bit a bit (cioè sostituire 0 con 1 e viceversa 1 con 0) e poi sommare la costante 1.
    - per esempio per -11 partiamo dalla rappresentazione del modulo 01011 (che equivale a 11); complemento a 1: 10100; incrementato di 1 diventa 10101.

Decimale	binario
+11	01011
-11	10101

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numeri corrisponderebbero se avessimo usato la codifica Complemento a 2?

0	0000	
1	0001	
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	
7	0111	
8	1000	
9	1001	
10	1010	
11	1011	
12	1100	
13	1101	
14	1110	
15	1111	

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numeri corrisponderebbero se avessimo usato la codifica Complemento a 2?

0	0000	+0
1	0001	+1
2	0010	+2
3	0011	+3
4	0100	+4
5	0101	+5
6	0110	+6
7	0111	+7
8	1000	-8
9	1001	-7
10	1010	-6
11	1011	-5
12	1100	-4
13	1101	-3
14	1110	-2
15	1111	-1

Rappresentazione in complemento a due

- Alcune proprietà interessanti
  - 0 si rappresenta con 000...00
  - -1 si rappresenta con 111...11
  - Il massimo numero positivo è 011…11
  - Il minimo numero negativo è 100…00

- Dato un numero negativo, scambiando 0 e 1 (operazione di complemento a 1) si ottiene il suo modulo diminuito di 1.
  - Es. su 4 bit: $-5_{10} = 1011_2$; $0100_2 = 4_{10}$.

Rappresentazione in complemento a due

- Caratteristiche:
  - Intervallo dei numeri rappresentabili su N bit: $[-(2^{N-1}), +(2^{N-1}-1)]$
  - Unica rappresentazione per lo zero
  - L'operazione di somma effettuata operando sulla rappresentazione del numero indipendentemente dal segno produce, a meno di overflow, sempre il risultato corretto.

rappresentazione in eccesso a 2^{N-1}

- Eccesso 2^{N-1} (per numeri rappresentati con N bit)

usando 8 cifre binarie si ha la rappresentazione in eccesso a 128 (cioè 2^7, 2^{N-1})

-128	00000000	(-128+128=0)
-127	00000001	(-127+128=1)
...		
-2	01111101	(-2+128=126)
...		
25	01011001	25+128=153
...		
127	11111111	(127+128=255)

rappresentazione in eccesso a 2^{N-1}

- La rappresentazione su N cifre binarie di un numero è ottenuta sommando 2^{N-1} al numero stesso e codificando il numero ottenuto (che è positivo) in binario.
  - Intervallo di rappresentazione dei numeri: $[-(2^{N-1}),+(2^{N-1}-1)]$
  - Vantaggio: l'ordinamento viene mantenuto nella codifica

eccesso a 2^7, o eccesso a 128

-127	00000001	(-127+128=1)
-2	01111101	(-2+128=126)
25	01011001	25+128=153

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numeri corrisponderebbero se avessimo usato la codifica in eccesso a 2^3?

0	0000	
1	0001	
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	
7	0111	
8	1000	
9	1001	
10	1010	
11	1011	
12	1100	
13	1101	
14	1110	
15	1111	

verifica

- Esercizio: scrivere tutti i numeri in binario puro da 0 a 15 su 4 bit. Queste 16 sequenze di bit a quali numeri corrisponderebbero se avessimo usato la codifica in eccesso a 2^3?

0	0000	-8
1	0001	-7
2	0010	-6
3	0011	-5
4	0100	-4
5	0101	-3
6	0110	-2
7	0111	-1
8	1000	0
9	1001	+1
10	1010	+2
11	1011	+3
12	1100	+4
13	1101	+5
14	1110	+6
15	1111	+7

conversioni fra basi

Dalla base r alla base 10

Base 2

cifre usate 0 e 1 (bit)

$\mathbf{1\ 0\ 0\ 1\ 0.01_2} = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$

$= 16 + 2 + 0.25 = 18.25_{10}$

Base 8

cifre usate 0, 1, 2, 3, 4, 5, 6, 7

$\mathbf{1\ 0\ 5\ 1\ 7.25} = 1 \times 8^4 + 0 \times 8^3 + 5 \times 8^2 + 1 \times 8^1 + 7 \times 8^0 + 2 \times 8^{-1} + 5 \times 8^{-2}$

$= 4096 + 320 + 8 + 7 + 0.25 + 0.078125 \quad = 4431.328125$

Base 16

cifre usate 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

$\mathbf{D\ E\ .\ C} = 13 \times 16^1 + 14 \times 16^0 + 12 \times 16^{-1}$

$= 208 + 14 + 0.75 = 222.\ 75$

Dalla base r alla base 10

Base 2

cifre usate 0 e 1 (bit)

1 1 0 . 1 0 1 1 $= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4}$

$= 4 + 2 + 0.5 + 0.125 + 0.0625 = 6.6875$

Base 8

cifre usate 0, 1, 2, 3, 4, 5, 6, 7

3 2 . 1 1 $= 3 \times 8^1 + 2 \times 8^0 + 1 \times 8^{-1} + 1 \times 8^{-2}$

$= 24 + 2 + 0.125 + 0.015625 = 26.140625$

Base 16

cifre usate 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

1A.F $= 1 \times 16^1 + 10 \times 16^0 + 15 \times 16^{-1}$

$= 16 + 10 + 0.9375 = 26.9375$

Dalla base 10 alla base r

Parte intera

1. Inizio;
2. Dividere il numero decimale per la base di arrivo r;
3. Il resto della divisione è una cifra nella nuova base a partire dalla cifra meno significativa ;
4. Il quoziente della divisione intera è il nuovo dividendo;
5. Se quoziente $\neq 0$ torna a 2);
6. Fine.

Dalla base 10 alla base r

- Consideriamo ad esempio il numero **13_{10}** e calcoliamo la sua rappresentazione in base due:

13/2	**=**	**6**	**resto**	**1**
6/2	**=**	**3**	**resto**	**0**
3/2	**=**	**1**	**resto**	**1**
1/2	**=**	**0**	**resto**	**1**

- Leggendo i resti dal basso verso l'alto, si ha che la rappresentazione binaria del numero **13_{10}** è **1101_2**

Dalla base 10 alla base r

- Consideriamo ad esempio il numero **42_{10}** e calcoliamo la sua rappresentazione in base due:

  42/2 = 21 resto 0
  21/2 = 10 resto 1
  10/2 = 5 resto 0
  5/2 = 2 resto 1
  2/2 = 1 resto 0
  1/2 = 0 resto 1

- Leggendo i resti dal basso verso l'alto, si ha che la rappresentazione binaria del numero **42_{10}** è **101010_2**

Dalla base 10 alla base r

- Consideriamo ad esempio il numero **345_{10}** e calcoliamo la sua rappresentazione in base due:

  345/2 = 172 resto 1

  172/2 = 86 resto 0
  86/2 = 43 resto 0
  43/2 = 21 resto 1
  21/2 = 10 resto 1
  10/2 = 5 resto 0
  5/2 = 2 resto 1
  2/2 = 1 resto 0
  1/2 = 0 resto 1

- Leggendo i resti dal basso verso l'alto, si ha che la rappresentazione binaria del numero **345_{10}** è **101011001_2**

Dalla base 10 alla base r

- Consideriamo ad esempio il numero 345_{10} e calcoliamo la sua rappresentazione in base sedici:

  345/16 = 21 resto 9

  21/16 = 1 resto 5

  1/16 = 0 resto 1
- Leggendo i resti dal basso verso l'alto, si ha che la rappresentazione esadecimale del numero 345_{10} è 159_{16}

Dalla base 10 alla base r

Parte frazionaria

1. Inizio;
2. Moltiplicare la parte frazionaria del numero decimale per la base di arrivo;
3. Separare parte intera e parte frazionaria;
4. La parte intera dà una cifra nella nuova base a partire dalla cifra più significativa ;
5. Se non ottengo 0 o non raggiungo la precisione richiesta torna a 2);
6. Fine.

Dalla base 10 alla base r

0.875	***1***.75
0.75	***1***.5
0.5	***1***.0

							0	.	1	1	1
7	6	5	4	3	2	1	0		-1	-2	-3

Dalla base 10 alla base r

0.24	***0***.48
0.48	***0***.96
0.96	***1***.92
0.92	***1***.84
0.84	***1***.68

					0	**.**	**0**	**0**	**1**	**1**	**1**
5	4	3	2	1	0		-1	-2	-3	-4	-5

Dalla base 10 alla base 16

0.24	***3***.84
0.84	***13***.44
0.44	***7***.04
0.04	***0***.64
0.64	***10***.24
….	….

0 . 3 D 7 0 A

5 4 3 2 1 0 -1 -2 -3 -4 -5

Esempio 35,59375

Per la parte intera

35	
17	1
8	1
4	0
2	0
1	0
0	1

0.59375	***1***.1875
0.1875	***0***.375
0.375	***0***.75
0.75	***1***.5
0.5	***1***.0

1	0	0	0	1	1	.	1	0	0	1	1
5	4	3	2	1	0		-1	-2	-3	-4	-5

Conversione tra basi: bin ↔ ott

Binario → Ottale

(62.875)

$$\underbrace{1\quad 1\quad 1}_{7}\quad \underbrace{1\quad 1\quad 0}_{6}\quad .\quad \underbrace{1\quad 1\quad 1}_{7}$$

(62.875)

Ottale → Binario

(82.5)

$$\overbrace{0\quad 0\quad 1}^{1}\quad \overbrace{0\quad 1\quad 0}^{2}\quad \overbrace{0\quad 1\quad 0}^{2}\quad .\quad \overbrace{1\quad 0\quad 0}^{4}$$

(82.5)

Conversione tra basi: bin ↔ esa

Binario → Esadecimale

$$\underbrace{1\quad 0\quad 0\quad 1}_{9}\quad \underbrace{0\quad 0\quad 1\quad 0}_{2}\ .\ \underbrace{1\quad 0\quad 0\quad 1}_{9}$$

(146.5625)

(146.5625)

Esadecimale → Binario

$$\overbrace{1\quad 1\quad 0\quad 0}^{C}\quad \overbrace{1\quad 1\quad 1\quad 1}^{F}\ .\ \overbrace{1\quad 0\quad 1\quad 0}^{A}$$

(207.625)

(207.625)