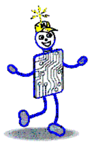


CAPITOLO 3: Sistemi Numerici

The Architecture of Computer Hardware, Systems Software & Networking: An Information Technology Approach

5th Edition, Irv Englander

John Wiley and Sons ©2013

Diapositive realizzate da Angela Clark, University of South Alabama

Diapositive per la 4° edizione realizzate da Wilson Wong, Bentley University

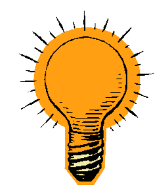
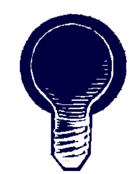
Diapositive per il CLEII tradotte e adattate da Gianluca Amato, Univ. CH-PI

Conteggi e aritmetica

- Sistema di numerazione decimale o in base 10
  - Origine: contare sulle dita
  - In inglese “digit” (cifra) deriva dalla parola latina *digitus* che vuol dire “dito”
- *Base*: il numero di cifre usate dal sistema numerico, incluso lo zero.
  - Esempio: la base 10 ha 10 cifre, da 0 a 9
- *Binario* o *base 2*
  - *Bit* (abbreviazione per *binary digit*): 2 cifre, 0 e 1
- *Ottale* o *base 8:* 8 cifre, da 0 a 7
- *Esadecimale o base 16:* 16 cifre, da 0 a F
- Esempi: $\mathbf{10}_{10} = \mathbf{A}_{16}$; $\mathbf{11}_{10} = \mathbf{B}_{16}$

Perché la base binaria?

- I primi computer usavano la base 10
  - Mark I e ENIAC
- John von Neumann propose di usare la numerazione binaria (1945)
  - Semplificava il progetto dei computer
  - Usata per le istruzioni e per i dati
- Relazione naturale tra lo stato acceso/spento degli interruttori e i calcoli usando la logica proposizionale (booleana).

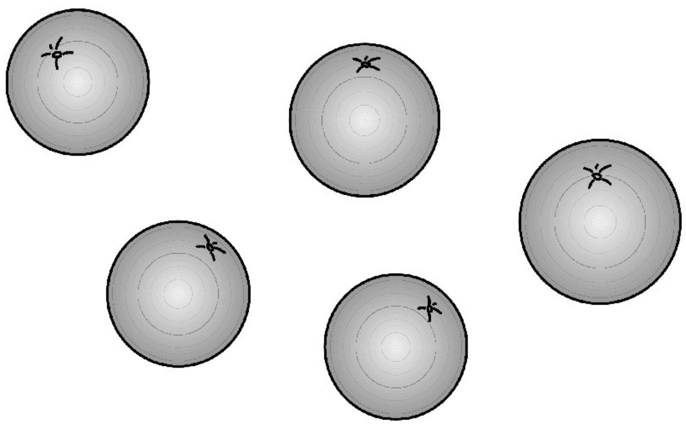
	
On	Off
Vero	Falso
Sì	No
1	0

Tenere traccia dei bit

- I bit sono comunemente manipolati e memorizzati in gruppi
  - 8 bit = 1 *byte*
  - 4 byte = 1 *word* o *parola* (in molti sistemi)
- Numero di bit usati nei calcoli:
  - Influenza l'accuratezza dei risultati
  - Limita la dimensione e l'intervallo dei numeri manipolati dal computer

Numeri: rappresentazione fisica

- Numerali differenti, stesso numero di arance
  - Cavernicoli: IIIII
  - Romani: V
  - Arabi: 5
- Basi differenti, stesso numero di arance
  - 5_{10}
  - 101_2
  - 12_3



Sistemi numerici

- Sistema romano: simboli *indipendenti* dalla posizione, semplice sistema *additivo*
  - Esempio: XII è un 10 seguito da due uno, ovvero 12
  - In realtà il sistema romano non è puramente additivo, perché in IV, il numero uno (I) viene sottratta da cinque (V), ma esistono sistemi numerici puramente additivi (i geroglifici egizi)
- Sistemi moderni: basati sulla notazione posizionale
  - Sistema decimale: basata sulle potenze di 10
  - Sistema binario: basato sulle potenze di 2
  - Sistema ottale: basato sulle potenze di 8
  - Sistema esadecimale: basato sulle potenze di 16

Sistema posizionale: base 10

$527_{10} = 5 \times 10^2 + 2 \times 10^1 + 7 \times 10^0$

100's place

10's place

1's place

Place	**10^2**	**10^1**	**10^0**
Value	**100**	**10**	**1**
Evaluate	**5 x 100**	**2 x 10**	**7 x1**
Sum	**500**	**20**	**7**

Sistema posizionale: ottale

$624_8 = 404_{10}$

64's place 8's place 1's place

Place	8^2	8^1	8^0
Value	64	8	1
Evaluate	6 x 64	2 x 8	4 x 1
Sum for Base 10	384	16	4

Sistema posizionale: esadecimale

$6{,}704_{16} = 26{,}372_{10}$

4,096's place 256's place 16's place 1's place

Place	16^3	16^2	16^1	16^0
Value	4,096	256	16	1
Evaluate	6 x 4,096	7 x 256	0 x 16	4 x 1
Sum for Base 10	24,576	1,792	0	4

Sistema posizionale: binario

$1101\ 0110_2 = 214_{10}$

Place	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Value	128	64	32	16	8	4	2	1
Evaluate	1 x 128	1 x 64	0 x 32	1 x16	0 x 8	1 x 4	1 x 2	0 x 1
Sum for Base 10	128	64	0	16	0	4	2	0

Intervallo di numeri possibili

- $R = B^K$
  - R = intervallo
  - B = base
  - K = numero di cifre
- Esempio #1: Base 10, 2 cifre
  - $R = 10^2 = 100$ numeri diversi (0…99)
- Example #2: Base 2, 16 cifre
  - $R = 2^{16} = 65{,}536$ o 64K
  - I numeri a 16-bit possono rappresentare 65,536 differenti valori numerici

Intervalli e numero di bit

Bits	Digits	Range
1	0+	2 (0 and 1)
4	1+	16 (0 to 15)
8	2+	256
10	3	1,024 (1K)
16	4+	65,536 (64K)
20	6	1,048,576 (1M)
32	9+	4,294,967,296 (4G)
64	19+	Approx. 1.6×10^{19}
128	38+	Approx. 2.6×10^{38}

Differenti basi numeriche

- Base:
  - Il numero di simboli diversi necessari per rappresentare qualunque numero

Più *grande* la base, *più* simboli sono richiesti

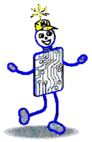
  - Base 10: 0,1,2,3,4,5,6,7,8,9
  - Base   2: 0,1
  - Base   8: 0,1,2,3,4,5,6,7
  - Base 16: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Numero di simboli vs numero di cifre

Per un dato numero, più *grande* la base

- *più* simboli sono richiesti
- ma *meno* cifre sono necessarie

- Esempio #1:
  - 65_{16} 101_{10} 145_{8} $110\ 0101_{2}$
- Esempio #2:
  - $11C_{16}$ 284_{10} 434_{8} $1\ 0001\ 1100_{2}$



Contare in base 2

Binary Number	Equivalent				Decimal Number
	8's (2^3)	4's (2^2)	2's (2^1)	1's (2^0)	
0				0×2^0	0
1				1×2^0	1
10			1×2^1	0×2^0	2
11			1×2^1	1×2^0	3
100		1×2^2			4
101		1×2^2		1×2^0	5
110		1×2^2	1×2^1		6
111		1×2^2	1×2^1	1×2^0	7
1000	1×2^3				8
1001	1×2^3			1×2^0	9
1010	1×2^3		1×2^1		10

Tabella addizione in base 10

$3_{10} + 6_{10} = 9_{10}$

+	0	1	2	3	4	5	6	7	8	9
0	0	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9	10
2	2	3	4	5	6	7	8	9	10	11
3	3	4	5	6	7	8	9	10	11	12
4	4	5	6	7	8	9	10	11	12	13

Tabella addizione in base 8

$3_8 + 6_8 = 11_8$

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	10
2	2	3	4	5	6	7	10	11
3	3	4	5	6	7	10	11	12
4	4	5	6	7	10	11	12	13
5	5	6	7	10	11	12	13	14
6	6	7	10	11	12	13	14	15
7	7	10	11	12	13	14	15	16

(no 8 or 9, of course)

Tabella moltiplic. in base 10

$3_{10} \times 6_{10} = 18_{10}$

x	0	1	2	3	4	5	6	7	8	9
0				0						
1		1	2	3	4	5	6	7	8	9
2		2	4	6	8	10	12	14	16	18
3		3	6	9	12	15	18	21	24	27
4	0	4	8	12	16	20	24	28	32	36
5		5	10	15	20	25	30	35	40	45
6		6	12	18	24	30	36	42	48	54
7		7	14	21	28	35	42	49	56	63

Tabella moltiplic. in base 8

3_8 x 6_8 = 22_8

x	0	1	2	3	4	5	6	7
0				0				
1		1	2	3	4	5	6	7
2		2	4	6	10	12	14	16
3	0	3	6	11	14	17	22	25
4		4	10	14	20	24	30	34
5		5	12	17	24	31	36	43
6		6	14	22	30	36	44	52
7		7	16	25	34	43	52	61

Addizione

Base	Problema	Cifra più grande
Decimal	6 +3	9
Octal	6 +1	7
Hexadecimal	6 +9	F
Binary	1 +0	1

Addizione

Base	Problema	Riporto	Risposta
Decimal	6 +4	Riporto 10	10
Octal	6 +2	Riporto 8	10
Hexadecimal	6 +A	Riporty 16	10
Binary	1 +1	Riporto 2	10

Aritmetica binaria

	1	1	1	1	1			
		1	1	0	1	1	0	1
+				1	0	1	1	0
	1	0	0	0	0	0	1	1

Aritmetica binaria

- Addizione
  - Booleana usando XOR e AND
- Moltiplicazione
  - AND
  - Scorrimento a *sinistra*
- Divisione
  - Sottrazione o complemento
  - Scorrimento a *destra*

+	0	1
0	0	1
1	1	10

x	0	1
0	0	0
1	0	1

Aritmetica binaria e logica

- Aritmetica realizzata tramite logica booleana
  - *O-ESCLUSIVO*
    - Produce “1” solo se uno degli input, ma ***non*** entrambi, sono “1”
  - *AND (bit di riporto)*
    - Produce “1” se e soltanto se entrambi gli input sono “1”

1	1	1	1	1			
	1	**1**	**0**	**1**	**1**	**0**	**1**
+			**1**	**0**	**1**	**1**	**0**
1	**0**	**0**	**0**	**0**	**0**	**1**	**1**

Moltiplicazione binaria

- Aritmetica realizzata tramite logica booleana
  - *AND (bit di riport)*
  - Produce “1” ↔ entrambi gli input sono “1”
  - *Spostamento (shifting)*
    - Spostare un numero a ***sinistra*** di una cifra ***moltiplica*** il suo valore per la base
    - Spostare un numero a ***destra*** di una cifra ***divide*** il suo valore per la base
    - Esempi:
      - 10_{10} shift ***left*** = 100_{10}
      - 10_{10} shift ***right*** = 1_{10}
      - 10_2 shift ***left*** = 100_2
      - 10_2 shift ***right*** = 1_2

Moltiplicazione binaria

			1	1	0	1	
				1	0	1	
			1	1	0	1	1's place
					0		2's place
	1	1	0	1			4's place (bits ***shifted*** to line up with 4's place of multiplier)
1	0	0	0	0	0	1	Result (***AND***)

Conversione dalla base 10

- Tabella delle potenze di 2

Power / Base	8	7	6	5	4	3	2	1	0
2	256	128	64	32	16	8	4	2	1
8				32,768	4,096	512	64	8	1
16					65,536	4,096	256	16	1

Dalla base 10 alla base 2

$42_{10} = 101010_2$

Power / Base	6	5	4	3	2	1	0
2	64	32	16	8	4	2	1
		1	0	1	0	1	0
Quoziente		42/32 = 1	10/16 = 0	10/8 = 1	2/4 = 0	2/2 = 1	0/1 = 0
Resto		10	10	2	2	0	0

Dalla base 10 alla base 2

Base 10 42

Resto

Quoziente

2)	42	(0		Cifra meno significativa
2)	21	(1		
2)	10	(0		
2)	5	(1		
2)	2	(0		
2)	1			Cifra più significativa

Base 2 101010

Dalla base 10 alla base 16

$5{,}735_{10} = 1667_{16}$

Power / Base	4	3	2	1	0
16	65,536	4,096	256	16	1
		1	**6**	**6**	**7**
Intero		5,735 / 4,096 = 1	1,639 / 256 = 6	103 /16 = 6	7
Resto		5,735 - 4,096 = 1,639	1,639 - 1,536 = 103	103 - 96 = 7	

Dalla base 10 alla base 16

Base 10 8,039

Resto

Quoziente

16)	8,039	(7	Cifra meno significativa
16)	502	(6	
16)	31	(15	
16)	1	(1	Cifra più significativa
16)	0		

Base 16 1F67

Dalla base 8 alla base 10

$7263_8 = 3{,}763_{10}$

Power	8^3	8^2	8^1	8^0
	512	64	8	1
	x 7	x 2	x 6	x 3
Sum for Base 10	**3,584**	**128**	**48**	**3**

Dalla base 8 alla base 10

$\mathbf{7263_8} = \mathbf{3{,}763_{10}}$

$$
\begin{array}{rcl}
\mathbf{7} & & \\
\underline{\times\ 8} & & \\
56 & +\ \mathbf{2} = & 58 \\
 & & \underline{\times\ 8} \\
 & & 464 \quad +\ \mathbf{6} = \quad 470 \\
 & & \qquad\qquad\qquad\quad \underline{\times\ 8} \\
 & & \qquad\qquad\qquad\quad 3760 \quad +\ \mathbf{3} = \quad \mathbf{3{,}763}
\end{array}
$$

Dalla base 16 alla base 2

- Perché l’esadecimale?
  - Più facile da leggere e scrivere rispetto al binario
  - I sistemi operativi e le reti moderne presentano una varietà di informazione di debugging in format esadecimale
- Approccio basato sul nibble

Base 16	1	F	6	7
Base 2	0001	1111	0110	0111

Frazioni

- Number point o radix point
  - Punto decimale in base 10
  - Punto binario in base 2
- Nessuna relazione esatta tra numeri frazionari in basi diverse
  - Conversioni esatte possono essere impossibili, come nel caso di 1/3 o 0.3333333…

Frazioni decimali

- Spostare la virgola un posto a destra
  - Effetto: moltiplica il numero per la base
  - Example: 139.0_{10} ➡ 1390_{10}
- Spostare la virgola un posto a sinistra
  - Effetto: divide il numero per la base
  - Example: 139.0_{10} ➡ 13.9_{10}

Frazioni: base 10 e base 2

$.2589_{10} = 2 \times 10^{-1} + 5 \times 10^{-2} + 8 \times 10^{-3} + 9 \times 10^{-4}$

Place	10^{-1}	10^{-2}	10^{-3}	10^{-4}
Value	1/10	1/100	1/1000	1/10000
Evaluate	2 x 1/10	5 x 1/100	8 x 1/1000	9 x1/1000
Sum	.2	.05	.008	.0009

$.101011_{2} = 0.671875_{10}$

Place	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}
Value	1/2	1/4	1/8	1/16	1/32	1/64
Evaluate	1 x 1/2	0 x 1/4	1x 1/8	0 x 1/16	1 x 1/32	1 x 1/64
Sum	.5		0.125		0.03125	0.015625

Frazioni: base 10 e base 2

- Nessuna relazione generale tra frazioni del tipo $1/10^k$ e $1/2^k$
  - Pertanto un numero rappresentabile in base 10 può non essere rappresentabile in base 2.
  - Ma l’opposto è diverso: tutte le frazioni della forma $1/2^k$ possono essere rappresentati in base 10
- Le conversioni di frazioni tra una base e un altra si interrompono:
  - Se c’è una soluzione esatta
  - Quando è stata ottenuta l’accuratezza desiderata

Conversione di numeri misti

- Parte intera e frazionaria vanno convertite separatamente
- Radix point: riferimento fisso per la converione
  - La cifra a sinistra della virgola rappresenta l’unità in qualunque base
  - B^0 è sempre uguale a 1 indipendentemente della base.

Copyright 2013 John Wiley & Sons