Sistemi di Numerazione

Corso di Laboratorio di Calcolo

Numeri e Numerali

Il numero “cinque”

5	V	_	Π	五
Arabo	Romano	Maya	Greco	Cinese

Il sistema decimale

0	10	20	100
1	11	21	101
2	12	22	102
3	13	23	103
4	14	24	104
5	15	25	105
6	16	26	106
7	17	27	107
8	18	28	108
9	19	29	109

Sistemi Posizionali

1492

$$1 \times 10^3 + 4 \times 10^2 + 9 \times 10^1 + 2 \times 10^0$$

Sistemi Posizionali

1492

$$1 \times 10^3 + 4 \times 10^2 + 9 \times 10^1 + 2 \times 10^0$$

Sistemi Posizionali

1492

$1 \times 10^3 + 4 \times 10^2 + 9 \times 10^1 + 2 \times 10^0$

Sistemi a base generica

- In un sistema in base n un numero è rappresentato da una sequenza di simboli della base (es.1492)
- Il valore di ciascun simbolo corrisponde a quello della cifra moltiplicato per una potenza di n corrispondente alla posizione che occupa il simbolo nel numerale, contando da 0.
- La potenza corrispondente si trova contando le cifre da destra a sinistra
- Un numerale a di M cifre in base *b* vale

$$a = \sum_{i=0}^{M-1} c_i b^i$$

Addizioni e sottrazioni

$$1492 + 48 =$$

$$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (2 \times 10^0) +$$

$$(4 \times 10^1) + (8 \times 10^0) =$$

Addizioni e sottrazioni

$$\begin{array}{r} 1492+ \\ 48= \\ \hline 10 \end{array}$$

$1492 + 48 =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (2 \times 10^0) +$

$(4 \times 10^1) + (8 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (10 \times 10^0) =$

Addizioni e sottrazioni

$1492 + 48 =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (2 \times 10^0) +$
$(4 \times 10^1) + (8 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (10 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (1 \times 10^1) + (0 \times 10^0) =$

$$\begin{array}{r} 1 \\ 1492+ \\ 48= \\ \hline 0 \end{array}$$

Addizioni e sottrazioni

$$\begin{array}{r} 1 \\ 1492+ \\ 48= \\ \hline 140 \end{array}$$

$1492 + 48 =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (2 \times 10^0) +$
$(4 \times 10^1) + (8 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (10 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (1 \times 10^1) + (0 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (14 \times 10^1) + (0 \times 10^0) =$

Addizioni e sottrazioni

$$
\begin{array}{r}
1 \\
1492+ \\
48= \\
\hline
40
\end{array}
$$

$1492 + 48 =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (2 \times 10^0) +$
$(4 \times 10^1) + (8 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (10 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (1 \times 10^1) + (0 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (14 \times 10^1) + (0 \times 10^0) =$

$(1 \times 10^3) + (4 \times 10^2) + (1 \times 10^2) + (4 \times 10^1) + (0 \times 10^0) =$

Addizioni e sottrazioni

$$1492 + 48 =$$

$$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (2 \times 10^0) + (4 \times 10^1) + (8 \times 10^0) =$$

$$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (10 \times 10^0) =$$

$$(1 \times 10^3) + (4 \times 10^2) + (9 \times 10^1) + (4 \times 10^1) + (1 \times 10^1) + (0 \times 10^0) =$$

$$(1 \times 10^3) + (4 \times 10^2) + (14 \times 10^1) + (0 \times 10^0) =$$

$$(1 \times 10^3) + (4 \times 10^2) + (1 \times 10^2) + (4 \times 10^1) + (0 \times 10^0) =$$

$$(1 \times 10^3) + (5 \times 10^2) + (4 \times 10^1) + (0 \times 10^0) =$$

$$1540$$

$$\begin{array}{r} 1492+ \\ 48= \\ \hline 1540 \end{array}$$

Il sistema binario

$$
\begin{array}{rcccccccccl}
0 & = & 0\times2^3 & + & 0\times2^2 & + & 0\times2^1 & + & 0\times2^0 & = & 0 \\
1 & = & 0\times2^3 & + & 0\times2^2 & + & 0\times2^1 & + & \color{red}{1\times2^0} & = & 1 \\
10 & = & 0\times2^3 & + & 0\times2^2 & + & \color{red}{1\times2^1} & + & 0\times2^0 & = & 2 \\
11 & = & 0\times2^3 & + & 0\times2^2 & + & \color{red}{1\times2^1} & + & \color{red}{1\times2^0} & = & 3 \\
100 & = & 0\times2^3 & + & \color{red}{1\times2^2} & + & 0\times2^1 & + & 0\times2^0 & = & 4 \\
101 & = & 0\times2^3 & + & \color{red}{1\times2^2} & + & 0\times2^1 & + & \color{red}{1\times2^0} & = & 5 \\
110 & = & 0\times2^3 & + & \color{red}{1\times2^2} & + & \color{red}{1\times2^1} & + & 0\times2^0 & = & 6 \\
111 & = & 0\times2^3 & + & \color{red}{1\times2^2} & + & \color{red}{1\times2^1} & + & \color{red}{1\times2^0} & = & 7 \\
1000 & = & \color{red}{1\times2^3} & + & 0\times2^2 & + & 0\times2^1 & + & 0\times2^0 & = & 8 \\
1001 & = & \color{red}{1\times2^3} & + & 0\times2^2 & + & 0\times2^1 & + & \color{red}{1\times2^0} & = & 9 \\
1010 & = & \color{red}{1\times2^3} & + & 0\times2^2 & + & \color{red}{1\times2^1} & + & 0\times2^0 & = & 10
\end{array}
$$

Il sistema esadecimale

Nel sistema esadecimale la base è **16**

Si devono inventare 6 nuovi simboli!

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

Il sistema esadecimale

- Nel sistema esadecimale le nuove cifre valgono:
  - **A = 10**
  - **B = 11**
  - **C = 12**
  - **D = 13**
  - **E = 14**
  - **F = 15**
- Il valore del numero si ottiene moltiplicando il valore della cifra per la potenza di 16 corrispondente alla posizione della cifra

Il sistema esadecimale

6E43AF

$$6*16^5+14*16^4+4*16^3+3*16^2+10*16^1+15*16^0$$

7 226 287

Eseguire le Somme

```
100110101+
001101010=
__________
```

Eseguire le Somme

```
100110101+
001101010=
__________
        1
```

Eseguire le Somme

```
100110101+
001101010=
__________
       11
```

Eseguire le Somme

```
100110101+
001101010=
__________
      111
```

Eseguire le Somme

```
100110101+
001101010=
__________
     1111
```

Eseguire le Somme

```
100110101+
001101010=
__________
    11111
```

Eseguire le Somme

```
100110101+
001101010=
__________
   1011111
```

Eseguire le Somme

```
  1
100110101+
001101010=
__________
    011111
```

Eseguire le Somme

```
   1
100110101+
001101010=
__________
  10011111
```

Eseguire le Somme

```
  11
100110101+
001101010=
__________
    0011111
```

Eseguire le Somme

```
  11
 100110101+
 001101010=
__________
  10011111
```

Eseguire le Somme

```
         11
309   100110101+
106   001101010=
___   __________
415   110011111
```

Overflow e Underflow

- Il numero di cifre utilizzate da un computer per rappresentare un numero è fissato (8, 16, 32 o 64)
  - ∃ un estremo superiore per l'insieme dei numeri
  - Overflow, underflow
  - Con 8 bit (massimo numero 255) la somma 200+100=44 perchè 300 = 100101100

Eseguire sottrazioni

- Sottrarre due numeri equivale a sommare al primo l’opposto del secondo $\Rightarrow$ numeri negativi

- Le operazioni vengono eseguite da circuiti elettrici nei computer
- Usando un trucco le sottrazioni si possono eseguire utilizzando lo stesso circuito delle addizioni

Numeri negativi

- Le sottrazioni possono essere convertite in addizioni se si usano i numeri negativi
- Numeri con segno (ad es. –2, -56.45, …)
  - Nel sistema binario si può riservare un bit per il segno (0 = +, 1= -)
- Numeri in complemento
  - Si ottengono sottraendo a b^n il valore assoluto del numero di n cifre
  - $-987 = 10^3-987 = 1000-987 = 13$
  - Si sfrutta il fatto che le cifre a disposizione sono limitate

Numeri in complemento a 10

- Il complemento di un numero si ottiene scrivendo il complemento delle cifre e aggiungendo 1
- Il complemento della cifra è il numero che manca per avere il valore della base –1
- Esempi:
  - $8 \rightarrow 1$
  - $3 \rightarrow 6$
  - $9 \rightarrow 0$
  - $0 \rightarrow 9$

```
 1849+
 8150=
 _____
 9999+
    1=
 _____
10000
```

Numeri in complemento a 10

- 8151 è il complemento a 10 di 1849 e rappresenta il numero –1849 poiché, sommato ad esso, dà 0 ignorando la cifra più significativa
- Nei computer le cifre usate per rappresentare i numeri sono fissate
- Lo stesso circuito può eseguire somme e sottrazioni

```
 1849+
 8150=
 _____
 9999+
    1=
 _____
10000
```

~~1~~

Numeri in complemento a 10

- Le sottrazioni si eseguono facilmente complementando i numeri
- 972 è il complemento di 27 e quindi rappresenta -27

```
153-         153+
 27=         972+
               1=
_____        _____
126         1126
```

Numeri in complemento a 2

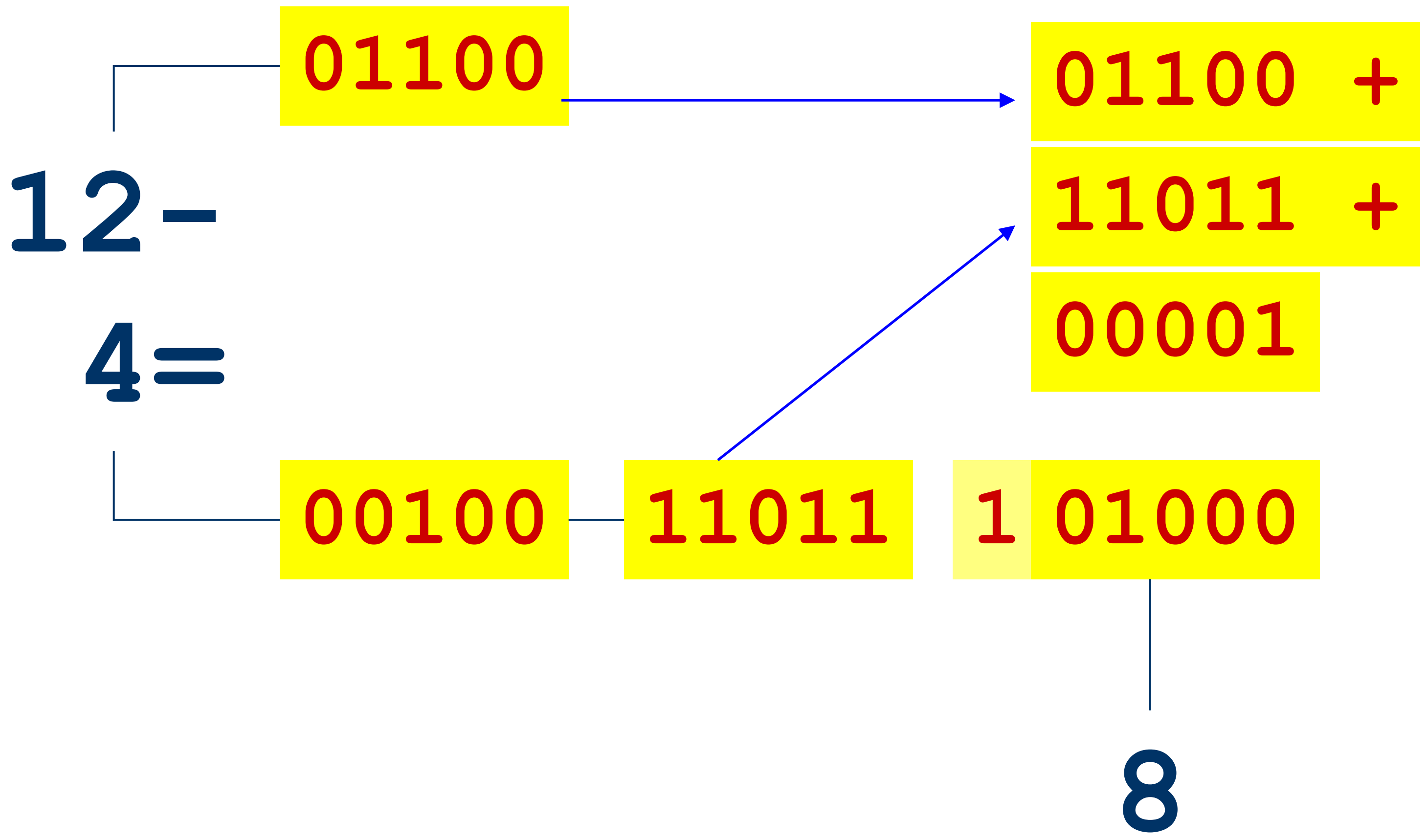
- I numeri negativi in base 2 si ottengono con la stessa tecnica

$$\mathbf{0} \rightarrow \mathbf{1}$$

$$\mathbf{1} \rightarrow \mathbf{0}$$

- Usando un numero di cifre adeguato si distinguono subito i numeri positivi (che iniziano con 0) dai numeri negativi (che iniziano con 1)

Un esempio



Gli operatori logici

- Operano su valori logici:
  - VERO $\Leftrightarrow$ 1
  - FALSO $\Leftrightarrow$ 0
- Restituiscono un valore logico (0 o 1)
- Godono delle proprietà
  - Commutativa
  - Associativa
- Proposizione = sequenza di valori logici

Gli operatori logici base

AND La proposizione è vera se tutti gli operandi sono veri

OR La proposizione è vera se almeno uno gli operandi è vero

NOT La proposizione viene negata

Le tavole delle verità

- Strumento utile per evidenziare il risultato di un’operazione logica
- In una griglia si dispongono tutte le possibili combinazioni di bit in ingresso e si aggiunge una colonna con il valore del risultato dell’operazione

La Tavola dell'AND

0	0	0
0	1	0
1	0	0
1	1	1

La Tavola dell'OR

0	0	0
0	1	1
1	0	1
1	1	1

La Tavola del NOT

0	1
1	0

Alcuni Esempi

```
100101 AND
010011
_____________
000001
```

```
100101 OR
010011
_____________
110111
```

```
NOT 010011 = 101100
```

Un’espressione Logica

- Rappresentiamo gli operatori AND, OR e NOT con i simboli &&, || e !

```
(1001||0110)&&((0111 && !0101)||0011)=?
```

Un’espressione Logica

- Rappresentiamo gli operatori AND, OR e NOT con i simboli &, | e !

```
(1001||0110)&&((0111 && !0101)||0011) =
(1111) && ((0111 && 1010) || 0011)
```

Un’espressione Logica

- Rappresentiamo gli operatori AND, OR e NOT con i simboli &, | e !

```
(1001||0110)&&((0111 && !0101)||0011) =
(1111) && ((0111 && 1010) || 0011)=
(1111) && ((0010) || 0011)
```

Un’espressione Logica

- Rappresentiamo gli operatori AND, OR e NOT con i simboli &, | e !

```
(1001||0110)&&((0111 && !0101)||0011) =
(1111) && ((0111 && 1010) || 0011)=
(1111) && ((0010) || 0011)=
(1111) && (0011)
```

Un'espressione Logica

- Rappresentiamo gli operatori AND, OR e NOT con i simboli &, | e !

```
(1001||0110)&&((0111 && !0101)||0011) =
(1111) && ((0111 && 1010) || 0011)=
(1111) && ((0010) || 0011)=
(1111) && (0011)=
(0011)
```

Rappresentare numeri razionali

- Il numero di bit utilizzati per rappresentare i numeri in un computer è fissato
  - Con 8 bit si rappresentano i numeri da 0 a 255 (oppure da –128 a 127)
- Rappresentare i numeri razionali con la virgola in posizione fissa è un inutile spreco di bit (34.56 = 0000034.5600000000)
- La precisione varia a seconda del valore del numero

Rappresentazione in Virgola Mobile

- I numeri si devono esprimere utilizzando il maggior numero di cifre significative $\rightarrow$ virgola mobile
- I numeri si rappresentano in notazione esponenziale con mantissa ed esponente
  $x = m \times 2^{E}$
- Si assegnano pochi bit all'esponente e il resto (m) alla mantissa
- Il bit più alto è il segno della mantissa
- La mantissa è compresa tra 0 e $1\text{-}2^{-m}$
- La mantissa è in forma normale C+m, dove C vale 0 o 1

Rappresentazione in virgola mobile

Segno della Mantissa

0	00000000	00000000000000000000000

Esponente

Mantissa

virgola mobile: un esempio

2

Rappresentazione in eccesso a 127

Segno della Mantissa

0	10000010	01100000000000000000000

Esponente

Mantissa

$$1*2^{-2}+1*2^{-3}=0.25+0.125=0.375$$

virgola mobile: un esempio

2

Segno della Mantissa

0	10000010	01100000000000000000000

Esponente

Mantissa

0.375

$$(1 + 0.375)*2^2=5.5$$

Caratteri

- Un carattere è un simbolo utilizzato per rappresentare un fonema o un numero
  - N.B. : il carattere 8 è concettualmente diverso dal valore 8 (che si può rappresentare, ad esempio, in numeri romani come VIII)

- Nei computer alcuni elementi pseudo-grafici sono rappresentati da caratteri
  - CR: Carriage Return (torna all'inizio della riga)
  - LF: Line Feed (avanza di una riga)
  - BEL: Bell (campanello)

Caratteri

- I caratteri alfabetici e quelli speciali (CR, LF, BEL, etc.) si rappresentano associando ad ognuno di essi un numero binario di 8 bit (codice ASCII) o 32 bit (Unicode)

- Alcuni esempi:

A ↔ 65	**a ↔ 97**
C ↔ 67	**c ↔ 99**
Z ↔ 90	**z ↔ 122**

Tipi di dati

- La stessa stringa di bit può rappresentare
  - Numeri interi positivi
  - Numeri interi con segno
  - Numeri razionali (reali)
  - Valori logici
  - Caratteri
- Per poterla interpretare è necessario conoscerne il tipo