10. Sistemi di numerazione

Definizioni
Un **sistema di numerazione** è l'insieme dei simboli e delle regole che permettono di esprimere i numeri. Un sistema di numerazione si dice **posizionale** se il valore di un simbolo dipende dalla posizione che esso occupa nella scrittura del numero.
Si chiamano **cifre** i simboli usati in un sistema di numerazione posizionale.
Esempio. Nel numero 1375 le cifre sono 1, 3, 7, 5.
Si dice **base** di un sistema di numerazione il numero di simboli che si usano per scrivere i numeri.
Sistema di numerazione a base n significa che ogni numero si esprime mediante l'uso di n cifre, ciascuna delle quali ha un valore che corrisponde alla potenza di base n ed esponente il posto k occupato da quella cifra contando a partire da destra verso sinistra.
Solitamente si usano i simboli 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 per i sistemi di numerazione a base minore o uguale a 10, per i sistemi di numerazione a base maggiore di 10, dopo il 9 si usano le lettere dell'alfabeto: A, B, C, D, E, F, …
Scrittura polinomiale di un numero. Nei sistemi posizionali ogni numero intero si esprime come somma di potenze che hanno per base la base del sistema di numerazione ed esponente la posizione occupata dalla cifra:

$$n = \sum_{k=1}^{m} a_k B^k \text{ , con base } B \geq 2 \text{ , cifre } 0 \leq a_k \leq B-1, a_k \in \mathbb{N}$$

10.1 Sistema decimale

Nel sistema di numerazione decimale si usano i dieci simboli: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
Scrittura polinomiale. Nella scrittura polinomiale un numero in base 10 si scrive come somma di potenze di 10 moltiplicate per le cifre che compongono il numero, la potenza del dieci che moltiplica ciascuna cifra è pari alla posizione occupata dalla cifra contando da destra verso sinistra e cominciando a contare da zero.
Esempio. $7408 = 7\cdot 10^3 + 4\cdot 10^2 + 0\cdot 10^1 + 8\cdot 10^0$
Numeri decimali. Un qualsiasi numero reale può essere scritto nella forma polinomiale

$$r = \pm\sum_{k=-\infty}^{n} a_k 10^k = \pm\left(\ldots + a_{-3}10^{-3} + a_{-2}10^{-2} + a_{-1}10^{-1} + a_0 10^0 + a_1 10^1 + a_2 10^2 + \ldots + a_n 10^n\right)$$

Dove *n* è un indice finito opportuno; le potenze negative di 10 possono essere infinite nel caso di un numero decimale periodico o irrazionale.
Esempio. $132{,}0735 = 5\cdot\frac{1}{10.000} + 3\cdot\frac{1}{1.000} + 7\cdot\frac{1}{100} + 0\cdot\frac{1}{10} + 2 + 3\cdot 10 + 1\cdot 100$

10.2 Sistema binario

Nel sistema di numerazione binario si usano due simboli, solitamente indicati con 0 e 1.
Cambiamento di base da 2 a 10. Per trasformare un numero dalla base 2 alla base 10 si scrive il numero nella forma polinomiale e si calcola il valore dell'espressione ottenuta considerando tutti i numeri come scritti in base 10.
Esempio

$$11101001_{(2)} = 1\cdot 2^0 + 0\cdot 2^1 + 0\cdot 2^2 + 1\cdot 2^3 + 0\cdot 2^4 + 1\cdot 2^5 + 1\cdot 2^6 + 1\cdot 2^7 =$$

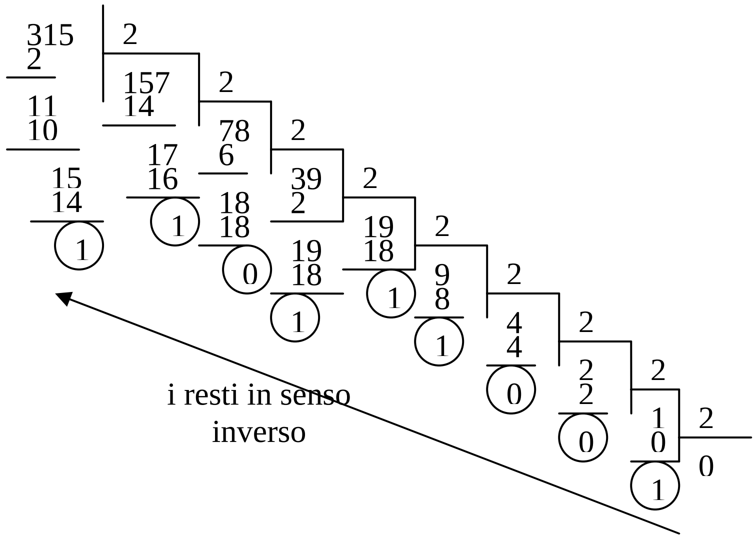
$$1\cdot 1 + 0\cdot 2 + 0\cdot 4 + 1\cdot 8 + 0\cdot 16 + 1\cdot 32 + 1\cdot 64 + 1\cdot 128 = 1 + 8 + 32 + 64 + 128 = 233_{(10)}$$

Si può utilizzare anche quest'altro metodo

$$1101001_{(2)} = \left(\left(\left(\left(\left(\left(1\cdot 2+1\right)\cdot 2+1\right)\cdot 2+0\right)\cdot 2+1\right)\cdot 2+0\right)\cdot 2+0\right)\cdot 2+1$$

Cambiamento di base da 10 a 2. Per trasformare un numero da base 10 a base 2 si eseguono le divisioni intere successive del numero dato per 2, e dei quozienti mano a mano ottenuti sempre dividendo per 2, fino ad avere come quoziente 0: il numero scritto nella base 2 si ottiene riportando i resti delle divisioni presi in senso inverso.

Esempio. Convertire in base 2 il numero 315.



$315_{(10)} = 100111011_{(2)}$

Addizione nel sistema binario. Si può applicare lo stesso algoritmo dell'addizione nel sistema decimale, occorre tenere presente che: 0+0=0; 0+1=1; 1+0=1; 1+1=10.

Esempio. $1100_{(2)} + 110_{(2)}$

		1	1	0	0	+	
			1	1	0	=	
	1	0	0	1	0		

Quindi $1100_{(2)} + 110_{(2)} = 10010_{(2)}$

Sottrazione nel sistema binario. Si può applicare lo stesso algoritmo della sottrazione nel sistema decimale, occorre tenere presente che 0-0=0; 1-0=1; 1-1=0; per 0-1 occorre 'prendere in prestito' una unità della cifra di ordine superiore, quindi 10-1=1.

Esempio. $11110_{(2)} - 1001_{(2)}$

					1		
	1	1	1	1	0	+	
		1	0	0	1	=	
	1	0	1	0	1		

Quindi $11110_{(2)} - 1001_{(2)} = 10101_{(2)}$

Moltiplicazione nel sistema binario. Si può applicare lo stesso algoritmo che si usa per il sistema decimale, tenendo presente che 0x0=0; 1x0=0; 0x1=0; 1x1=1.

Esempio. $111_{(2)} \times 101_{(2)}$

				1	1	1	x	
				1	0	1	=	
				1	1	1		
			0	0	0			
		1	1	1				
	1	0	0	0	1	1		

Quindi $111_{(2)} \times 101_{(2)} = 100011_{(2)}$.

Divisione nel sistema binario. Si può eseguire in modo analogo alla divisione nel sistema decimale.

Esempio. $11010_{(2)}:101_{(2)}$

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

Quindi $11010_{(2)}:101_{(2)}=11_{(2)}$ con resto $1_{(2)}$

10.3 Il sistema binario nell'informatica

Il sistema binario viene anche utilizzato per rappresentare l'informazione digitale:

- il **bit** (binary digit) è l'unità di informazione e corrisponde a 1 cifra del sistema binario;
- 1 **byte** = 8 bit, ossia 8 cifre;
- 1 KB (**kilobyte**) = 2^{10} B=1.024 byte solitamente approssimato con 1000
- 1 MB (**megabyte**) = 2^{20} KB = 1.048.576 byte, approssimato con 1.000.000
- 1GB (**gigabyte**) = 2^{30} MB = 1.073.741.824 byte, approssimato con 1.000.000.000
- 1 TB (**terabyte**) = 2^{40} GB = 1.099.511.627.776 byte, approssimato con 1.000.000.000.000

Nel campo informatico, per motivi esclusivamente pratici, si preferisce rappresentare sia i numeri negativi, sia i numeri decimali utilizzando esclusivamente 0 e 1, senza cioè usare i simboli +, - né la virgola per i numeri decimali.

Rappresentazione dei numeri negativi. Per rappresentare i numeri negativi usando esclusivamente 0 e 1 si utilizzano due convenzioni, entrambe si basano sul fatto che si utilizzano un numero limitato di cifre, sempre uguale, per rappresentare i numeri.

Il metodo della **rappresentazione con modulo e segno** consiste nell'utilizzare la prima cifra per rappresentare il segno, 0 per rappresentare un numero positivo e 1 per rappresentare un numero negativo.

Esempio. Utilizzando 8 bit si ha:

$$-12_{(10)} = 10001100_{(2)} \qquad +12_{(10)} = 00001100_{(2)}$$

Questo metodo ha come svantaggio il fatto che esistono due modi per rappresentare il numero 0, precisamente 00000000 e 10000000.

Complemento a 2. E' il sistema più diffuso per rappresentare i numeri negativi in informatica. Anche in questa rappresentazione il primo bit è 0 per i numeri positivi e 1 per i numeri negativi; nel caso di numero negativo gli altri numeri si ottengono invertendo 0 con 1.

Esempio. Utilizzando 8 bit si ha:

$$+12_{(10)}=00001100_{(2)} \qquad -12_{(10)}=11110011_{(2)}$$

Con questo metodo il numero 0 ha un'unica rappresentazione.

Rappresentazione dei numeri decimali. Per rappresentare i numeri decimali nel sistema binario senza usare il simbolo della virgola ma solo i simboli 0 e 1, si usano due metodi.

La **rappresentazione a virgola fissa** si ottiene fissando il numero di bit da usare per la parte intera e quelli da utilizzare per la parte decimale. Per esempio, utilizzando 16 bit si usano i primi 8 bit per rappresentare la parte intera e gli ultimi 8 bit per rappresentare la parte decimale.

Nella **rappresentazione a virgola mobile** si fa uso della rappresentazione scientifica dei numeri. In notazione scientifica un numero a si rappresenta come $a = M \times b^E$, dove M è la mantissa, b la base del sistema di numerazione, E l'esponente.

Per esempio, se si rappresentano i numeri a precisione singola con 32 bit, si utilizza il primo bit per il segno, 8 bit per l'esponente e 23 bit per la mantissa.

Esempio. Per rappresentare il numero -118,625

- Il segno è negativo, quindi il primo bit è 1;
- $118{,}625_{(10)} = 1110110{,}101_{(2)}$ scrivendolo in notazione scientifica diventa $1{,}110110101 \times 2^6$, la mantissa, portata a 23 bit, è 11011010100000000000000;
- All’esponente 6 occorre aggiungere 127 (per i numeri a precisione singola) =133 che nel sistema binario è 10000101.

In definitiva il numero è

$$-118{,}625_{(10)} = \underset{\text{segno}}{1}\ \underbrace{10000101}_{\text{esponente}}\ \underbrace{11011010100000000000000}_{\text{mantissa}}{}_{(2)}$$

10.3 Sistema esadecimale

Nel sistema esadecimale, abbreviato con **esa** o **hex**, si utilizzano i seguenti 16 simboli:

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

Conversione da sistema esadecimale a decimale. Si scrive il numero in forma polinomiale e si eseguono le operazioni, il numero ottenuto è in forma decimale.

Esempio.

$$23A1F_{(16)} = F \cdot 16^0 + 1 \cdot 16^1 + A \cdot 16^2 + 3 \cdot 16^3 + 2 \cdot 16^4 =$$

$$15 \cdot 1 + 1 \cdot 16 + 10 \cdot 256 + 3 \cdot 4096 + 2 \cdot 65536 = 145951_{(10)}$$

Conversione da decimale a esadecimale. Si divide il numero per 16 ripetutamente fino a ottenere quoziente 0, quindi si riportano i resti in ordine inverso.

Esempio. Convertire 1204 nel sistema esadecimale

- 1204 diviso 16 = 75 con resto 4 $=4_{(16)}$;
- 75 diviso 16 = 4 con resto $11=B_{(16)}$;
- 4 diviso 16 =0 con resto 4, poiché il quoziente è 0 si interrompe la divisione;

$$1204_{(10)} = 4B4_{(16)}$$

10.4 Sistema di numerazione romano

L’antico sistema di numerazione romano è usato occasionalmente ancora oggi (per indicare numeri ordinali, secoli, a volte le pagine dell’introduzione di un libro,…). Questo sistema di numerazione non è posizionale ma additivo.

Si utilizzano i seguenti simboli di base:

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

E i seguenti gruppi di base:

IV	IX	XL	XC	CD	CM
4	9	40	90	400	900

Tutti i numeri si scrivono utilizzando i simboli di base, ripetuti al massimo tre volte, secondo una logica additiva (II=2, XXX=30, …) e i gruppi di base, che utilizzano una logica sottrattiva.

I simboli I, X, C, M possono essere ripetuti fino a un massimo di tre volte.

I simboli V, L, D non possono essere ripetuti.

I gruppi di base sono gli unici in cui si utilizza la sottrazione.

Non è corretto per esempio scrivere IL per indicare 49, occorre scrivere XLIX.

Non è corretto scrivere IC per indicare 99, occorre scrivere XCIX.

Per i numeri che superano MMM, cioè 3000, si pone una lineetta sul simbolo di base per indicare che va moltiplicato per 1000. $\overline{V} = 5000$, $\overline{C} = 100000$.