

Elementi di Informatica

(Lezione III , parte I)

Rappresentazione delle informazioni :

La codifica dei numeri

Rappresentazione dei numeri

- All'interno dei computer, a causa dei vincoli tecnologici, per rappresentare qualsiasi tipo di numero, si *utilizzano sempre un numero fisso di cifre binarie.*
- Su tutti i computer si utilizzano:
 - 16 bit (2 byte)**
 - 32 bit (4 byte)**
- In alcuni casi si può arrivare a **64 bit (8 byte)** o più a seconda del tipo di processore.

Rappresentazione dei numeri (cont.)

- Tutti i numeri vengono distinti in tre categorie:
 - Interi senza segno (interi positivi).
 - Interi con segno (interi positivi e negativi).
 - Reali (numeri positivi e negativi con virgola).

- Ogni categoria viene rappresentata in modo differente.

Rappresentazione con numero fisso di cifre (1)

- Per comprendere il meccanismo alla base della rappresentazione con un numero fisso di cifre partiamo da un esempio:
 - **Qual è il numero più grande rappresentabile con 4 cifre?**

In base 10: **9999**

In base 2: **1111 = 15₁₀**

In base 16: **FFFF = 65535₁₀**

In base 8: **7777 = 4095₁₀**

Rappresentazione con numero fisso di cifre (2)

➤ In generale si avrà, con n cifre:

$$\boxed{b^n - 1}$$

In base 10: **9999 = 10⁴ - 1**

In base 2: **1111 = 2⁴ - 1**

In base 16: **FFFF = 16⁴ - 1**

In base 8: **7777 = 8⁴ - 1**

Rappresentazione con numero fisso di cifre (3)

Quindi vale la seguente *regola* nel caso di numeri interi positivi:

Nella base di numerazione b disponendo di n cifre si possono rappresentare soltanto i numeri

$$\text{da } 0 \text{ a } b^n - 1$$

Rappresentazione dei numeri interi senza segno

➤ Per calcolare il valore massimo ammesso occorre applicare la regola $2^n - 1$ (dove n vale 16 o 32)

- Nella rappresentazione a 16 bit i possibili valori saranno compresi tra

0 e 65.535

- Nella rappresentazione a 32 bit i possibili valori saranno compresi tra

0 e 4.294.967.295

Numeri interi con segno (1)

➤ Per rappresentare i numeri con il loro segno (interi positivi e negativi) esistono due possibili modi. Il primo è il seguente:

- Dati n bit, un bit si riserva al segno e gli altri $n - 1$ sono destinati al numero
- Ad esempio, considerando 8 bit e ponendo il primo bit a sinistra 0 per il + e 1 per il - avremo:

$$0000\ 0101_2 = + 5_{10}$$

$$1000\ 0101_2 = - 5_{10}$$

Numeri interi con segno (2)

➤ Questo tipo di rappresentazione prende il nome di:

Rappresentazione in modulo e segno

➤ Anche se semplice, possiede però un grosso difetto: esistono due zeri.

1	0	0	0	0	0	1	1	=	-3
1	0	0	0	0	0	1	0	=	-2
1	0	0	0	0	0	0	1	=	-1
1	0	0	0	0	0	0	0	=	-0
0	0	0	0	0	0	0	0	=	+0
0	0	0	0	0	0	0	1	=	+1
0	0	0	0	0	0	1	0	=	+2
0	0	0	0	0	0	1	1	=	+3

Numeri interi con segno (3)

➤ Utilizzando n bit e riservandone uno al segno, l'applicazione della formula precedente porterà:

$$\mathbf{da} \quad -(2^{n-1} - 1) \quad \mathbf{a} \quad 2^{n-1} - 1$$

(dove n vale al solito 16 o 32).

Numeri interi con segno (4)

- Seguendo il ragionamento precedente, i possibili valori nel caso di 16 bit saranno quindi compresi tra

$$\mathbf{-32.767 \text{ e } +32.767}$$

- Nel caso dei 32 bit si avrà:

$$\mathbf{-2.147.483.647 \text{ e } +2.147.483.647}$$

Numeri interi con segno (5)

- Il secondo modo per rappresentare i numeri con il loro segno (interi positivi e negativi) è quello del *complemento a due*:

- Con n bit si possono rappresentare i numeri da

$$\mathbf{-2^{n-1} \text{ a } 2^{n-1} - 1}$$

- Dato un numero composto da n bit, la rappresentazione in complemento a due si ottiene invertendo gli 1 in 0 e gli 0 in 1, e poi sommando 1 al risultato ottenuto.

Numeri interi con segno (6)

$$n = 3$$

$$1\ 0\ 0 = -4$$

$$1\ 0\ 1 = -3$$

$$1\ 1\ 0 = -2$$

$$1\ 1\ 1 = -1$$

$$0\ 0\ 0 = 0$$

$$0\ 0\ 1 = +1$$

$$0\ 1\ 0 = +2$$

$$0\ 1\ 1 = +3$$

- È presente un solo zero!
- Tutti i numeri positivi cominciano con 0 mentre quelli negativi cominciano con 1.
- Complementando si passa dal valore positivo allo stesso valore negativo e viceversa (ad esclusione del -4).
 - Codificare -3 in complemento a due:
 - si considera +3 in binario,
 - si invertono le cifre : 100,
 - si somma 1, ottenendo 101;
 - e viceversa da 101 si ottiene 011.

L'overflow

- Per questioni tecnologiche tutti i computer, senza alcuna eccezione, trattano i numeri sempre con un numero fisso di cifre binarie (ad esempio 16, 32 o più).
- Quando l'elaboratore esegue un'operazione il cui risultato eccede il numero di cifre permesso, la computazione si arresta immediatamente e viene segnalato l'errore di **OVERFLOW**.

L'overflow (cont.)

- Ad esempio se la rappresentazione è a 32 bit senza segno e si vuole eseguire la seguente operazione:

$$\begin{array}{r} 3.000.000.000 + \\ \underline{2.000.000.000} = \end{array}$$

***** Errore di **OVERFLOW**

La computazione si arresta immediatamente.

Rappresentazione dei numeri con virgola

- I numeri con la virgola vengono rappresentati mediante la *notazione scientifica* o in virgola mobile.
- Anche i numeri interi possono sempre essere espressi come numeri con virgola attraverso la notazione scientifica.

Rappresentazione dei numeri con virgola (cont.)

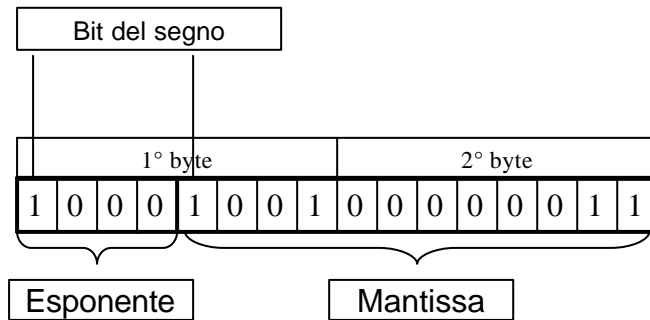
Numero	Notazione Scientifica	Parte decimale (mantissa)	Exp
250	$0,25 \times 10^3$	25	3
-83,76	$-0,8376 \times 10^2$	-8376	2
0,05	$0,5 \times 10^{-1}$	5	-1
55.640.350	$0,5564035 \times 10^8$	5564035	8

Rappresentazione dei numeri con virgola (cont.)

Parte decimale (mantissa)	Exp
25	3
-8376	2
5	-1
5564035	8

Con la notazione scientifica ogni numero viene memorizzato solo come mantissa (parte decimale senza lo 0 e la virgola) e con l'esponente (senza la base): si risparmia spazio.

Rappresentazione dei numeri con virgola (cont.)



Fine

Elementi di Informatica

(Lezione III , parte II)

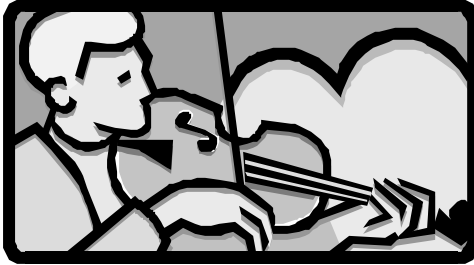
Rappresentazione delle informazioni :

Codifica di dati multimediali

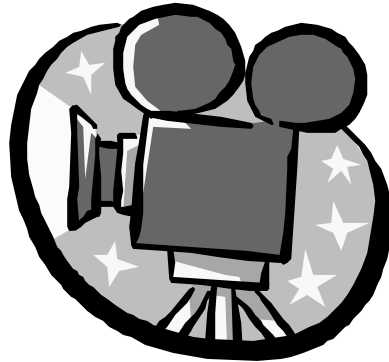
Ma il mondo non è tutto “scritto” !

- I caratteri alfanumerici non costituiscono le uniche informazioni utilizzate dagli elaboratori.
 - Le applicazioni *multimediali* utilizzano ed elaborano informazioni contenenti:
 - immagini,
 - suoni,
 - filmati.

La codifica dei dati multimediali



L'informazione per gli umani ha un carattere analogico....



In termini tecnici si tratta di “SEGNALI”

➤ Un segnale analogico si può:

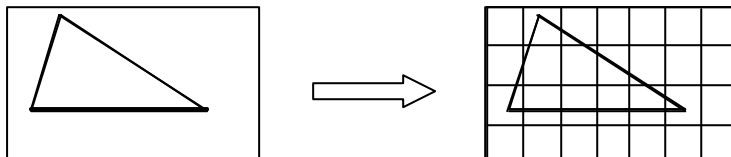
- CAMPIONARE
 - Per un suono: misurare l'intensità ogni centesimo di secondo.
 - Per una immagine : misurare i colori ogni millimetro quadrato.
- RAPPRESENTARE con un numero.

La codifica delle immagini (1)

- Esistono numerose tecniche che vengono utilizzate per la memorizzazione e l'elaborazione di un'immagine.
- Consideriamo un'immagine in bianco e nero, senza ombreggiature o livelli di chiaroscuro.

La codifica delle immagini (2)

L'immagine viene suddivisa mediante una griglia formata da righe orizzontali e verticali a distanza costante.

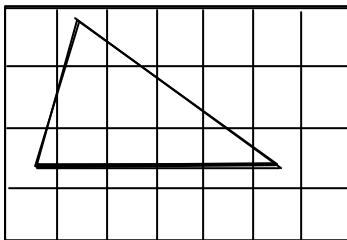


La codifica delle immagini (3)

- Ogni quadrato prende il nome di **pixel (picture element)** e viene codificato in binario secondo la seguente convenzione:
- Il simbolo 0 viene utilizzato per la codifica di un pixel corrispondente ad un quadrato in cui il bianco è predominante.
 - Il simbolo 1 viene utilizzato per la codifica di un pixel corrispondente ad un quadrato in cui il nero è predominante.

La codifica delle immagini (4)

- Per convenzione la griglia dei pixel è ordinata dal basso verso l'alto e da sinistra verso destra.



0 ₂₂	1 ₂₃	0 ₂₄	0 ₂₅	0 ₂₆	0 ₂₇	0 ₂₈
0 ₁₅	1 ₁₆	1 ₁₇	0 ₁₈	0 ₁₉	0 ₂₀	0 ₂₁
0 ₈	1 ₉	1 ₁₀	1 ₁₁	1 ₁₂	0 ₁₃	0 ₁₄
0 ₁	0 ₂	0 ₃	0 ₄	0 ₅	0 ₆	0 ₇

La figura sarà rappresentata dalla stringa binaria:

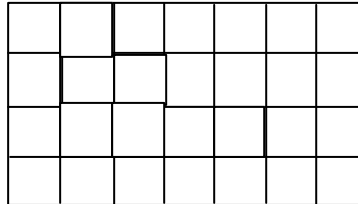
000000 0111100 0110000 0100000

La codifica delle immagini (5)

➤ Dato che il contorno della figura non sempre coincide con la griglia si ottiene un'approssimazione della figura originaria.

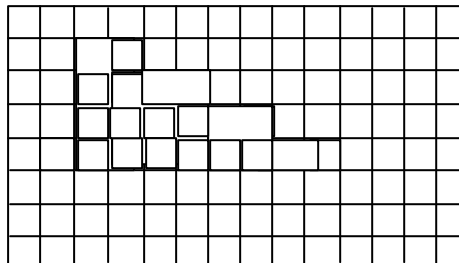
- Riconvertendo la stringa:
– 0000000 0111100 0110000 0100000

si avrà:



La codifica delle immagini (6)

➤ La rappresentazione sarà più fedele all'aumentare del numero di pixel.



➤ La *dimensione* dell'immagine è espressa come numero di pixel nel formato *righe x colonne*.

Gradazioni di grigio

- Per codificare immagini con diversi livelli di grigio si usa una rappresentazione binaria: ad ogni livello di grigio corrisponde una sequenza di bit.
 - Ad esempio, utilizzando quattro bit si possono rappresentare $2^4 = 16$ livelli di grigio, mentre con otto bit ne possiamo distinguere $2^8 = 256$.

Esempi di toni di grigio



1 bit
2 toni



2 bit
4 toni



8 bit
256 toni

L'uso del colore

- Il colore viene generato dalla composizione di tre *colori primari*: Red, Green, Blue (video RGB)
- Ad ogni colore primario si associa una possibile sfumatura o *gradazione* mediante un'opportuna sequenza di bit.
 - Utilizzando 2 bit per ogni colore primario si possono ottenere 4 gradazioni per ognuno di essi, cioè 64 colori.
 - Un pixel richiede quindi un byte circa (6 bit) in questa maniera.

L'uso del colore (cont.)

- Utilizzando 4 bit per ogni colore primario si ottengono 16 gradazioni per ognuno di essi, cioè 4096 colori differenti.
 - Un pixel richiede quindi circa due byte (12 bit) di informazione.
- Utilizzando 8 bit per ogni colore primario si ottengono 256 gradazioni per ognuno di essi, cioè 16,8 milioni di colori circa.
 - Un pixel richiede quindi tre byte di informazione.

Risoluzione

- Il numero di pixel per unità di area indica la “risoluzione” con cui si è campionata l’immagine.
- Il numero di pixel presenti sul video (*colonne x righe*) prende il nome di risoluzione dello schermo.
 - Risoluzioni tipiche sono: 640x480 800x600 1024x768
 - La dimensione dell’immagine sarà:

	16 bit	32 bit
640x480	600 KB	~ 1,2 MB
800x600	~ 938 KB	~ 1,9 MB
1024x768	~ 1,6 MB	3MB

Formati delle immagini

- Codifica *raster* o *bitmap*.
 - Ogni punto rappresenta un’informazione indipendente.
 - GIF,
 - JPEG,
 - BMP.
- Codifiche *ibride (raster/vettoriale)*.
 - Ogni elemento geometrico primitivo viene specificato individualmente.
 - Postscript,
 - PDF (Portable Document Format).

Codifica di filmati video

- Un filmato è una sequenza di immagini statiche (dette fotogrammi o *frames*).
 - Codifica normale (dei singoli frames).
 - 3 minuti di video con 24 frames/sec,
 - ❖ minimo 16 frame/s per non percepire i singoli fotogrammi.
 - Risoluzione singolo frame : 200x100, 16 bit/pixel.
 - Memoria necessaria: $(3*60*24) (200*100*2) \sim 165$ MB.
 - Codifica differenziale.
 - È inefficiente codificare completamente ogni frame.
 - ❖ Alcuni frames si codificano interamente, altri solo nelle parti che differiscono da quelli adiacenti.

Formati video

- MPEG (Moving Picture Experts Group)
 - Costituisce uno standard.
 - Molto efficiente ma complesso.
- QuickTime
 - Proposto da Apple.
 - Meno efficiente ma più semplice.
- Indeo - AVI
 - Proposto da Intel, usato da MicroSoft.
 - È il più inefficiente.

Codifica di suoni

➤ Il segnale acustico viene digitalizzato.

- Dimensioni medie:

- un minuto di audio con qualità CD musicale stereo occupa da 1MB a 10MB a seconda della codifica impiegata.

➤ Codifiche standard:

- WAV (MS-Windows),
- MIDI
- MP3

Formati sonori

➤ MIDI:

- Codifica le note e gli strumenti che devono eseguirle.
- Solo musica, non voce.
- Richiede un sintetizzatore o “campioni” per la riproduzione (non utilizzabile “direttamente”).
- Molto efficiente.

➤ MP3:

- MPEG - layer 3: variante MPEG per i suoni.
- Grande diffusione.
- Molto efficiente.

Ma quanto spazio ci vuole?

- I bit costano!
- Trasmettere i bit costa!
- Gli uomini però hanno linguaggi “ridondanti”
 - Esempio : Dmn è mrcl d e nn c sn lzn d prgrmmzn
 - È un messaggio poco chiaro ma decodificabile...
E occupa meno caratteri (e meno bit) di “Domani è mercoledì e non ci sono lezioni di programmazione”.
- IDEA: compressione!!!!

Fine