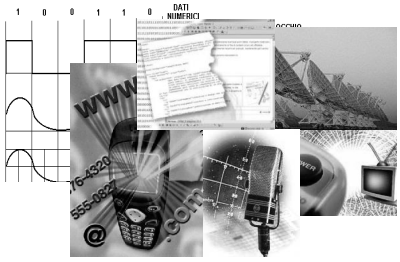
La codifica dell'informazione

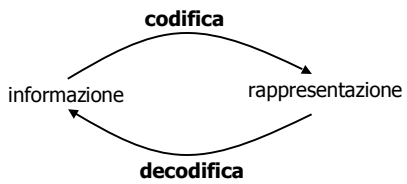
Tipi di informazione

- Esistono vari tipi di informazione, di natura e forma diversa, così come rappresentazioni diverse della stessa informazione



Problema

- Come possiamo rappresentare le informazioni all'interno di un sistema di calcolo?
- Si introduce il concetto di **codifica**



Codifica (o Codice)

- **X**: insieme degli oggetti che si vogliono rappresentare
- **A**: alfabeto di simboli
- **A***: insieme di tutte le possibili sequenze, finite e infinite, costruite su A

- **cod**: X ⟶ A* **decod**: A* ⟶ X U {errore}
- ∀ y∈X : decod(cod(y)) = y

- L'insieme **X**, l'alfabeto **A**, le funzioni **cod** e **decod** formano un **Codice** per la rappresentazione degli elementi di X

Codifica (o Codice)

- Nel caso dei sistemi di calcolo è stata introdotta la rappresentazione **digitale**
  - **bit** (**b**inary dig**it** - cifra binaria): **0** o **1**
- Per poter rappresentare un numero maggiore di informazioni si usano **sequenze** di bit

00 01 10 11

Rappresentazione digitale

- Il processo secondo cui si fa corrispondere ad una informazione una configurazione di bit prende il nome di **codifica dell'informazione**
- ***Esempio:*** *un esame può avere quattro possibili esiti*
  - *cod(ottimo)* → *00*
  - *cod(discreto)* → *01*
  - *cod(sufficiente)* → *10*
  - *cod(insufficiente)* → *11*

Codifica dell'informazione

- Con 2 bit si codificano 4 informazioni (2^2)
- Con 3 bit si codificano 8 informazioni (2^3)
- ……
- Con N bit si codificano **2^N** informazioni

Codifica dell'informazione

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

……..

1	1	1	1	1	1	1	1

Codifica dell'informazione

- Problema inverso: quanti bit ci vogliono per rappresentare **M** informazioni diverse?

$$2^N >= M$$

- ***Esempio:*** *dovendo rappresentare 1.000 informazioni diverse dobbiamo avere a disposizione* ***N=10*** *bit per la codifica*

$$2^{10} = 1024$$

NB*. "avanzano" delle configurazioni ma non è possibile usare solo 9 bit per 1000 informazioni*

Codifica dei caratteri

- Alfabeto anglosassone → per codificare ogni simbolo sono sufficienti **7** bit (ASCII standard)
- **8** bit (ASCII esteso)
- **16** bit (UNICODE)
- MS Windows usa un codice proprietario a16 bit per carattere, simile ad UNICODE

ASCII = American Standard Code for Information Interchange

Codifica dei caratteri (ASCII)

ASCII	Simb.	ASCII	Simb.	ASCII	Simb.
00000000	NUL	00001110	SO	00011100	FS
00000001	SOH	00001111	SI	00011101	GS
00000010	STX	00010000	DLE	00011110	RS
00000011	ETX	00010001	DC1	00011111	US
00000100	EOT	00010010	DC2	00100000	SP
00000101	ENQ	00010011	DC3	00100001	!
00000110	ACK	00010011	DC4	00100010	"
00000111	BEL	00010101	NAK	00100011	#
00001000	BS	00010110	SYN	00100100	$
00001001	HT	00010111	ETB	00100101	%
00001010	NL	00011000	CAN	00100110	&
00001011	VT	00011001	EM	00100111	'
00001100	NP	00011010	SUB	00101000	(
00001101	CR	00011011	ESC	00101001	)

Codifica dei caratteri (ASCII)

ASCII	Simb.	ASCII	Simb.	ASCII	Simb.
00101010	*	00111001	9	01000111	G
00101011	+	00111010	:	01001000	H
00101100	,	00111011	;	01001001	I
00101101	-	00111100	<	01001010	J
00101110	.	00111101	=	01001011	K
00101111	/	00111110	>	01001100	L
00110000	**0**	00111111	?	01001101	M
00110001	**1**	01000000	@	01001110	N
00110010	**2**	**01000001**	**A**	01001111	O
00110011	3	**01000010**	**B**	01010000	P
00110100	4	**01000011**	**C**	01010001	Q
00110101	5	01000100	D	01010010	R
00110110	6	01000101	E	01010011	S
00111000	8	01000110	F	01010100	T

Codifica delle parole

- … e le parole? Sono sequenze di caratteri

- *Esempio: informatica generale*

01101001 01101110 01100110 01101111 01110010 01101101 01100001 01110100 01101001 →
i n f o r m a t i

01100011 01100001 00000000 01100111 01100101 01101110 01100101 01110010 01100001 →
c a g e n e r a

01101100 01100101
l e

Codifica dell'informazione: verifica

1. *Nell'alfabeto di Marte sono previsti 300 simboli; quanti bit si devono utilizzare per rappresentarli tutti?*
2. *Quanti byte occupa la frase "**biologia marina**" se la si codifica utilizzando il codice ASCII esteso?*
3. *Quanti byte occupa la stessa frase scritta in codice UNICODE?*
4. *Dati 12 bit per la codifica, quante informazioni distinte si possono rappresentare?*

Codifica dell'informazione: soluzione

1. *Nell'alfabeto di Marte sono previsti 300 simboli; quanti bit si devono utilizzare per rappresentarli tutti?*

L'esercizio richiede di trovare il numero di bit che sono necessari per codificare **300 informazioni diverse**.

Dobbiamo quindi applicare la formula $\mathbf{2^N >= M}$ e ricavare N

$2^N >= 300$ se N=9

Codifica dell'informazione: soluzione

2. *Quanti byte occupa la frase "**biologia marina**" se la si codifica utilizzando il codice ASCII esteso?*

   Poichè sappiamo che ogni carattere in codice ASCII esteso occupa **un byte** dobbiamo **contare il numero di caratteri** (inclusi gli spazi bianchi) che formano la frase "biologia marina" e moltiplicare per 1

   15 caratteri → 15 byte

3. *Quanti byte occupa la stessa frase scritta in codice UNICODE?*

   Poichè ogni carattere in codice UNICODE occupa **due byte** avremo

   15 caratteri → 15 x 2 byte = 30 byte

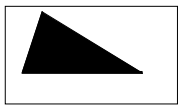
Codifica dell'informazione: soluzione

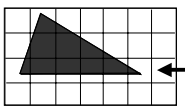
4. *Dati 12 bit per la codifica, quante informazioni distinte si possono rappresentare?*

   In questo caso conosciamo la **lunghezza delle sequenze** di bit che sono usate per la codifica dell'informazione e basterà applicare la formula 2^N per trovare il numero di informazioni distinte che si possono rappresentare

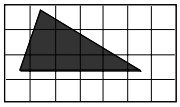
   $2^{12} = 4096$

Codifica delle immagini





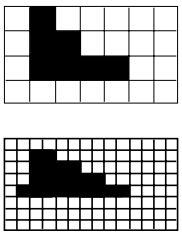
Codifica delle immagini



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

Codifica delle immagini

00000000111100011000001000000



Codifica delle immagini

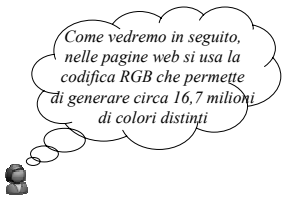
- Assegnando un bit ad ogni pixel è possibile codificare solo immagini in bianco e nero
- Per codificare le immagini con diversi livelli di grigio oppure a colori si usa la stessa tecnica: per ogni pixel viene assegnata una sequenza di bit
- Per memorizzare un pixel non è più sufficiente un solo bit. Ad esempio, se utilizziamo 4 bit possiamo rappresentare 2^4=16 livelli di grigio o 16 colori diversi, mentre con 8 bit ne possiamo distinguere 2^8=256, ecc.

L'uso del colore

- Il colore può essere generato componendo 3 colori: Red, Green, Blue (RGB)
- Ad ogni colore si associa una possibile sfumatura
- Usando 2 bit per ogni colore si possono ottenere 4 sfumature per il rosso, 4 per il blu e 4 per il verde che, combinate insieme, danno origine a 64 colori diversi
- Ogni pixel per essere memorizzato richiede 6 bit

L'uso del colore

- Usando 8 bit per ogni colore si possono ottenere 256 sfumature per il rosso, 256 per il blu e 256 per il verde che, combinate insieme, danno origine a circa 16,7 milioni di colori diversi (precisamente 16777216 colori)
- Ogni pixel per essere memorizzato richiede 3 byte



Colori e risoluzione

- Il numero di pixel presenti sullo schermo (colonne x righe) prende il nome di **risoluzione**
- Risoluzioni tipiche sono

800 x 600
1024 x 768
1280 x 1024

Codifica delle immagini: verifica

1. *Quanti byte occupa un'immagine di 100x100 pixel in bianco e nero?*
2. *Quanti byte occupa un'immagine di 100x100 pixel a 256 colori?*
3. *Se un'immagine a 16,7 milioni di colori occupa 2400 byte, da quanti pixel sarà composta?*

Codifica delle immagini: soluzione

1. *Quanti byte occupa un'immagine di 100x100 pixel in bianco e nero?*

Conoscendo la risoluzione dell'immagine possiamo trovare il **numero di pixel** che la compongono: 100x100 = 10000 pixel.

Inoltre, nel caso di immagini in bianco e nero basta **un solo bit** per codificare il colore di ogni pixel e quindi saranno necessari 10000 bit per memorizzare l'immagine.

Per trovare il numero di byte basta fare 10000 / 8 = 1250 byte

Codifica delle immagini: soluzione

2. *Quanti byte occupa un'immagine di 100x100 pixel a 256 colori?*

Rispetto all'esercizio precedente, in questo cambia lo spazio occupato da ciascun pixel. Sappiamo che l'immagine è a 256 colori. Per poter rappresentare 256 configurazioni diverse sono necessari **8 bit**, ovvero **1 byte**

L'immagine occuperà quindi 10000 x 1 byte = 10000 byte

Codifica delle immagini: soluzione

3. *Se un'immagine a 16,7 milioni di colori occupa 2400 byte, da quanti pixel sarà composta?*

In questo caso le informazioni fornite dall'esercizio sono il numero colori e lo spazio occupato dall'immagine.

Dal numero di colori ricaviamo lo spazio occupato da ciascun pixel, calcolando il valore **N** nell'espressione 2^N**>=16,7 milioni**.
Il risultato è 24 bit, ovvero 3 byte.

Se ogni pixel richiede 3 byte e l'immagine occupa 2400 byte, sarà composta da 2400 / 3 = 800 pixel.

Unità di misura

Di solito si usano i multipli del byte

Kilo	KB	2^{10} (~ un migliaio, 1024)
Mega	MB	2^{20} (~ un milione, 1KBx1024)
Giga	GB	2^{30} (~ un miliardo, 1MBx1024)
Tera	TB	2^{40} (~ mille miliardi, 1GBx1024)

Grafica bitmap

- Le immagini codificate pixel per pixel sono dette immagini in grafica bitmap
- La grafica bitmap va bene per immagini complesse o irregolari. I formati più conosciuti sono: BITMAP (.bmp), GIF (.gif), JPEG (.jpg)

Nelle pagine web si usano principalmente le immagini in formato GIF o JPEG (recentemente anche PNG)

GIF (Graphics Interchange Format) *JPEG (Joint Photographic Expert Group)*

Grafica bitmap

- Le immagini bitmap occupano parecchio spazio
- Esistono delle tecniche di compressione che permettono di ridurne le dimensioni
- Ad esempio, se più punti vicini di un'immagine assumono lo stesso colore, si può memorizzare la codifica del colore una sola volta e poi ricordare per quante volte deve essere ripetuta
- GIF e JPEG sono formati compressi

Grafica vettoriale

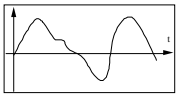
- Se le immagini sono regolari si può usare una codifica di tipo **vettoriale** in cui non si specificano le informazioni di colore dei singoli pixel ma ogni elemento geometrico primitivo viene specificato individualmente
- Le immagini vengono costruite a partire dalla descrizione degli elementi che le compongono mediante un linguaggio testuale o delle formule geometriche
- Spesso occupano meno spazio rispetto alle immagini bitmap

Codifica dei filmati video

- Un filmato è una sequenza di immagini statiche (dette **fotogrammi** o **frame**)
- Per codificare un filmato si digitalizzano i suoi fotogrammi
- Esempi di formati per il video sono AVI (Audio Video Interleave, Microsoft) e MOV (noto come QuickTime, Apple)
- Compressione: MPEG (Moving Picture Expert Group), differenza tra fotogrammi

Codifica dei suoni

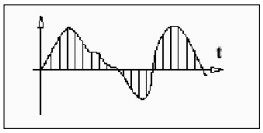
- Fisicamente un suono è rappresentato come un'onda che descrive la variazione della pressione dell'aria nel tempo (onda sonora)



- Sull'asse delle ascisse viene posto il tempo t e sull'asse delle ordinate la variazione della pressione corrispondente

Codifica dei suoni

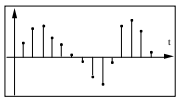
- Si effettuano dei campionamenti sull'onda (cioè si misura il valore dell'onda ad intervalli di tempo costanti) e si codificano in forma digitale le informazioni estratte da tali campionamenti



- La sequenza dei valori numerici ottenuta dai campioni può essere facilmente codificata

Codifica dei suoni

- Quanto più frequentemente il valore dell'onda viene campionato, tanto più precisa sarà la sua rappresentazione



- Il numero di campioni raccolti per ogni secondo definisce la frequenza di campionamento che si misura in Hertz (Hz)

Codifica dei suoni

- Esempi di formati sono
  - WAV (Microsoft)
  - AIFF (Audio Interchange File Format, Apple)
- Compressione: MP3 (estensione audio per MPEG)

Codifica dei suoni: verifica

1. *Quanto spazio occupa un suono della duranta di 10 secondi campionato a 100 Hz, in cui ogni campione occupa 4 byte?*
2. *Un secondo di suono campionato a 512 Hz occupa 1KB. Quanti valori distinti si possono avere per i campioni?*

Codifica dei suoni: soluzione

1. *Quanto spazio occupa un suono della duranta di 10 secondi campionato a 100 Hz, in cui ogni campione occupa 4 byte?*

La frequenza di campionamento ci dice quanti campioni di suono vengono memorizzati in un secondo, 100 in questo caso. Avendo 10 secondi di suono avremo 10 x 100 = **1000 campioni**.

Poichè ogni campione richiede 4 byte, il suono occuperà 1000 x 4 = 4000 byte (che corrispondono a circa 4 KB)

Codifica dei suoni: soluzione

2. *Un secondo di suono campionato a 512 Hz occupa 1KB. Quanti valori distinti si possono avere per i campioni?*

Poichè vengono memorizzati 512 campioni al secondo, avremo in tutto 512 campioni (stiamo considerando un solo secondo di suono).

Il file sonoro occupa 1 KB, cioè 1024 byte e quindi ogni singolo campione occuperà 1024 / 512 = 2 byte, ovvero 16 bit.

Si potranno quindi avere 2^{16} = 65536 valori distinti

Osservazione

- Per calcolare lo spazio occupato da un file di testo, da un'immagine, da un file audio, la tecnica è sempre la stessa
  - Si trova lo **spazio occupato da ogni unità elementare** costituente il file *(un carattere per il testo, un pixel per l'immagine, un campione per il file audio)*
  - Si trova il **numero di unità elementari** che costituiscono il file *(il numero di caratteri per il testo, il numero di pixel per l'immagine - sfruttando la risoluzione, il numero di campioni per il file audio - sfruttando la frequenza di campionamento)*
  - Si **moltiplicano** queste due quantità