Codifica dell'informazione

- Il calcolatore memorizza ed elabora vari tipi di informazioni
  - Numeri, testi, immagini, suoni
- Occorre rappresentare tale informazione in formato facilmente manipolabile dall'elaboratore

Rappresentazione delle informazioni

Idea di fondo

- usare presenza/assenza di carica elettrica
- usare passaggio/non passaggio di corrente/luce

Usiamo cioè una rappresentazione binaria (a due valori) dell'informazione

L'unità minimale di rappresentazione è il BIT (BInary digiT – cifra digitale): 0 o 1

Informazioni complesse

Con 1 bit rappresentiamo solo 2 diverse informazioni:

si / no - on / off - 0 / 1

Mettendo insieme più bit possiamo rappresentare più informazioni:

00 / 01 / 10 / 11

Informazioni complesse si memorizzano come sequenze di bit

Informazioni complesse

- Per codificare i nomi delle 4 stagioni bastano 2 bit

- Ad esempio:
  - 0 0 per rappresentare Inverno
  - 0 1 per rappresentare Primavera
  - 1 0 per rappresentare Estate
  - 1 1 per rappresentare Autunno

- Quanti bit per codificare i nomi dei giorni della settimana?

Informazioni complesse

In generale, con N bit, ognuno dei quali può assumere 2 valori, possiamo rappresentare 2^N informazioni diverse (tutte le possibili combinazioni di 0 e 1 su N posizioni)

viceversa

Per rappresentare M informazioni dobbiamo usare N bit, in modo che $2^N >= M$

Esempio

Per rappresentare 57 informazioni diverse dobbiamo usare gruppi di almeno 6 bit. Infatti:

$$2^6 = 64 > 57$$

Cioè un gruppo di 6 bit può assumere 64 configurazioni diverse:

000000 / 000001 / 000010 .../ 111110 / 111111

Il Byte

- Una sequenza di 8 bit viene chiamata *Byte*
  - 0 0 0 0 0 0 0 0
  - 0 0 0 0 0 0 0 1
  - .....

byte = 8 bit = 2^8 = 256 informazioni diverse

Usato come unità di misura per indicare

- le dimensioni della memoria
- la velocità di trasmissione
- la potenza di un elaboratore

Usando sequenze di byte (e quindi di bit) si possono rappresentare caratteri, numeri immagini, suoni.

Altre unità di misura

- KiloByte (KB), MegaByte (MB), GigaByte (GB)
- Per ragioni storiche in informatica Kilo, Mega, e Giga indicano però le *potenze di 2* che più si avvicinano alle corrispondenti potenze di 10
- Più precisamente
  - 1 KB = 1024 x 1 byte = $2^{10} \sim 10^3$ byte
  - 1 MB = 1024 x 1 KB = $2^{20} \sim 10^6$ byte
  - 1 GB = 1024 x 1 MB = $2^{30} \sim 10^9$ byte
- I multipli del byte vengono utilizzati come unità di misura per la capacità della memoria di un elaboratore

Il sistema decimale

- 10 cifre di base: 0, 1, 2, …, 9
- **Notazione posizionale:** la posizione di una cifra in un numero indica il suo **peso** in potenze di **10**. I pesi sono:
  - unità = $10^0 = 1$ (posiz. 0-esima)
  - decine = $10^1 = 10$ (posiz. 1-esima)
  - centinaia = $10^2 = 100$ (posiz. 2-esima)
  - migliaia = $10^3 = 1000$ (posiz. 3-esima)
  - … .. .. …

Esempio di numero rappresentato in notazione decimale

Il numerale 2304 in notazione decimale (o in base 10) rappresenta la quantità:

$$2304 = 2*10^3 + 3*10^2 + 0*10^1 + 4*10^0 =$$

$$2000 + 300 + 0 + 4 = 2304 \text{ (numero)}$$

Nota: numero e numerale qui coincidono, perché il sistema decimale e quello adottato come sistema di riferimento.

Il sistema binario

- 2 Cifre di base: 0 e 1.
- Notazione posizionale: la posizione di una cifra in un numero binario indica il suo peso in potenze di 2. I pesi sono:
  - $2^0 = 1$ (posiz. 0-esima)
  - $2^1 = 2$ (posiz. 1-esima)
  - $2^2 = 4$ (posiz. 2-esima)
  - $2^3=8$; $2^4=16$; $2^5=32$; $2^6=64$; $2^7=128$; $2^8=256$; $2^9=512$; $2^{10} = 1024$; $2^{11}=2048$, $2^{12}=4096$;...

Esempio di numero rappresentato in notazione binaria

Il numerale 10100101 in notazione binaria (o in base 2) rappresenta la quantità:

$$10100101$$

$$1*2^7+0*2^6+1*2^5+0*2^4+0*2^3+1*2^2+0*2^1+1*2^0$$

$$128 + 0 + 32 + 0 + 0 + 4 + 0 + 1 =$$

165 (numero)

Il numero più grande rappresentato con N cifre

- Sist. Decimale = 99...99 = $10^N - 1$
- Sist. Binario= 11..11 = $2^N - 1$
- Esempio: 11111111 (8 bit binari) = 2^8 -1 = 255. Per rappresentare il n. 256 ci vuole un bit in più: 100000000 = $1*2^8$ = 256.

Quindi...

Fissate quante cifre (bit) sono usate per rappresentare i numeri, si fissa anche il numero più grande che si può rappresentare:

- con 16 bit: $2^{16} - 1 = 65.535$
- con 32 bit: $2^{32} - 1 = 4.294.967.295$
- con 64 bit: $2^{64} - 1 =$ circa $1{,}84 * 10^{19}$

Conversione da base 2 a base 10

Basta moltiplicare ogni bit per il suo peso e sommare il tutto:

Esempio:

$$10100$$

$$1*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 0*2^0 =$$

$$16 + 4 = 20$$

la conversione e' una somma di potenze

(N.B. se il numero binario termina per 1 e' dispari altrimenti e' pari).

Conversione da base 10 a base 2

- Dividere il numero per 2 ripetutamente finche` il risultato non e` 0
- Scrivere i resti in ordine inverso.

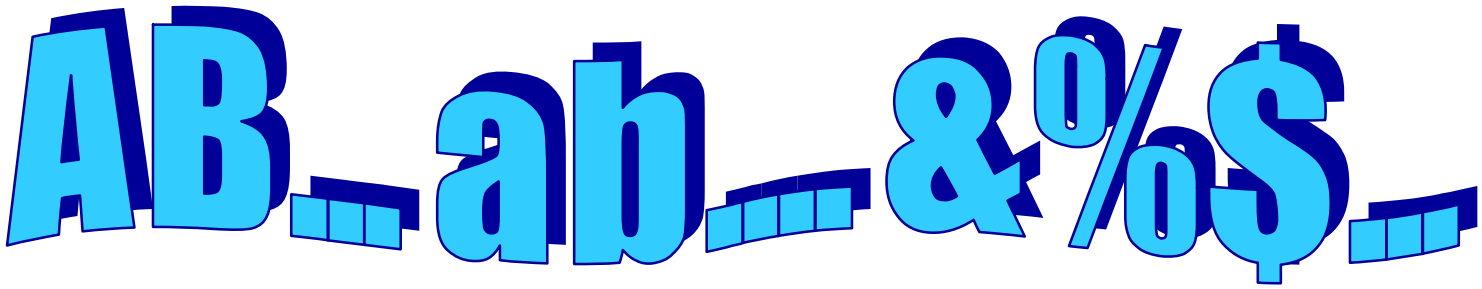
Esempio: conversione del numero **12**

Divisioni: 12/2 = 6/2 = 3/2 = 1/2 = 0

Resti: 0 0 1 1

12 = 1100

La Codifica dei Caratteri



Codici per i simboli dell'alfabeto

- **Per rappresentare i simboli dell'alfabeto anglosassone (0 1 2 … A B … A b …) bastano 7 bit**
  - **Nota: *B* e *b* sono simboli diversi**
  - **26 maiuscole + 26 minuscole + 10 cifre + 30 segni di interpunzione+… -> circa 120 oggetti**

- **Per l'alfabeto esteso con simboli quali &, %, $, … bastano 8 bit come nella codifica accettata universalmente chiamata ASCII esteso**
- **Per manipolare un numero maggiore di simboli si utilizza la codifica UNICODE a 16 bit**

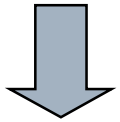
Codifica ASCII

- La codifica **ASCII** (**A**merican **S**tandard **C**ode for **I**nterchange **C**ode) utilizza codici su 7 bit
  ($2^7 = 128$ caratteri diversi)
- Ad esempio
  - 1 0 0 0 0 0 1 rappresenta A
  - 1 0 0 0 0 1 0 rappresenta B
  - 10 0 0 0 1 1 rappresenta C

- Le parole si codificano utilizzando sequenze di byte
  - 1000010 1000001 1000010 1000001
    B A B A

Numeri in ASCII

Le cifre 0..9 rappresentate in Ascii sono simboli o caratteri NON quantità numeriche



Non possiamo usarle per indicare quantità e per le operazioni aritmetiche. (Anche nella vita di tutti giorni usiamo i numeri come simboli e non come quantità: i n. telefonici)

Codifica di immagini

Codifica di immagini

- Un’immagine è un insieme continuo di informazioni
  - A differenza delle cifre e dei caratteri alfanumerici, per le immagini non esiste un'unità minima di riferimento
- Problema: rendere digitale una informazione prettamente analogica

Codifica di immagini

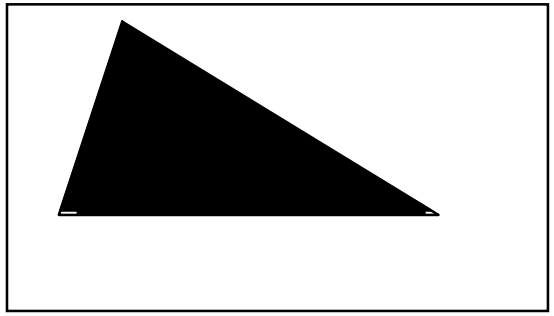
- **Esistono numerose tecniche per la memorizzazione digitale e l'elaborazione di un'immagine**
  - **una prevede la scomposizione dell'immagine in una *griglia* di tanti elementi (punti) che sono l'unità *minima* di memorizzazione;**
  - **La seconda strada prevede la presenza di strutture elementari di natura più complessa, quali *linee*, *circonferenze*, *archi*, etc.**

Codifica delle immagini B/N

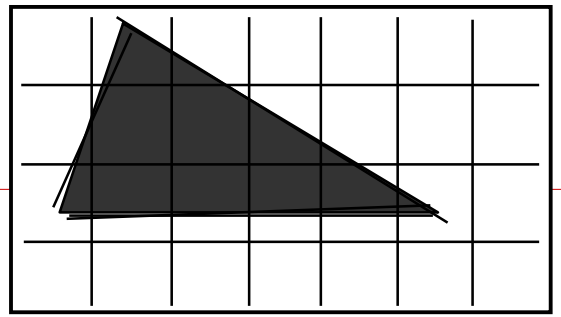
- Dividere l'immagine in una griglia a righe orizzontali e verticali
- Ogni quadratino della griglia è un **pixel** (picture element)
- Codificare ogni pixel con:
  - 0 se il pixel è bianco
  - 1 se il pixel è nero
- Convenire un ordinamento per i bit usati nella codifica

Codifica delle immagini B/N

- Consideriamo un'immagine in bianco e nero, senza ombreggiature o livelli di chiaroscuro



- Suddividiamo l'immagine mediante una griglia formata da righe orizzontali e verticali a distanza costante



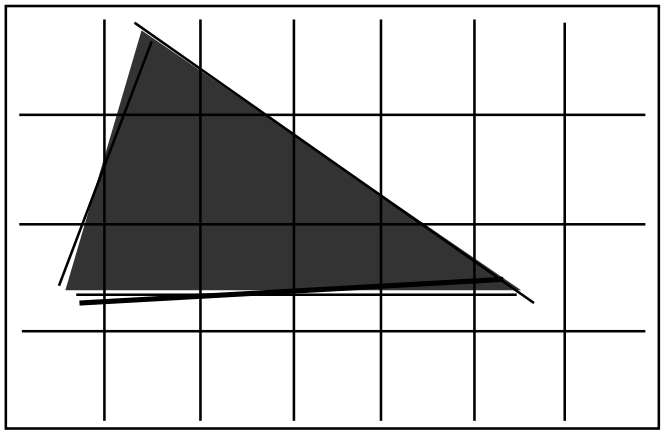
Codifica delle immagini B/N

- Ogni quadratino derivante da tale suddivisione prende il nome di **pixel** (picture element) e può essere codificato in binario secondo la seguente convenzione:
  - il simbolo “**0**” viene utilizzato per la codifica di un pixel corrispondente ad un quadratino bianco (in cui il bianco è predominante)
  - il simbolo “**1**” viene utilizzato per la codifica di un pixel corrispondente ad un quadratino nero (in cui il nero è predominante)

Codifica delle immagini B/N

Poiché una sequenza di bit è lineare, si deve definire una convenzione per ordinare i pixel della griglia

Hp: assumiamo che i pixel siano ordinati dal basso verso l'alto e da sinistra verso destra

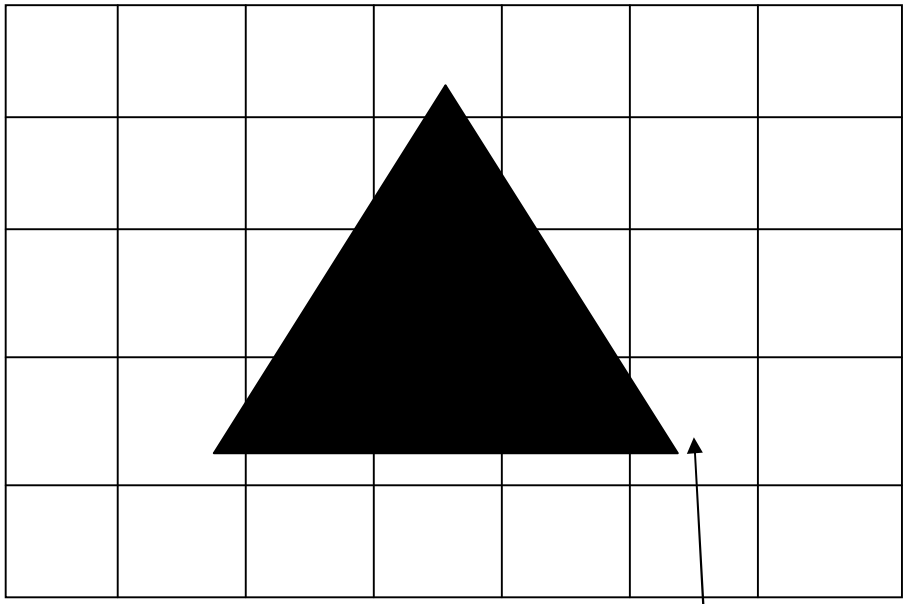


0_{22}	1_{23}	0_{24}	0_{25}	0_{26}	0_{27}	0_{28}
0_{15}	1_{16}	1_{17}	0_{18}	0_{19}	0_{20}	0_{21}
0_{8}	1_{9}	1_{10}	1_{11}	1_{12}	0_{13}	0_{14}
0_{1}	0_{2}	0_{3}	0_{4}	0_{5}	0_{6}	0_{7}

La rappresentazione della figura è data dalla stringa binaria

0000000 0111100 0110000 0100000

Codifica di un'immagine B/N



Pixel = 1

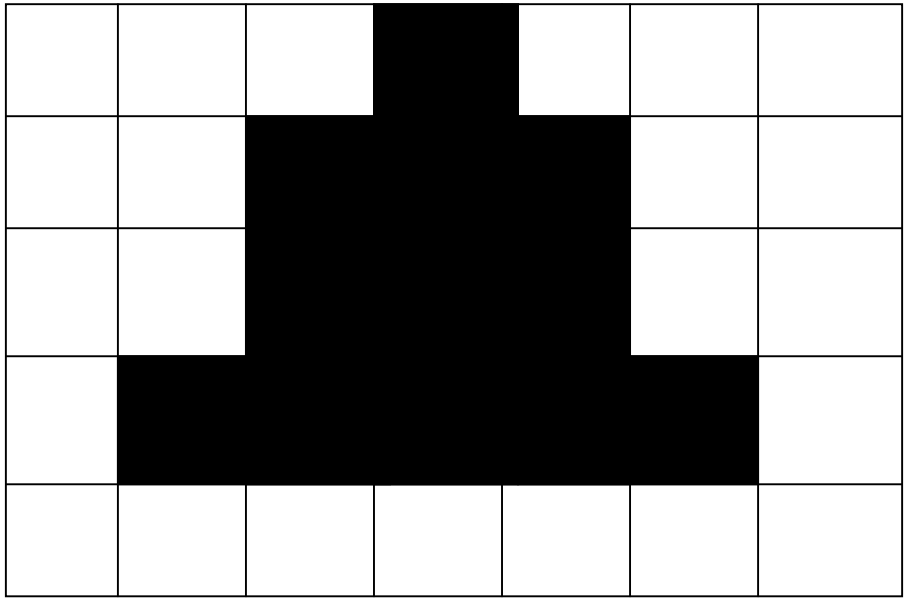
0 0 0 1 0 0 0
0 0 1 1 1 0 0
0 0 1 1 1 0 0
0 1 1 1 1 1 0
0 0 0 0 0 0 0

codifica

Decodifica

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

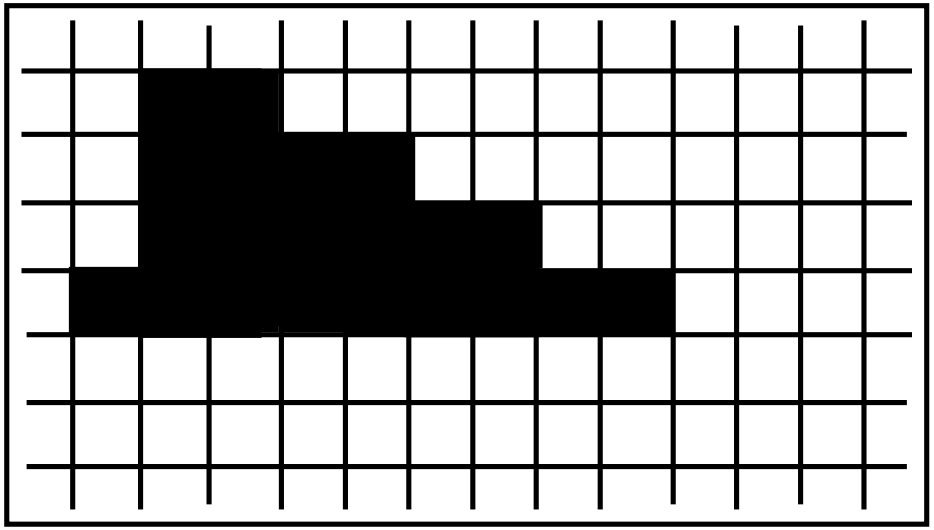
Codifica



Immagine

Codifica delle immagini B/N

- **Non sempre il contorno della figura coincide con le linee della griglia**
  - **nella codifica si ottiene un'approssimazione della figura originaria**
- **La rappresentazione sarà più fedele all'aumentare del numero di pixel**
  - **ossia al diminuire delle dimensioni dei quadratini della griglia in cui è suddivisa l'immagine**



Codifica delle immagini B/N

Quindi: le immagini sono rappresentate con un certo livello di approssimazione, o meglio, di risoluzione, ossia il numero di pixel usati per riprodurre l'immagine.

Risoluzioni tipiche

- 640 x 480 pixel; 800 x 600 pixel
- 1024 x 768 pixel; 1280 x 1024 pixel

Immagini in toni di grigio

- Le immagini in bianco e nero hanno delle sfumature, o **livelli di intensità di grigio**

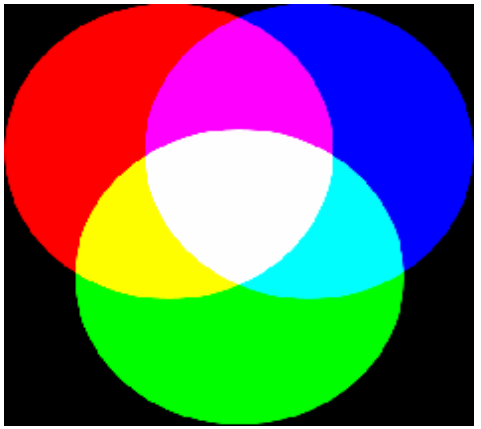
- Per codificare immagini con sfumature:
  - si fissa un insieme di livelli (*toni*) di grigio, cui si assegna convenzionalmente una rappresentazione binaria
  - per ogni pixel si stabilisce il livello medio di grigio e si memorizza la codifica corrispondente a tale livello

- Per memorizzare un pixel non è più sufficiente 1 bit.
  - con 4 bit si possono rappresentare $2^4 = 16$ livelli di grigio
  - con 8 bit ne possiamo distinguere $2^8 = 256$,
  - con K bit ne possiamo distinguere 2^K

Immagini a colori

- Analogamente possono essere codificate le immagini a colori:
  - bisogna definire un insieme di sfumature di colore differenti e rappresentarle mediante una opportuna sequenza di bit

- Nella codifica RGB si utilizzano tre colori
  - rosso (Red), verde (Green) e blu (Blue)



- Ad ogni colore si associa un certo numero di sfumature codificate su N bit
  (2^N possibili sfumature)

- Esempio
  - con 2 bit per colore si ottengono 4 sfumature per colore
  - con 8 bit per colore si ottengono 256 sfumature per colore e 256^3(16 milioni) possibili colori

Immagini a colori

- La qualità dell'immagine dipende
  - dal numero di punti in cui viene suddivisa (*risoluzione*)
  - dai toni di colore permessi dalla codifica;

Bitmap

- La rappresentazione di un'immagine mediante la codifica a pixel viene chiamata **bitmap**
- Il numero di byte richiesti per memorizzare un bitmap dipende dalla risoluzione e dal numero di colori
- Esempio
  - se la risoluzione è 640x480 con 256 colori occorrono 2.457.600 bit = 300 KB

Bitmap

- I formati bitmap più conosciuti sono
  - BITMAP (.bmp),
  - GIF (.gif),
  - JPEG (.jpg)
  - TIFF (.tiff)
- In tali formati si utilizzano metodi di *compressione* per ridurre lo spazio di memorizzazione
  - Aree dello stesso colore si rappresentano in modo "abbreviato".
- E' in genere possibile passare da un formato ad un altro

Codifica dei filmati

- **Immagini in movimento sono memorizzate come sequenze di fotogrammi**
  - **Si sfrutta la limitatezza della capacità percettiva dell'occhio umano**
  - **la sequenza continua di immagini viene *discretizzata* ottenendo una serie di immagini (frame) che variano velocemente, ma a intervalli stabiliti**

- **In genere si tratta di sequenze compresse di immagini**
  - **ad esempio si possono registrare solo le variazioni tra un fotogramma e l'altro**

Codifica dei filmati

- **Esistono vari formati (comprendente il sonoro):**
  - ***mpeg*** **(il piu' usato)**
  - ***avi*** **(microsoft)**
  - ***quicktime*** **(apple)**
  - ***mov***
- **E' possibile ritoccare i singoli fotogrammi**

Codifica dei suoni

- Si effettuano dei campionamenti su dati analogici
  - L'onda sonora viene misurata (campionata) ad intervalli regolari
- Si rappresentano i valori campionati con valori digitali
- La frequenza del campionamento determina la fedeltà della riproduzione del suono
  - Minore è l'intervallo di campionamento e maggiore è la qualità del suono

CD musicali: 44000 campionamenti al secondo, 16 bit per campione

Codifica dei suoni

Alcuni formati:

.mov

.wav

.mpeg

.avi

.midi - usato per l'elaborazione della musica al computer