



# Fondamenti di Informatica

---

- Accademia di Belle Arti di Verona
- Università degli Studi di Verona
- A.A. 2020-2021
  
- Docente - Vincenzo Giannotti

# CAPITOLO 4 – CODIFICA DELL'INFORMAZIONE

# I Codici Binari

In questo capitolo ci viene chiesto di abbandonare per un attimo il nostro consueto modo di pensare i numeri per poterli rappresentare come vengono «visti» all'interno di un Processore.

Vedremo dunque come diversi elementi di informazione possono essere rappresentati all'interno di un computer affinché questi possa elaborarli.

Noi parleremo soprattutto di «numeri» e di come questi possano essere manipolati. I numeri infatti stanno alla base di tutte le tipologie di informazione che normalmente consideriamo siano elaborate dal nostro computer: immagini, testi, suoni etc..

Numeri, ovvero sequenze di bit che possono essere trattati come numeri, sono anche le combinazioni di segnali che la Control Unit trasmette ai diversi apparati per «comandarli».

# I Codici Binari

Il primo problema che ci poniamo è il seguente: come possiamo codificare un certo numero di oggetti, utilizzando i simboli che ha a disposizione il computer?

Il problema di come codificare degli oggetti è ben noto, visto tutti i giorni noi ci confrontiamo con l'esigenza di codificare dei concetti e degli oggetti, tanto che non ci facciamo nemmeno caso. A ben guardare però tutto viene codificato nella nostra società:

1. i beni di una società hanno un numero di inventario;
2. una bottiglia di vino ha un suo proprio codice che ne consente il tracciamento;
3. noi stessi siamo codificati attraverso il nostro codice fiscale
4. i nostri pensieri vengono codificati tramite il linguaggio o la scrittura

# I Codici Binari

Ora però ci troviamo all'interno di un computer e quindi per poter codificare qualsiasi cosa dobbiamo utilizzare i simboli che abbiamo a disposizione e che sono, come abbiamo visto:

**0 e 1**

Dunque, per rappresentare un dato oggetto o una data istruzione dovremo utilizzare una opportuna sequenza di 0 e di 1 che ovviamente deve essere univoca per ciascun oggetto.

Se abbiamo due soli oggetti la questione è banale:

Alla mela il codice 0



e alla pera il codice 1





# I Codici Binari

Ma se abbiamo un cesto di frutta? Come debbono essere combinati tra loro i BIT per poter rappresentare tutta l'informazione di cui abbiamo bisogno?

Come con l'alfabeto, che ci mette a disposizione 21 caratteri con cui siamo stati capaci di scrivere tutta la nostra storia, così, allo stesso modo, pur con soli due BIT, noi possiamo fare altrettanto.

**000 => 0**



**001 => 1**



**010 => 2**



**011 => 3**



**100 => 4**



**101 => 5**



**110 => 6**



**111 => 7**



# I Codici Binari

L'esempio appena visto ci mostra che con tre BIT siamo capaci di codificare otto diversi oggetti.

Da questo si deduce la seguente regola generale:

**con «n» BIT possiamo rappresentare  $2^n$  oggetti**

Per esempio, il codice **ASCII** (American Standard Code for Information Interchange) è uno standard utilizzato per la codifica dei caratteri. Inizialmente venivano utilizzati 7 bit (più un bit di controllo) per rappresentare 128 caratteri; in seguito con Extended ASCII si ritenne più utile utilizzare l'intero byte per rappresentare 256 caratteri, includendo caratteri speciali in diverse lingue.

01100101	101	65	e	Lettera minuscola e
01100110	102	66	f	Lettera minuscola f
01100111	103	67	g	Lettera minuscola g
01101000	104	68	h	Lettera minuscola h
01101001	105	69	i	Lettera minuscola i
01101010	106	6A	j	Lettera minuscola j
01101011	107	6B	k	Lettera minuscola k
01101100	108	6C	l	Lettera minuscola l
01101101	109	6D	m	Lettera minuscola m
01101110	110	6E	n	Lettera minuscola n
01101111	111	6F	o	Lettera minuscola o
01110000	112	70	p	Lettera minuscola p
01110001	113	71	q	Lettera minuscola q
01110010	114	72	r	Lettera minuscola r
01110011	115	73	s	Lettera minuscola s
01110100	116	74	t	Lettera minuscola t
01110101	117	75	u	Lettera minuscola u
01110110	118	76	v	Lettera minuscola v
01110111	119	77	w	Lettera minuscola w
01111000	120	78	x	Lettera minuscola x
01111001	121	79	y	Lettera minuscola y
01111010	122	7A	z	Lettera minuscola z
01111011	123	7B	{	Parentesi graffa aperta
01111100	124	7C		Barra verticale
01111101	125	7D	}	Parentesi graffa chiusa
01111110	126	7E	~	Tilde
01111111	127	7F	DEL	Delete (cancellazione)

# Sistemi di Numerazione

Ora, noi non siamo interessati all'applicazione di alcun criterio di associazione tra oggetti e sequenze di BIT, ma semplicemente vogliamo capire come, una sequenza di BIT, possa essere interpretata come un numero che siamo in grado di capire «al volo».

Tuttavia, prima di questo passaggio, dobbiamo fare una premessa su cosa sia un «Sistema di Numerazione».

Migliaia	Centinaia	Decine	Unità'
1000 = <b>M</b>	100 = <b>C</b>	10 = <b>X</b>	1 = <b>I</b>
2000 = MM	200 = CC	20 = XX	2 = II
3000 = MMM	300 = CCC	30 = XXX	3 = III
	400 = CD	40 = XL	4 = IV
	500 = <b>D</b>	50 = <b>L</b>	5 = <b>V</b>
	600 = DC	60 = LX	6 = VI
	700 = DCC	70 = LXX	7 = VII
	800 = DCCC	80 = LXXX	8 = VIII
	900 = CM	90 = XC	9 = IX

	I	II	III	IIII	V	VI	VII	VIII	IX	X	C
Nell'impero romano	I	II	III	IIII	V	VI	VII	VIII	IX	X	C
Nell'impero cinese	一	二	三	四	五	六	七	八	九	十	百
Nell'impero indiano	१	२	३	४	५	६	७	८	९	१०	१००
Presso i maya (America centrale)	.	..	...	....	—	÷	÷÷	÷÷÷	÷÷÷÷	=	∞



# Sistemi di Numerazione

Un sistema di numerazione si può considerare costituito da:

1. Un insieme di «Simboli elementari» (le cifre);
2. Un insieme di «Regole» che definiscano come rappresentare un numero con le cifre a disposizione.

Sui numeri così definiti, deve essere possibile applicare un insieme di «Operazioni».

Nella nostra vita quotidiana noi siamo abituati ad utilizzare il «**Sistema Decimale**» e lo facciamo perché da sempre, per far di conto, si utilizzano le dieci dita delle mani.

Il nostro attuale sistema di numerazione però non è l'unico possibile e un esempio alternativo molto noto è quello del sistema utilizzato nell'antica Roma.

# Sistema di Numerazione posizionale

I Romani avevano più difficoltà di noi a far di conto perché il loro sistema era di tipo «**non posizionale**».

Vediamo cosa significa questo, con un esempio:

- 3 => III in questo caso la rappresentazione avviene con 3 simboli
- 9 => IX in questo la rappresentazione avviene con 2 simboli dove la I cifra viene sottratta alla II cifra
- 11 => XI in questo la rappresentazione avviene ancora con 2 simboli ma la I cifra viene sommata alla II cifra

Quindi per sapere cosa fare (sommare o sottrarre) noi dobbiamo conoscere come sono disposte le cifre.

La complicazione di questo metodo di rappresentazione è dato dal fatto che alle cifre non viene attribuito un «peso» dato dalla loro posizione.

# Sistema di Numerazione posizionale

Invece, in un «**Sistema Posizionale**» ogni cifra ha un suo ben preciso peso.

Se per esempio prendiamo il numero **324** la cifra più a destra rappresenta le unità, quella centrale le decine, quella più a sinistra le centinaia.

Il sistema di numerazione in base 10 è stato introdotto in Europa dagli Arabi e ha determinato un drastico miglioramento delle capacità di fare calcoli.

La procedura (l'algoritmo) per la moltiplicazione dei numeri romani è infatti molto più complicata di quelle dei numeri arabi.

# Sistema di Numerazione posizionale

Per capire esattamente cosa sia un sistema posizionale e come questo poi possa essere esteso anche ai codici binari, diamo queste semplici regole.

Un sistema posizionale è definito da:

- Un numero «b» detto «Base» : 10 nel caso del sistema decimale
- «b» cifre : 0,1,...9 nel caso del sistema decimale.

A ciascuna cifra viene dato un peso che dipende solo dalla sua posizione nella sequenza. Il «**valore**» di un qualsiasi numero è dato da, calcolando da destra verso sinistra:

cifra in posizione 1 => unità ( $10^0$ ) + cifra in posizione 2 => decine ( $10^1$ ) + cifra in posizione 3 => centinaia ( $10^2$ ) + cifra in posizione 4 => migliaia ( $10^3$ ) e così via...

$$324 \Rightarrow 3 \times 10^2 + 2 \times 10^1 + 4 \times 10^0$$



# Conversione di Base

A questo punto abbiamo gli strumenti per riportare questo stesso ragionamento alla «**base 2**» che sappiamo essere quella adottata dal computer, potendo contare sui soli due simboli 0 e 1 (le cifre). Le regole infatti sono le stesse.

Allora, dato che le regole per calcolare il valore di un numero binario sono le stesse di quelle viste per i numeri decimali, è lecito chiedersi:

- Come possiamo convertire un numero rappresentato in «base 10» in uno rappresentato in «base 2»?
- Come possiamo viceversa convertire un numero rappresentato in «base 2» in uno rappresentato in «base 10»?

Questo tipo di conversione si rende necessaria proprio per comprendere «al volo» il numero con cui abbiamo a che fare.

# Conversione di base

Prendiamo per esempio il seguente numero in base dieci:

$$108_{10} = ???_2$$

Il procedimento di conversione è il seguente: si divide  $108/2$  e si mette da parte il resto; si prende il risultato e si ripete la stessa operazione fino a quando il risultato della divisione sia pari a 0.

$$108/2 = 54 \quad \text{resto} = 0$$

$$54/2 = 27 \quad \text{resto} = 0$$

$$27/2 = 13 \quad \text{resto} = 1$$

$$13/2 = 6 \quad \text{resto} = 1$$

$$6/2 = 3 \quad \text{resto} = 0$$

$$3/2 = 1 \quad \text{resto} = 1$$

$$1/2 = 0 \quad \text{resto} = 1$$

$$108_{10} = 1101100_2$$

# Conversione di base

Prendiamo ora lo stesso numero in base due e cerchiamo di fare il procedimento inverso:

$$1101100_2 \Rightarrow ???_{10}$$

Associamo a ciascuna cifra il suo peso, ricordando però che ora lavoriamo in base due:

**1 1 0 1 1 0 0**

↓ ↓ ↓ ↓ ↓ ↓ ↓

$2^6$   $2^5$   $2^4$   $2^3$   $2^2$   $2^1$   $2^0$

il numero è sempre lo stesso, che sia rappresentato in base 2 o in base 10

↓ ↓ ↓ ↓ ↓ ↓ ↓

64 32 16 8 4 2 1

↓ ↓ ↓ ↓ ↓ ↓ ↓

$$64 + 32 + 0 + 8 + 4 + 0 + 0 = \mathbf{108}_{10}$$

# Notazione esadecimale

Considerando l'attività interna di un computer vediamo però che quasi mai i bit vengono utilizzati uno alla volta; normalmente, per comodità, si utilizzano stringhe di bit anche in funzione della capacità di trasporto dell'informazione.

Quando tra le specifiche del nostro computer leggiamo «sistema a 32 bit» o «sistema a 64 bit», significa che la capacità di trasporto (bus) e di elaborazione (processore) del nostro sistema è di 32 o di 64 bit alla volta.

È molto difficile tuttavia leggere e comprendere velocemente stringhe di bit molto lunghe del tipo 1011010100110001 e di conseguenza è molto facile commettere degli errori.

Per questo si preferisce utilizzare la «**notazione esadecimale**».



SEQUENZA BIT	RAPPRESENTAZIONE ESADECIMALE
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

# Notazione esadecimale

Nella notazione esadecimale ad una cifra corrisponde una sequenza di 4 bit (ricordiamo che  $2^4=16$ ). Questo significa che una stringa come quella precedente:

**1011010100110001**

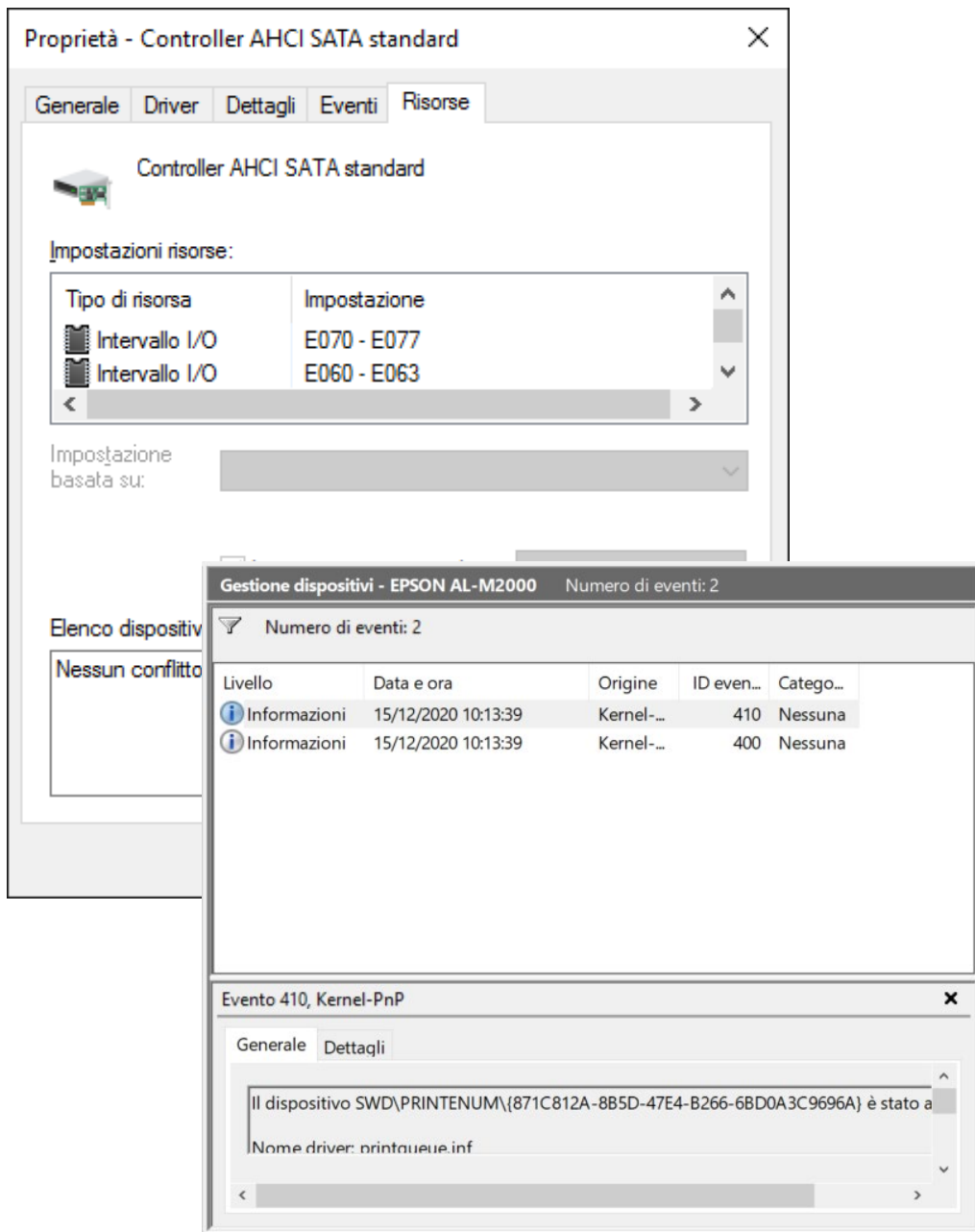
può essere rappresentata con 4 simboli esadecimali:

**B531**

per ottenere questo risultato, la sequenza viene suddivisa in pacchetti di 4 bit e ciascun pacchetto viene convertito nella sua rappresentazione esadecimale, cioè in base 16.

**1011 0101 0011 0001**

**B     5     3     1**



# Notazione esadecimale

Questa notazione è molto utilizzata sia per rappresentare alcune informazioni di sistema del nostro computer, come per esempio quelle che riguardano le impostazioni dei dispositivi o dei registri, sia in programmazione, essendo in grado, come detto, di mettere in relazione diretta una sola cifra esadecimale con quattro cifre binarie.

# Esercizio di conversione

Prendiamo il numero **242** e convertiamolo in binario

Prendiamo il risultato binario e riconvertiamolo in decimale

Prendiamo il risultato binario e convertiamolo in esadecimale

# Esercizio di conversione

242

242/2    121    r0

121/2    60    r1

60/2    30    r0

30/2    15    r0

15/2    7    r1

7/2    3    r1

3/2    1    r1

1/2    0    r1

1 1 1 1 0 0 1 0

F        2

128+64+32+16+2=242



# La codifica di dati multimediali

Le informazioni multimediali (immagini, video, suoni) hanno per l'uomo un carattere «**analogico**».

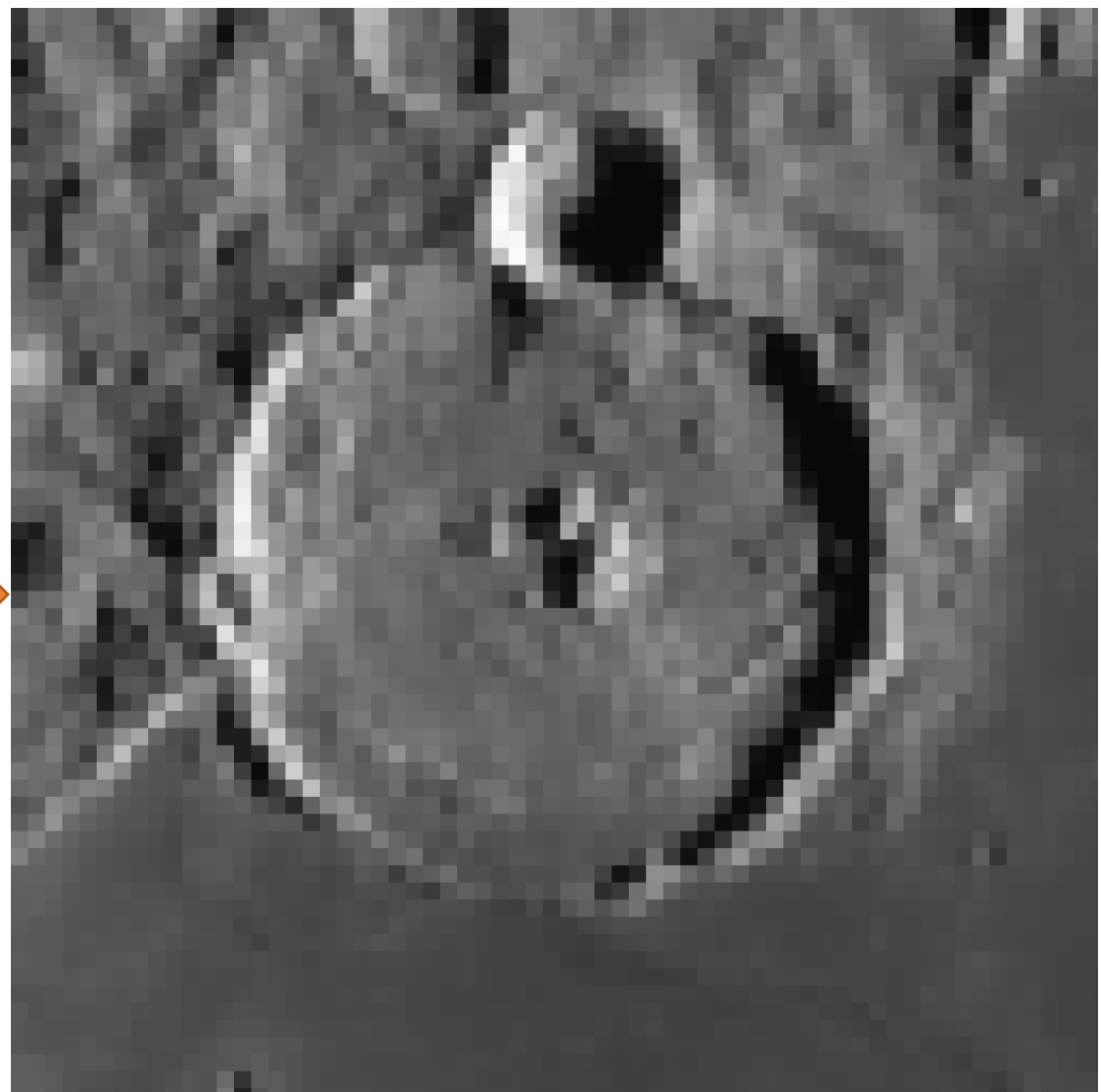
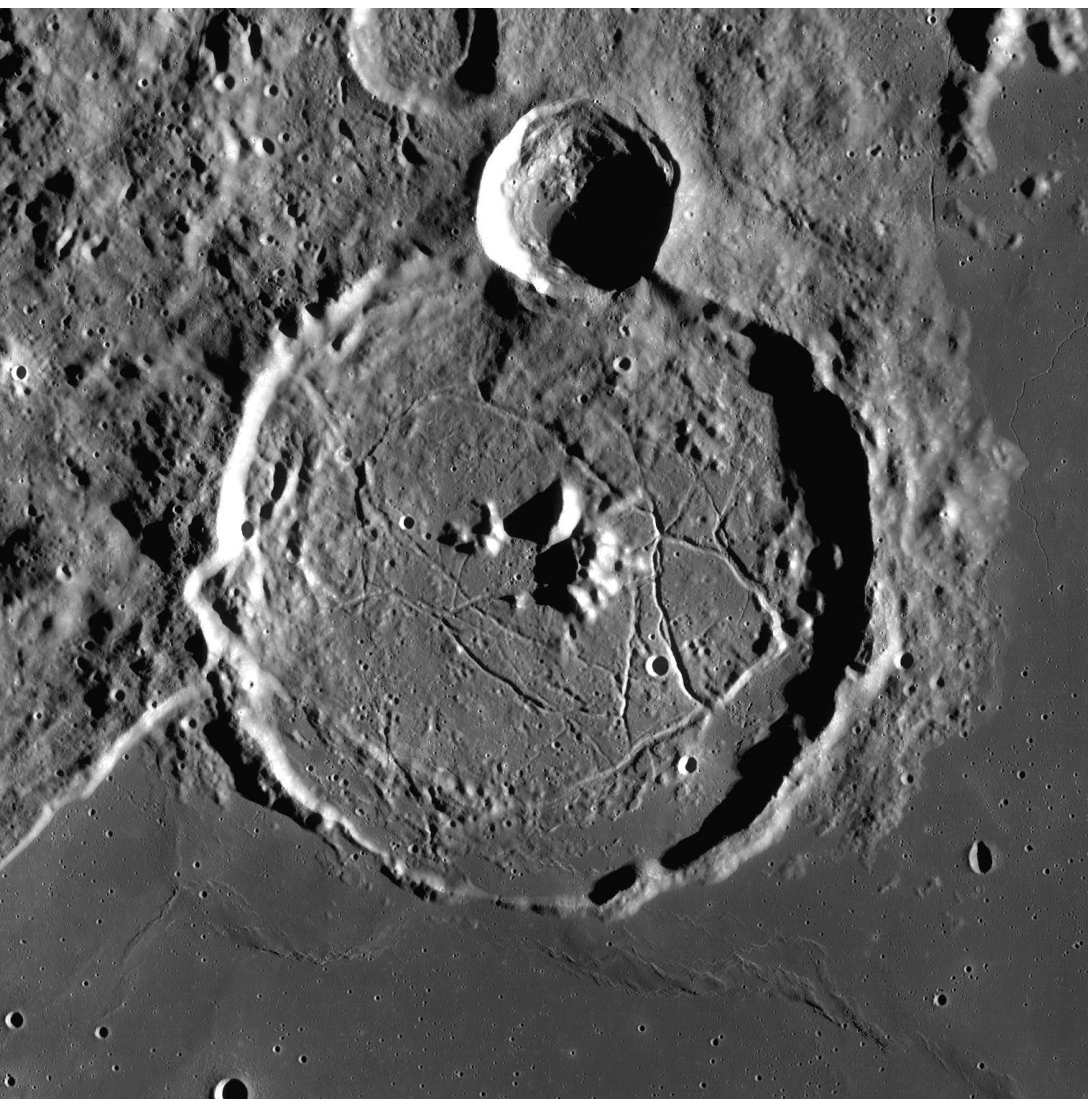
Questo significa che tali informazioni possono essere rappresentate da grandezze continue che a loro volta possono essere rappresentate tramite numeri reali (quindi con precisione teoricamente infinita).

Quando si passa al mondo digitale, che è per sua natura, finito, una grandezza analogica deve essere «campionata» così da ricavarne dei valori approssimati:

- per esempio, di una immagine si può rilevare la composizione del colore in ogni mmq
- di un suono può essere rilevata l'altezza ogni millesimo di secondo

I valori che rileviamo in questo modo, possono essere convertiti in numeri binari ed essere trasferiti nella memoria di un computer.



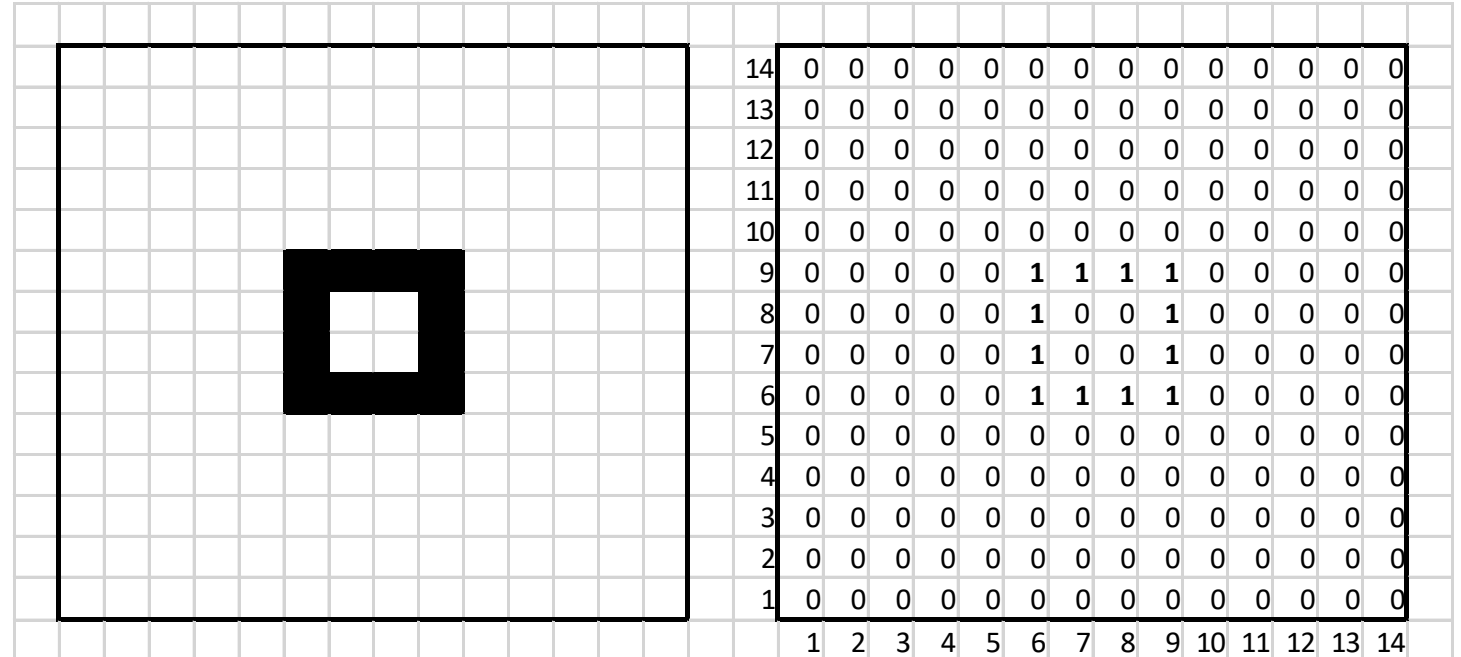


# La codifica delle immagini

Se per esempio prendiamo una immagine in bianco e nero senza gradazioni di grigio, ciascun elemento di immagine (picture element = **PIXEL**) può essere codificato con:

- 0=> bianco predominante
- 1=> nero predominante

Per convenzione l'immagine viene ricomposta su una griglia ordinata dal basso verso l'alto e da sinistra verso destra.



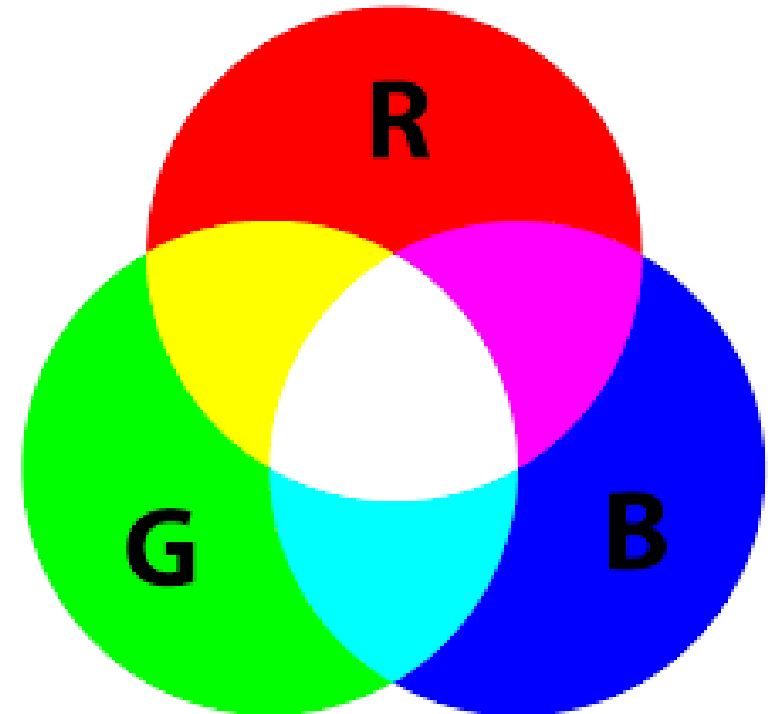


# La codifica delle immagini

- La qualità dell'immagine dipende pertanto dalla fittezza del campionamento e quindi dal numero di pixel/cm<sup>2</sup> di cui è composta. Questo rapporto fornisce la «**risoluzione**» dell'immagine.
- La qualità dell'immagine dipende anche dalla sua «**profondità**», ossia dal numero di bit utilizzati per rappresentare un singolo pixel dell'immagine.
- Il numero di bit richiesti per memorizzare un'immagine dipende dalla risoluzione e dalla profondità, mentre la dimensione dell'immagine è espressa in numero di pixel nella forma *righe X colonne*.
- Per esempio, se abbiamo la necessità di codificare diverse gradazioni di grigio, dobbiamo utilizzare più bit per ciascun pixel: con 8 bit possiamo rappresentare  $2^8=256$  livelli di grigio; con 12 bit possiamo rappresentare  $2^{12}=4096$  livelli di grigio.
- Se invece vogliamo poter trattare l'informazione del «**colore**» dobbiamo tenere presente che questo è dato dalla composizione dei tre colori primari: Red, Green, Blue (RGB).

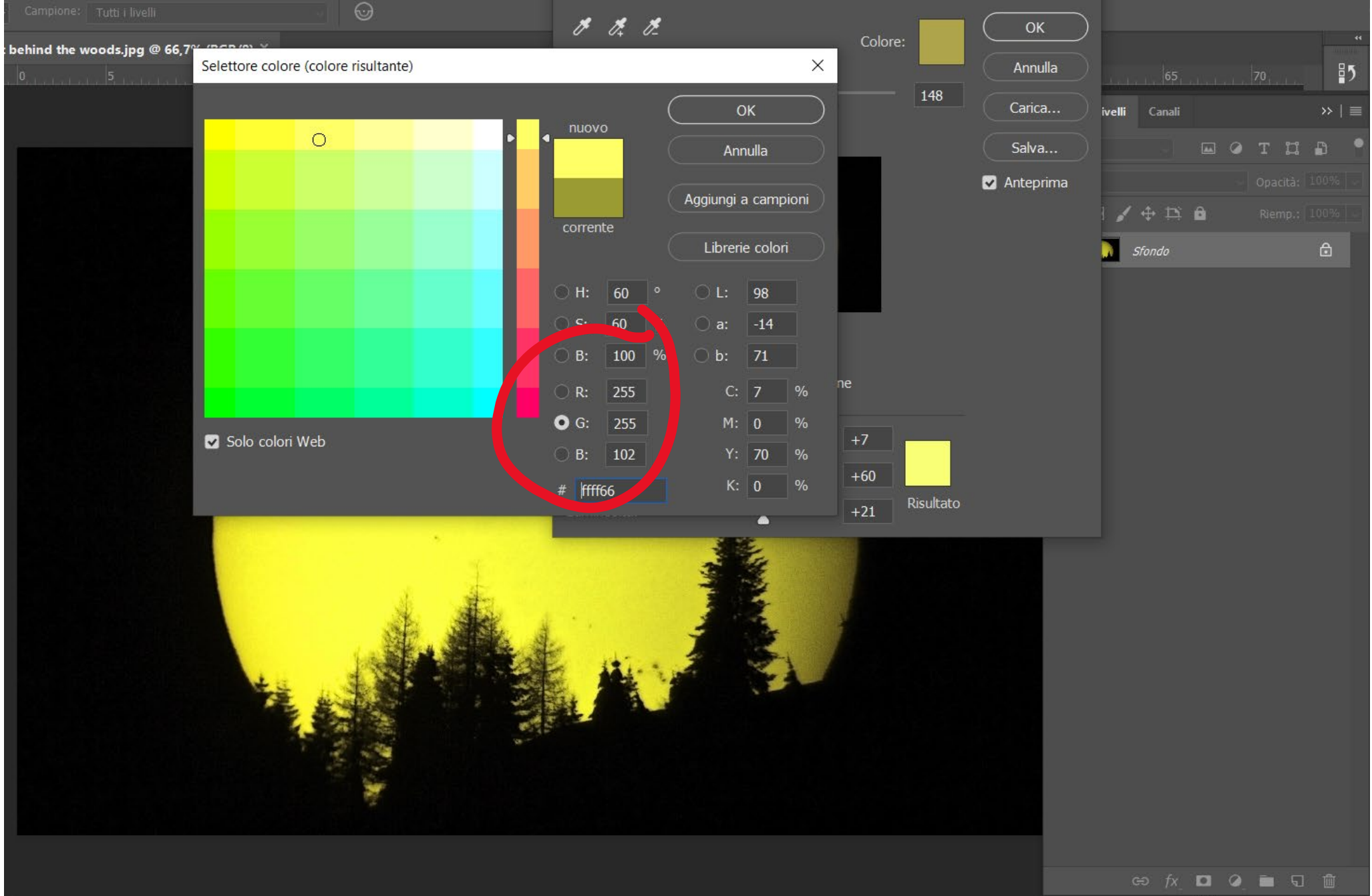
# La codifica delle immagini a colori

- RGB è un modello di colore di tipo additivo utilizzato in informatica, dove la somma dei colori Rosso, Verde e Blu (delle loro lunghezze d'onda) produce un aumento della luminosità fino a che si ottiene il bianco (tricromia).
- Ciascuno dei tre colori è a sua volta dato da una gradazione che il più delle volte viene espressa con 8 bit (0-255).
- Nel bianco abbiamo le tre componenti a 255; nel nero le tre componenti valgono tutte 0.
- Con questa tecnica è possibile rappresentare fino a 16,8 milioni di colori (Truecolor).



# La codifica delle immagini a colori

- Il modello CMYK (Cyan, Magenta, Yellow, Black), da non confondere col RGB, si basa invece sulla cosiddetta «**sintesi sottrattiva**» e riguarda la definizione del colore nei dispositivi di stampa.
- In questo caso i pigmenti vengono miscelati in maniera da produrre un colore riflesso risultante dalla sottrazione dallo spettro luminoso, delle frequenze assorbite da uno strato di sostanze opache.
- La riproduzione del colore in stampa tramite questi quattro inchiostri è detta stampa in quadricromia.
- Nella pittura tradizionale si utilizzano anche in questo caso tre colori primari sottrattivi che sono il rosso, il blu e il giallo.

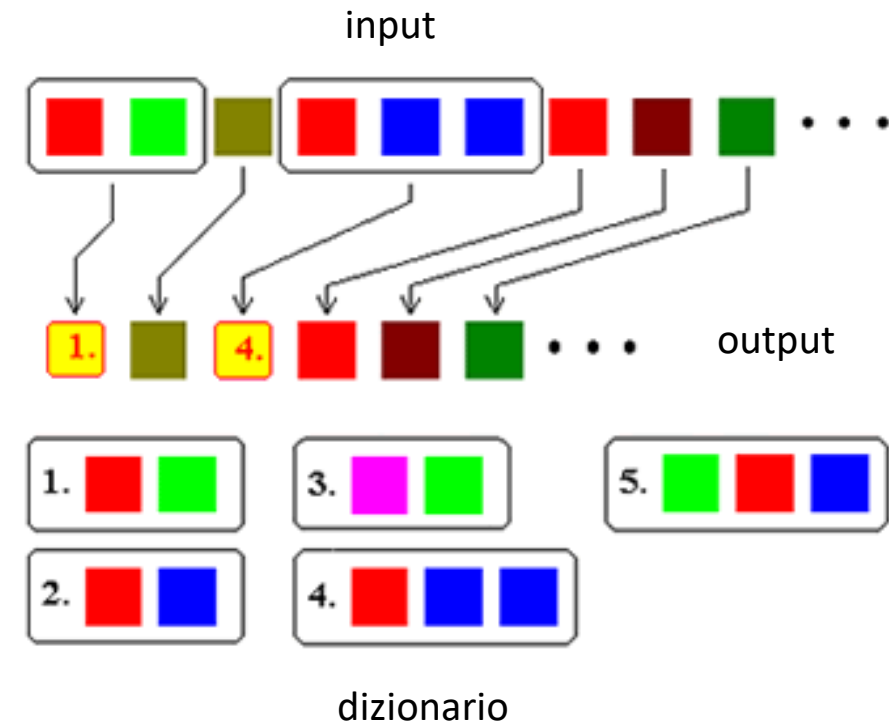


# La codifica delle immagini a colori

- Abbiamo visto che un'immagine a colori è normalmente caratterizzata da tre Byte per ciascun pixel.
- Poiché una immagine di buona qualità, catturata per esempio con una camera digitale, può tranquillamente essere composta da oltre 20-50-100 milioni di pixel, capiamo subito che memorizzare anche una singola immagine può comportare l'impiego di notevoli risorse.
- Per ottimizzare lo spazio di memoria utilizzato possono essere adottati degli algoritmi di compressione:
  - Senza perdita di informazione
  - Con perdita di informazione

# La compressione delle immagini

- Algoritmi senza perdita di informazione (**Lossless**). Questi impiegano delle tecniche del tipo:
- Viene costruito una sorta di dizionario contenente gruppi di punti che si ripetono frequentemente, quindi l'immagine viene ricodificata utilizzando il dizionario. Per esempio, se abbiamo 345 pixel di colore A seguiti da 514 di colore B e da 18 di colore C, possiamo codificare quella parte di immagine con 345A514B18C.
- Alcuni noti formati lossless sono il PNG, TIFF, GIF (quest'ultimo per immagini fino a 256 colori).





# La compressione delle immagini

- Nel caso ci serva una maggior compressione, possiamo utilizzare degli algoritmi con perdita di informazione (**Lossy**). Questi sono impiegati quando abbiamo a che fare con immagini con milioni di colori dove i metodi lossless risultano meno efficienti a causa delle frequenti variazioni.
- Tra i metodi lossy, il più noto è **JPEG** che utilizza una combinazione di varie tecniche progettate per sfruttare le limitazioni dell'occhio umano:
- Una prima tecnica, considera il fatto che l'occhio umano è più sensibile alle variazioni di luminosità che ai cambiamenti di colore. Così, a partire da un'immagine che è codificata in termini di componenti di luminanza e di cromaticanza, il primo passo è quello di mediare i valori di cromaticanza su una matrice quadrata di due per due pixel. Questo riduce l'informazione di cromaticanza di un fattore quattro conservando però tutta l'informazione originale di luminosità. Il risultato è un significativo grado di compressione senza una perdita rilevante di qualità dell'immagine.

# La compressione delle immagini

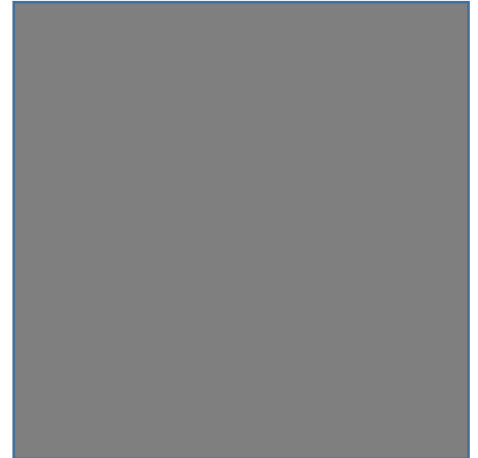
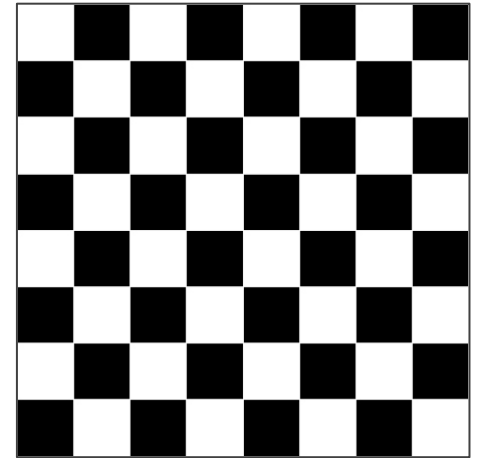
- Il passo successivo è quello di dividere l'immagine in blocchi da otto per otto pixel e di comprimere i dati di ciascun blocco separatamente. Questo viene fatto applicando una tecnica matematica nota come «Trasformata Discreta del Coseno» i cui dettagli non ci interessano. Quello che invece interessa è che questa trasformazione converte il blocco originale otto per otto calcolando una funzione i cui coefficienti rispecchiano la relazione che interviene tra i pixel del blocco piuttosto che il loro reale valore. In questa funzione i valori inferiori ad una data soglia sono posti a zero in quanto i relativi valori sarebbero non percepibili dall'occhio umano.

# La compressione delle immagini

Per esempio, se il blocco originale conteneva un motivo a scacchi, il nuovo elemento potrebbe riflettere un'uniforme colore medio senza che debba essere rispettata la distribuzione originale dei pixel.

A questo punto possono essere applicate le tecniche lossless più tradizionali che abbiamo già visto, con lo scopo di comprimere ulteriormente l'immagine.

Tutte insieme le tecniche standard di base del JPEG riescono normalmente a comprimere immagini a colori di un fattore almeno 10 arrivando spesso anche a fattori 30 senza che vi sia una sostanziale perdita di qualità.



Full res  
1500kB



Half res 118kB



Low res 36kB



# Compressione Audio e Video

- Gli standard più comunemente utilizzati per la codifica e la compressione di audio e di video sono stati sviluppati dal **Motion Picture Experts Group** sotto la guida di ISO. L'acronimo dello standard deriva proprio dal nome del gruppo che l'ha sviluppato. MPEG comprende una varietà di standard per le diverse applicazioni. Per esempio, le necessità per le trasmissioni TV ad alta definizione sono diverse da quelle per le videoconferenze e così saranno diversi gli standard di compressione dei segnali. Le tecniche impiegate da MPEG sono piuttosto complicate e a noi non interessa dettagliarle; basti sapere che queste sono basate sul fatto che il video non è altro che una sequenza di immagini che fornisce l'impressione del movimento. Per comprimere tali sequenze, solo alcune delle immagini (gli I-frame) sono codificate nella loro interezza (normalmente in formati simili al Jpeg) mentre delle immagini che si trovano tra gli I-frame vengono codificate solo le differenze con l'immagine di riferimento.



# Compressione Audio e Video

Per quanto riguarda l'Audio il sistema più noto per la compressione è il MP3, che fu sviluppato all'interno degli standard MPEG. Infatti, l'acronimo MP3 non è altro che l'abbreviazione di MPEG layer 3.

MP3 sfrutta le proprietà e le limitazioni dell'orecchio umano, rimuovendo quei dettagli che non è in grado di percepire:

- Una delle tecniche adottate è chiamata «temporal masking» e si basa sul fatto che per un breve periodo, dopo un suono di forte intensità, l'orecchio umano non è in grado di rilevare i suoni più tenui che altrimenti in condizioni normali sarebbero udibili.
- Un'altra, chiamata «frequency masking» considera che un suono ad una data frequenza tende a mascherare i suoni più tenui che ricadono nell'intorno della sua frequenza.

Sfruttando queste caratteristiche, MP3 può essere utilizzato per ottenere una notevole compressione dei segnali audio, mantenendo tuttavia il suono nei pressi della qualità di un CD.

# Compressione Audio e Video

Vale la pena di puntualizzare che tuttavia, nella compressione audio e video, non è detto che l'obiettivo principale sia il contenimento dello spazio di archiviazione. Molto spesso ciò che conta in questi casi è la «**trasmissione**» dell'informazione (es. *streaming* audio o video).

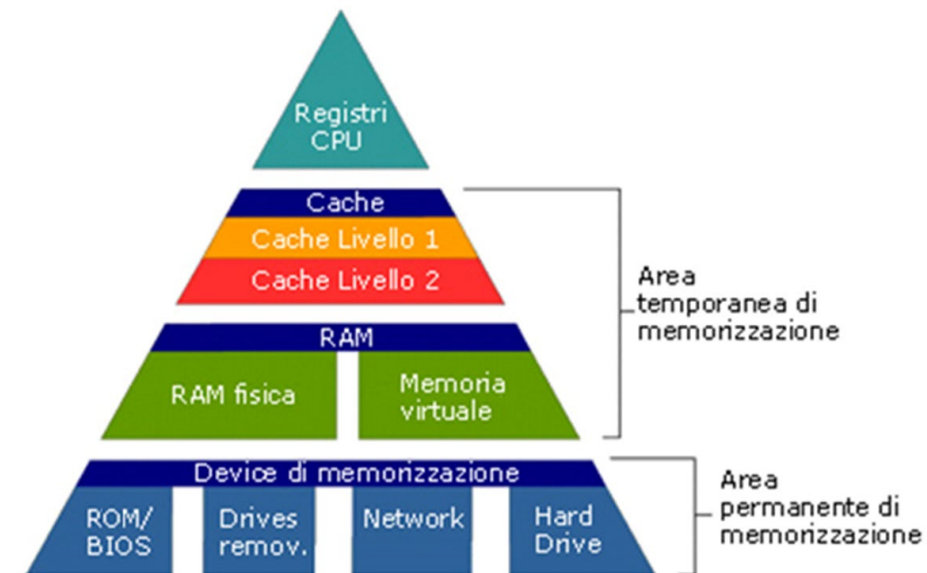
- Se abbiamo un video in cui ciascun frame pesa 1 MB e una linea di trasmissione che non sopporta più di 10 Kb/s non ci sarà alcuna possibilità di successo.
- Normalmente la trasmissione video richiede un «**transfer rate**» di alcuni Mb/s mentre per la trasmissione audio sono sufficienti 64 Kb/s.



# L'organizzazione della Memoria

Ora che abbiamo visto come viene codificata l'informazione, torniamo ad occuparci, ora anche da un punto di vista tecnologico, di una componente fondamentale del nostro computer, la **Memoria**, dove i dati codificati vengono posti prima che siano elaborati:

- Registri e Cache memory – memorie di supporto alle attività di elaborazione
- La memoria Centrale RAM (Random Access Memory) – volatile
- La memoria ROM (Read Only Memory) – per memorizzare il programma di avvio del computer
- Memorie di massa (HD, penne, Cdrom etc..) – per memorizzare in maniera permanente grandi quantità di informazioni



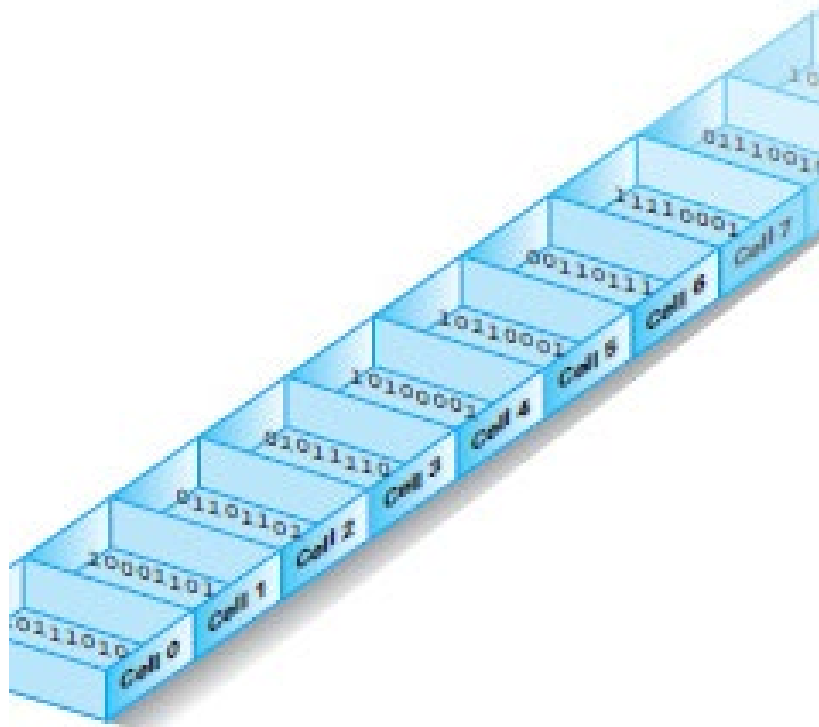
# La Memoria Centrale

- Quando si parla della Memoria Centrale del Computer ci si riferisce ad una serie di dispositivi in grado di immagazzinare dati e di trasferirli ad elevata velocità, dialogando direttamente con la CPU. L'appellativo di «**Memoria Centrale**» serve a differenziare tali dispositivi rispetto a dispositivi esterni, di maggiore capacità e minore velocità, utilizzati per archiviare dati in grande quantità.
- I dispositivi di Memoria Centrale sono usualmente di tipo «semiconduttore» e possono essere:
  - Le memorie del primo tipo sono le cosiddette «**Memorie RAM**» - dove RAM sta per Random Access Memory - e la principale caratteristica di queste è la volatilità: i dati sono memorizzati soltanto se il computer è alimentato; quando il computer viene spento, la memoria RAM si azzerà.
  - Le memorie del secondo tipo sono le cosiddette «**Memorie ROM**» (Read Only Memory) e queste sono in grado di mantenere le informazioni memorizzate anche dopo lo spegnimento del computer.

# La Memoria Centrale

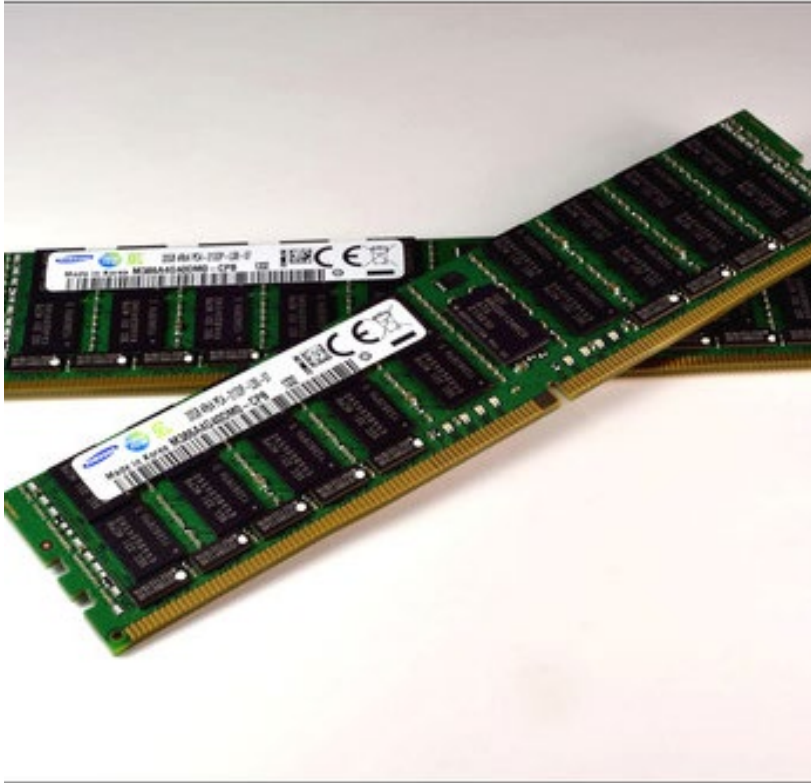
- Nella memoria Centrale ciascun bit viene memorizzato utilizzando un apposito circuito elettronico miniaturizzato. Tipicamente essa è organizzata in celle in grado di memorizzare 8 bit (un byte). Ciò consente di facilitare le operazioni di lettura e di scrittura.
- Sebbene il concetto di destra e sinistra non abbia alcun significato all'interno di un circuito elettronico, tuttavia per convenzione l'organizzazione dei bit all'interno della cella viene realizzata in maniera tale che a ciascun bit possa essere attribuito un «peso» cosicché sia possibile calcolare un valore numerico per ciascuna stringa di bit.

$$01101100_2 \Rightarrow 108_{10}$$



# La Memoria Centrale RAM

- Ad ogni cella viene quindi attribuito un **indirizzo** numerico attraverso il quale noi siamo in grado di individuare univocamente qualsiasi posizione di memoria. Questa tecnica consente sia di saltare con precisione da una cella a un'altra, sia di muoversi tra celle contigue.
- In questo modo non abbiamo teoricamente più limiti quando vogliamo, per esempio, rappresentare numeri molto grandi per i quali non basta un'unica cella (che ricordiamo può rappresentare numeri da 0 a 255).



# La Memoria Centrale RAM

Poiché la memoria principale del nostro computer è composta da celle indirizzate e accessibili indipendentemente l'una dall'altra, in maniera casuale, essa viene detta ad accesso casuale, da cui l'acronimo **RAM**.

- Talvolta, in funzione della tecnologia utilizzata per costruire le celle, le memorie RAM possono avere denominazioni diverse:
  - SRAM (Static RAM)
  - DRAM (Dynamic RAM)
  - SDRAM (Synchronous DRAM)

# La Memoria Centrale RAM - Tecnologie

- SRAM (Static RAM) : Consentono di mantenere le informazioni per un tempo infinito (finché sono alimentate), sono molto veloci e consumano poco e quindi dissipano poco calore. La circuiteria però è più complessa e ciò le rende costose e poco capienti. Per questi motivi sono solitamente usate per le memorie cache, dove sono necessarie elevate velocità in abbinamento a ridotti consumi e capienze non troppo elevate.
- DRAM (Dynamic RAM) : la circuiteria è molto semplice e tuttavia non sono in grado di mantenere l'informazione a lungo per cui debbono essere periodicamente «rinfrescate». Durante il processo di refresh rimangono inutilizzabili per cui risultano più lente delle SRAM.
- SDRAM (Synchronous DRAM) : come le DRAM ma sincrone (accesso comandato dal clock).

# La Memoria Centrale ROM

Fanno parte della memoria principale anche le memorie **ROM** (Read-Only Memory ) di tipo non volatile.

- In questo tipo di memorie il procedimento di scrittura dei dati è normalmente piuttosto difficoltoso e lento (come nel caso delle memorie EPROM - Erasable Programmable Read-Only Memory) e talvolta addirittura impossibile (come per le memorie PROM - Programmable Read-Only Memory): per questo motivo, a differenza delle memorie RAM, non possono essere impiegate come «memorie di lavoro» e il loro scopo è piuttosto quello di memorizzare il cosiddetto «**firmware**», ossia un software che viene realizzato ad hoc per uno specifico hardware.
- Il **BIOS** (Basic Input/output System) per esempio, è il firmware che viene utilizzato nei computer IBM PC compatibili per «far partire il sistema» inizializzando i componenti hardware e caricando il Sistema Operativo.



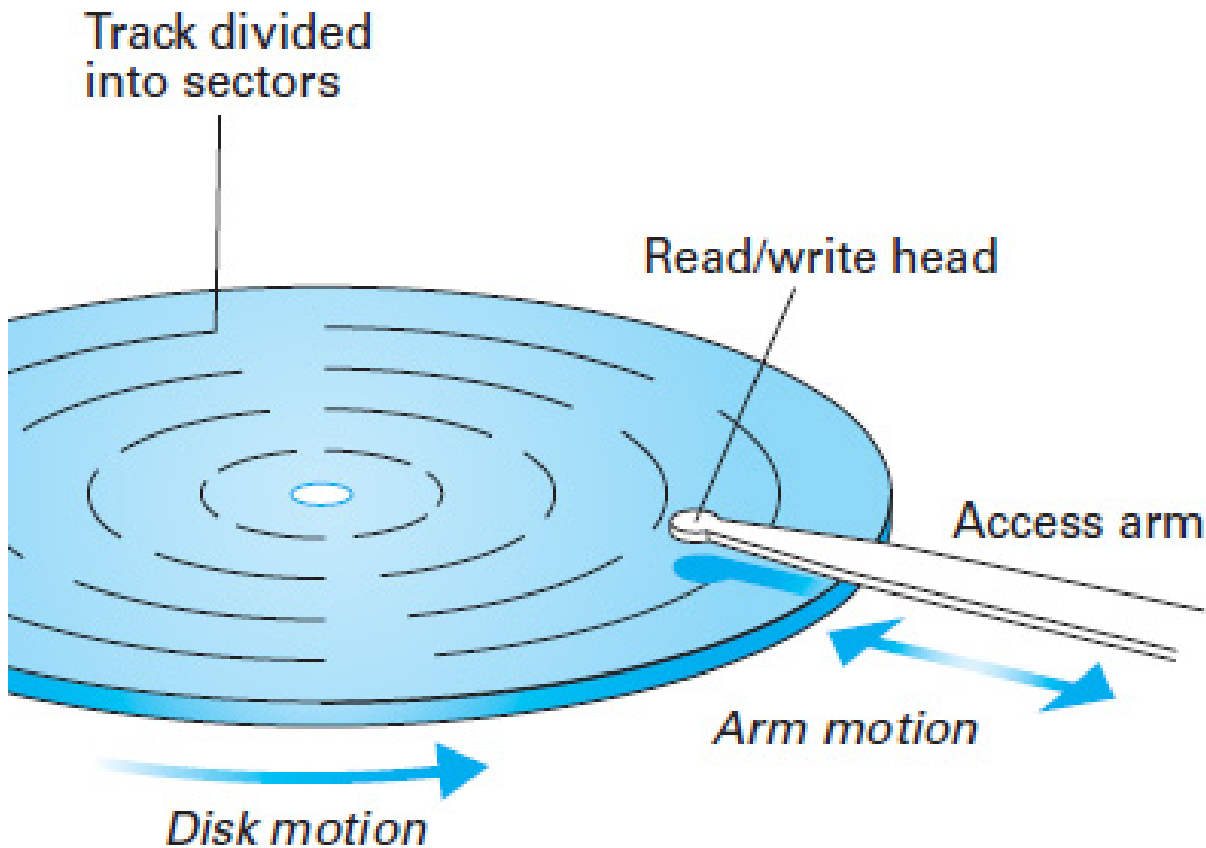
# Memorie di Massa

- A causa del fatto che le memorie RAM sono volatili ed hanno una capacità limitata di memorizzazione e un costo elevato, la maggior parte dei computer utilizza dei dispositivi aggiuntivi per lo *storage* di grandi quantità di dati.
- Alcuni di questi dispositivi, a noi ben noti, sono gli Hard Disk, i CD, i DVD.
- Il maggiore svantaggio di questi dispositivi rispetto alle memorie RAM risiede nel fatto che essi utilizzano degli elementi meccanici per posizionarsi ad un dato indirizzo; ciò comporta prestazioni molto inferiori in termini di «tempi di accesso» all'informazione.
- Le memorie FLASH, a differenza delle precedenti, sono memorie di massa di tipo elettronico a semiconduttore e non hanno componenti meccanici. Questo tipo di memorie, utilizzate per esempio negli SSD (Soli State Drive), sono senz'altro preferibili alle precedenti le quali tuttavia mantengono ancora un buon mercato in quanto più economiche soprattutto per l'archiviazione di grandi moli di dati.



# Dischi Magnetici

- I dischi magnetici (**Hard Disk**) sono da molti anni il dispositivo più utilizzato per memorizzare e leggere, in maniera efficiente, grandi moli di dati. Delle testine poste sopra la superficie magnetica del disco sono in grado di leggere o scrivere dei dati.
- Mentre il disco gira velocemente, la testina posta trasversalmente può spostarsi da una «**traccia**» all'altra del disco. Normalmente un hard disk è composto da più dischi impilati uno sull'altro, distanziati tra loro per far passare le testine di lettura/scrittura che si muovono all'unisono.



## Dischi Magnetici

- Ciascuna traccia a sua volta è suddivisa in «**settori**» dove le informazioni sono memorizzate come sequenze continue di bit (un valore tipico è 512).
- Il processo di inizializzazione di un disco, nel quale vengono allocate tracce e settori, è detto «**formattazione**».
- Tra i parametri utilizzati per specificare le prestazioni di un disco ci sono il tempo di accesso e la velocità di rotazione: un disco da 7200 rpm (che è un tipico valore dei dischi commerciali) ha un tempo di accesso di circa 9ms.



## Dischi Ottici

- I dischi ottici (**CD e DVD**) sono realizzati utilizzando un materiale riflettente ricoperto da una superficie protettiva. Le informazioni sono memorizzate creando una variazione sulla superficie riflettente che può essere letta con un laser in grado di rilevare le irregolarità sulla superficie del disco, mentre questo ruota. I tradizionali CD hanno una capacità di memorizzazione di circa 600-700 MB. Il DVD (Digital Versatile Disk) è composto di più strati sovrapposti e può consentire di memorizzare diversi GB (il laser può venire focalizzato sui diversi strati).
- Infine abbiamo il **Blu-ray Disk**; questo può memorizzare fino a 5 volte i dati memorizzabili dal DVD utilizzando dei laser più precisi in luce blu.



# Memorie Flash

- Sia i dischi magnetici, sia i dischi ottici, hanno la limitazione di dover utilizzare degli elementi meccanici per leggere/scrivere i dati e questo riduce notevolmente le prestazioni di questi dispositivi.
- La più recente tecnologia delle Memorie Flash ha di fatto eliminato questo problema.
- Nelle Memorie Flash i bit sono memorizzati inviando dei segnali elettrici ad un supporto dove gli elettroni sono intrappolati in piccole camere di diossido di silicio. Questa alterazione può essere mantenuta per diversi anni.



# Memorie Flash

Questo tipo di memoria ha però una limitazione data dal fatto che la ripetuta cancellazione e sovrascrittura di dati causa dei micro danneggiamenti che ne impediscono l'utilizzazione come memorie principali (che fanno un lavoro continuo).

Per mantenere un buon livello di affidabilità debbono essere supportate da algoritmi ECC (Error Correcting Code) che verificano la correttezza delle informazioni memorizzate.



# Prossimo Capitolo

## Reti di computer e Internet

Nel prossimo capitolo vedremo come funziona una rete di calcolatori, come è nato Internet e come funziona il World Wide Web.