

INTRODUZIONE AI CALCOLATORI ELETTRONICI

**CORSO DI LAUREA IN ECONOMIA AZIENDALE
ING. PAOLA LAPADULA - UNIVERSITÀ DEGLI STUDI
DELLA BASILICATA**

A.A. 2024/2025

RAPPRESENTAZIONE DEI NUMERI

- Numeri rappresentabili all'interno di un calcolatore sono di due tipi

Numeri Interi:

- utilizzando il sistema di numerazione binario,
- qualsiasi numero intero può essere rappresentato tramite una sequenza di bit di opportuna lunghezza

RAPPRESENTAZIONE DEI NUMERI

- Numeri rappresentabili all'interno di un calcolatore sono di due tipi (cont.)

Numeri Con la Virgola:

- Scomponendo il numero in parte intera e parte frazionaria
- e sempre utilizzando il sistema di numerazione binario,
- è possibile ottenere una rappresentazione anche per questo tipo di numeri

CONVERSIONI

- Per codificare i numeri esiste un sistema di numerazione simile a quello tradizionale (sistema decimale e posizionale)
 - Ogni cifra ha un valore che dipende dalla posizione che occupa all'interno del numero
 - Per esempio

2 1 0 Posizione

645

$$6 * (10^2) + 4 * (10^1) + 5 * (10^0) = 600 + 40 + 5 = 645$$

CONVERSIONI

- Conversione dal sistema binario al sistema decimale
 - Per esempio

3 2 1 0 Posizione

1 1 0 1

$$1 * (2^3) + 1 * (2^2) + 0 * (2^1) + 1 * (2^0) = 8+4+1 = 13_{10}$$

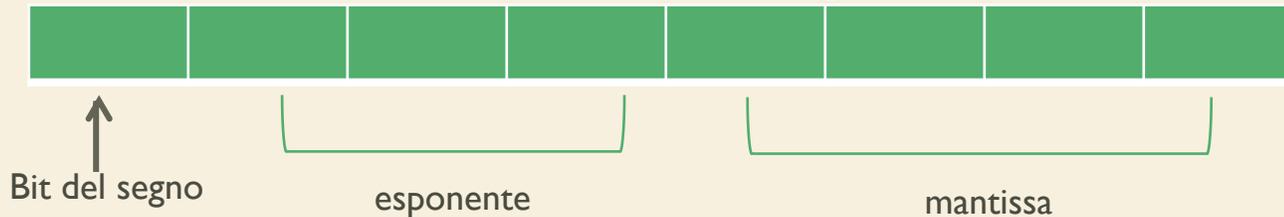
CONVERSIONI

- Per ottenere il valore binario di un numero intero si procede utilizzando successive divisioni per 2
 - Per esempio

$$\begin{array}{r} 26_{10} \\ 26:2 = 0 \\ 13:2 = 1 \\ 6:2 = 0 \\ 3:2 = 1 \\ 1:2 = 1 \end{array} \begin{array}{c} \uparrow \\ \text{Resto della} \\ \text{divisione per 2} \end{array} 11010_2$$

CONVERSIONI PER I NUMERI REALI

- Codifica in virgola mobile



Nel registro saranno prefissate zone diverse per la mantissa e per l'esponente

- Fissata la base B (es. 10 o 2)
- il numero viene rappresentato attraverso due altri numeri **mantissa** ed **esponente**
 - $N = m B^e$, con $B^{i-1} \leq m < B^i$ (i dipende dal numero di cifre da rappresentare)
 - es: $23.315 = 0.23315 \times 10^2$

$$10110.010 = 10.110010 \times 2^3$$

CONVERSIONI PER I NUMERI REALI

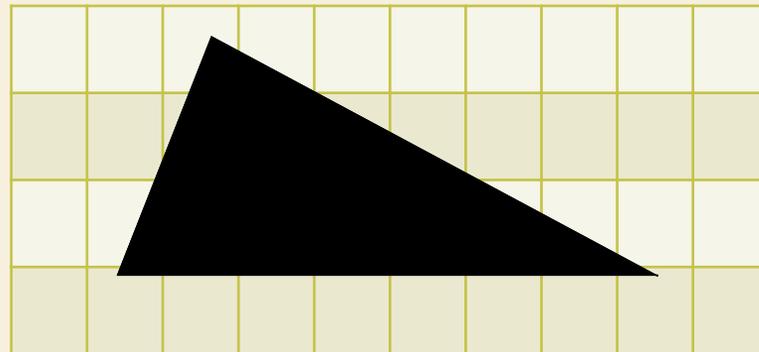
- Lo Standard per la presentazione di numeri reali in singola precisione
 - è utilizzato, ad esempio, per rappresentare il tipo `float` del C++ che occupa 4 byte (32 bit): 1 bit per il segno, 8 bit per l'esponente e 23 bit per la mantissa

CONVERSIONI

- Trasformare i seguenti numeri dal sistema decimale a quello binario:
 - 67
 - 124
 - 65
 - 79
- Verificare la conversione ritrasformando le stringhe di byte ottenute nuovamente nel sistema decimale

RAPPRESENTAZIONE DELLE IMMAGINI

- Supponiamo di avere un triangolo
- Suddividiamo l'immagine mediante una griglia formata da righe orizzontali e verticali a distanza costante



CODIFICA DELLE IMMAGINI

- Ogni cella derivante da tale suddivisione prende il nome di **PIXEL**
 - da “PICtureELeMent” identifica una piccola porzione rettangolare dello schermo
 - Rappresenta un'unità logica elementare di riferimento per la rappresentazione delle immagini digitali

CODIFICA DELLE IMMAGINI

- Ogni cella (pixel)
 - può essere codificata in binario secondo la seguente convenzione:
 - il simbolo 0 per le celle dove il bianco è predominante
 - il simbolo 1 per le celle in cui il nero è predominante

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

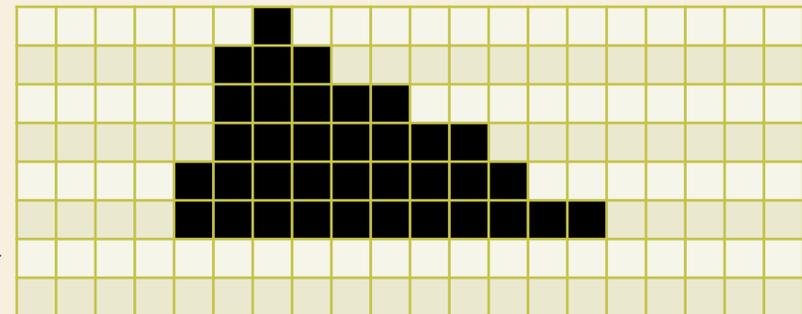
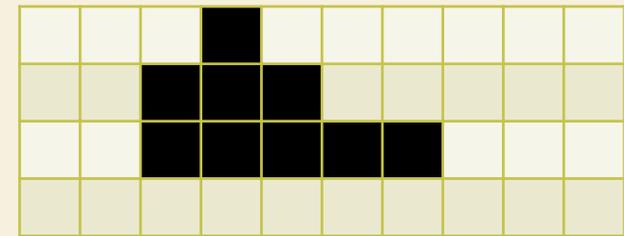
CODIFICA DELLE IMMAGINI

- Poiché una sequenza di bit è lineare,
- è necessario definire convenzioni per ordinare la griglia dei pixel in una sequenza
- Assumiamo che i pixel siano ordinati dal basso verso l'alto e da sinistra verso destra
- 0000000000 0011111000 0011100000 0001000000

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

CODIFICA DELLE IMMAGINI

- Non sempre il contorno della figura coincide con le linee della griglia
- nella codifica si ottiene un'approssimazione della figura originaria
- Se riconvertiamo la sequenza di stringhe
0000000000 00| | | | 000
00| | 00000 000|000000
in immagine otteniamo
- Aumentando il numero di pixel



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

CODIFICA DELLE IMMAGINI

- Per memorizzare un pixel non è più sufficiente un solo bit
 - Per esempio, se utilizziamo quattro bit possiamo rappresentare
 - $2^4 = 16$ livelli di grigio o 16 colori diversi
- Mentre con otto bit ne possiamo distinguere
 - $2^8 = 256$,
 - ecc.

CODIFICA DELLE IMMAGINI

- Il colore può essere generato componendo tre 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 blue e 4 per il verde che, combinate insieme, danno origine a 64 colori diversi
 - ogni pixel per essere memorizzato richiede 6 bit

CODIFICA DELLE IMMAGINI

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

RISOLUZIONE

- Il numero di pixel presenti sullo schermo (colonne x righe) prende il nome di **risoluzione**
- Risoluzioni Comuni Oggi:
 - **Full HD**: 1920x1080 - Standard per la maggior parte dei display e contenuti video.
 - **Quad HD (QHD)**: 2560x1440 - Comune nei monitor di fascia alta per computer.
 - **Ultra HD (UHD/4K)**: 3840x2160 - Predominante nei televisori e monitor ad alta definizione.
 - **5K e oltre: 5120x2880** - Usato nei monitor professionali per grafica e video editing.

RISOLUZIONE

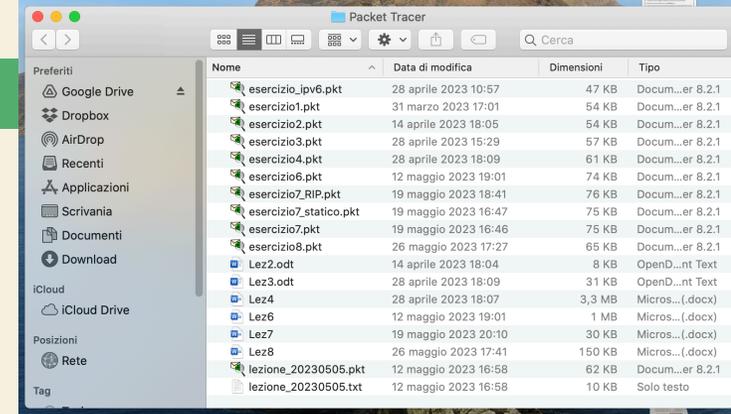
- La codifica di un'immagine di 256 colori, formata da 640 x 480 pixel,
 - richiederà $640 \times 480 = 307.200$ byte
- Codifica delle Immagini Moderna:
 - Una immagine Full HD (1920x1080) in JPEG con alta qualità può variare da 1 a 5 MB, a seconda del livello di compressione e del contenuto dell'immagine.
 - Dimensione file (byte) = numero di pixel x byte per pixel =
 $1920 \times 1080 \times 3 = 6.220.800$ byte ≈ 6.22 MB

Byte per pixel (3 byte/pixel): Ogni pixel ha bisogno di 3 byte per rappresentare un colore nel formato RGB.

RISOLUZIONE E DPI

- **Risoluzione**
 - indica le dimensioni dell'immagine espresse in pixel (es. 640x480)
- **DPI (dots per inch - punti per pollice)**
 - è un parametro relativo che, abbinato alla risoluzione, definisce le dimensioni dell'immagine in fase di acquisizione (tramite scanner) e di stampa

LE BITMAP



- Le immagini **bitmap** sono rappresentazioni digitali di immagini pixel per pixel
- Ogni pixel contiene informazioni di colore specifiche.
- Queste immagini possono richiedere notevoli quantità di spazio su disco a causa della loro rappresentazione dettagliata dei pixel.
 - Ad esempio, un'immagine Full HD (1920x1080) a colori completi (24 bit per pixel) può occupare circa 6.22 MB non compressi
- Ideali per l'editing foto dettagliato, la grafica per la stampa e applicazioni che richiedono precisione visiva.

LE BITMAP

- Maggiore è il numero di pixel e la profondità di colore, maggiore sarà la qualità e la dimensione del file dell'immagine.
- Tuttavia, tecniche di compressione avanzate possono ridurre efficacemente la dimensione del file mantenendo una qualità visiva accettabile.
- La compressione può essere lossless, preservando esattamente i dettagli originali (es. PNG), o lossy, dove alcuni dettagli possono essere persi per ridurre drasticamente la dimensione del file (es. JPEG).
 - Nella compressione JPEG, aree dell'immagine con colori simili possono essere compresse più efficacemente riducendo la variazione di colore tra i pixel vicini."

ORIGINAL
745 KB



LOSSLESS
676 KB



LOSSY
231 KB

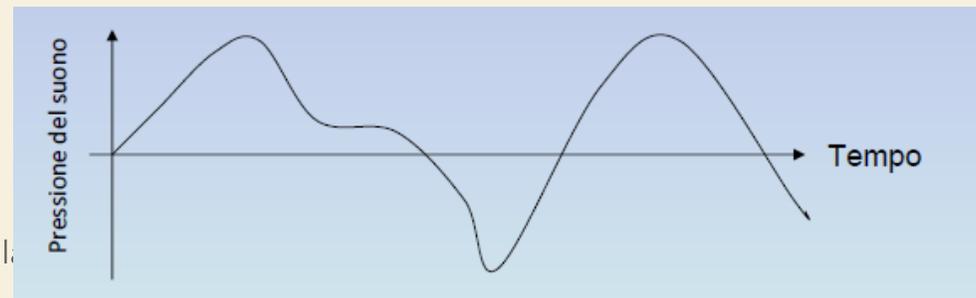


I VARI FORMATI

- I formati come GIF, JPEG e PNG sono formati compressi di immagini bitmap
- Per esempio: rispetto al bitmap, il formato JPEG dedica meno bit alla descrizione delle sfumature cromatiche di un'immagine
- Un'immagine screenshotata al PC, in formato JPEG occupa circa 850 KB

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 rappresentato il tempo
- e sull'asse delle ordinate viene rappresentata la variazione di pressione corrispondente al suono stesso

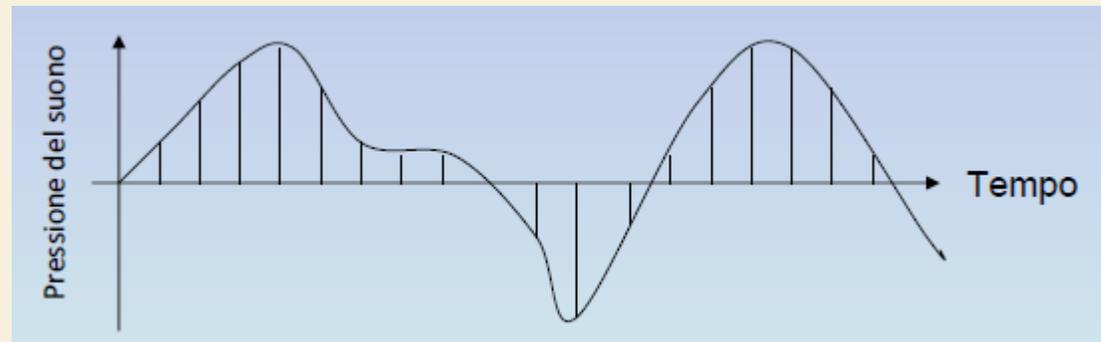


CODIFICA DEI SUONI

- Un'idea chiave: il suono è continuo (la pressione varia senza salti)
- È possibile ottenere una registrazione analogica continua dell'onda
- Una registrazione analogica memorizza ogni variazione dell'onda originale (almeno in linea di principio)
- La rappresentazione digitale di un suono funziona in un modo diverso

LA RAPPRESENTAZIONE DIGITALE

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

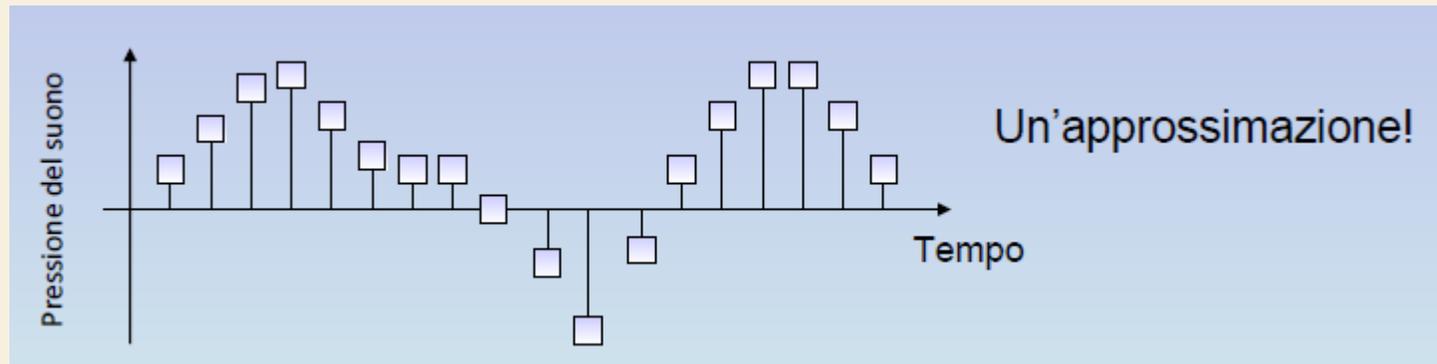


LA RAPPRESENTAZIONE DIGITALE

- (cont.) Quanto più frequentemente il valore di intensità 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, numero di campionamento ogni secondo)

LA RAPPRESENTAZIONE DIGITALE

- (cont.) La sequenza dei valori numerici ottenuti dai campioni può essere facilmente codificata con sequenze di bit



CODIFICHE STANDARD

- **CODIFICHE AUDIO STANDARD**

- **MP3**: Altamente diffuso per la musica digitale, bilancia efficacemente la compressione e la qualità audio.
- **WAV**: Formato non compresso ideale per produzioni professionali che richiedono la massima qualità sonora.
- **AIFF**: Simile a WAV, usato principalmente su dispositivi Apple, senza perdita di dati audio.
- **FLAC**: Codec audio lossless che riduce la dimensione dei file senza perdere informazioni sonore, perfetto per archivi audio di alta qualità.
- **AAC**: Oltrepassa il MP3 in efficienza, ampiamente utilizzato in streaming e dispositivi mobili per la sua superiore qualità a parità di bit rate.

CODIFICHE STANDARD

- **Tecnologie di Compressione:**

- **Lossy (MP3, AAC):** Comprime riducendo la qualità audio ma minimizzando il dimensionamento del file, adatto per uso generale e streaming.
- **Lossless (FLAC, WAV, AIFF):** Mantiene l'integrità del suono originale, ideale per l'editing audio e l'archiviazione a lungo termine.

- **MIDI:**

- **Funzione:** Non è un formato audio, ma un protocollo che registra e riproduce le azioni degli strumenti musicali in forma digitale, essenziale per la composizione musicale e la produzione digitale.

SOMMARIO

- Elementi di Base dell'ICT
 - Rappresentazione delle informazioni
 - Rappresentazione dei caratteri
 - Rappresentazione dei numeri
 - Rappresentazione di immagini e video

TERMINI DELLA LICENZA

- This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/1.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
- Questo lavoro viene concesso in uso secondo i termini della licenza “Attribution-ShareAlike” di Creative Commons. Per ottenere una copia della licenza, è possibile visitare <http://creativecommons.org/licenses/by-sa/1.0/> oppure inviare una lettera all'indirizzo Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.