



Introduzione ai Calcolatori Elettronici

2 Crediti Formativi (CFU)

**Corso di Laurea in Economia
Aziendale**

A.A. 2022/2023

Ing. Paola Lapadula
Università degli Studi della Basilicata



Sommario

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



Rappresentazione delle Informazioni

- In un calcolatore, le informazioni sono codificate in forma binaria - sistema binario
 - ossia in una sequenza finita di 0 e di 1
- La più piccola unità di informazione memorizzabile o elaborabile da un calcolatore è il bit (binary digit) → 8 bit = 1 Byte
- L'alfabeto con cui un calcolatore **codifica** ogni informazione consiste di due soli simboli 0 e 1



Rappresentazione delle Informazioni

- Oltre al sistema binario, in base 2, esistono altri sistemi
 - Sistema esadecimale è un sistema numerico posizionale in base 16
 - utilizza 16 simboli invece dei 10 del sistema numerico tradizionale
 - si usano simboli da 0 a 9 per le prime dieci cifre, e poi le lettere da A a F per le successive sei cifre, per un totale di 16 simboli



Rappresentazione delle Informazioni

- Manipolare bit significa manipolare valori vero/falso
 - bit 0: valore falso
 - bit 1: valore vero
- Per mettere in relazione le istruzioni contenute nei linguaggi di program., si utilizzano i **connettivi logici** o **operatori booleani**



Gli operatori booleani

- Le Operazioni booleane (mat. George Boole 1815-1864) e operatori combinano due valori di ingresso (input) per produrre un terzo valore in uscita (output)
- Ciascun operatore prende in input uno o due booleani e restituisce in output un altro booleano



A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

Gli operatori booleani

- Gli operatori più importanti sono: AND, OR e NOT
- **AND**, prende in input due operandi e produce in output un booleano, attenendosi al seguente comportamento:
 - Se entrambi gli operandi sono true allora l'output è true; in tutti gli altri casi l'output è uguale a false



A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

Gli operatori booleani

- (cont.) **OR**, prende in input due operandi e produce in output un booleano, attenendosi al seguente comportamento
 - Se almeno uno degli operandi è uguale a true, l'output è true;
 - altrimenti, se nessuno dei due operandi è uguale a true l'output sarà false



Gli operatori booleani

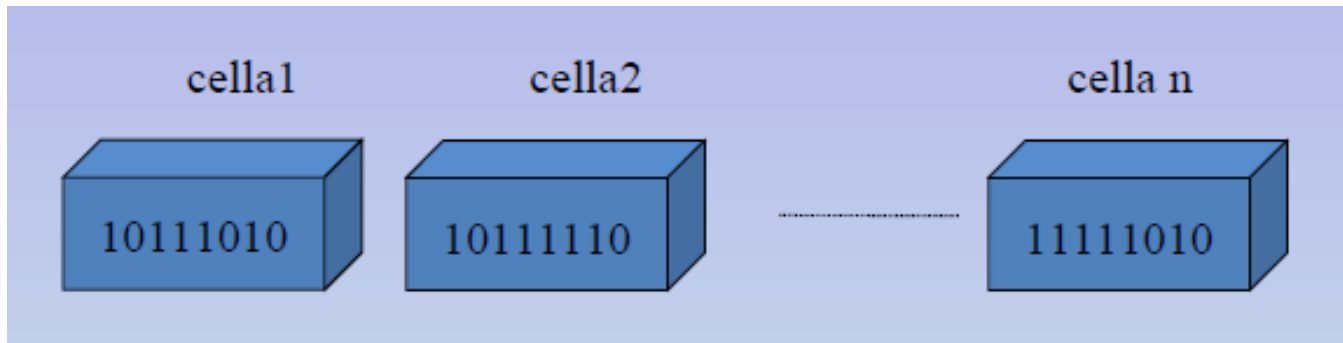
A	Not A
0	1
1	0

- **NOT**, prende in input un solo operando e produce in output un booleano, attenendosi al seguente comportamento:
 - Se l'operando di input è true allora l'output sarà false
 - Se, invece l'operando di input è false, allora l'output sarà uguale a true
 - In altri termini, l'operatore di NOT prende un input e ne restituisce l'esatto contrario



Memorizzazione e rappresentazione dei dati

- Per archiviare i dati, un computer contiene un gran numero di circuiti flip-flop,
- ognuno in grado di memorizzare un singolo bit
- I circuiti di memoria sono organizzati in unità chiamate celle di memoria





Memorizzazione e rappresentazione dei dati

- Per identificare le singole celle di memoria ad ognuna di esse è assegnata una denominazione univoca chiamata **indirizzo**
- Poiché la memoria principale di un computer è organizzata sotto forma di singole celle dotate di indirizzo, è possibile accedere a tali celle a secondo della necessità



Codifica delle Informazioni

- Analizziamo il metodo con cui è possibile codificare le informazioni tramite bit: testi, immagini, suoni
- Le informazioni per poter essere interpretate dal calcolatore devono essere “trasformate” in sequenze di bit
 - La “trasformazione” viene detta CODIFICA
- Codificare: trasformare un messaggio di un alfabeto in un messaggio di un altro alfabeto mediante l'applicazione di opportune regole dette regole di codifica



Codifica delle Informazioni

- Esistono diverse convenzioni per codificare i diversi tipi di dati
- Nello specifico... processo secondo cui si fa corrispondere ad un'informazione una configurazione di cifre binarie
- Con un certo numero di bit possiamo codificare un certo numero di informazioni



Codifica delle Informazioni

- Con una sequenza di n bit si possono rappresentare 2^n informazioni diverse

$n = 1 \longrightarrow 0, 1$

$n = 2 \longrightarrow 00, 01, 10, 11$

$n = 3 \longrightarrow 000, 001, 010, 011, 100, 101, 110, 111$

- Con 8 bit si possono rappresentare 256 informazioni:

- $2^8 = 256$

- Si usano tutte le combinazioni possibili tra 00000000 e 11111111



Unità di misura

- BIT (binary digit)
- BYTE = 8 bit
- 1Kb (kilobit) 1024 bit
- 1KB (kilobyte) 1024 byte (2^{10} byte)
- 1Mb (megabit) 1024 Kb
- 1MB (megabyte) 1024 KB (2^{20} byte)
- 1GB (gigabyte) 1024 MB
- 1TB (terabyte) 1024 GB
- 1PB (petabyte) 1024 TB



Rappresentazione dei caratteri

- Codice ASCII (American Standard Code for Information Interchange)
 - codice di traduzione per i simboli alfanumerici
 - ad ogni carattere è associato un numero
 - la rappresentazione del carattere coincide con quella del numero
 - vale anche per le cifre (0-9) ed i caratteri speciali (es: *, #, \$)



Rappresentazione dei caratteri

Tabella dei Codici ASCII Estesi

000	NUL	033	!	066	B	099	c	132	ä	165	Ñ	198	ã	231	þ
001	Start Of Header	034	"	067	C	100	d	133	å	166	ª	199	Ä	232	ß
002	Start Of Text	035	#	068	D	101	e	134	ä	167	º	200	Å	233	Û
003	End Of Text	036	\$	069	E	102	f	135	ç	168	¸	201	Æ	234	Ü
004	End Of Transmission	037	%	070	F	103	g	136	è	169	©	202	Ç	235	Ù
005	Enquiry	038	&	071	G	104	h	137	é	170	¬	203	Ð	236	Ý
006	Acknowledge	039		072	H	105	i	138	è	171	½	204	É	237	ÿ
007	Bell	040	(073	I	106	j	139	ï	172	¼	205	Ê	238	–
008	Backspace	041)	074	J	107	k	140	î	173	¸	206	Ë	239	·
009	Horizontal Tab	042	*	075	K	108	l	141	ì	174	«	207	Ì	240	-
010	Line Feed	043	+	076	L	109	m	142	Á	175	»	208	Í	241	±
011	Vertical Tab	044	,	077	M	110	n	143	Â	176	∴	209	Î	242	_
012	Form Feed	045	-	078	N	111	o	144	É	177	∑	210	Ï	243	¼
013	Carriage Return	046	.	079	O	112	p	145	Ê	178	⊞	211	Ï	244	¶
014	Shift Out	047	/	080	P	113	q	146	Æ	179		212	È	245	§
015	Shift In	048	0	081	Q	114	r	147	ø	180	†	213	É	246	÷
016	Delete	049	1	082	R	115	s	148	ö	181	À	214	Í	247	,
017	-- frei --	050	2	083	S	116	t	149	ò	182	Á	215	Î	248	°
018	-- frei --	051	3	084	T	117	u	150	ù	183	Â	216	Ï	249	ˆ
019	-- frei --	052	4	085	U	118	v	151	û	184	Ë	217	¸	250	˙
020	-- frei --	053	5	086	V	119	w	152	ÿ	185	Ë	218	¸	251	¸
021	Negative Acknowledge	054	6	087	W	120	x	153	Ö	186	Ë	219	■	252	¸
022	Synchronous Idle	055	7	088	X	121	y	154	Ü	187	¸	220	■	253	¸
023	End Of Transmission Block	056	8	089	Y	122	z	155	ø	188	¸	221	¸	254	■
024	Cancel	057	9	090	Z	123	{	156	£	189	¢	222	¸	255	
025	End Of Medium	058	:	091	[124		157	∅	190	¥	223	■		
026	Substitute	059	;	092	\	125	}	158	×	191	¸	224	Ó		
027	Escape	060	<	093]	126	~	159	f	192	¸	225	Ô		
028	File Separator	061	=	094	^	127	¸	160	á	193	¸	226	Õ		
029	Group Separator	062	>	095	_	128	Ç	161	í	194	¸	227	Ö		
030	Record Separator	063	?	096	`	129	ü	162	ó	195	¸	228	Ø		
031	Unit Separator	064	@	097	a	130	é	163	ú	196	–	229	Ö		
032		065	A	098	b	131	â	164	ñ	197	+	230	µ		



Rappresentazione dei caratteri

- Il codice ASCII a 7 bit è limitato:
 - non include lettere accentate
- Codice ASCII a 8 bit ("ASCII esteso"):
 - al carattere è associato un num. tra 0 e 255
maggior numero di caratteri
rappresentabili
- Al giorno d'oggi codice UNICODE (16 bit o superiore)



Rappresentazione dei caratteri

- **Unicode** è un sistema per rappresentare i caratteri di tutti i differenti linguaggi del mondo
- Il codice assegnato al carattere viene rappresentato con U+, seguito dalle quattro (o sei) cifre esadecimali del numero che lo individua
- Es: Il carattere **a** ha codice Unicode **U+0061**



Rappresentazione dei caratteri

Ricapitolando

- Codice ASCII (American Standard Code for Information Interchange)
 - ad ogni carattere è associato un numero
 - Codice ASCII a 7 bit
 - Codice ASCII a 8 bit ("ASCII esteso")
- Unicode a 16 bit o superiore
 - viene supportato dai moderni standard della programmazione e del markup come XML, Java, JavaScript, LDAP, CORBA 3.0, e da vari sistemi operativi



I File

- Il file è un contenitore di informazione digitalizzata
- Le informazioni codificate al suo interno sono leggibili solo da software
- I dati codificati in un file sono organizzati come una sequenza di byte, immagazzinati come un solo elemento su una memoria di massa



I File

- Immaginate un file come un foglietto a quadretti nel quale scrivete i codici corrispondenti alle lettere che voi digitate dalla tastiera:
- Scrivo

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

- Ad esempio per scrivere C scrivo 01000011,
per scrivere A scrivo 01000001 ...



I File

- Ogni programma salva il file in un determinato formato identificato univocamente dall'estensione del programma
- Il nome del file è del tipo:
 - nome . estensione
 - Estensione e icona identificano il programma
 - Es: foto.gif, documento.doc, calcolo.xls



I File

- Alcune estensioni molto note:
 - ZIP/TAR -> Winzip
 - XLSX -> Excel
 - PPT -> Power Point -> Presentazioni multimediali
 - MDB -> Access -> DataBase
 - EXE -> File eseguibili
 - MP3 -> Winamp / Windows Media Player -> File musicale
 - HTML -> Browser -> File Internet Pagina web



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
1101

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



Conversioni

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

26_{10}	$26:2 = 0$	 Resto della divisione per 2	11010_2
	$13:2 = 1$		
	$6:2 = 0$		
	$3:2 = 1$		
	$1:2 = 1$		



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 $B^{i-1} \leq m < B^1$ (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



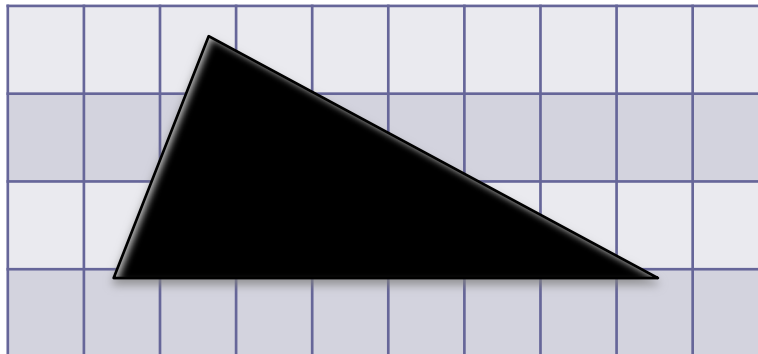
Conversioni

- 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

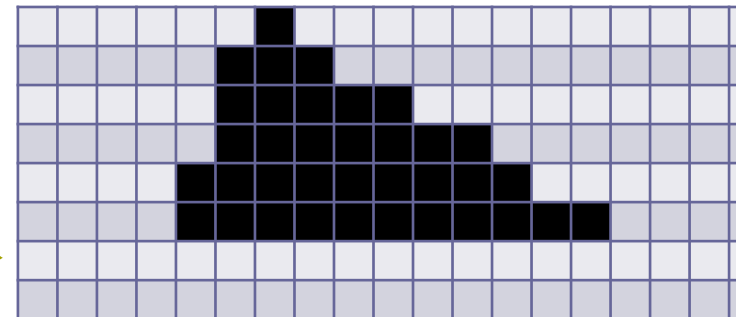
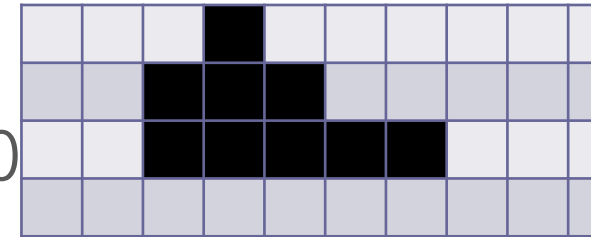
■ 0000000000 0011111000 0011100000
0001000000

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 0011111000`
`0011100000 0001000000` 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**
- Risoluzione tipiche sono
 - 640x480, 800x600, 1024x768, 1280x1024, 1280x960, 1600x1200
- La codifica di un'immagine di 256 colori, formata da 640 x 480 pixel,
 - richiederà $640 \times 480 = 307.200$ byte



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



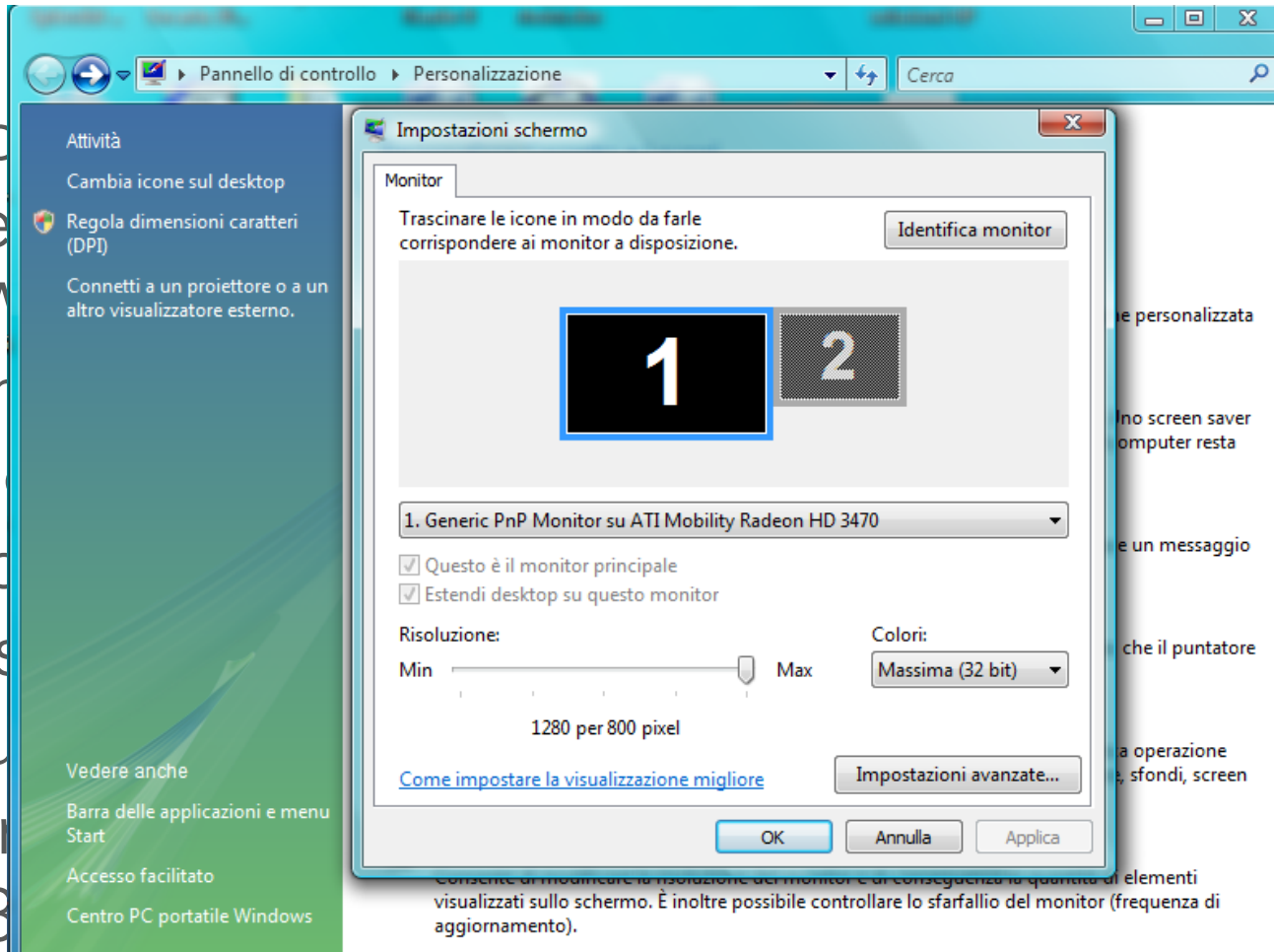
Risoluzione e Monitor

- Per modificare i numeri di colori (il numero di pixel sullo schermo):
 - (SO Windows) Start, Pannello di controllo, Schermo, Impostazione (scheda)
- Un colore sul monitor è specificato da tre componenti: rosso, verde e blu
- l'intensità di ogni colore è rappresentata da una quantità (da 0 a 255)
- Ogni intensità RGB è rappresentata da un byte (8 bit)



Risoluzione e Monitor

- Per mo
- di pixe
- (SO W
- Scher
- Un col
- compo
- l'intens
- una qu
- Ogni in
- byte (8



Personalizzata
No screen saver
computer resta
e un messaggio
che il puntatore
a operazione
e sfondi, screen



Le bitmap

- Le immagini codificate pixel per pixel sono dette immagini **bitmap**
- Le immagini di questo tipo occupano molto spazio
- Per esempio, l'immagine della finestra "Proprietà - Schermo" sulla diapositiva precedente (in cui 3 byte sono usati per i colori) richiede 493.674 byte (circa 482 KB)



Le bitmap

- Maggiore è il numero di pixel maggiore è la qualità dell'img, maggiore è la dimensione
- Esistono delle tecniche di compressione che permettono di ridurre 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



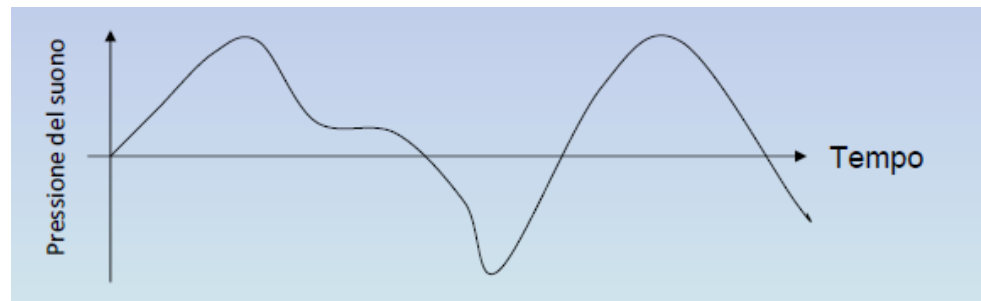
I vari formati

- 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
- L'immagine della finestra "Proprietà - Schermo" in formato JPEG occupa 30.401 byte (circa 29,6 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





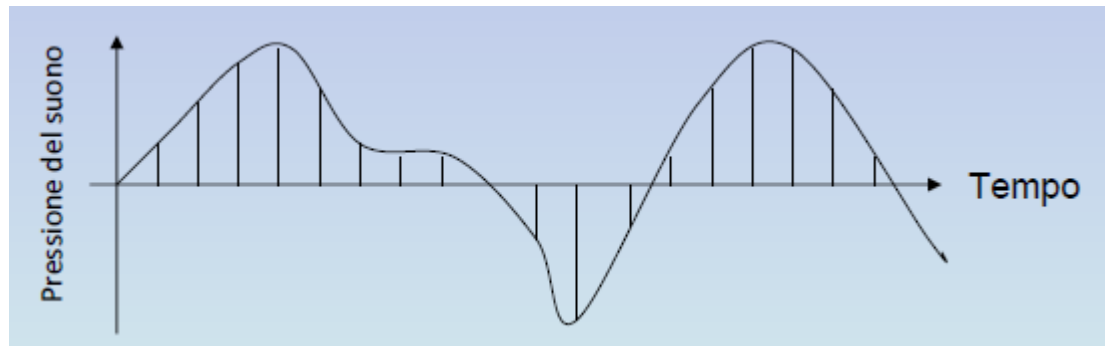
Codifica dei suoni

- 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





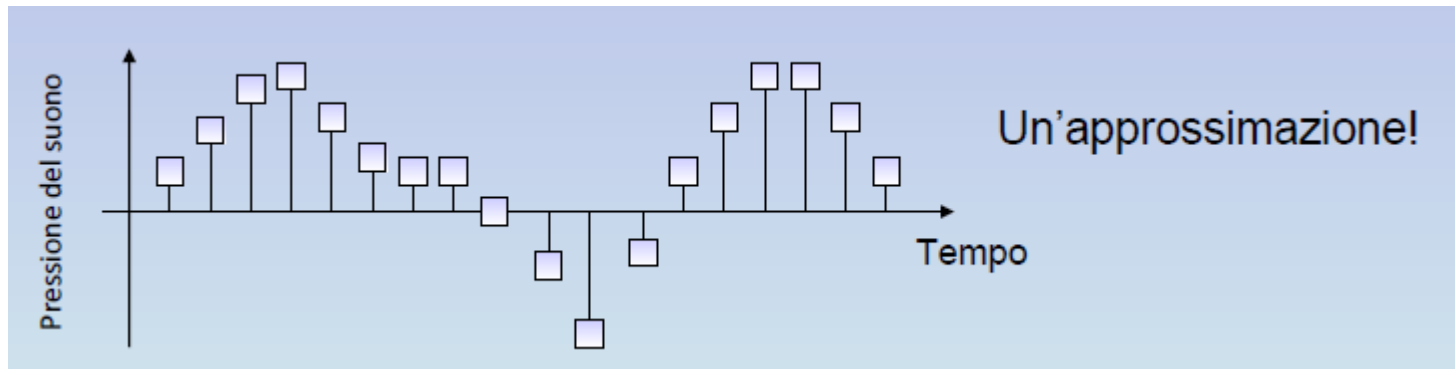
La rappresentazione digitale

- (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 standard
 - MP3, WAV (MS-Windows), AIFF (Audio Interchange File Format, Apple)
- MPEG-3:
 - variante MPEG per suoni
 - Grande diffusione, molto efficiente (fattore di compressione circa 10:1)
- MIDI:
 - codifica le note e gli strumenti che devono eseguirle
 - Efficiente, ma solo musica, non voce



Sommario

- Elementi di Base dell'ICT
 - Rappresentazione delle informazioni
 - Rappresentazione dei caratteri
 - Rappresentazione dei numeri
 - Rappresentazione di immagini e video
- Parte del materiale di questa lezione è stato sviluppato a partire dalle lezioni della Dott.ssa Domenica Sileo.



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.