

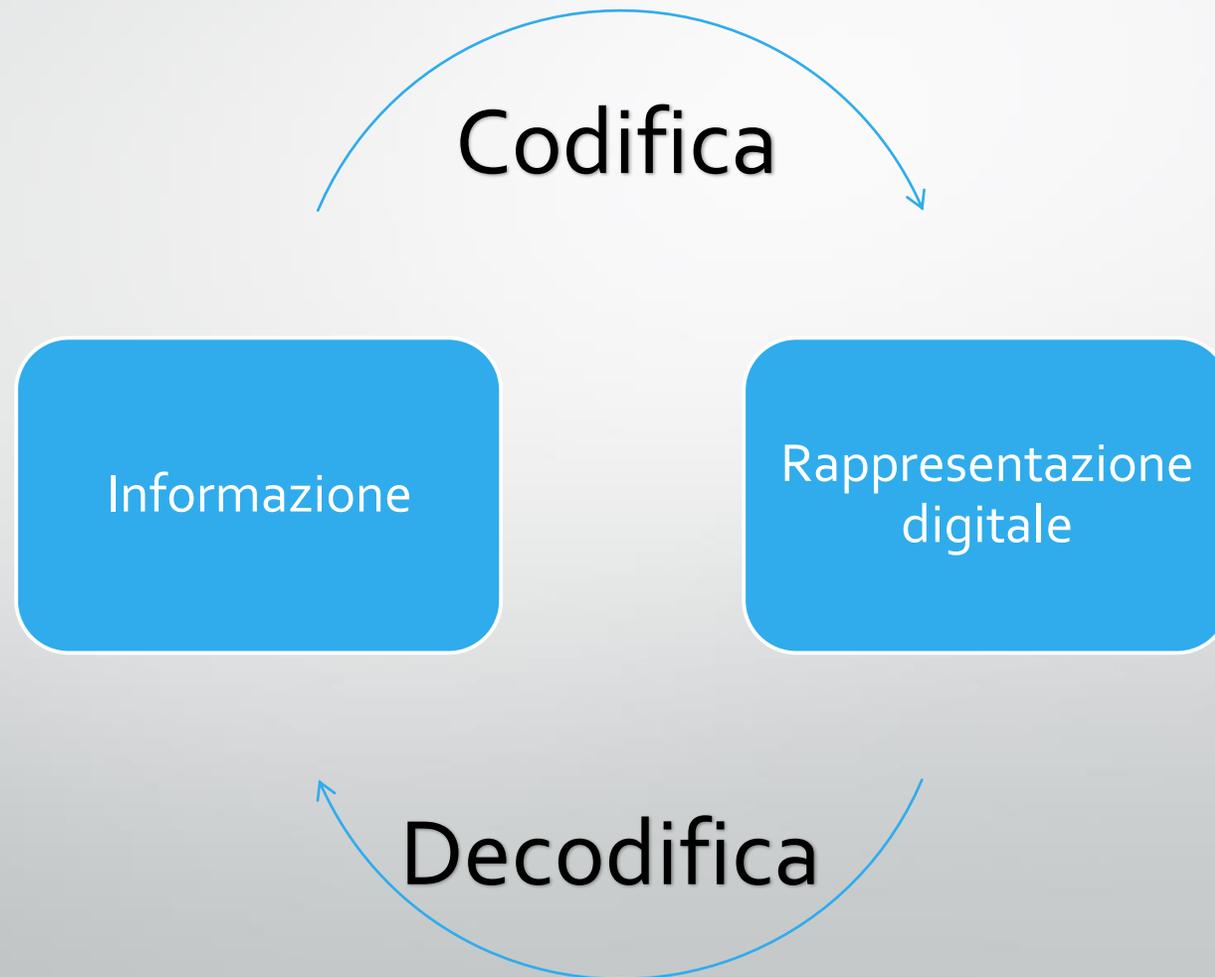


SISTEMA NUMERICO BINARIO

INFORMAZIONI: tipi

- Le informazioni sono concetti astratti che esistono indipendentemente dalla loro rappresentazione
- Tutto ciò che ci circonda è informazione
- Qualche esempio:
 - Numeri
 - Testi
 - Immagini
 - Suoni
 - Video

INFORMAZIONI



CODIFICA BINARIA: BIT

- Usiamo una rappresentazione binaria:
 - Presenza/assenza di carica elettrica
 - Passaggio/non passaggio di corrente
- BIT: BInary digiT
 - La più piccola informazione che possiamo memorizzare
 - Sì/no – 0/1 – on/off – vero/falso

CODIFICA BINARIA: BIT

- BIT: Binary digiT
 - La più piccola informazione che possiamo memorizzare
 - Si/no – 0/1 – on/off – vero/falso
- Quante informazioni possiamo rappresentare con 1 bit?
 - Solo 2, come abbiamo appena visto!
 - 0 / 1

CODIFICA BINARIA: BIT

- Possiamo combinare più bit per rappresentare più informazioni.
- Quante informazioni possiamo rappresentare con 2 bit?
 - 4: 00 / 01 / 10 / 11
- E con 3 bit?
 - 8: 000 / 001 / 010 / 011 / 100 / 101 / 110 / 111
- Regola generale: con N bit possiamo rappresentare 2^N informazioni

CODIFICA BINARIA: BYTE

- In informatica ha assunto particolare importanza il concetto di BYTE
- 1 byte = 8 bit = 2^8 informazioni = 256 informazioni
- BYTE = BinarY ocTEt
- Perché proprio 8 bit? Semplicemente perché per l'indirizzamento della memoria si utilizza la notazione esadecimale...che è compatibile con la notazione binaria

CODIFICA BINARIA: BYTE

MULTIPLI DEL BYTE:

- Nel Sistema Internazionale
 - kB = kilobyte = 10^3 byte = 1000 byte
 - MB = megabyte = 10^6 byte = 1000 kB
 - GB = gibabyte = 10^9 byte = 1000 MB
 - TB = terabyte = 10^{12} byte = 1000 GB
 - PB = petabyte = 10^{15} byte = 1000 TB

CODIFICA BINARIA: BYTE

MULTIPLI DEL BYTE:

- Utilizzando i prefissi binari

- kiB = kibibyte = 2^{10} byte = 1024 byte
- MiB = mebibyte = 2^{20} byte = 2^{10} kB = $1024 * 1024$ byte
- GiB = gibibyte = 2^{30} byte = 2^{10} MB = $1024 * 1024 * 1024$ byte
- TiB = tebibyte = 2^{40} byte = 2^{10} GB = $1024 * 1024 * 1024 * 1024$ byte
- PiB = pebibyte = 2^{50} byte = 2^{10} TB = $1024 * 1024 * 1024 * 1024 * 1024$ byte

CODIFICA BINARIA: Numeri

- Come codifichiamo un numero in binario?
- Effettuando la conversione tra il sistema decimale e quello binario:

$$12_{10} = 1100_2 = C_{16}$$

- Da dove esce la lettera C?

CODIFICA BINARIA: Numeri

decimal	hexadecimal	binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

CODIFICA BINARIA: Testi

- Come codifichiamo una parola utilizzando un sistema binario?
- Si utilizza la tabella ASCII :
 - American Standard Code for Information Interchange

CODIFICA BINARIA: tabella ASCII

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

CODIFICA BINARIA: tabella ASCII

- Ad esempio, la parola INFORMATICA si traduce in:

	I	N	F	O	R	M	A
Hx	49	4E	46	4F	52	4D	41
Bn	01001001	01001110	01000110	01001111	01010010	01001110	01000001
	T	I	C	A			
Hx	54	49	43	41			
Bn	01010100	01001001	01000011	01000001			



IMMAGINI DIGITALI

Nozioni di base

IMMAGINI DIGITALI

- Un'immagine è una rappresentazione visiva, non solida, della realtà.
- Può rappresentare la realtà fisica (in modo più o meno realistico) oppure una realtà fittizia o astratta.
- Un'immagine GRAFICA è un'immagine fissata su un supporto fisico oppure visualizzata (in modo più o meno prolungato nel tempo) su uno schermo di visualizzazione.

IMMAGINI DIGITALI

- Perché le immagini siano elaborate al computer occorre trasformarle in una rappresentazione numerica/digitale, attraverso un processo chiamato **DIGITALIZZAZIONE**.
- Un'immagine digitale è la rappresentazione numerica di una immagine bidimensionale.
- La rappresentazione può essere di tipo **RASTER** o **VETTORIALE**.
 - **RASTER**: l'immagine è composta da una matrice di punti, detti pixel, la cui colorazione è definita (codificata) tramite uno o più valori numerici (bit).
 - **VETTORIALE**: sono descritti degli elementi primitivi, quali linee o poligoni, che vanno a comporre l'immagine

IMMAGINI DIGITALI

IMMAGINE RASTER o BITMAP:

- nelle immagini a COLORI, viene memorizzato solitamente il livello di intensità dei colori fondamentali (nel modello di colore RGB, uno dei più usati, sono tre: rosso, verde e blu. Un altro esempio è CMYK, usato per la stampa, basato su quattro colori fondamentali: ciano, magenta, giallo e nero.)
- nelle immagini MONOCROMATICHE in scala di grigio (dette impropriamente bianco e nero) il valore indica l'intensità del grigio, che varia dal nero al bianco.

IMMAGINI DIGITALI

IMMAGINE RASTER o BITMAP:



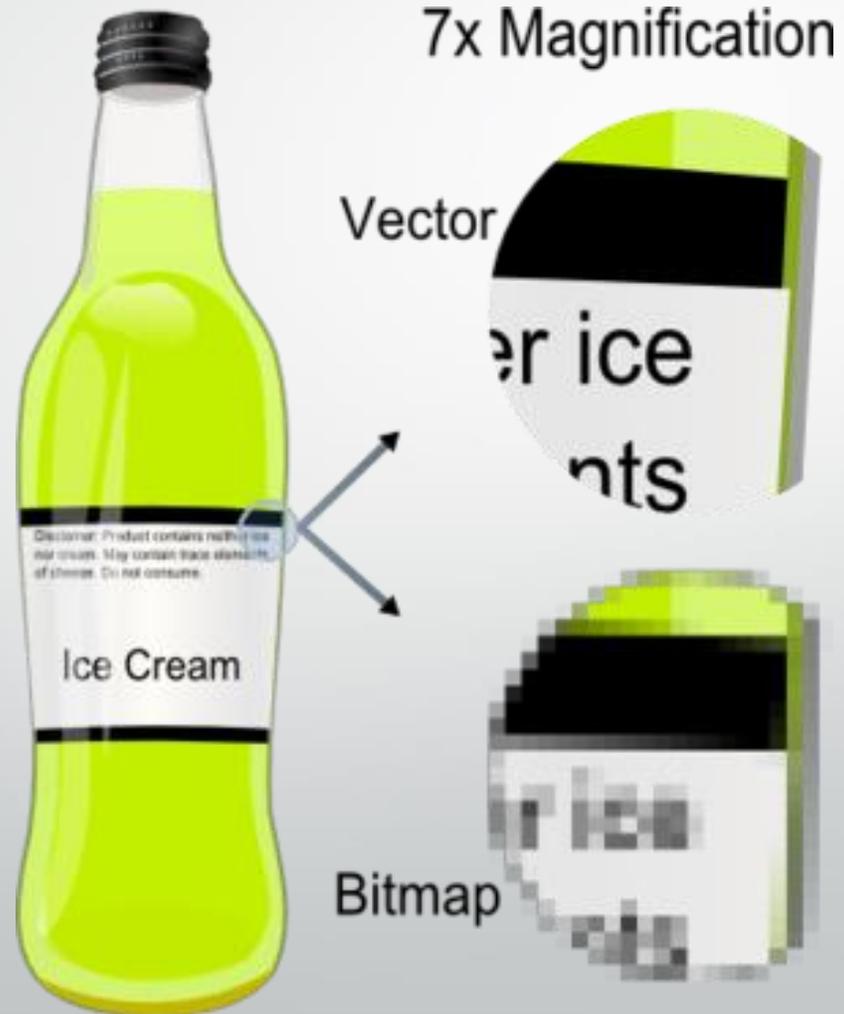
IMMAGINI DIGITALI

IMMAGINE RASTER o BITMAP:

- Le immagini BITMAP possono essere memorizzate in vari formati, basati su algoritmi di compressione, che possono essere:
 - LOSSY – con perdita
 - Bitmap – Jpeg
 - LOSSLESS – senza perdita
 - GIF – PNG

IMMAGINI DIGITALI

IMMAGINE VETTORIALE:



IMMAGINI DIGITALI

IMMAGINE VETTORIALE:

- Le immagini vettoriali sono un tipo di immagine facilmente scalabile e ruotabile, ottenuto dall'unione di un certo numero di punti o nodi, che formano linee e poligoni, a loro volta uniti in strutture più complesse, fino a formare l'immagine voluta.
- Questo tipo di immagine è utilizzato nel disegno tecnico per la progettazione architettonica ed industriale, nella rappresentazione di certi font, nella grafica per la creazione di loghi e marchi o altri oggetti, eccetera.

IMMAGINI DIGITALI

IMMAGINE VETTORIALE:

- Vantaggi:
 - Possibilità di esprimere i dati in una forma direttamente comprensibile ad un essere umano
 - Possibilità di esprimere i dati in un formato che occupi (molto) meno spazio rispetto all'equivalente raster
 - Possibilità di ingrandire l'immagine arbitrariamente, senza che si verifichi una perdita di risoluzione dell'immagine stessa
- Svantaggi:
 - Difficoltà di utilizzo dei software

IMMAGINI DIGITALI

PRINCIPALI FORMATI:

- Analogici
 - 333×480 : VHS
 - 580×480 : Super VHS, LaserDisc
 - 768×576 : Trasmissioni televisive (PAL)
- Digitali
 - 720×480 : DVD
 - 1280×720 : Blu-ray
 - 1920×1080 : FullHD, Blu-ray
 - 3840×2160 : 4K UHD TV

CODIFICA BINARIA: Conversione

CONVERSIONE IN BINARIO PARTENDO DA BASE 10

- Si utilizza il metodo delle divisioni successive:
 - Si prende il numero da convertire e lo si divide per la base (2)
 - Si prende il resto della divisione
 - Si continua a dividere il quoziente ottenuto per la base fino ad ottenere il valore zero
 - Il numero in base 2 è la successione dei resti ottenuti partendo dall'ultimo

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 23 (base 10)

DIVIDENDO	DIVISORE	QUOZIENTE INTERO	RESTO
23	2	11	1
11	2	5	1
5	2	2	1
2	2	1	0
1	2	0	1

La rappresentazione binaria è 10111

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 164

DIVIDENDO	DIVISORE	QUOZIENTE INTERO	RESTO
164	2	82	0
82	2	41	0
41	2	20	1
20	2	10	0
10	2	5	0
5	2	2	1
2	2	1	0
1	2	0	1

La rappresentazione binaria è: 10100100

CODIFICA BINARIA: Conversione

CONVERSIONE IN ESADECIMALE

- Si utilizza il metodo delle divisioni successive:
 - Si prende il numero da convertire e lo si divide per la base (16)
 - Si prende il resto della divisione
 - Si continua a dividere il quoziente ottenuto per la base fino ad ottenere il valore zero
 - Il numero in base 16 è la successione dei resti ottenuti partendo dall'ultimo

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 23

DIVIDENDO	DIVISORE	QUOZIENTE INTERO	RESTO
23	16	1	7
1	16	0	1

La rappresentazione esadecimale è: 17

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 16_4

DIVIDENDO	DIVISORE	QUOZIENTE INTERO	RESTO	CODICE ESADECIMALE
16_4	16	10	4	4
10	16	0	10	A

La rappresentazione esadecimale è: A_4

CODIFICA BINARIA: Conversione

CONVERSIONE IN BASE 10 A PARTIRE DA QUALSIASI BASE

- Si utilizza il metodo della notazione posizionale:
 - Partendo da destra si associa ad ogni cifra binaria la potenza corrispondente:
Si inizia SEMPRE con potenza ZERO e come BASE quella di partenza (2 – 16)
 - Si calcolano le potenze
 - Si moltiplicano le potenze con le cifre binarie
 - Si sommano i numeri ottenuti

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 10111 (base 2)

Bn	1	0	1	1	1
Potenze	2^4	2^3	2^2	2^1	2^0
Valore	16	8	4	2	1
Moltip.	$16*1$	$8*0$	$4*1$	$2*1$	$1*1$
Somma	$16+4+2+1 = 23$				

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 10100100 (base 2)

Bn	1	0	1	0	0	1	0	0
Potenze	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Valore	128	64	32	16	8	4	2	1
Moltip.	$128*1$	$64*0$	$32*1$	$16*0$	$8*0$	$4*1$	$2*0$	$1*0$
Somma	$128+32+4 = 164$							

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 17 (base 16)

HEX	1	7
Potenze	16^1	16^0
Valore	16	1
Moltip.	$16*1$	$1*7$
Somma	$16+7 = 23$	

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero A_4 (base 16)

HEX	A	4
Potenze	16^1	16^0
Valore	16	1
Moltip.	$16 * A = 16 * 10$	$1 * 4$
Somma	$160 + 4 = 164$	

CODIFICA BINARIA: Conversione

CONVERSIONE TRA BASE 2 E 16

Come già detto queste 2 basi sono compatibili:

- 16 è una potenza di 2: $16 = 2^4$

DA BINARIO A ESADECIMALE:

- partendo da destra possiamo sostituire ogni 4 cifre binarie con una esadecimale

DA ESADECIMALE A BINARIO:

- Sempre partendo da destra sostituiamo ogni cifra esadecimale con 4 cifre binarie

CODIFICA BINARIA: Conversione

ESEMPIO: consideriamo il numero 10111 (base 2)

BN	0001	0111
HEX	1	7

ESEMPIO: consideriamo il numero 10100100 (base 2)

BN	1010	0100
HEX	A	4

CODIFICA BINARIA: Operazioni aritmetiche

ADDIZIONE IN BINARIO

- Si effettua esattamente come l'addizione in base 10

SOTTRAZIONE IN BINARIO

- Si effettua esattamente come la sottrazione in base 10

CODIFICA BINARIA: Operazioni aritmetiche

Esempio ADDIZIONE: 10100100 + 10111

Primo addendo	1	0	1	0	0	1	0	0
Secondo addendo	0	0	0	1	0	1	1	1
Riporto					1			
Somma	1	0	1	1	1	0	1	1

CODIFICA BINARIA: Operazioni aritmetiche

Esempio SOTTRAZIONE: 10100100 - 10111

Minuendo	1	0	1	0	0	1	0	0
Sottraendo	0	0	0	1	0	1	1	1
Differenza								

CODIFICA BINARIA: Operazioni aritmetiche

Esempio SOTTRAZIONE: 10100100 - 10111

Prestito						0	1	10
Minuendo	1	0	1	0	0	1	0	0
Sottraendo	0	0	0	1	0	1	1	1
Differenza							0	1

CODIFICA BINARIA: Operazioni aritmetiche

Esempio SOTTRAZIONE: 10100100 - 10111

Prestito			0	1	1	10	0	1	10
Minuendo	1	0	1	0	0	1	0	0	
Sottraendo	0	0	0	1	0	1	1	1	
Differenza				0	1	1	0	1	

CODIFICA BINARIA: Operazioni aritmetiche

Esempio SOTTRAZIONE: 10100100 - 10111

Prestito			0	1	1	10	0	1	10
Minuendo	1	0	1	0	0	1	0	0	
Sottraendo	0	0	0	1	0	1	1	1	
Differenza	1	0	0	0	1	1	0	1	