

Abilità Informatiche

Luigi Catuogno

[luigi.catuogno@uniparthenope.it]

Corso di Laurea in Economia e Commercio - Anno Accademico 2022-23

1

Libro di testo

[IdB]

Dennis P. Curtin, Kim Foley, Kunal Sen, Cathleen Morin

Informatica di base

VII edizione (2016), MacGraw Hill Education

ISBN: 978-88-386-1537-5

2

Altro materiale di utile consultazione

[Sli]

Slides, appunti e altro materiale distribuito dal docente

[Misc]

Altra fonte diversamente specificata di volta in volta

3

Codifica dell'informazione
non numerica

4

Rappresentazione del testo

5

Rappresentazione del Testo

Il testo è probabilmente l'informazione più diffusa e trattata dai calcolatori elettronici.

L	a		c	a	s	a

6

Rappresentazione del Testo

Un testo è rappresentato con una sequenza (stringa) di simboli alfanumerici ad ognuno dei quali è associato un codice binario in base a uno standard prefissato.

L	a		c	a	s	a
01001100	01100001	00100000	01100011	01100001	01110011	01100001

7

Rappresentazione del Testo

Dispositivi di input per l'informazione testuale traducono l'azione ottica o meccanica per l'acquisizione dei simboli alfanumerici nei codici corrispondenti.



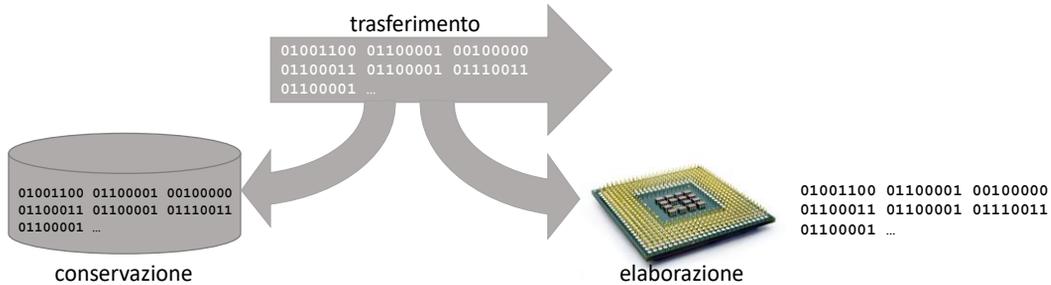
codifica →

```
01001100 01100001 00100000
01100011 01100001 01110011
01100001 ...
```

8

Rappresentazione del Testo

L'informazione testuale è trasferita, conservata ed elaborata in questa forma.



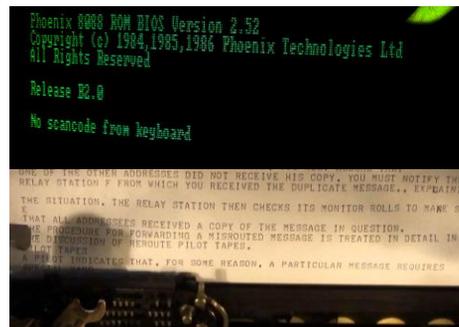
9

Rappresentazione del Testo

Dispositivi di output traducono il codice che rappresenta ciascun carattere ricevuto e ne «riproducono l'aspetto», ciascuno sul proprio «mezzo».

```
01001100 01100001 00100000
01100011 01100001 01110011
01100001 ...
```

decodifica



10

Standard per la codifica dei caratteri

- La definizione di un alfabeto comune al maggior numero possibile di sistemi, e di una codifica standard per rappresentarlo è essenziale per l'elaborazione e la trasmissione delle informazioni
- Tra gli standard più diffusi:
 - **EBCDIC** (*Extended Binary Coded Decimal Interchange Code*)
 - **ASCII** (*American Standard Code for Information Interchange*)

11

Standard per la codifica dei caratteri

- **EBCDIC** (*Extended Binary Coded Decimal Interchange Code*)
 - codifica dei caratteri alfanumerici in stringhe binarie di 8 bit;
 - Sviluppato da IBM negli anni sessanta a partire da un codice precedente, in uso sulle schede perforate.
 - E' ancora utilizzato da diversi sistemi mainframe IBM



12

Standard per la codifica dei caratteri



- ASCII (American Standard Code for Information Interchange)
 - Sviluppato negli anni sessanta a partire da una codifica utilizzata per le telescriventi
 - La prima definizione dello standard (1963) introduce una codifica a 7 bit.
 - I 128 simboli originari comprendevano: caratteri «di controllo», numeri, lettere, segni di interpunzione, simboli matematici

13



Il Codice ASCII

14

Il codice ASCII

- I caratteri sono disposti in una tabella di 8 righe per 7 colonne.
- La codifica ASCII del carattere posto nella casella (i, j) è una sequenza di sette bit in cui:
 - I tre bit più significativi sono dati dalla conversione in binario del numero di colonna
 - I quattro bit meno significativi sono dati dalla conversione in binario del numero di riga.
- Un simile metodo consentiva di utilizzare agevolmente la tabella per effettuare le operazioni di codifica e di decodifica

15

Il codice ASCII

USASCII code chart

Bits		Column							
		0	1	2	3	4	5	6	7
Row	0	1	2	3	4	5	6	7	
0 0 0 0	0	NUL	DLE	SP	0	@	P	\	p
0 0 0 1	1	SOH	DC1	!	1	A	Q	a	q
0 0 1 0	2	STX	DC2	"	2	B	R	b	r
0 0 1 1	3	ETX	DC3	#	3	C	S	c	s
0 1 0 0	4	EOT	DC4	\$	4	D	T	d	t
0 1 0 1	5	ENQ	NAK	%	5	E	U	e	u
0 1 1 0	6	ACK	SYN	B	6	F	V	f	v
0 1 1 1	7	BEL	ETB	'	7	G	W	g	w
1 0 0 0	8	BS	CAN	(8	H	X	h	x
1 0 0 1	9	HT	EM)	9	I	Y	i	y
1 0 1 0	A	LF	SUB	*	:	J	Z	j	z
1 0 1 1	B	VT	ESC	+	;	K	[k	{
1 1 0 0	C	FF	FS	.	<	L	\	l	
1 1 0 1	D	CR	GS	-	=	M]	m	}
1 1 1 0	E	SO	RS	.	>	N	^	n	~
1 1 1 1	F	SJ	US	/	?	O	_	o	DEL

- l'ordine in cui i caratteri sono inseriti in tabella riprende in parte l'ordine alfabetico e, in parte è arbitrario.
 - I primi 32 sono i caratteri di controllo:
 - Non richiedono la stampa di alcun carattere
 - Danno informazioni o «direttive» alla telescrivente

16

Codice ASCII: codifica/decodifica

How alive?

USASCII code chart

Bits					Column											
b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	b ₀	0	0	0	0	1	1	1	1	
Row	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	0	1	1	1	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	0	1	0	2	2	2	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	3	3	3	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	4	4	4	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	5	5	5	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	6	6	6	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	7	7	7	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	8	8	8	8	BS	CAN	(8	H	X	h	x
1	0	0	1	9	9	9	9	9	HT	EM)	9	I	Y	i	y
1	0	1	0	A	A	A	A	A	LF	SUB	*	:	J	Z	j	z
1	0	1	1	B	B	B	B	B	VT	ESC	+	;	K	[k	{
1	1	0	0	C	C	C	C	C	FF	FS	.	<	L	\	l	
1	1	0	1	D	D	D	D	D	CR	GS	-	=	M]	m	}
1	1	1	0	E	E	E	E	E	SO	RS	.	>	N	^	n	~
1	1	1	1	F	F	F	F	F	SI	US	/	?	O	_	o	DEL

17

Codice ASCII: codifica/decodifica

How alive?

USASCII code chart

Bits					Column											
b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	b ₀	0	0	0	0	1	1	1	1	
Row	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	0	1	1	1	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	0	1	0	2	2	2	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	3	3	3	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	4	4	4	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	5	5	5	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	6	6	6	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	7	7	7	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	8	8	8	8	BS	CAN	(8	H	X	h	x
1	0	0	1	9	9	9	9	9	HT	EM)	9	I	Y	i	y
1	0	1	0	A	A	A	A	A	LF	SUB	*	:	J	Z	j	z
1	0	1	1	B	B	B	B	B	VT	ESC	+	;	K	[k	{
1	1	0	0	C	C	C	C	C	FF	FS	.	<	L	\	l	
1	1	0	1	D	D	D	D	D	CR	GS	-	=	M]	m	}
1	1	1	0	E	E	E	E	E	SO	RS	.	>	N	^	n	~
1	1	1	1	F	F	F	F	F	SI	US	/	?	O	_	o	DEL

18

Codice ASCII: codifica/decodifica

How alive?

1 0 0 1 0 0 0

USASCII code chart

Bits					Column											
b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	b ₀	0	1	2	3	4	5	6	7	
0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p	
0	0	0	0	0	0	1	0	SOH	DC1	!	1	A	Q	a	q	
0	0	0	0	1	0	0	0	STX	DC2	"	2	B	R	b	r	
0	0	0	0	1	0	1	0	ETX	DC3	#	3	C	S	c	s	
0	0	0	0	1	0	1	1	EOT	DC4	\$	4	D	T	d	t	
0	0	0	1	0	0	0	0	ENQ	NAK	%	5	E	U	e	u	
0	0	0	1	0	0	1	0	ACK	SYN	&	6	F	V	f	v	
0	0	0	1	0	1	0	0	BEL	ETB	'	7	G	W	w	w	
0	0	0	1	0	1	1	0	BS	CAN	(8	H	X	x	x	
0	0	0	1	0	1	1	1	HT	EM)	9	I	Y	y	y	
0	0	1	0	0	0	0	0	A	LF	SUB	*	:	J	Z	j	z
0	0	1	0	0	0	1	0	B	VT	ESC	+	;	K	[k	{
0	0	1	0	0	0	1	1	C	FF	FS	.	<	L	\	l	
0	0	1	0	0	1	0	0	D	CR	GS	-	=	M]	m	}
0	0	1	0	0	1	1	0	E	SO	RS	.	>	N	^	n	~
0	0	1	0	0	1	1	1	F	SI	US	/	?	O	_	o	DEL

19

Codice ASCII: codifica/decodifica

How alive?

1 0 0 1 0 0 0

Hex: 4 8

USASCII code chart

Bits					Column											
b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	b ₀	0	1	2	3	4	5	6	7	
0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p	
0	0	0	0	0	0	1	0	SOH	DC1	!	1	A	Q	a	q	
0	0	0	0	1	0	0	0	STX	DC2	"	2	B	R	b	r	
0	0	0	0	1	0	1	0	ETX	DC3	#	3	C	S	c	s	
0	0	0	0	1	0	1	1	EOT	DC4	\$	4	D	T	d	t	
0	0	0	1	0	0	0	0	ENQ	NAK	%	5	E	U	e	u	
0	0	0	1	0	0	1	0	ACK	SYN	&	6	F	V	f	v	
0	0	0	1	0	1	0	0	BEL	ETB	'	7	G	W	w	w	
0	0	0	1	0	1	1	0	BS	CAN	(8	H	X	x	x	
0	0	0	1	0	1	1	1	HT	EM)	9	I	Y	y	y	
0	0	1	0	0	0	0	0	A	LF	SUB	*	:	J	Z	j	z
0	0	1	0	0	0	1	0	B	VT	ESC	+	;	K	[k	{
0	0	1	0	0	0	1	1	C	FF	FS	.	<	L	\	l	
0	0	1	0	0	1	0	0	D	CR	GS	-	=	M]	m	}
0	0	1	0	0	1	1	0	E	SO	RS	.	>	N	^	n	~
0	0	1	0	0	1	1	1	F	SI	US	/	?	O	_	o	DEL

20

Esercizio

H	o	w	
48			

a	l	i	v	e	?

21

Ok! Esercizio

H	o	w	
48	6f	77	20

a	l	i	v	e	?
61	6c	69	76	65	3f



Sopravvissuto – The Martian (The Martian), USA 2015, diretto da Ridley Scott, basato su un romanzo di Andy Weir, con Matt Damon, Jeff Daniels, Jessica Chastain.

22

Esercizio

43	69	61	6f	21

23

Esercizio

USASCII code chart

Bits					Column							
					0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b ₄	b ₃	b ₂	b ₁	Row	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(8	H	X	h	x
1	0	0	1	9	HT	EM)	9	I	Y	i	y
1	0	1	0	A	LF	SUB	*	:	J	Z	j	z
1	0	1	1	B	VT	ESC	+	;	K	[k	{
1	1	0	0	C	FF	FS	,	<	L	\	l	
1	1	0	1	D	CR	GS	-	=	M]	m	}
1	1	1	0	E	SO	RS	.	>	N	^	n	~
1	1	1	1	F	SI	US	/	?	O	_	o	DEL

24

Esercizio

01000011
 01101001
 01100001
 00100001

USASCII code chart

Bits					Columns										
b7	b6	b5	b4	b3	b2	b1	b0	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	0	0	1	0	0	SOH	DC1	!	1	A	Q	a	q
0	0	0	0	1	0	1	0	STX	DC2	"	2	B	R	b	r
0	0	0	1	0	0	0	0	ETX	DC3	#	3	C	S	c	s
0	0	0	1	0	0	1	0	EOT	DC4	\$	4	D	T	d	t
0	0	0	1	0	1	0	1	ENQ	NAK	%	5	E	U	e	u
0	0	0	1	0	1	1	0	ACK	SYN	&	6	F	V	f	v
0	0	0	1	1	0	0	0	BEL	ETB	'	7	G	W	g	w
0	0	0	1	1	0	1	0	BS	CAN	(8	H	X	h	x
0	0	0	1	1	0	1	1	HT	EM)	9	I	Y	i	y
0	0	1	0	0	0	0	0	LF	SUB	*	:	J	Z	j	z
0	0	1	0	0	0	1	0	VT	ESC	+	;	K	[k	{
0	0	1	0	0	1	0	0	FF	FS	,	<	L	\	l	
0	0	1	0	0	1	0	1	CR	GS	-	=	M]	m	}
0	0	1	0	1	0	0	0	SO	RS	.	>	N	^	n	~
0	0	1	0	1	0	1	1	SI	US	/	?	O	_	o	DEL

25

Ok! Esercizio

C	i	a	o	!
43	69	61	6f	21

26

Il codice ASCII: caratteri di controllo

La telescrivente visualizza i caratteri «stampabili» da sinistra verso destra una riga alla volta (ogni posizione nella riga è detta colonna). Se la telescrivente raggiunge la fine della riga (l'ultima colonna), inizierà a scrivere i caratteri a partire dalla colonna più a sinistra della riga successiva. Questo modus operandi può essere alterato dai caratteri di controllo.

08	BS (Back Space)	La telescrivente riprende a stampare dalla colonna precedente a quella corrente
09	HT (Horizontal Tab)	Tabulazione orizzontale. Il prossimo carattere sarà stampato nella prossima colonna del tabulatore
12	LF (Line feed)	Nuova riga. Il prossimo carattere sarà stampato sulla colonna successiva a quella corrente ma sulla riga successiva
13	CR (Carriage Return)	Il prossimo carattere sarà stampato sulla prima colonna della riga corrente.
07	BEL	La telescrivente suona il campanello (Il computer fa bip)

27

Il codice ASCII: caratteri di controllo

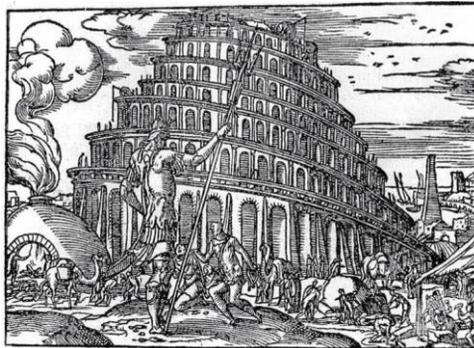
NUL	Null	DLE	Data link escape
SOH	Start of heading	DC1	Device control 1
STX	Start of text	DC2	Device control 2
ETX	End of text	DC3	Device control 3
EOT	End of transmission	DC4	Device control 4
ENQ	Enquiry	NAK	Negative acknowledge
ACK	Acknowledge	SYN	Synchronous idle
BEL	Bell	ETB	End of transmission block
BS	Backspace	CAN	Cancel
HT (TAB)	Horizontal tabulation	EM	End of medium
LF	Line feed	SUB	Substitute
VT	Vertical tabulation	ESC	Escape
FF	Form feed	FS	File separator
CR	Carriage return	GS	Group separator
SO	Shift out	RS	Record separator
SI	Shift in	US	Unit separator
SP	Space	DEL	Delete

28

ANSI ASCII

- La American Standard Association (ASA) rilasciò la prima edizione dello standard nel 1963. Una profonda revisione dello standard fu rilasciata nel 1967.
 - Intanto, la ASA era diventata ANSI (American National Standards Institute).
 - L'ultima revisione dello standard è del 1986.
- Per ASCII «*Esteso*» si intendono alcune varianti (per lo più proprietarie) che estendevano la codifica standard utilizzando l'ottavo bit, in modo da introdurre alcuni simboli fonetici estranei alla lingua inglese (alcune vocali accentate)
- Sostanzialmente orientato all'alfabeto latino (e alcune sue varianti)

29



UNICODE

30

UNICODE

- Uno standard che si propone di affrontare il problema del multilinguismo è Unicode (Universal Encoding).
 - il suo scopo è quello di creare una codifica della scrittura a livello universale
 - assegna un numero univoco ad ogni simbolo in maniera indipendente dal programma, dalla piattaforma e dalla lingua (e relativo alfabeto)
 - È compatibile con il codice ASCII,
 - Originariamente si basava su una codifica a 16 bit che dava la possibilità di codificare più di 65 mila caratteri.

31

UNICODE



- Introdotta dall'Unicode Consortium nel 1991
 - Il consorzio include: Adobe, Apple, IBM, Microsoft, ...
 - Standard «congiunto» con ISO 10646 Universal Code Character Set (UCS)
- Sono previste diverse codifiche di caratteri: *UCS Transformation Format (UTF)*: UTF-8, UTF-16 e UTF-32
 - Parole codice di lunghezze crescente

32

UNICODE

- Lo standard Unicode raggruppa i set di caratteri rappresentati in «planes», in base alla «diffusione» dei gruppi di lingue che li utilizzano.
 - Più diffuse sono le lingue, più compatti sono i codici assegnati ai simboli del loro alfabeto.
 - E.g. il «Basic Multilingual Plane» (BMP) raccoglie le codifiche dei caratteri di quasi tutte le lingue moderne, più molti simboli.
 - ASCII
 - Alfabeti latini
 - Alfabeto greco-copto
 - Alfabeto cirillico
 - ...

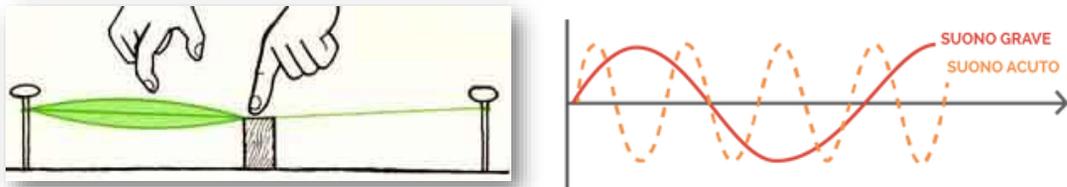
33

Codifica delle grandezze *analogiche*

34

Codifica dei suoni

- Il suono è un classico esempio di segnale analogico per il quale è necessario un processo di digitalizzazione
- Il suono è funzione del tempo. Consiste in vibrazioni prodotte meccanicamente nel mezzo di trasmissione (l'aria) che formano un'onda, caratterizzata da *ampiezza* e *frequenza*.



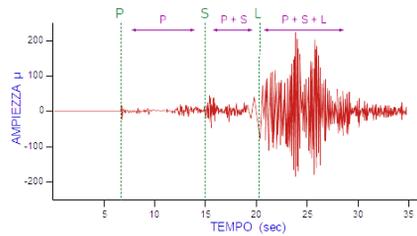
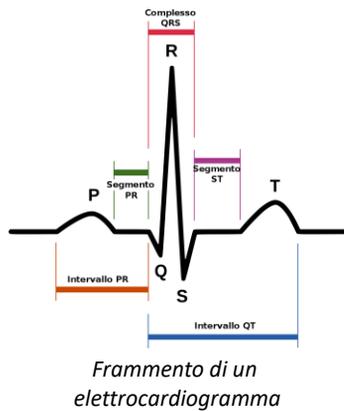
35

Codifica dei suoni

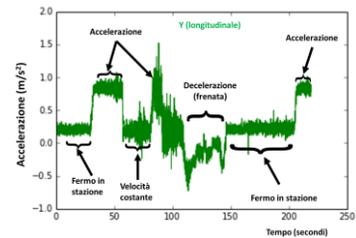
- *L'orecchio umano*, percepisce
 - l'ampiezza dell'onda come «*volume*»
 - La frequenza dell'onda come «*nota*»
- *Un microfono* è in grado di convertire le vibrazioni in segnale elettrico di ampiezza «proporzionale» e di uguale frequenza

36

Codifica delle grandezze *analogiche*



Rilevazioni dell'accelerometro di un veicolo in movimento



37

Analogico vs. Digitale

- Una informazione **analogica** può essere considerata il risultato di una «misura» in natura
 - La frequenza di un suono
 - Il colore di una superficie
 - I contorni di una immagine
 - ...
- può essere rappresentata con funzioni continue del tempo, di coordinate spaziali, ...
- La sua rappresentazione **digitale** la «riassume» utilizzando valori *discreti*.

38

Da analogico a digitale

- Un calcolatore elettronico non può elaborare informazioni analogiche se non sono prima convertite in digitale (*digitalizzate*)
- La trasformazione da analogico a digitale si realizza in due fasi:
 - **Campionamento:** è il processo di selezione del sottoinsieme discreto di informazioni da rappresentare in digitale. È il partizionamento di un flusso continuo di informazione in quantità discrete, rispetto al tempo, allo spazio o ad entrambi.
 - **Quantizzazione:** è il processo che misura le caratteristiche (ad esempio grandezza, intensità, colore) dei campioni selezionati attribuendo loro un valore numerico.

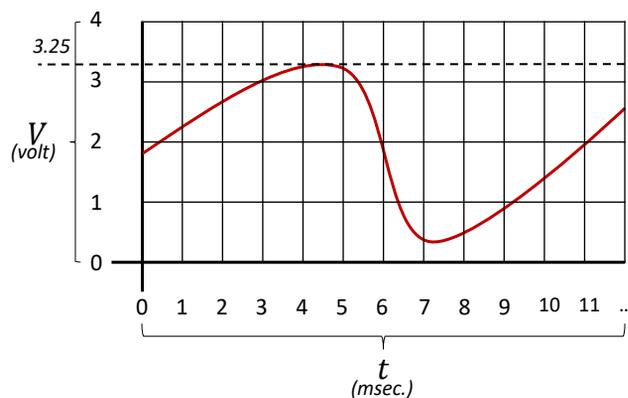
39

Campionamento

Consideriamo la variazione di ampiezza di un segnale nel tempo rappresentato dalla funzione:

$$V(t)$$

Il primo passo per la digitalizzazione di questa informazione consiste nello stabilire un insieme finito di punti (istanti di tempo) in cui misurare il valore della funzione



40

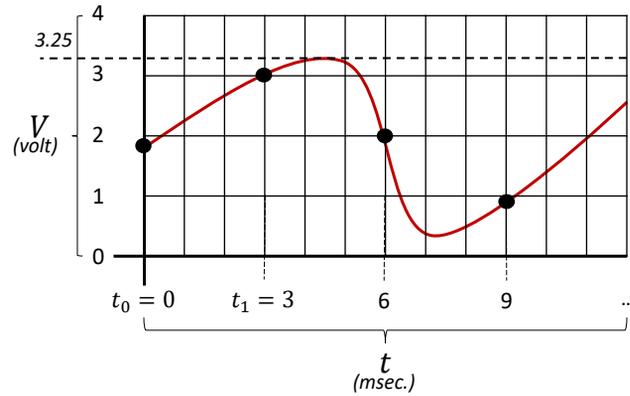
Campionamento

Generalmente si scelgono valori «equidistanti», distanziati da un certo scostamento fisso;

Per esempio: procediamo alla misura ogni 3 ms,

Misuriamo quindi i valori assunti da $V(t_i)$ nei punti

$$t_i = (t_0 + i * 3)$$



41

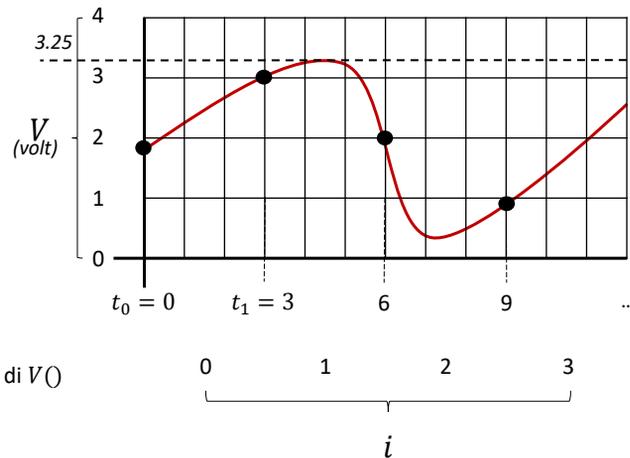
Campionamento

i	t	V
0	0	1.9
1	3	3
2	6	2
3	9	0.9

Misuriamo quindi i valori assunti da $V(t_i)$ negli istanti

$$t_i = (t_0 + i * 3)$$

t_i è l' i -esimo intervallo di campionamento di $V()$

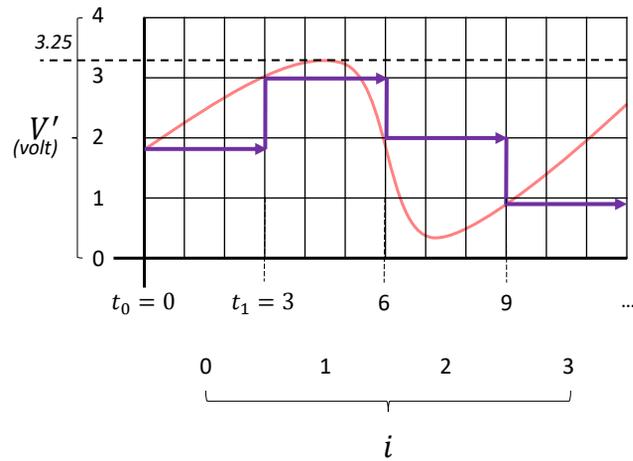


42

Campionamento

Rappresentiamo la funzione $V(t)$ con una nuova funzione «discreta» $V'(i)$ che:

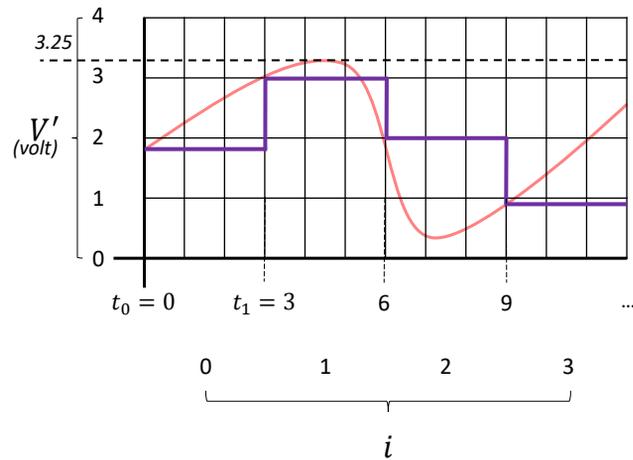
Per ciascun intervallo di valori di t , abbia come risultato il valore di $V(t)$ misurato nel i -esimo istante di campionamento



43

Campionamento

i	t	V'
0	0	1.9
	1	1.9
	2	1.9
1	3	3
	4	3
	5	3
2	6	2
	7	2
...

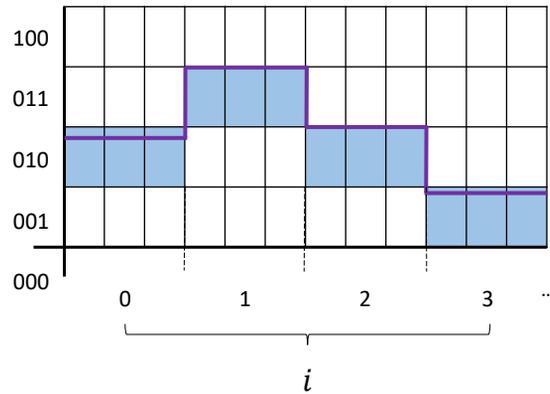


44

Quantizzazione

Rappresentiamo i valori del campione con una **codifica binaria**. Otteniamo la funzione «digitalizzata» $D()$

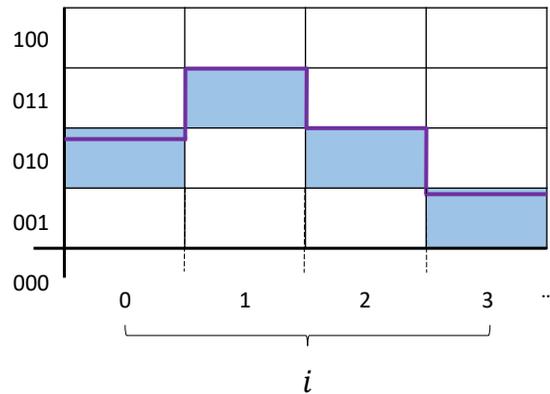
$V'(t_i)$	$D(i)$
$0 < V'(i) \leq 1$	001
$1 < V'(i) \leq 2$	010
$2 < V'(i) \leq 3$	011
$3 < V'(i) \leq 4$	100



45

Quantizzazione

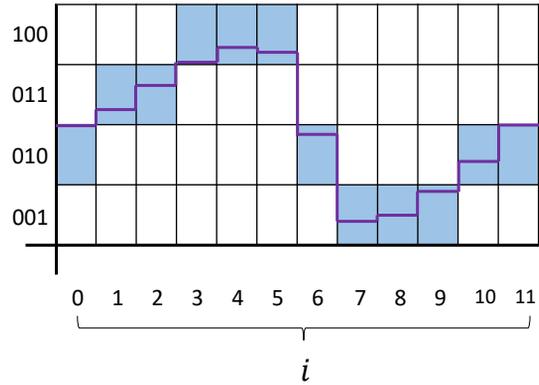
i	$D(i)$
0	010
1	011
2	010
3	001



46

Campionamento

Variando le ampiezze degli intervalli di campionamento, l'approssimazione della funzione analogica è migliore.



47

Codifica delle immagini

48

Codifica delle Immagini

- Nel mondo reale, una immagine è un insieme continuo di informazioni
 - luce, colore
- Per codificare un'immagine è necessario scomporne le informazioni in un insieme discreto di elementi che verranno codificati con sequenze di bit
- La scomposizione più ovvia consiste nel suddividere l'immagine in un reticolo di punti detti **pixel** (*picture element*)

49

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

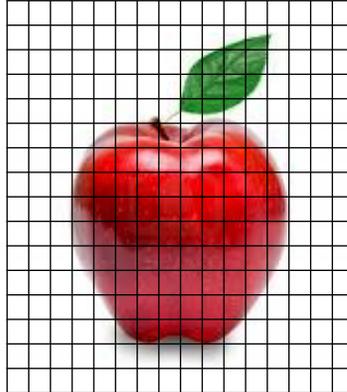


50

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

2. Vi sovrapponiamo una «griglia» di dimensioni arbitrarie (per esempio: 16x16). Ciascuna cella contiene un singolo bit.



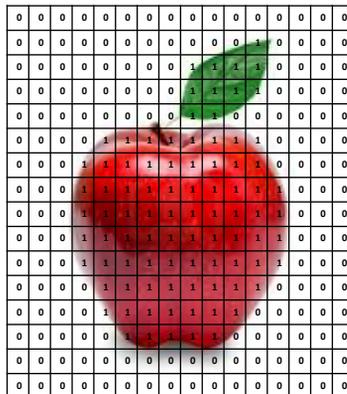
51

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

2. Vi sovrapponiamo una «griglia» di dimensioni arbitrarie (per esempio: 16x16). Ciascuna cella contiene un singolo bit.

3. Poniamo ciascuna cella della griglia a **1** se «*la maggior parte*» della cella è occupata da un elemento dell'immagine diverso dallo sfondo; **0** altrimenti.



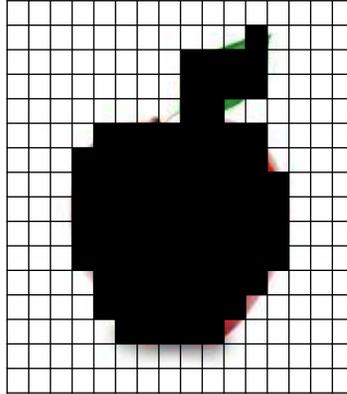
52

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

2. Vi sovrapponiamo una «griglia» di dimensioni arbitrarie (per esempio: 16x16). Ciascuna cella contiene un singolo bit.

3. Poniamo ciascuna cella della griglia a **1** se «la maggior parte» della cella è occupata da un elemento dell'immagine diverso dallo sfondo; **0** altrimenti.



4. La griglia rappresenta la versione «digitalizzata» dell'immagine. Le celle poste a 1 codificano il «profilo» dell'oggetto, con la «capacità di risoluzione» data (16x16).

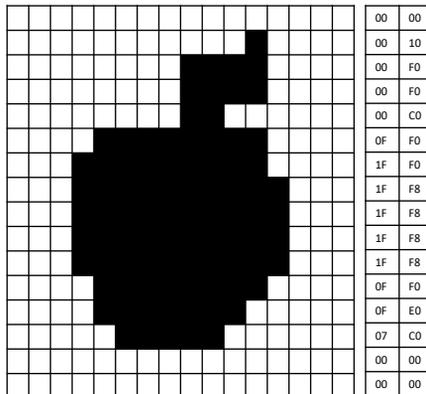
53

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

2. Vi sovrapponiamo una «griglia» di dimensioni arbitrarie (per esempio: 16x16). Ciascuna cella contiene un singolo bit.

3. Poniamo ciascuna cella della griglia a **1** se «la maggior parte» della cella è occupata da un elemento dell'immagine diverso dallo sfondo; **0** altrimenti.



4. La griglia rappresenta la versione «digitalizzata» dell'immagine. Le celle poste a 1 codificano il «profilo» dell'oggetto, con la «capacità di risoluzione» data (16x16).

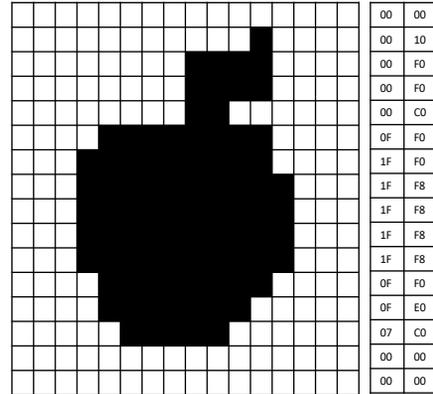
5. Estraiamo da ogni riga due byte e otteniamo la codifica *bitmap* della nostra mela...

54

Codifica delle immagini

Abbiamo codificato la nostra immagine associando a ciascun punto un solo bit di informazione, ottenendo una rappresentazione (in due colori) della mela.

Per la codifica abbiamo scomposto l'immagine in 16 righe ciascuna composta da 16 bit (2 byte)
 $16 \times 2 = 32 \text{ bytes}$



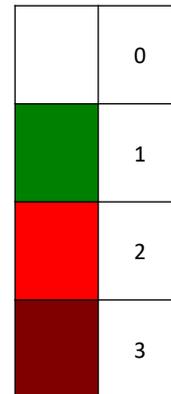
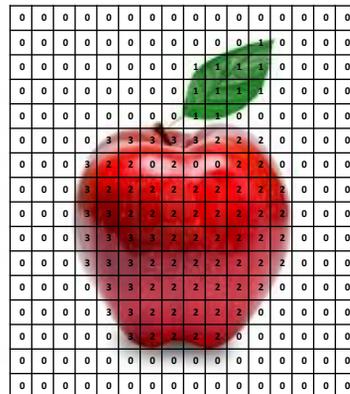
55

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

2. Vi sovrapponiamo una «griglia» di dimensioni arbitrarie (per esempio: 16x16). Ciascuna cella contiene due bit.

3. Poniamo ciascuna cella della griglia a **0** a **3** secondo una codifica dei colori a due bit



56

Codifica delle immagini

1. Acquisiamo l'immagine di una mela:

2. Vi sovrapponiamo una «griglia» di dimensioni arbitrarie (per esempio: 16x16). Ciascuna cella contiene due bit.

3. Poniamo ciascuna cella della griglia a **0** a **3** secondo una arbitraria codifica dei colori a due bit

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	0
	1
	2
	3

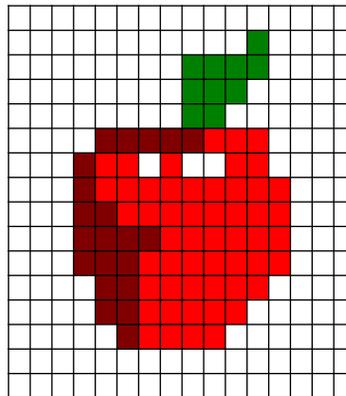
57

Codifica delle immagini

Abbiamo codificato la nostra immagine associando a ciascun punto due bit di informazione, ottenendo ora una rappresentazione in quattro colori della nostra mela.

Per la codifica abbiamo scomposto l'immagine in 16 righe ciascuna composta da 32 bit (4 byte)

$$16 \times 4 = 64 \text{ bytes}$$



	0
	1
	2
	3

58

Dispositivi di acquisizione e *risoluzione*

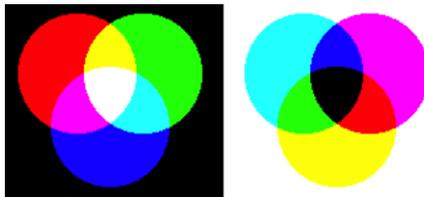
- Camere digitali, scanner, etc. acquisiscono una «areola» del mondo, registrandone le caratteristiche di colore, luminosità, etc,
 - Quanto più piccole, vicine e numerose sono tra loro, tanto migliore è la «qualità» dell'immagine acquisita
- Il concetto di **risoluzione** è legato a quanto sono «*fitti*» i punti che visualizzano l'immagine.
 - Maggiore è la risoluzione dell'immagine, maggiore è la possibilità di distinguere dettagli in essa presenti.
- Tutti i pixel contenuti in una immagine digitale
 - hanno dimensioni identiche.
 - La loro dimensione è determinata dalla risoluzione alla quale l'immagine viene digitalizzata(es. la risoluzione di 600 dpi indica che ciascun pixel misura 1/600 di pollice).

59

Le immagini a colori

- La *colorimetria* spiega che un colore può essere ottenuto tramite combinazione di almeno tre colori base detti primari
- Se i tre colori base sono il Rosso, il Verde ed il Blu si ha lo spazio RGB

$$\text{Color} = a R + b G + c B$$
- Con 8 bit/colore base, per ogni colore si useranno 24 bit, ovvero circa 16 milioni di colori diversi



60

Esempio di Immagini Digitali



61

ESERCIZI

- Quanti byte occorrono per codificare una immagine...
 - 16x16 a 16 colori
 - 32x23 a 2 colori
 - 160x120 a 16 milioni di colori (cod. RGB)

62

I formati BITMAP

- Ciascuna immagine viene memorizzata con diversi formati bitmap
 - (Il formato è la codifica delle informazioni su contorni, colore, etc.)
- La rappresentazione di immagini di grandi dimensioni o a una elevata risoluzione può richiedere una gran quantità di informazioni
- Perciò, alcuni formati prevedono forme di *compressione dati*
 - (Compressione: tecniche avanzate di teoria dei codici per eliminare/ridurre la ridondanza nella codifica delle informazioni)
- Tra i formati più comuni,
 - Tagged Image File Format **TIFF**
 - Graphics Interchange Format **GIF**
 - Joint Photographers Expert Group **JPEG**
 - Microsoft Bit Map **BMP** e Device Independent BitMap **DIB**
 - PC Paintbrush **PCX**

63

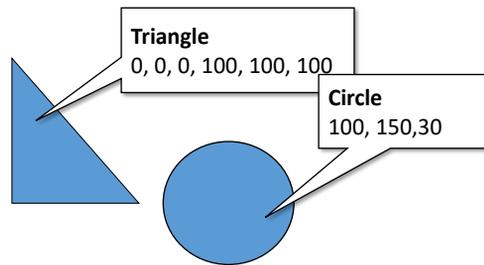
Dimensione dei Bitmap: un esempio

Immagine	Definizione	Colori	Bit	Taglia
TV	720x625	256	8	440KB
SVGA	1024x768	65536	16	1.5MB
Fotografica	15000x10000	16M	25	430MB

64

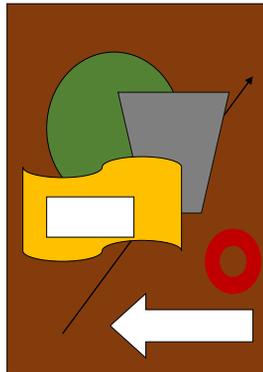
Le Immagini Vettoriali

- Una immagine viene descritta in modo astratto attraverso gli elementi grafici di alto livello (*linee, archi, colori*) che la costituiscono
 - Ciascuna singola componente è identificabile
 - Ogni «oggetto» può essere modificato, variando i parametri che lo contraddistinguono
 - Dimensioni, posizione...



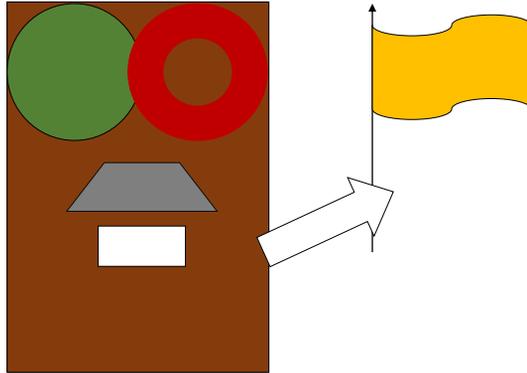
65

Le immagini vettoriali



66

Le immagini vettoriali



67

I formati Vettoriali

- Tra i formati grafici più diffusi, vanno ricordati
 - PostScript PS
 - EPS
 - PDF
 - Drawing eXchange Format (DXF)
 - Initial Graphics Exchange Specifications (IGES)

68

Le immagini in Movimento

- L'uomo «percepisce» il movimento se osserva una successione sufficientemente rapida di immagini fisse
 - *Al cinema sono generalmente proiettati circa 25 fotogrammi al secondo*
 - *In televisione, tra i 25 e i trenta fotogrammi al secondo*



- La sequenza continua di immagini della realtà viene quindi discretizzata ottenendo una serie di immagini (detti *frame*) che variano velocemente, ma a intervalli stabili.
- Il *frame-rate* è il numero di frame mostrati per secondo (*fps*).

69

Le immagini in movimento

- La rappresentazione di informazioni audio/video può essere piuttosto «corposa»
 - Per ottenere una riproduzione di qualità «realistica» dell'informazione, occorre che la granularità dell'acquisizione delle immagini sia molto fine (frame-rate alto)
 - Un frame-rate alto, implica la memorizzazione di una quantità di dati maggiore



70

Le immagini in movimento

- Lo standard MPEG (Moving Picture Expert Group) fornisce una codifica per rappresentare:
 - la sequenza dei frame
 - I suoni,
- Parametri e tecniche di *compressione dati*.



71

Compressione dati

- Per risolvere i problemi connessi alle dimensioni elevate sono stati introdotti processi di compressione che:
 - riducono lo spazio occupato mediante o la diminuzione del numero di bit necessari per codificare una singola informazione (compressione *entropica*)
 - riducono il numero di informazioni da memorizzare o trasmettere (compressione *differenziale*, compressione *semantica*).

72

Compressione dati

- La compressione può conservare integralmente o no il contenuto della rappresentazione originale secondo due tecniche principali:
 - la compressione senza perdita di informazione (**lossless, reversibile**) che sfrutta le ridondanze nella codifica del dato;
 - la compressione con perdita di informazione (**lossy, irreversibile**) che invece sfrutta le ridondanze nella percezione dell'informazione.

73

Compressione Lossless

- La compressione lossless avviene tramite una classe di algoritmi che consentono di ricostruire tutta l'informazione iniziale partendo da quella compressa.
- Non sempre si ottengono riduzioni significative.
- Tra le tecniche di compressione lossless si ricordano:
 - la *Run-length encoding (RLE)* che codifica sequenze di valori uguali premettendo un indicatore di ripetizioni al valore codificato;
 - la codifica di *Huffman* che assegna un numero inferiore di bit alle sequenze più probabili attraverso un vettore di codifica;
 - la compressione *Lempel-Ziv-Welch (LZW)* che costruisce dinamicamente una tabella di codifica con numero variabile di bit sulla base delle sequenze incontrate;
 - la *codifica differenziale* in cui ogni dato è rappresentato come differenza rispetto al dato precedente.

74

Compressione Lossy

- I metodi lossy comportano riduzioni notevoli delle dimensioni, ma la ricostruzione dell'informazione da quella compressa non è però identica a quella iniziale.
- Tali metodi rimuovono parti che possono non essere percepite come avviene nel caso di immagini, video e suoni.
 - Ad esempio gli algoritmi di compressione usati nei formati GIF e JPEG per immagini fisse sfruttano la caratteristica dell'occhio umano di essere poco sensibile a lievi cambiamenti di colore in punti contigui, e quindi eliminano questi lievi cambiamenti appiattendolo il colore dell'immagine.

75

Compressione Lossy

- Tra le tecniche di compressione lossy si ricordano:
 - la compressione **JPEG** per le immagini che applica una trasformata nel dominio delle frequenze (*Discrete Cosine Transform*) che permette di sopprimere dettagli irrilevanti riducendo il numero di bit necessari per la codifica;
 - la compressione **MPEG** per i video che codifica parte dei frame come differenze rispetto ai valori previsti in base ad una interpolazione;
 - la compressione **MP3** per l'audio che si basa alle proprietà psicoacustiche dell'udito umano per sopprimere le informazioni inutili.

76

Mappa

Per lo studio e l'approfondimento degli argomenti trattati

77

Mappa

[Idb] Capitolo 2: sezioni 2.3 (dicotomia analogico/digitale) e 2.5 (codifica del testo)

Per le codifiche binarie dei numeri interi con segno, si veda la Sezione 2.2.5

[Sli] Esempi discussi a lezione su codifica e decodifica di testo e immagini.

78

Approfondimenti (*facoltativi*)

[Misc] Molte versioni della tabella ASCII, sono disponibili sulla rete, tra queste:

<https://www.asciitable.it/>

La home page ufficiale dell'UNICODE consortium è all'URL

<https://home.unicode.org>

Per molto tempo, i caratteri alfanumerici sono stati l'unico strumento per fare computer grafica. Ne è fiorita una insolita forma d'arte: ASCII Art, che ha ancora molti estimatori. Uno tra i tanti archivi di opere si trova all'URL <https://asciiart.eu>

«**Sopravvissuto – The Martian**» (The Martian), USA 2015, film diretto da Ridley Scott, basato su un romanzo di Andy Weir, con Matt Damon, Jeff Daniels, Jessica Chastain.