

Capitolo 3 – Parte 1

La codifica dell'informazione

Informatica e Informazione
La codifica dell'informazione
Analogico vs digitale

Informazione e Informatica

Informatica e telecomunicazione: quale definizione?

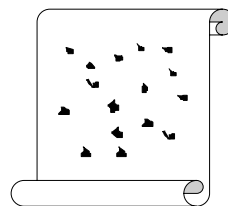
- Cos'è l'**informatica**?
 - lo studio sistematico degli algoritmi che descrivono e trasformano l'informazione: la loro teoria, analisi, progetto, efficienza, realizzazione e applicazione [ACM – Association for Computing Machinery]
 - la scienza della rappresentazione e dell'elaborazione dell'informazione
- Cos'è la **telecomunicazione**?
 - la trasmissione rapida a distanza dell'informazione
- **Attenzione:**
 - Non si parla di tecnologia dei calcolatori !
 - Si attribuisce ruolo centrale al concetto di informazione !

04/04/02

Introduzione ai sistemi informatici

3

Il concetto di informazione



Configurazione 1



Configurazione 2

04/04/02

Introduzione ai sistemi informatici

4

Non esiste informazione senza supporto

- L'informazione è "portata da", o "trasmessa su", o "memorizzata in", o "contenuta in" qualcosa; questo "qualcosa" però non è l'informazione stessa.
- Ogni supporto ha le sue caratteristiche in quanto mezzo su cui può essere scritta dell'informazione
 - alcuni supporti sono particolarmente adatti alla trasmissione di informazione, ma non alla sua memorizzazione (aria),
 - per altri supporti vale il viceversa (compact disc)

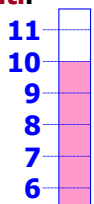
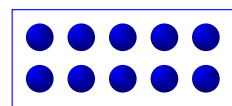
04/04/02

Introduzione ai sistemi informatici

5

Informazione e supporti (1)

La stessa informazione può essere scritta su supporti differenti.



04/04/02

Introduzione ai sistemi informatici

6

Informazione e supporti (2)

Lo **stesso supporto** può portare **informazioni differenti**.



04/04/02

Introduzione ai sistemi informatici

7

Informazione e supporto (3)

- Distinguere informazione e supporto fisico è distinguere tra "**entità logiche**" ed "**entità fisiche**":
 - l'informazione richiede un supporto fisico, ma non coincide con esso;
 - l'informazione è un'entità extra-fisica, non interpretabile in termini di materia-energia e sottoposta alle leggi della fisica solo perché basata su un supporto fisico.
- **L'informazione si può creare e distruggere.**

04/04/02

Introduzione ai sistemi informatici

8

Quali caratteristiche deve avere un sistema fisico per supportare informazioni?

- Si ottiene informazione quando, dato un insieme di alternative possibili, la lettura del supporto ne elimina alcune e ne seleziona altre.
- **Condizione necessaria** perché un supporto possa portare informazione è che possa assumere **configurazioni differenti**, a ognuna delle quali venga associata una differente **entità di informazione**.
 - Un supporto che possa presentarsi sempre e comunque in un unico modo **non può** portare alcuna informazione.
 - Il caso più semplice è quello in cui **le configurazioni** possibili per il supporto sono **due**.

04/04/02

Introduzione ai sistemi informatici

9

Supporto fisico: 1a condizione

- Deve consentire di potere identificare delle differenze
 - Es: voglio rappresentare 2 alternative
- Cosa **rappresenta** ciascuna configurazione?



04/04/02

Introduzione ai sistemi informatici

10

Configurazioni e codici

- A ogni configurazione del supporto deve essere associata un'entità di informazione:
 - foglio senza macchia = Divina Commedia;
 - foglio con macchia = I Promessi Sposi.
- Per interpretare le differenti configurazioni del supporto in termini di informazione è necessario conoscere il **codice** (cioè la regola) che a ogni **configurazione ammessa del supporto** associa un'entità di informazione.
- La definizione di un codice comporta che siano identificati in modo non ambiguo l'insieme delle **possibili configurazioni del supporto** e l'insieme delle **possibili entità di informazione** a cui ci si vuole riferire.
- **Variando il codice è possibile riferirsi a entità di informazione differenti utilizzando uno stesso supporto fisico.**

04/04/02

Introduzione ai sistemi informatici

11

Supporto fisico: 2a condizione

Deve essere **condivisa** una regola per attribuire un **significato** a ciascuna configurazione



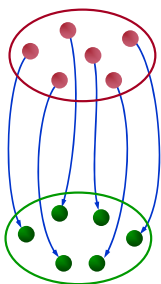
04/04/02

Introduzione ai sistemi informatici

12

Definire un codice

- Identificare
 - { **Configurazioni** }
 - { **Entità informazione** }
- Associare gli elementi dei 2 insiemi



04/04/02

Introduzione ai sistemi informatici

13

Livelli di informazione: il telegrafo

- **Supporto fisico**: conduttore in cui transita corrente continua
- **Entità di informazione**: "punti" e "linee", che rappresentano le lettere dell'alfabeto
- **Codice** = regola che specifica la corrispondenza:
 - transita per 1 s \Rightarrow punto
 - transita per 2 s \Rightarrow linea
 - non transita per 1 s \Rightarrow separa punti e linee di una stessa lettera
 - non transita per 2 s \Rightarrow separa due lettere
- "c'è per 1 s, non c'è per 1 s, c'è per 2 s" = "punto linea".
- **Codice Morse** = relazione tra lettere e sequenze di punti linee:
 - punto linea \Rightarrow lettera "A"
 - linea punto punto punto \Rightarrow lettera "B"
 - linea punto linea punto \Rightarrow lettera "C"
 - ecc.

- **Abbiamo associato simboli ad altri simboli!!**

04/04/02

Introduzione ai sistemi informatici

14

Relazione tra messaggi e configurazione del supporto

- **Configurazione del supporto fisico** (livello fisico, **LF**):
 - c'è per 2 s, non c'è per 1 s, c'è per 1 s, non c'è per 2 s, c'è per 2 s, non c'è per 1 s, c'è per 2 s, non c'è per 1 s, c'è per 2 s
- **messaggio**, al livello logico 1 (**LL1**):
 - linea punto, separazione di lettera, linea linea linea
- **messaggio**, al livello Logico 2 (**LL2**):
 - "N" "O"
- **messaggio**, al livello Logico 3 (**LL3**):
 - "NO", con il significato (per esempio) di "incontro non confermato"
- Le relazioni tra questi livelli definiscono ognuna un codice per interpretare il significato delle entità che compaiono al livello precedente in termini delle entità del livello successivo: sono relazioni di significazione.

04/04/02

Introduzione ai sistemi informatici

15

Informazione e incertezza

- La presenza di informazione è condizionata dal fatto che il supporto sia in grado di **assumere diverse configurazioni**.
- Se la nostra ignoranza, o più formalmente la nostra incertezza, circa l'effettiva configurazione del supporto viene ridotta dall'accesso al supporto, allora sembra del tutto ragionevole assumere che si possa affermare che tale atto **ci ha portato dell'informazione**.
- Se fossimo in grado di misurare il grado di incertezza in cui ci trovavamo prima della lettura e quello successivo a essa, la quantità di informazione portata dalla configurazione che abbiamo letto sul supporto potrebbe essere definita proprio dalla differenza tra tali gradi di incertezza.

04/04/02

Introduzione ai sistemi informatici

16

Informazione e incertezza

- Ipotesi:
 - si conoscono a priori le configurazioni che il supporto può assumere,
 - non si sa quale di queste sia quella effettivamente assunta.
- La lettura del supporto è come la selezione di un elemento da un insieme di composizione nota (e.g. estrai una pallina da un'urna di cui conosci il contenuto)
 - quanto è più numeroso l'insieme, tanto maggiore sarà l'informazione portata dalla lettura
 - quanto meno è probabile che si presenti una configurazione, tanto maggiore è l'informazione che essa porta.
- La quantità di informazione che si ottiene selezionando una configurazione da un insieme che ne contiene due è l'unità elementare (**bit**).
 - la risposta "sì" oppure "no" a una domanda porta 1 bit di informazione;
 - la risposta a due domande di tal genere (che corrisponde in effetti alla selezione tra 4 alternative: "sì-sì", "sì-no", "no-sì", "no-no") porta 2 bit;
 - ...

04/04/02

Introduzione ai sistemi informatici

17

La codifica dell'informazione

Codifica dati e istruzioni

- Algoritmi = **istruzioni** che operano su **dati**.
- Per scrivere un programma è necessario rappresentare **istruzioni** e **dati** in un formato tale che l'esecutore automatico sia capace di
 - **memorizzare** istruzioni e dati;
 - **manipolare** istruzioni e dati.

04/04/02

Introduzione ai sistemi informatici

19

Codifica dati e istruzioni

- **Alfabeto dei simboli**
 - cifre "0", "1", ..., "9", separatore decimale ("."), separatore delle migliaia (" ") e segni positivo ("+") o negativo ("-").
- **Regole di composizione** (sintassi), che definiscono le successioni "ben formate"
 - "1.234,5" è la rappresentazione di un numero;
 - "1,23,45" non lo è.
- **Codice** (semantica)
 - "1.234,5" = $1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 + 5 \times 10^{-1}$
 - "1,23,45" = ??
- Lo stesso alfabeto può essere utilizzato con codici diversi:
 - "123,456" = $1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2} + 6 \times 10^{-3}$, [IT]
 - "123,456" = $1 \times 10^5 + 2 \times 10^4 + 3 \times 10^3 + 4 \times 10^2 + 5 \times 10^1 + 6 \times 10^0$, [UK]

04/04/02

Introduzione ai sistemi informatici

20

Codifica Binaria

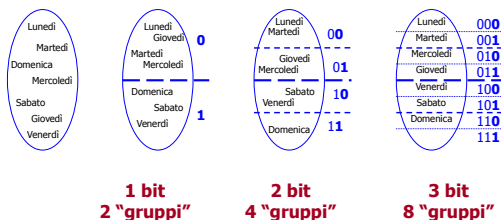
- **Alfabeto binario**: usiamo dispositivi con solo due stati
- Problema: assegnare un **codice univoco** a tutti gli oggetti compresi in un insieme predefinito (e.g. studenti)
- Quanti **oggetti** posso codificare con **k bit**:
 - 1 bit \Rightarrow 2 stati (0, 1) \Rightarrow 2 oggetti (e.g. Vero/Falso)
 - 2 bit \Rightarrow 4 stati (00, 01, 10, 11) \Rightarrow 4 oggetti
 - 3 bit \Rightarrow 8 stati (000, 001, ..., 111) \Rightarrow 8 oggetti
 - ...
 - **k bit \Rightarrow 2^k stati \Rightarrow 2^k oggetti**
- Quanti **bit** mi servono per codificare **N** oggetti:
 - $N \leq 2^k \Rightarrow k \geq \log_2 N \Rightarrow k = \lceil \log_2 N \rceil$ (intero superiore)
- Attenzione: c'è l'ipotesi implicita che i codici abbiano tutti la **stessa lunghezza**

04/04/02

Introduzione ai sistemi informatici

21

I giorni della settimana in binario



04/04/02

Introduzione ai sistemi informatici

22

Codifica binaria dei caratteri

- Quanti sono gli oggetti compresi nell'insieme?
 - 26 lettere maiuscole + 26 minuscole \Rightarrow 52
 - 10 cifre
 - Circa 30 segni d'interpunzione
 - Circa 30 caratteri di controllo (EOF, CR, LF, ...)
- circa 120 oggetti complessivi $\Rightarrow k = \lceil \log_2 120 \rceil = 7$
- Codice ASCII: utilizza 7 bit e quindi può rappresentare al massimo $2^7 = 128$ caratteri
 - Con 8 bit (= byte) rappresento 256 caratteri (ASCII esteso)
 - Si stanno diffondendo codici più estesi (e.g. UNICODE) per rappresentare anche i caratteri delle lingue orientali

04/04/02

Introduzione ai sistemi informatici

23

bit, Byte, KiloByte, MegaByte, ...

- bit** = solo due stati, "0" oppure "1".
- Byte** = 8 bit, quindi $2^8 = 256$ stati
- KiloByte [KB]** = 2^{10} Byte = 1024 Byte $\sim 10^3$ Byte
- MegaByte [MB]** = 2^{20} Byte = 1 048 576 Byte $\sim 10^6$ Byte
- GigaByte [GB]** = 2^{30} Byte $\sim 10^9$ Byte
- TeraByte [TB]** = 2^{40} Byte $\sim 10^{12}$ Byte
- PetaByte [PB]** = 2^{50} Byte $\sim 10^{15}$ Byte
- ExaByte [EB]** = 2^{60} Byte $\sim 10^{18}$ Byte

04/04/02

Introduzione ai sistemi informatici

24

ASCII su 7 bit

	000	001	010	011	100	101	110	111
010	sp	!	"	#	\$	%	&	'
011	0	1	2	3	4	5	6	7
100	@	A	B	C	D	E	F	G
101	P	Q	R	S	T	U	V	W
110	`	a	b	c	d	e	f	g
111	p	q	r	s	t	u	v	w

04/04/02

Introduzione ai sistemi informatici

25

La codifica delle istruzioni

➤ Si segue lo schema presentato per i caratteri alfanumerici:

- **quali e quante** sono le istruzioni da codificare?
- qual è la **lunghezza** delle successioni di bit da utilizzare?
- qual è la **corrispondenza** tra istruzioni e successioni di bit?

Istruzioni aritmetico-logiche	Istruzioni per il trasferimento dati	Istruzioni di controllo
Codice Istruzione	Codice Istruzione	Codice Istruzione
01100000 ADD	00010101 LOAD	10011001 IF_EQ
01100100 SUB	00110110 STORE	10110110 GOSUB
01111110 AND	...	10101100 RETURN
...

04/04/02

Introduzione ai sistemi informatici

26

Oltre al codice operativo

- ... è necessario far riferimento ai **dati** necessari per completare l'esecuzione dell'istruzione,
- e.g. addizione: è necessario che sia specificato (anche **implicitamente**) dove leggere i due operandi da sommare e dove scrivere il risultato;
- il **numero** dei dati da specificare è variabile, in funzione delle istruzioni.

Codice Operativo

Codice Operativo Indirizzo 1

Codice Operativo Indirizzo 1 Indirizzo 2

Codice Operativo Indirizzo 1 Indirizzo 2 Indirizzo 3

04/04/02

Introduzione ai sistemi informatici

27

Numeri naturali

➤ Sistema di numerazione posizionale in base **b**

- $c_k c_{k-1} \dots c_0$ rappresenta $c_k \times b^k + c_{k-1} \times b^{k-1} + \dots + c_0 \times b^0$
- $b=10 \Rightarrow 1101_{\text{dieci}}$ indica $1 \times 10^3 + 1 \times 10^2 + 0 \times 10 + 1 \times 10^0$

➤ Conversione **binario** \Rightarrow **decimale**

- basta scrivere il numero secondo la notazione posizionale utilizzando già il sistema decimale
- $b=2 \Rightarrow 1101_{\text{due}}$ indica $1 \times 2^3 + 1 \times 2^2 + 0 \times 2 + 1 \times 2^0 = 13_{\text{dieci}}$

➤ Conversione **decimale** \Rightarrow **binario**

- Si potrebbe utilizzare lo stesso metodo indicato sopra, ma è molto complesso
- $b=10 \Rightarrow 345_{\text{dieci}}$ indica $11 \times 1010^{10} + 100 \times 1010^1 + 101 \times 1010^0$

04/04/02

Introduzione ai sistemi informatici

28

Conversione decimale binaria

Si calcolano i resti delle divisioni per due

18 : 2 = 9	resto 0	137 : 2 = 68	resto 1
9 : 2 = 4	resto 1	68 : 2 = 34	resto 0
4 : 2 = 2	resto 0	34 : 2 = 17	resto 0
2 : 2 = 1	resto 0	17 : 2 = 8	resto 1
1 : 2 = 0	resto 1	8 : 2 = 4	resto 0
		4 : 2 = 2	resto 0
		2 : 2 = 1	resto 0
		1 : 2 = 0	resto 1

10010

10001001

04/04/02

Introduzione ai sistemi informatici

29

Numeri interi

➤ Alfabeto binario

- anche il segno è rappresentato da 0 o 1
- è indispensabile indicare il numero **k** di bit utilizzati

➤ **Modulo e segno**

- **1** bit di segno (0 positivo, 1 negativo)
- **k-1** bit di modulo
 - Esempio: $+6_{\text{dieci}} = 0110_{\text{ms}}$ $-6_{\text{dieci}} = 1110_{\text{ms}}$
- si rappresentano i valori da $-2^{k-1}+1$ a $2^{k-1}-1$
 - con 4 bit i valori vanno da -7 a +7
 - con 8 bit i valori vanno da -127 a +127
- Attenzione: ci sono due rappresentazioni dello 0
 - con 4 bit sono $+0_{\text{dieci}} = 0000_{\text{ms}}$ $-0_{\text{dieci}} = 1000_{\text{ms}}$

04/04/02

Introduzione ai sistemi informatici

30

Diverse codifiche/interpretazioni

Codice	Nat	MS
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

Codice	Nat	MS
1000	8	-0
1001	9	-1
1010	10	-2
1011	11	-3
1100	12	-4
1101	13	-5
1110	14	-6
1111	15	-7

04/04/02

Introduzione ai sistemi informatici

31

Ottali ed esadecimali

➤ Utili per rappresentare sinteticamente i valori binari

➤ Ottali (base b = 8)

- Alfabeto ottale: cifre comprese tra 0 e 7
 - $354_{\text{otto}} = 3 \times 8^2 + 5 \times 8^1 + 4 \times 8^0 = 192 + 40 + 4 = 236_{\text{dieci}}$
 - $1461_{\text{otto}} = 1 \times 8^3 + 4 \times 8^2 + 6 \times 8^1 + 1 \times 8^0 = 512 + 256 + 48 + 1 = 817_{\text{dieci}}$
- Ogni cifra ottale corrisponde a tre cifre binarie:
 - $11101100_{\text{due}} = [11] [101] [100] = 354_{\text{otto}}$
 - $1100110001_{\text{due}} = [1] [100] [110] [001] = 1461_{\text{otto}}$

➤ Esadecimali (base b = 16)

- Alfabeto esadecimale: cifre 0 - 9 + lettere A - F
 - $EC_{\text{sedici}} = 14 \times 16^1 + 12 \times 16^0 = 224 + 12 = 236_{\text{dieci}}$
 - $331_{\text{sedici}} = 3 \times 16^2 + 3 \times 16^1 + 1 \times 16^0 = 768 + 48 + 1 = 817_{\text{dieci}}$
- Ogni cifra esadecimale corrisponde a quattro cifre binarie:
 - $11101100_{\text{due}} = [1110] [1100] = EC_{\text{sedici}}$
 - $1100110001_{\text{due}} = [11] [0011] [0001] = 331_{\text{sedici}}$

04/04/02

Introduzione ai sistemi informatici

32

Numeri razionali

➤ Rappresentazione in **virgola fissa**

- $0.1011_{\text{due}} = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4}$
 $= 0.5 + 0.125 + 0.0625 = 0.6875_{\text{dieci}}$
- $11.101_{\text{due}} = 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$
 $= 2 + 1 + 0.5 + 0.125 = 3.625_{\text{dieci}}$
- Il numero di cifre prima e dopo la "virgola" è **fisso**!!

➤ Rappresentazione in **virgola mobile** (float)

- usata spesso anche in decimale per rappresentare numeri molto grandi o molto piccoli: 0.1357×10^{64}
 - mantissa** - parte frazionaria compresa tra 0 e 1 [0.1357]
 - esponente** - numero intero (≥ 0)
- utilizza 1 bit per il segno (s), h bit per l'esponente (e) e k bit per la mantissa (m): **$R = s \times m \times 2^e$**

04/04/02

Introduzione ai sistemi informatici

33

Approfondimento: il complemento a 2

Numeri interi in complemento a 2

➤ Alfabeto binario

- anche il segno è rappresentato da 0 o 1
- è indispensabile indicare il numero **k** di bit utilizzati

➤ Complemento a 2

- X** corrisponde al binario naturale di **$2^k + X$**
 - $+6_{\text{dieci}} \Rightarrow 2^4 + 6 = 22 \Rightarrow [1]0110 \Rightarrow 0110_{C_2}$
 - $-6_{\text{dieci}} \Rightarrow 2^4 - 6 = 10 \Rightarrow [0]1010 \Rightarrow 1010_{C_2}$
- si rappresentano i valori da -2^{k-1} a $2^{k-1}-1$
 - con 4 bit i valori vanno da -8 a +7
 - con 8 bit i valori vanno da -128 a +127
- Attenzione: c'è una sola rappresentazione dello 0
 - con 4 bit è $+0_{\text{dieci}} = 0000_{C_2}$ mentre $1000_{C_2} = -8_{\text{dieci}}$

04/04/02

Introduzione ai sistemi informatici

35

Il complemento a 2

➤ Metodi alternativi per calcolare la rappresentazione di -X a partire da quella di X

- Effettuare il complemento di ogni bit di X e aggiungere poi 1
 - rappresentazione di $+6_{\text{dieci}} = 0110_{C_2}$ (NB ci vogliono 4 bit!!)
 - complemento di tutti i bit $\Rightarrow 1001_{C_2}$ (corrisponderebbe a -6_{dieci})
 - aggiungere 1 $\Rightarrow 1010_{C_2}$ (che corrisponde a -6_{dieci})
- Partendo da destra e andando verso sinistra, lasciare invariati tutti i bit fino al primo 1 compreso, complementare tutti gli altri bit.
 - rappresentazione di $+6_{\text{dieci}} = 0110_{C_2}$ (NB ci vogliono 4 bit!!)
 - gli ultimi due bit ($\dots 1.0$) rimangono invariati
 - gli altri due bit vengono complementati (**1.0.1.0**_{C2})

04/04/02

Introduzione ai sistemi informatici

36

Complemento a 2 – Alcune osservazioni

- i numeri positivi iniziano con **0**, quelli negativi con **1**
- data la rappresentazione di un numero su **k** bit, la rappresentazione dello stesso numero su **k+1** bit si ottiene aggiungendo (a sinistra) un bit uguale al primo (**estensione del "segno"**)
 - Rappresentazione di -6 su 4 bit = 1010
 - Rappresentazione di -6 su 5 bit = 11010
 - Rappresentazione di -6 su 8 bit = 11111010
- la sottrazione si effettua come somma algebrica
 - $4 - 6 = +4 + (-6) = 0100 + 1010 = 1110 = -2$
 - $9 - 6 = +9 + (-6) = 01001 + 11010 = [1]00011 = +3$

04/04/02

Introduzione ai sistemi informatici

37

Diverse codifiche/interpretazioni

Codice	Nat	MS	C2
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7

Codice	Nat	MS	C2
1000	8	-0	-8
1001	9	-1	-7
1010	10	-2	-6
1011	11	-3	-5
1100	12	-4	-4
1101	13	-5	-3
1110	14	-6	-2
1111	15	-7	-1

04/04/02

Introduzione ai sistemi informatici

38

Approfondimento: Operazioni Logiche (Booleane)

Algebra di Boole

- L'algebra di Boole (dal suo inventore G. Boole) serve a descrivere le operazioni logiche.
- Componenti dell'algebra di Boole:
 - Operatori booleani
 - Regole di trasformazione ed equivalenza tra operatori booleani
- Gli **operandi** booleani assumono solo due valori:

Vero/Falso True/False 1/0 Sì/No ...

04/04/02

Introduzione ai sistemi informatici

40

Operatori e tavole di verità

A	not A
0	1
1	0

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

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

A B	A xor B
0 0	0
0 1	1
1 0	1
1 1	0

A B	A ↔ B
0 0	1
0 1	0
1 0	0
1 1	1

A B	A nand B
0 0	1
0 1	1
1 0	1
1 1	0

A B	A nor B
0 0	1
0 1	0
1 0	0
1 1	0

04/04/02

Introduzione ai sistemi informatici

41

Espressioni booleane

- Equivalenza
 - Due espressioni booleane sono **equivalenti** se hanno la medesima tavola di verità
- Tautologia
 - Un'espressione booleana è una **tautologia** se è sempre vera
Esempio: **A OR (NOT A)**
- Contraddizione
 - Un'espressione booleana è una **contraddizione** se è sempre falsa
Esempio: **A AND (NOT A)**

04/04/02

Introduzione ai sistemi informatici

42

Notazione

➤ Esistono convenzioni diverse:

- Negazione **not A** $\neg A$ $A !$ $- A$
- Congiunzione **A and B** $A \wedge B$ $A \& B$ $A \times B$
- Disgiunzione **A or B** $A \vee B$ $A | B$ $A + B$
- Disgiunzione esclusiva **A xor B** $A \wedge B$ $A \oplus B$
[equivale a $(A \text{ and } (\text{not } B)) \text{ or } ((\text{not } A) \text{ and } B)$]
- Implicazione (se ... allora) $A \rightarrow B$ $A \supset B$ $A \Rightarrow B$
- Doppia implicazione (se e solo se) $A \leftrightarrow B$ $A \equiv B$ $A \Leftrightarrow B$

04/04/02

Introduzione ai sistemi informatici

43

Proprietà degli operatori booleani

Proprietà	AND (×)	OR (+)
Identità	$A \times 1 = A$	$A + 0 = A$
Elemento nullo	$A \times 0 = 0$	$A + 1 = 1$
Idempotenza	$A \times A = A$	$A + A = A$
Inverso	$A \times (\neg A) = 0$	$A + (\neg A) = 1$
Commutativa	$A \times B = B \times A$	$A + B = B + A$
Associativa	$A \times (B \times C) = (A \times B) \times C$	$A + (B + C) = (A + B) + C$
Distributiva	$A \times (B + C) = (A \times B) + (A \times C)$	$A + (B \times C) = (A + B) \times (A + C)$
Assorbimento	$A \times (A + B) = A$	$A + (A \times B) = A$
De Morgan	$\neg(A \times B) = (\neg A) + (\neg B)$	$\neg(A + B) = (\neg A) \times (\neg B)$

04/04/02

Introduzione ai sistemi informatici

44

Analogico vs digitale

Informazione "classificatoria" e più che "classificatoria"

- Informazione **classificatoria**:
"è questo, ma avrebbe potuto essere quest'altro".
 - fondata sulla possibilità di distinzione;
 - utile per discriminare tra elementi compresi in un insieme finito.
- Informazione **più che classificatoria**:
 - riconoscere distinzioni;
 - stabilire una relazione d'ordine ("questo è maggiore di quest'altro");
 - stabilire una metrica ("questo è distante un certo valore da quest'altro").
- L'insieme delle entità di informazione ha una **struttura**.
 - La struttura dice "cosa si può fare" con le entità di informazione dell'insieme, in termini di operazioni di combinazione e di confronto.
 - Si tratta di informazione su informazione (**meta-informazione**) in presenza della quale l'insieme delle entità di informazione diventa un **sistema**, cioè appunto un "insieme con struttura".

04/04/02

Introduzione ai sistemi informatici

46

Due alternative

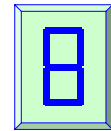
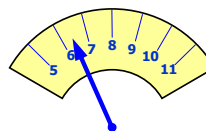
- Meta-informazione esplicita nel supporto:
 - il supporto ha una struttura corrispondente a quella presente tra entità di informazione.
 - Esempio, per indicare il voto conseguito da uno studente si utilizza un rettangolo la cui superficie è proporzionale al voto: tra le dimensioni dei rettangoli è stabilita una relazione di ordine analoga alla relazione d'ordine che esiste tra i voti.
- Meta-informazione implicita nella regola di codifica:
 - al supporto si richiede solo di avere configurazioni molteplici e distinguibili l'una dall'altra,
 - la meta-informazione è definita in modo estensionale nella regola di codifica.
 - Esempio, per indicare il voto ottenuto da uno studente si usa una configurazione convenzionalmente scelta in un insieme, per esempio {■, □, ✕, ■, ■, ■} se i voti diversi da comunicare fossero al più 6.
- Cosa succede nei due casi se si aggiungono i "mezzi punti"?

04/04/02

Introduzione ai sistemi informatici

47

Analogico e digitale



- Meta-informazione esplicita nel supporto
- Meta-informazione implicita nella codifica

- **Codifica ANALOGICA**
- **Codifica DIGITALE**

04/04/02

Introduzione ai sistemi informatici

48

Analógico vs digitale

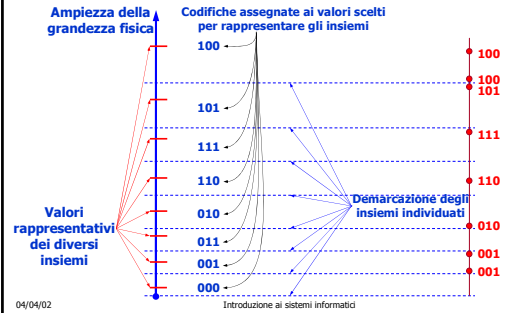
	Analógico	Digitale
Definizione della codifica	Intensionale	Estensionale
Ridefinizione codifica vs variazione entità informazione	Nessuna ridefinizione	Estendere il processo definito in precedenza
Applicabilità vs cardinalità entità informazione	OK anche per cardinalità non finita	Solo per cardinalità finita e con entità note a priori
Applicabilità vs struttura entità informazione	Solo se c'è una struttura	OK in ogni caso
Condizioni sul supporto	Solo se il supporto ha una struttura	OK in ogni caso

04/04/02

Introduzione ai sistemi informatici

49

Da analogico a digitale: 1. la quantizzazione



04/04/02

Introduzione ai sistemi informatici

50

Da analogico a digitale: 2. il campionamento

- La grandezza varia nel tempo e non può essere rappresentata da un solo valore.
- I valori di riferimento debbono essere rilevati in diversi istanti di tempo (frequenza di campionamento).
- La quantizzazione deve poi essere ripetuta per ogni valore campionato.

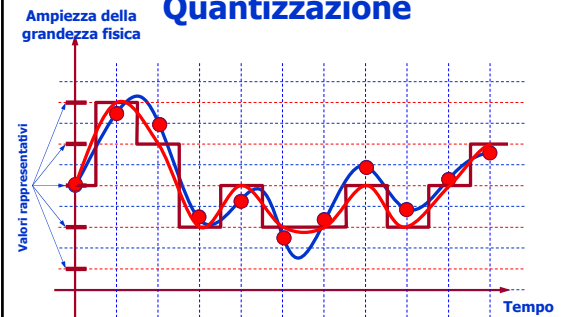


04/04/02

Introduzione ai sistemi informatici

51

Quantizzazione



04/04/02

Introduzione ai sistemi informatici

52

Il successo del digitale

- Rumore:** effetto dell'ambiente sul supporto.
- Quanto un supporto è "immune" al rumore?
 - Codifica analogica: ogni configurazione è lecita dal punto di vista informativo e quindi risulta impossibile distinguere il rumore dal segnale.
 - Codifica digitale: un valore binario è associato a un insieme di configurazioni valide quindi si può
 - riconoscere il rumore che porta in configurazioni non lecite
 - trascurare il rumore che non fa uscire il segnale dall'insieme associato alla stessa configurazione



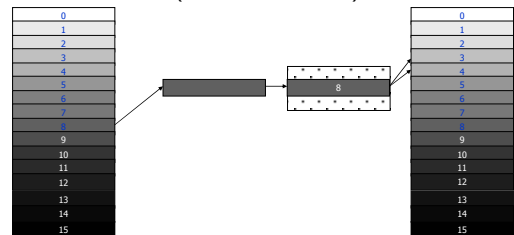
04/04/02

Introduzione ai sistemi informatici

53

Le ragioni del digitale

Al crescere del numero di possibili configurazioni aumenta la probabilità di non interpretare correttamente l'informazione (es. a causa di disturbi)



04/04/02

Introduzione ai sistemi informatici

54

Le ragioni del digitale

Minimizzando il numero di possibili configurazioni...

