

Introduzione all'Informatica

Francesco Folino

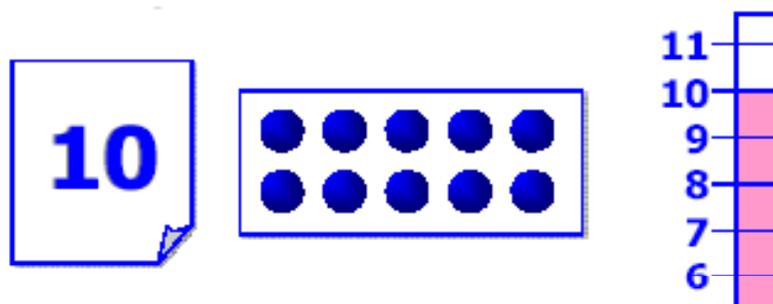
Codifica dell'Informazione

CODIFICA DI DATI E ISTRUZIONI

- ◆ Algoritmi
 - Istruzioni che operano su dati
- ◆ Per scrivere un programma è necessario rappresentare dati e istruzioni in un formato tale che l'esecutore automatico sia in grado di
 - Memorizzare istruzioni e dati
 - Manipolare istruzioni e dati
- ◆ Le informazioni gestite dai sistemi di elaborazione devono essere codificate
 - per poter essere memorizzate, elaborate, scambiate,...

CODIFICA DELL'INFORMAZIONE

- ◆ La stessa informazione si può rappresentare in modi differenti



- ◆ Stessa rappresentazione per informazioni differenti



SISTEMI DI CODIFICA

Sistema di codifica (o *codifica*, o *codice*)

- Usa un insieme di simboli di base (**alfabeto**)
- I simboli dell'alfabeto possono essere combinati ottenendo differenti **configurazioni** (o *codici*, o *stati*), distinguibili l'una dall'altra
- Associa ogni configurazione ad una particolare entità di informazione (la configurazione diventa un modo per rappresentarla)

SISTEMI DI CODIFICA: NUMERI INTERI (DECIMALI)

- ◆ Alfabeto
 - Cifre “0”, “1”, “2”, ..., “9”
 - separatore decimale (“,”)
 - separatore delle migliaia (“.”)
 - Segni positivo (“+”) e negativo (“-”)
- ◆ Regole di composizione (**sintassi**)
 - Definiscono le combinazioni ben formate
 - 12.318,43
 - 12,318.43
- ◆ Codice (**semantica**)
 - Associano ad ogni configurazione un’entità di informazione
 - $12.318,43 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 1 \times 10^1 + 1 \times 10^0 + 4 \times 10^{-1} + 3 \times 10^{-2}$
- ◆ Lo stesso alfabeto può essere usato per 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]

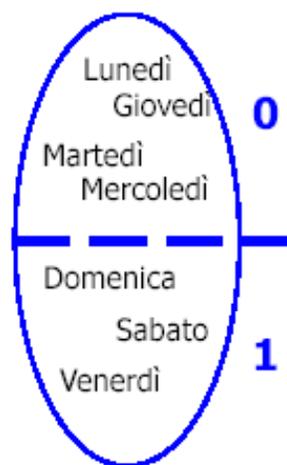
CODIFICA BINARIA

- ◆ Codifica binaria: usa un alfabeto di **2** simboli
- ◆ Utilizzata nei sistemi informatici
 - Si utilizza una grandezza fisica (luminosità, tensione elettrica, la corrente elettrica), per rappresentare informazione
 - Si divide in due intervalli l'insieme dei valori che la grandezza può assumere: ogni intervallo corrisponde ad un simbolo
- ◆ Solo 2 simboli al fine di ridurre la probabilità di errore
 - Tanto più simboli si devono distinguere e tanto meno la rivelazione sarà affidabile (gli intervalli della grandezza fisica saranno meno ampi)

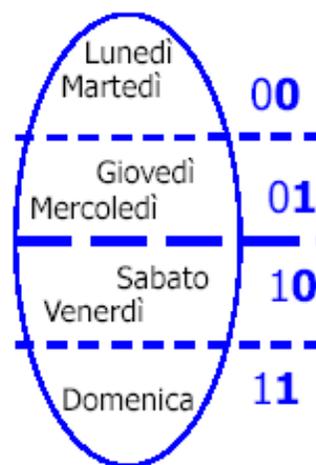
CODIFICA BINARIA

- ◆ **BIT (BInary digiT)**
 - unità elementare di informazione rappresentabile con dispositivi elettronici
 - con 1 bit si possono rappresentare 2 stati
 - 0/1, on/off, si/no
- ◆ Combinando più bit si può codificare un numero maggiore di stati
 - con 2 bit possono rappresentare 4 stati
 - con **K** bit si possono rappresentare **2^K** stati
- ◆ Quanti bit sono necessari per codificare N oggetti?
 - $N \leq 2^K \rightarrow K \geq \log_2 N \rightarrow \mathbf{K = \lceil \log_2 N \rceil}$

I GIORNI DELLA SETTIMANA IN BINARIO



1 bit
2 "gruppi"



2 bit
4 "gruppi"



3 bit
8 "gruppi"

CODIFICA BINARIA: UNITA' DERIVATE

- ◆ **Byte** = 8 bit
 - può rappresentare $2^8 = 256$ stati
- ◆ **KiloByte** (KB) = 2^{10} byte = 1.024 byte $\cong 10^3$ byte
- ◆ **MegaByte** (MB) = 2^{20} byte = 1.048.576 byte $\cong 10^6$ byte
- ◆ **GigaByte** (GB) = 2^{30} byte = 1.073.741.824 byte $\cong 10^9$ byte
- ◆ **TeraByte** (TB) = 2^{40} byte = 1.099.511.627.776 byte $\cong 10^{12}$ byte

CODIFICA DEI NUMERI NATURALI

- ◆ Sistema di numerazione posizionale con **base β**
 - β simboli (**cifre**) corrispondono ai numeri da 0 a $\beta-1$
 - i numeri naturali maggiori o uguali a β possono essere rappresentati da una sequenza di cifre
- ◆ Se un numero naturale N è rappresentato in base β dalla sequenza di n cifre

$$a_{n-1} a_{n-2} a_{n-3} \dots a_1 a_0$$

allora N può essere espresso come segue:

$$N = \sum_{i=0}^{n-1} a_i \beta^i = a_{n-1} \beta^{n-1} + a_{n-2} \beta^{n-2} + \dots + a_2 \beta^2 + a_1 \beta + a_0$$

CODIFICA DEI NUMERI NATURALI

- ◆ *Esempio:* **13** può essere espresso in funzione delle potenze di 2 come:

$$13 = 1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1$$

cioè può essere rappresentato dalla sequenza di bit

1 1 0 1

CONVERSIONE DECIMALE-BINARIO

- ◆ Si calcolano i resti della divisione per 2

$$\begin{array}{ll} 18 : 2 = 9 & \text{resto } 0 \\ 9 : 2 = 4 & \text{resto } 1 \\ 4 : 2 = 2 & \text{resto } 0 \\ 2 : 2 = 1 & \text{resto } 0 \\ 1 : 2 = 0 & \text{resto } 1 \end{array}$$

10010

$$\begin{array}{ll} 137 : 2 = 68 & \text{resto } 1 \\ 68 : 2 = 34 & \text{resto } 0 \\ 34 : 2 = 17 & \text{resto } 0 \\ 17 : 2 = 8 & \text{resto } 1 \\ 8 : 2 = 4 & \text{resto } 0 \\ 4 : 2 = 2 & \text{resto } 0 \\ 2 : 2 = 1 & \text{resto } 0 \\ 1 : 2 = 0 & \text{resto } 1 \end{array}$$

10001001

CODIFICA DEI NUMERI INTERI

◆ Modulo e segno

- Il bit più a sinistra rappresenta il segno del numero (0 = '+', 1 = '-')
- Esempio: +7 = 0111, -7 = 1111
- Valori da $-2^{k-1}+1$ a $2^{k-1}-1$
- Con k=4 bit: da $-2^3+1=-7$ a $2^3-1=+7$
- Attenzione ci sono due zeri!
+0=0000 e -0=1000

CODIFICA DEI NUMERI INTERI

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

Complemento a 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 $+0_{\text{dieci}} = 0000_{\text{C2}}$ mentre $1000_{\text{C2}} = -8_{\text{dieci}}$

CODIFICA DEI NUMERI INTERI

◆ Complemento a 2

Metodi 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 $+6_{\text{dieci}} = 0110_{\text{C2}}$ (NB ci vogliono 4 bit!!)
 - complemento di tutti i bit $\Rightarrow 1001_{\text{C2}}$ (corrisponderebbe a -7_{dieci})
 - aggiungere 1 $\Rightarrow 1010_{\text{C2}}$ (che corrisponde a -6_{dieci})

CODIFICA DEI NUMERI INTERI

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

CODIFICA DEI NUMERI INTERI

◆ Complemento a 2

- 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
 - $4 - 6 = +4 + (-6) = 0100 + 1010 = 1110 = -2$
 - $9 - 6 = +9 + (-6) = 01001 + 11010 = [1]00011 = +3$

CODIFICA DI CARATTERI

- ◆ Associando un simbolo dell'alfabeto ad ogni numero possiamo codificare tutte le lettere
- ◆ Codifica **ASCII** (*American Standard Code for Information Interchange*):
 - Caratteri speciali, punteggiatura, a-z, A-Z, 0-9
 - Utilizza 7 bit (128 caratteri)
 - I codici ASCII estesi usano 8 bit (256 caratteri)
- ◆ Codifica **EBCDIC** (*Extended Binary-Coded Decimal Interchange Code*)
 - Utilizza 8 bit (256 caratteri)
- ◆ Codifica **UNICODE**
 - Utilizza 16 bit (65536 caratteri)
 - I primi 128 caratteri di UNICODE sono gli stessi di ASCII
 - I successivi corrispondono ad altri alfabeti (greco, cirillico, ebraico, ...)
 - Non riesce a coprire i simboli (oltre 200.000) di tutte le lingue!