
Informatica Applicata introduzione

INFORMATICA

E' la scienza che si propone di raccogliere, organizzare, elaborare e conservare le informazioni gestendole in modo automatico.

Il termine deriva dalla fusione delle parole:

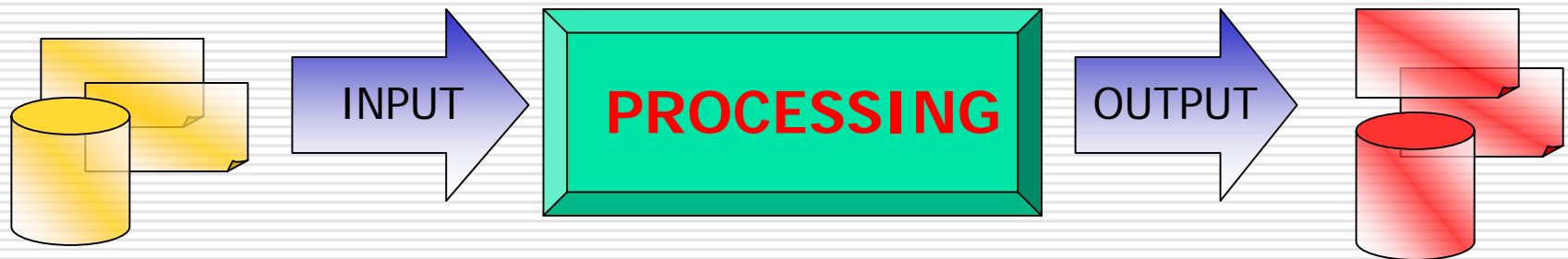
INFORMazione

autoMATICA



AUTOMA

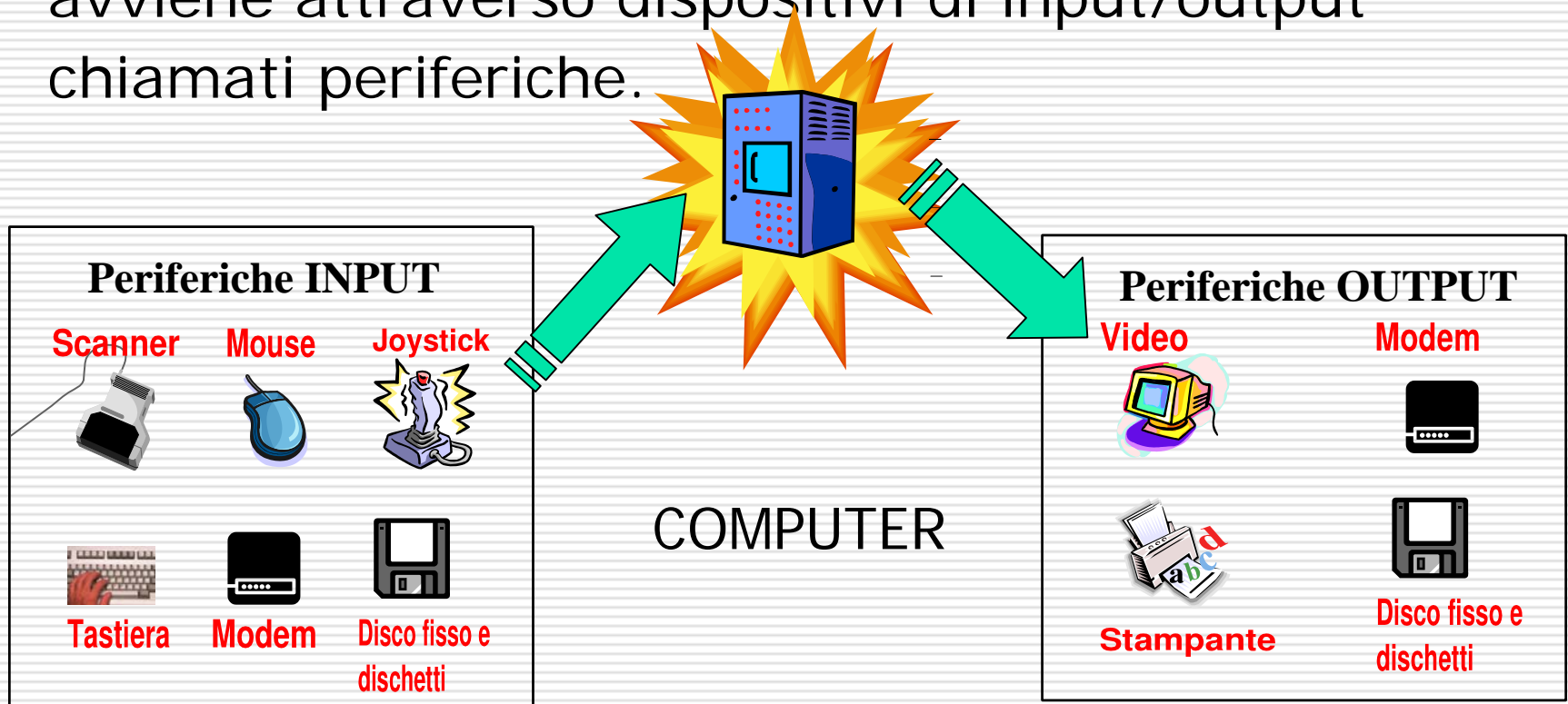
L'automa è un sistema, che imita il comportamento umano, in grado di "ricevere informazioni dall'esterno (input), reagire alle stesse elaborandole (processing), e inviare informazioni di nuovo all'esterno (output)".



Il computer è un tipo di automa composto da componenti elettronici

PERIFERICHE

La comunicazione da e verso l'esterno avviene attraverso dispositivi di input/output chiamati periferiche.

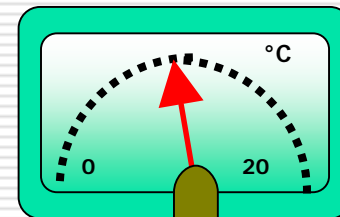


INFORMAZIONE ANALOGICA

Si definisce *analogico* un procedimento che rappresenta un fenomeno con continuità, per esempio: un orologio classico che con il moto regolare della lancetta segna il trascorrere del tempo in modo continuo:

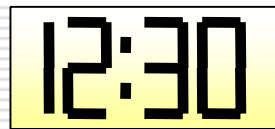
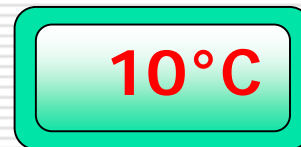


oppure un termometro che rappresenta la temperatura con una lancetta:



INFORMAZIONE DIGITALE

E' *digitale* un procedimento che rappresenta lo stesso fenomeno traducendolo **in cifre** (dall'inglese *digit*, cifra) e quindi in modo discontinuo, come per esempio avviene in un orologio a cristalli liquidi numerico nel quale la stessa durata temporale viene misurata da una successione di scatti.

A digital clock display showing the time 12:30. The digits are black on a yellow background.A digital temperature display showing 10°C. The text is red on a green background.

Analogico e digitale

Contrariamente a quanto si potrebbe credere la registrazione digitale, pur essendo "a salti", può essere più precisa di quella analogica in quanto non soggetta ad interferenze e disturbi.

Occorre però che il numero di valori utilizzati sia molto elevato, in modo da cogliere ogni più piccola sfumatura!

Rappresentazione dell'informazione

Il principio di funzionamento di un computer si basa sulla logica binaria.

Per comprenderne l'idea di base, possiamo pensare a un interruttore, che può essere aperto o chiuso, o ad una scheda che può essere forata o meno in un punto, o ad un suono che può essere presente o no, o ancora ad una riflessione ottica che può verificarsi o meno.

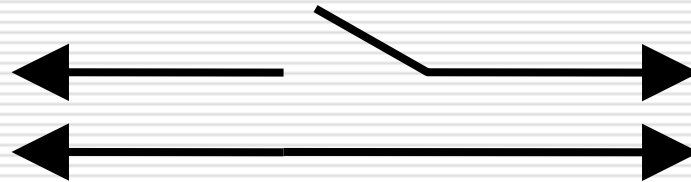


Bit

Ciò che accomuna tutti questi fenomeni è la caratteristica di poter assumere solamente **due stati**:

0 = aperto

1 = chiuso



Questa variabile che assume solo due stati (0 e 1) si chiama **bit** ("binary digit", cifra binaria) ed è l'unità minima di informazione e la base dell'algebra binaria.

Byte

Un computer "ragiona" unicamente interpretando gruppi di bit, cioè comandi rappresentati da sequenze di "uno" e di "zero" (per esempio, 00101100).

Convenzionalmente, **8 bit** costituiscono **1 byte**.

10011010

00011101

11001010

Rappresentazione di caratteri

Il byte è stato inventato come un gruppo di bit sufficienti a rappresentare un carattere.

I caratteri dell'alfabeto inglese sono 26, quindi come minimo un byte deve consentire 26 codici distinti. Per fare questo non basterebbe un byte di 4 bit ($2^4 = 16$), ma uno di 5 bit sì ($2^5 = 32 > 26$).

Rappresentazione di caratteri

In realtà:

- ▣ rappresentare solo i caratteri dell'alfabeto è limitativo.
- ▣ Vorrei poter rappresentare anche le cifre.
- ▣ Vorrei poter rappresentare anche la punteggiatura.
- ▣ Vorrei poter rappresentare anche qualche simbolo (p.es. aritmetici).
- ▣ Vorrei poter aggiungere "caratteri" speciali che marchino p.es. la fine di una riga, l'inizio e la fine di una trasmissione, e svolgano altre funzioni di controllo.
- ▣ Vorrei anche distinguere le 26 lettere in maiuscolo dalle stesse in minuscolo.

Il byte di 8 bit

Essendo un byte formato da 8 bit, sarà possibile rappresentare in un byte un numero compreso tra 0 e 255.

Grazie a questo ampio intervallo sarà possibile rappresentare in un computer anche **caratteri** e **simboli** associando ad ogni carattere un numero compreso tra 0 e 255.

Possiamo pertanto identificare il byte come l'ingombro di memoria di un singolo carattere.

TAVOLE DEI CARATTERI

Le più famose tavole di codifica dei caratteri sono:

- ▣ **ASCII** (American Standard Code for Information Interchange)
- ▣ **EBCDIC** (Extended Binary Coded Decimal Interchange Code)
- ▣ **ANSI** (American National Standard Institute)
- ▣ Per poter rappresentare i caratteri di qualunque lingua è stato creato uno schema, chiamato **UNICODE**, formato da 34168 caratteri.

TAVOLE DEI CARATTERI

Vediamo di fianco una parte
della tavola ASCII:

NUM	CAR	NUM	CAR	NUM	CAR	NUM	CAR	NUM	CAR	NUM	CAR	NUM	CAR
33	!	48	0	63	?	78	N	93]	108	l		
34	"	49	1	64	@	79	O	94	^	109	m		
35	#	50	2	65	A	80	P	95	_	110	n		
36	\$	51	3	66	B	81	Q	96	`	111	o		
37	%	52	4	67	C	82	R	97	a	112	p		
38	&	53	5	68	D	83	S	98	b	113	q		
39	'	54	6	69	E	84	T	99	c	114	r		
40	(55	7	70	F	85	U	100	d	115	s		
41)	56	8	71	G	86	V	101	e	116	t		
42	*	57	9	72	H	87	W	102	f	117	u		
43	+	58	:	73	I	88	X	103	g	118	v		
44	,	59	;	74	J	89	Y	104	h	119	w		
45	-	60	<	75	K	90	Z	105	i	120	x		
46	.	61	=	76	L	91	[106	j	121	y		
47	/	62	>	77	M	92	\	107	k	122	z		

Forma polinomiale del numero

Il sistema numerico da noi adottato è posizionale: il valore della cifra dipende dalla posizione in cui si trova all'interno del numero

Ogni posizione indica un peso legato alla base.

Ad es:

$$321_{10} = 3 \times 10^2 + 2 \times 10^1 + 1 \times 10^0 = 3 \times 100 + 2 \times 10 + 1 \times 1 \\ = 321_{10}$$

Il numero di cifre usate da un sistema numerico prende il nome di **base**.

Il sistema numerico in base 10 utilizza le cifre da 0 a 9

Numerazione binaria

Affiancando un sufficiente numero di bit è possibile rappresentare qualunque numero.

Ogni **posizione** rappresenta una potenza crescente di 2:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	0	1	0	1	1	0

$$= 128 * 1 + 64 * 0 + 32 * 0 + 16 * 1 + 8 * 0 + 4 * 1 + 2 * 1 + 1 * 0 = 150 \text{ decimale}$$

Questo procedimento consente di convertire un numero binario in decimale

Numerazione binaria

Il procedimento inverso (da decimale a binario), si ottiene dividendo i resti per le potenze di 2:

$$150/128=1$$

$$r: 22/64=0$$

$$r: 22/32=0$$

$$r: 22/16=1$$

$$r: 6/8=0$$

$$r: 6/4=1$$

$$r: 2/2=1$$

$$r: 0$$

Gli stessi principi valgono per qualunque BASE di numerazione!!!

Binario risultante: 01101001

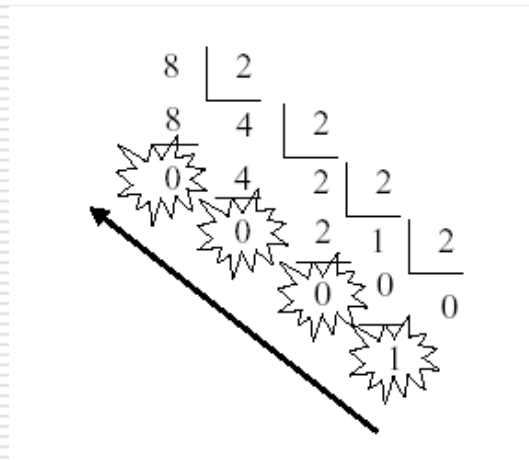
Numerazione binaria

In modo più semplice, è sufficiente dividere il numero di partenza per la base di destinazione, tante volte finché il quoziente si annulla.

Il resto delle divisioni letto in senso inverso è il numero nella base di destinazione.

Ad es:

$8_{10} \longrightarrow (?)_2$



$$8_{10} = 1000_2$$

Numerazione ottale

In ambito informatico rivestono particolare rilevanza altre due numerazioni: quella **ottale**, che utilizza solo 8 cifre (da 0 a 7),

8^4	8^3	8^2	8^1	8^0
3	7	0	4	5

che in decimale è =

$$4096 * 3 + 512 * 7 + 64 * 0 + 8 * 4 + 5 = 15909$$

...ed esadecimale

E la numerazione **esadecimale**, che utilizza 16 cifre (da 0 a 9 e le lettere da A a F) che corrispondono ai valori da 0 a 15 decimali.

16^3	16^2	16^1	16^0
2	5	E	F

che in decimale è = $4096 * 2 + 256 * 5 + 16 * 14 + 15 = 9711$

Per la conversione inversa da decimale a ottale (o esadecimale) si utilizza lo stesso metodo di divisione dei resti visto per la numerazione binaria.

Confronti

Decimale	Binario	Ottale	Esadecimale
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13

CONVERSIONE TRA BASI

Essendo le basi 8 e 16 potenze di 2, esiste un meccanismo semplice di conversione tra esse e la base 2:

Per i **numeri ottali** è sufficiente suddividere il numero binario in gruppi di 3 bit (a partire da destra) e convertire ogni singolo gruppo nel corrispondente ottale:

1	0	1	1	1	0	1	0	0
---	---	---	---	---	---	---	---	---

 = 5 6 4 ottale

CONVERSIONE TRA BASI

Essendo le basi 8 e 16 potenze di 2, esiste un meccanismo semplice di conversione tra esse e la base 2:

Per i **numeri esadecimali** è sufficiente suddividere il numero binario in gruppi di 4 bit (a partire da destra) e convertire ogni gruppo nel corrispondente esadecimale:

1	0	1	0	0	0	1	1
---	---	---	---	---	---	---	---

 = **A 3** esadecimale

Misura dell'informazione

La quantità di memoria di un computer si misura mediante i multipli del byte, ossia:

Multiplo	Sigla	Valore	Approssimazione
Kilo	k	$2^{10} = 1024$	$\approx 10^3$
Mega	M	$2^{20} = 1024^2 = 1024 \text{ k}$	$\approx 10^6$
Giga	G	$2^{30} = 1024^3 = 1024 \text{ M}$	$\approx 10^9$
Tera	T	$2^{40} = 1024^4 = 1024 \text{ G}$	$\approx 10^{12}$

E' per via del sistema binario che un kilobyte è composto da 1024 byte e non 1000, infatti 1024 è una potenza di 2 (1000 lo è di 10).

Gli attuali computer sono dotati di quantità di memoria **RAM** nell'ordine delle centinaia di megabyte, mentre i **dischi fissi** hanno capacità nell'ordine delle decine di gigabyte.

Stringhe e Vettori (Array)

Una stringa è una serie ordinata di caratteri alfanumerici.

Ogni singola parola e ogni frase complessa sono quindi da considerarsi delle stringhe.

Il vettore (o array) è la generalizzazione del concetto di stringa:

"sequenza omogenea di informazioni di uno **stesso tipo**" (numeri, caratteri o altro).