

Lezione 1

1. Concetti di base dell'informatica

1.1 Informazione: tipo, valore e semantica

L'informatica consiste nella gestione delle informazioni operata dal calcolatore (computer).

Il concetto fondamentale dell'informatica è dunque l'*informazione*, essa è costituita da un dato (ad es. un numero) che può assumere diversi valori di un determinato tipo e dal suo significato (semantica del dato).

Informazione	Tipi	Numeri interi
		Numeri non interi
		Alfanumerici
		Strutture (combinazioni di tipi numerici e alfanumerici)
	Valore	Dato
Semantica		

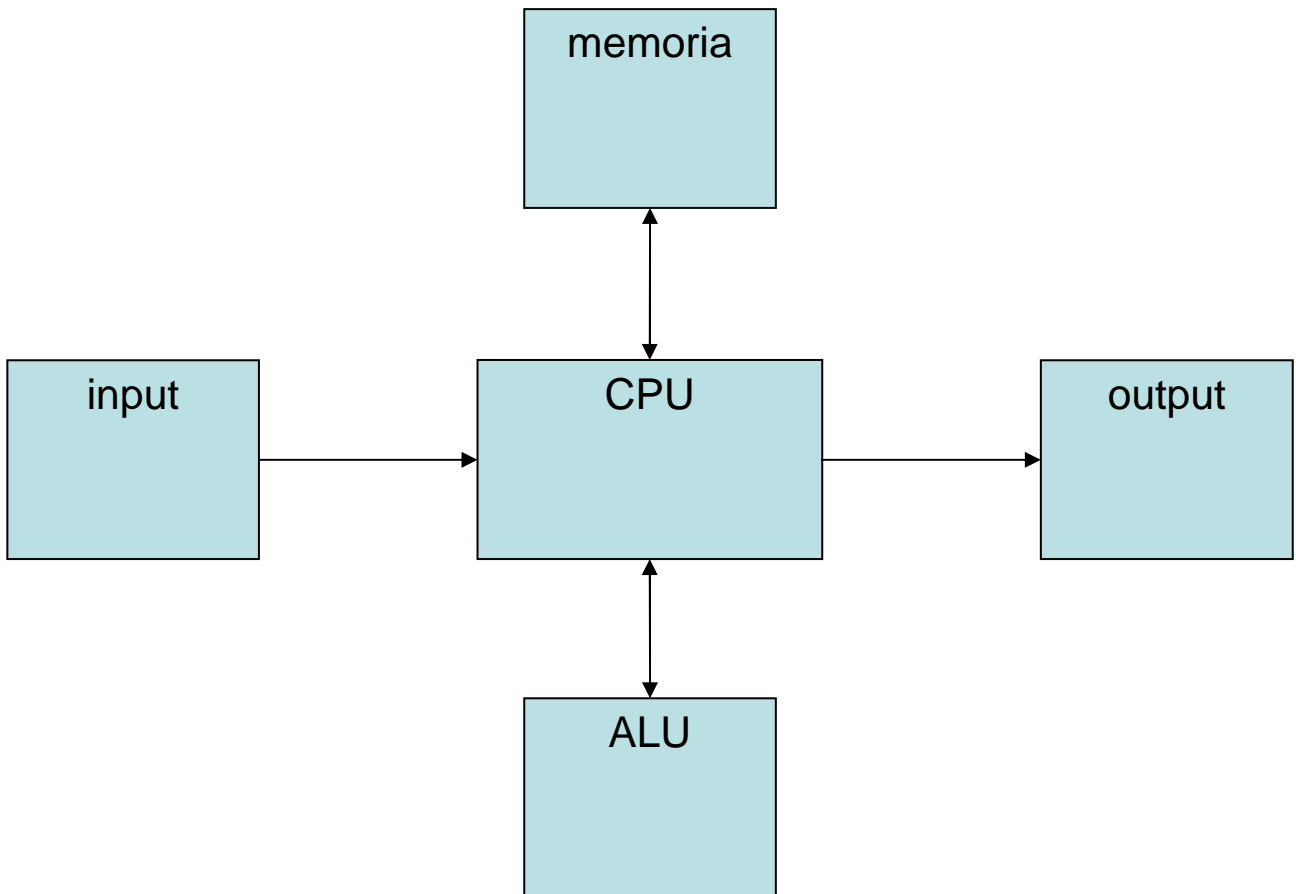
1.1.1 Algoritmo

Sequenza di istruzioni che bisogna fornire al soggetto dell'azione per fargliela svolgere allo scopo di risolvere un determinato problema. Il grado di scomposizione delle azioni in azioni elementari che costituiscono il procedimento per risolvere il problema dipendono dalle conoscenze del soggetto che deve seguire lo stesso algoritmo. Il design di un algoritmo viene infatti costruito attraverso una progettazione top-down ovvero suddividendo il problema in sotto-problemi.

INPUT → immettere una sequenza di dati o informazioni, verso il computer

OUTPUT → il computer mette a disposizione un dato

Il calcolatore comunica con l'ambiente esterno mediante delle interfacce di ingresso/uscita o input/output chiamate periferiche, che hanno il compito di tradurre i segnali che giungono dal calcolatore in informazioni comprensibili alle periferiche e viceversa.



Periferiche INPUT

Lettore CD
Mouse
Tastiere
Scanne
Internet
Microfono
Modem
Touch screen
Webcam
Firewire

Periferiche OUTPUT

Stampante
Monitor
USB
Audio
CD

Le informazioni vengono immesse nel calcolatore tramite le periferiche input per poi passare per la CPU che esegue gli algoritmi dalla memoria avvalendosi di una memoria temporanea e dell'ALU, unità in grado di svolgere problemi logico-aritmetici (**ARITHMETIC LOGIC UNIT**).

1.2 SISTEMI DI RAPPRESENTAZIONE

Alla base della formulazione di un algoritmo c'è il problema dell'elaborazione dell'informazione e quindi della rappresentazione dell'informazione.

L'informazione consiste nella ricezione di un messaggio dagli infiniti possibili significati che per essere codificato deve essere associato ad un numero finito di simboli.

1.2.1 Differenza tra sistema di numerazione romano ed arabo

Due dei più comuni sistemi di rappresentazione sono i sistemi numerazionali arabo e romano, attraverso i quali associamo dei simboli a delle quantità, processo fondamentale per affrontare problemi quali contare misurare etc

La differenza tra questi due sistemi di rappresentazione è che quello romano è un sistema di numerazione additivo il che significa che ad ogni numero viene associato un simbolo e il valore è dato dalla loro somma (es. CL: C= 100 L=50 \Rightarrow CL= 150), mentre quello Arabo è un sistema detto decimale posizionale ovvero ove la quantità viene associata non solo ai simboli ma anche alla loro posizione.

-Esercizi con rappresentazioni numerali e alfanumeriche posizionali in diverse basi

Premessa

Per “Base di un Sistema Matematico” si intende il numero di Simboli che esso ci mette a disposizione per rappresentare delle quantità.

Per esempio, il nostro Sistema Decimale si dice “in Base 10” perché vengono utilizzati i seguenti 10 simboli : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Le diverse combinazioni di questi pochi Simboli (cifre) ci permettono di esprimere quantità all'infinito perché il valore effettivo che ognuno di essi rappresenta è determinato dalla posizione che impegna all'interno del numero, così che, per esempio, nel numero **111** lo stesso simbolo “**1**” assume tre valori diversi (**1** centinaia, **1** decine, **1** unità) a seconda della colonna che occupa.

ES. Cento in base dieci: $100_{10} = 1 \times 10^2 + 0 \times 10^1 + 0 \times 10^0 = 100 + 0 + 0 = 100$

Dove l'elevazione del numero dieci alla zero alla prima e alla seconda sta ad indicare rispettivamente le unità le decine e le centinaia e quindi che scrivendo il numero 100 in realtà stiamo dicendo che nel numero sono presenti un elemento di valore 100 (10^2), zero elementi di valore **10** (10^1), zero elementi di valore **1** (10^0) del Sistema Decimale

E' importante quindi chiarire che il simbolo 100 sta ad indicare la quantità cento solo se ci riferiamo al sistema decimale, ovvero in base dieci.

Per essere chiari, lo stesso numero in base 16 starà ad indicare un'altra quantità

ES. $100_{16} = 1 \times 16^2 + 0 \times 16^1 + 0 \times 16^0 = 256$

Allo stesso modo potremmo utilizzare, nell'ambito dello stesso sistema in base 16, altri simboli come ad esempio i numeri arabi da 1 a 9 e i caratteri dell'alfabeto da A ad F (sistema esadecimale)

Con queste premesse calcoliamo la quantità riferita al simbolo FF.

$$\text{ES. } FF_{16} = 15 \times 16^1 + 15 \times 16^0 = 255$$

Possiamo quindi renderci conto che anche cambiando il tipo di rappresentazione il meccanismo rimane lo stesso.

Diventa quindi semplice capire che il numero 100 in un sistema che utilizza due simboli corrisponderà al numero 4 nel nostro posizionale decimale.

$$100_2 = 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 4$$

1.2.2 Sistema di numerazione binaria

DEFINIZIONE: Il sistema binario è un sistema posizionale in base due nel quale vengono utilizzati i due simboli zero ed uno.

premessa

Il sistema numerico binario è il sistema numerico posizionale che è stato scelto per essere utilizzato in informatica per la sua semplicità data dalla caratteristica di essere rintracciabile in natura con la corrente elettrica.

E' infatti possibile rappresentare i due simboli zero e uno rispettivamente con un circuito elettrico aperto o chiuso .

Il calcolatore dunque utilizza il sistema binario perché in esso la trasmissione dei dati avviene tramite circuiti elettrici.

Un Computer infatti, per poter funzionare, deve ricevere energia elettrica e lo scorrere di questo flusso può essere controllato da un interruttore il quale può assumere solo 2 posizioni (acceso/spento), così il calcolatore riesce ad interpretare quelli che per noi sono i numeri 1 e zero facendoli corrispondere a due concetti fondamentalmente diversi ovvero 0 = negativo o spento
1 = positivo o acceso.

Quando un circuito è chiuso, cioè percorso da corrente elettrica, la cifra binaria comunicata è 1; quando invece la corrente non passa, il computer trasmette 0.

Così una serie di impulsi elettrici sposta dati da un dispositivo del computer all'altro.

Circuiti elettrici

1.2.3 Bit e multipli

Avendo quindi premesso che i calcolatori utilizzano il sistema binario chiameremo **Bit (Binary Digit, cifra binaria)** ognuna di queste due cifre.

Il bit è dunque è la più piccola unità di informazione che un computer può elaborare o memorizzare, ogni bit che porta corrente è uguale ad uno (acceso) mentre i pacchetti che non portano corrente (spento) equivalgono a zero.

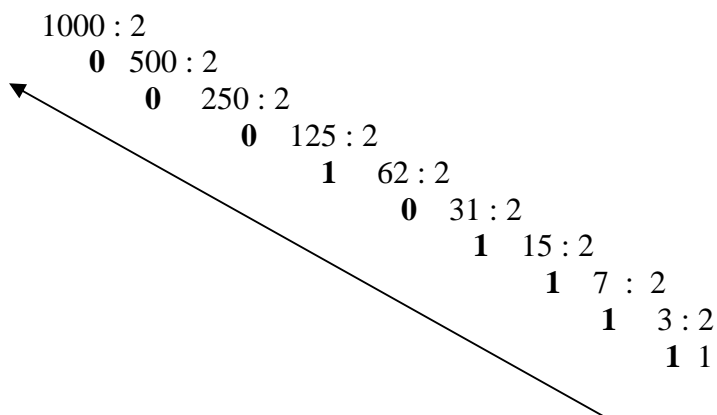
Pacchetti di otto bit costituiscono i byte, che codificano informazioni elementari quali le lettere dell'alfabeto e le cifre numeriche.

L'impacchettamento di 8 bit in un byte ci da infatti la possibilità di rappresentare 256 diversi simboli.

1.2.4 CONVERSIONE DECIMALE → BINARIO

La conversione di un numero decimale (es. 1000) in binario si effettua tramite divisione, dividendo successivamente per 2 (base del sistema binario) il numero decimale da convertire e considerando i resti al contrario.

ES. $1000 = 111101000$



1.2.5 RAPPRESENTAZIONI CON I BYTE

Rappresentazione dei numeri positivi e negativi

Il bit iniziale (più a sinistra) del numero indica il segno mentre il modulo si ottiene invertendo il valore dei singoli bit e aggiungendo 1 al numero binario risultante.

Se n è il numero di bit utilizzati, si possono rappresentare 2^n numeri. Così, considerando un pacchetto di 8 bit si possono rappresentare i numeri compresi tra -128 e +127.

Rappresentazione di numeri frazionari in “virgola mobile” (Floating point)

Nei numeri a virgola mobile la informazione relativa alle cifre significative viene archiviata con un gruppo di bit che viene detto **mantissa**. Il numero di zeri da aggiungere è conservato separatamente e viene detto **esponente**. Sia la mantissa che l'esponente sono quindi dei numeri interi e vengono

memorizzati come tali all'interno del pacchetto di bit.

Rappresentazione di numeri frazionari in "virgola fissa"

Il numero frazionario è rappresentato come una coppia di numeri interi: la parte intera e la parte decimale.

Rappresentazione caratteri alfanumerici

Per quanto riguarda i caratteri alfanumerici è stata adottata una convenzione che permette di tradurre l'intero byte grazie ad un codice chiamato [ASCII \(American Standard Code for Information Interchange\)](#), elaborato negli anni '70, tramite il quale viene associato ad ogni carattere un byte.

Questo codice dà la possibilità di tradurre 256 simboli (marca sempre solo un byte = 8 bit → $2^8 = 256$) tra cui le interpunzioni e le lettere maiuscole e minuscole.

ASCII Tabella dei caratteri

La tabella seguente è relativa al codice US ASCII, ANSI X3.4-1986 (*ISO 646 International Reference Version*). I codici decimali da 0 a 31 e il 127 sono caratteri non stampabili (*codici di controllo*). Il 32 corrisponde al carattere di "spazio". I codici dal 32 al 126 sono caratteri stampabili.

Legenda:

- Char - La rappresentazione stampabile del carattere, se presente
- Dec - Il codice decimale del carattere
- Row/Col - La rappresentazione decimale riga/colonna del carattere
- Oct - Il codice [ottale](#) del carattere
- Hex - Il codice [esadecimale](#) del carattere

Char	Dec	Col/Row	Oct	Hex	Nome	Descrizione
	0	00/00	0	0x00	NUL (Ctrl-@)	NULL
	1	00/01	1	0x01	SOH (Ctrl-A)	START OF HEADING
	2	00/02	2	0x02	STX (Ctrl-B)	START OF TEXT
	3	00/03	3	0x03	ETX (Ctrl-C)	END OF TEXT
	4	00/04	4	0x04	EOT (Ctrl-D)	END OF TRANSMISSION

	5	00/05	5	0x05	ENQ (Ctrl-E)	ENQUIRY
	6	00/06	6	0x06	ACK (Ctrl-F)	ACKNOWLEDGE
	7	00/07	7	0x07	BEL (Ctrl-G)	BELL (Beep)
	8	00/08	10	0x08	BS (Ctrl-H)	BACKSPACE
	9	00/09	11	0x09	HT (Ctrl-I)	HORIZONTAL TAB
	10	00/10	12	0x0A	LF (Ctrl-J)	LINE FEED
	11	00/11	13	0x0B	VT (Ctrl-K)	VERTICAL TAB
	12	00/12	14	0x0C	FF (Ctrl-L)	FORM FEED
	13	00/13	15	0x0D	CR (Ctrl-M)	CARRIAGE RETURN
	14	00/14	16	0x0E	SO (Ctrl-N)	SHIFT OUT
	15	00/15	17	0x0F	SI (Ctrl-O)	SHIFT IN
	16	01/00	20	0x10	DLE (Ctrl-P)	DATA LINK ESCAPE
	17	01/01	21	0x11	DC1 (Ctrl-Q)	DEVICE CONTROL 1 (XON)
	18	01/02	22	0x12	DC2 (Ctrl-R)	DEVICE CONTROL 2
	19	01/03	23	0x13	DC3 (Ctrl-S)	DEVICE CONTROL 3 (XOFF)
	20	01/04	24	0x14	DC4 (Ctrl-T)	DEVICE CONTROL 4
	21	01/05	25	0x15	NAK (Ctrl-U)	NEGATIVE ACKNOWLEDGE
	22	01/06	26	0x16	SYN (Ctrl-V)	SYNCHRONOUS IDLE
	23	01/07	27	0x17	ETB (Ctrl-W)	END OF TRANSMISSION BLOCK
	24	01/08	30	0x18	CAN (Ctrl-X)	CANCEL

	25	01/09	31	0x19	EM (Ctrl-Y)	END OF MEDIUM
	26	01/10	32	0x1A	SUB (Ctrl-Z)	SUBSTITUTE
	27	01/11	33	0x1B	ESC (Ctrl-])	ESCAPE
	28	01/12	34	0x1C	FS (Ctrl-)	FILE SEPARATOR
	29	01/13	35	0x1D	GS (Ctrl-])	GROUP SEPARATOR
	30	01/14	36	0x1E	RS (Ctrl-^)	RECORD SEPARATOR
	31	01/15	37	0x1F	US (Ctrl-_)	UNIT SEPARATOR
	32	02/00	40	0x20		SPACE
!	33	02/01	41	0x21		EXCLAMATION MARK
"	34	02/02	42	0x22		QUOTATION MARK
#	35	02/03	43	0x23		NUMBER SIGN
\$	36	02/04	44	0x24		DOLLAR SIGN
%	37	02/05	45	0x25		PERCENT SIGN
&	38	02/06	46	0x26		AMPERSAND
'	39	02/07	47	0x27		APOSTROPHE
(40	02/08	50	0x28		LEFT PARENTHESIS
)	41	02/09	51	0x29		RIGHT PARENTHESIS
*	42	02/10	52	0x2A		ASTERISK
+	43	02/11	53	0x2B		PLUS SIGN
,	44	02/12	54	0x2C		COMMA
-	45	02/13	55	0x2D		HYPHEN, MINUS SIGN
.	46	02/14	56	0x2E		PERIOD, FULL STOP

/	47	02/15	57	0x2F	SOLIDUS, SLASH
0	48	03/00	60	0x30	DIGIT ZERO
1	49	03/01	61	0x31	DIGIT ONE
2	50	03/02	62	0x32	DIGIT TWO
3	51	03/03	63	0x33	DIGIT THREE
4	52	03/04	64	0x34	DIGIT FOUR
5	53	03/05	65	0x35	DIGIT FIVE
6	54	03/06	66	0x36	DIGIT SIX
7	55	03/07	67	0x37	DIGIT SEVEN
8	56	03/08	70	0x38	DIGIT EIGHT
9	57	03/09	71	0x39	DIGIT NINE
:	58	03/10	72	0x3A	COLON
;	59	03/11	73	0x3B	SEMICOLON
<	60	03/12	74	0x3C	LESS-THAN SIGN, LEFT ANGLE BRACKET
=	61	03/13	75	0x3D	EQUALS SIGN
>	62	03/14	76	0x3E	GREATER-THAN SIGN, RIGHT ANGLE BRACKET
?	63	03/15	77	0x3F	QUESTION MARK
@	64	04/00	100	0x40	COMMERCIAL AT SIGN
A	65	04/01	101	0x41	CAPITAL LETTER A
B	66	04/02	102	0x42	CAPITAL LETTER B
C	67	04/03	103	0x43	CAPITAL LETTER C
D	68	04/04	104	0x44	CAPITAL LETTER D

E	69	04/05	105	0x45		CAPITAL LETTER E
F	70	04/06	106	0x46		CAPITAL LETTER F
G	71	04/07	107	0x47		CAPITAL LETTER G
H	72	04/08	110	0x48		CAPITAL LETTER H
I	73	04/09	111	0x49		CAPITAL LETTER I
J	74	04/10	112	0x4A		CAPITAL LETTER J
K	75	04/11	113	0x4B		CAPITAL LETTER K
L	76	04/12	114	0x4C		CAPITAL LETTER L
M	77	04/13	115	0x4D		CAPITAL LETTER M
N	78	04/14	116	0x4E		CAPITAL LETTER N
O	79	04/15	117	0x4F		CAPITAL LETTER O
P	80	05/00	120	0x50		CAPITAL LETTER P
Q	81	05/01	121	0x51		CAPITAL LETTER Q
R	82	05/02	122	0x52		CAPITAL LETTER R
S	83	05/03	123	0x53		CAPITAL LETTER S
T	84	05/04	124	0x54		CAPITAL LETTER T
U	85	05/05	125	0x55		CAPITAL LETTER U
V	86	05/06	126	0x56		CAPITAL LETTER V
W	87	05/07	127	0x57		CAPITAL LETTER W
X	88	05/08	130	0x58		CAPITAL LETTER X
Y	89	05/09	131	0x59		CAPITAL LETTER Y
Z	90	05/10	132	0x5A		CAPITAL LETTER Z

[91	05/11	133	0x5B	LEFT SQUARE BRACKET
\	92	05/12	134	0x5C	REVERSE SOLIDUS (BACKSLASH)
]	93	05/13	135	0x5D	RIGHT SQUARE BRACKET
^	94	05/14	136	0x5E	CIRCUMFLEX ACCENT
_	95	05/15	137	0x5F	LOW LINE, UNDERLINE
`	96	06/00	140	0x60	GRAVE ACCENT
a	97	06/01	141	0x61	SMALL LETTER a
b	98	06/02	142	0x62	SMALL LETTER b
c	99	06/03	143	0x63	SMALL LETTER c
d	100	06/04	144	0x64	SMALL LETTER d
e	101	06/05	145	0x65	SMALL LETTER e
f	102	06/06	146	0x66	SMALL LETTER f
g	103	06/07	147	0x67	SMALL LETTER g
h	104	06/08	150	0x68	SMALL LETTER h
i	105	06/09	151	0x69	SMALL LETTER i
j	106	06/10	152	0x6A	SMALL LETTER j
k	107	06/11	153	0x6B	SMALL LETTER k
l	108	06/12	154	0x6C	SMALL LETTER l
m	109	06/13	155	0x6D	SMALL LETTER m
n	110	06/14	156	0x6E	SMALL LETTER n
o	111	06/15	157	0x6F	SMALL LETTER o
p	112	07/00	160	0x70	SMALL LETTER p

q	113	07/01	161	0x71		SMALL LETTER q
r	114	07/02	162	0x72		SMALL LETTER r
s	115	07/03	163	0x73		SMALL LETTER s
t	116	07/04	164	0x74		SMALL LETTER t
u	117	07/05	165	0x75		SMALL LETTER u
v	118	07/06	166	0x76		SMALL LETTER v
w	119	07/07	167	0x77		SMALL LETTER w
x	120	07/08	170	0x78		SMALL LETTER x
y	121	07/09	171	0x79		SMALL LETTER y
z	122	07/10	172	0x7A		SMALL LETTER z
{	123	07/11	173	0x7B		LEFT CURLY BRACKET, LEFT BRACE
	124	07/12	174	0x7C		VERTICAL LINE, VERTICAL BAR
}	125	07/13	175	0x7D		RIGHT CURLY BRACKET, RIGHT BRACE
~	126	07/14	176	0x7E		TILDE
	127	07/15	177	0x7F		DELETE

Negli anni '90 con la globalizzazione è diventato evidente che la tabella ASCII, codificando solo 256 caratteri, non era sufficiente per coprire altri alfabeti come ad esempio quello giapponese. Per risolvere questo problema si è quindi passati ad una rappresentazione con due byte, raggiungendo la possibilità di codificare fino a 16536 simboli, fino all'attuale estensione da due a quattro byte rappresentata da un codice chiamato [UNICODE](#).

Rappresentazioni per dati strutturati

I caratteri strutturati sono quelli rappresentati da caratteri numerici e alfanumerici come ad esempio le date, codici fiscali (a lunghezza fissa) l'e-mail, i numeri telefonici (a lunghezza variabile) etc. In questo caso è l'algoritmo stesso che decide il metodo di rappresentazione del carattere. Tipicamente per quanto riguarda i caratteri strutturati a lunghezza variabile si utilizza una sequenza di byte in cui il primo porta l'informazione della lunghezza della parola in modo tale che si andranno poi a leggere solo il numero di byte necessari.