

**Università degli Studi della Calabria**

**Facoltà di Ingegneria**

**CORSO DI**

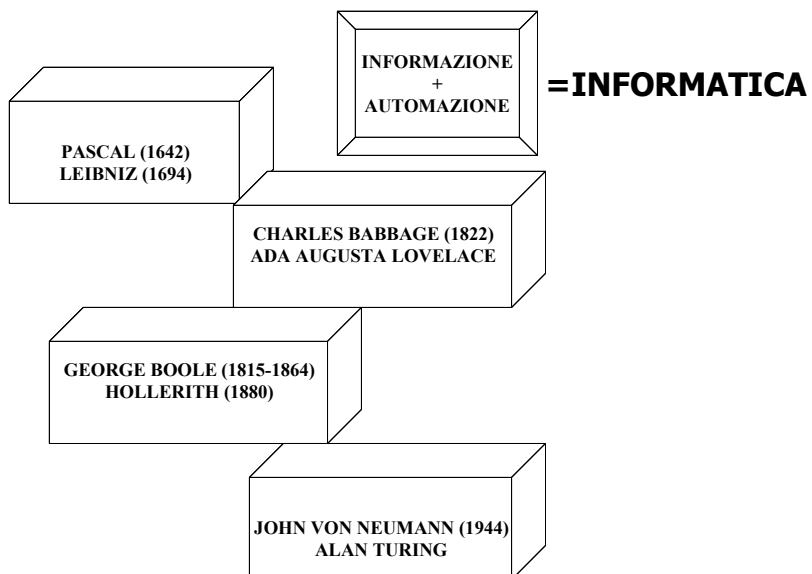
**INTRODUZIONE ALL' INFORMATICA**

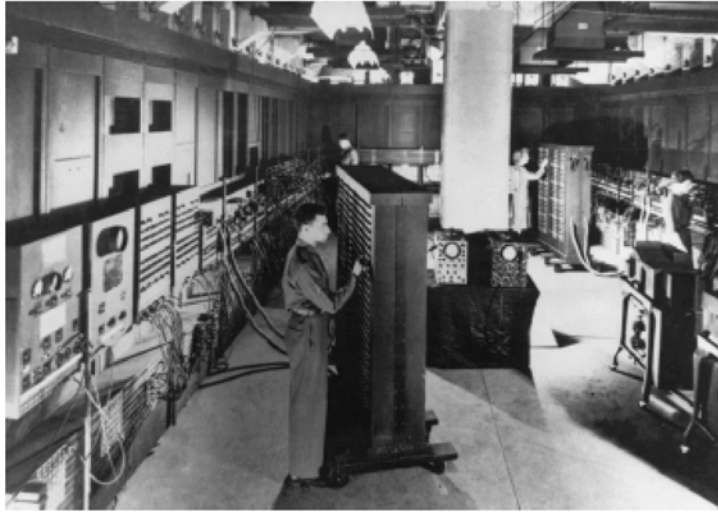
**A.A 2002/2003**

Docente : Domenico Talia (talia@deis.unical.it)

Esercitori: P. Trunfio, A. Congiusta

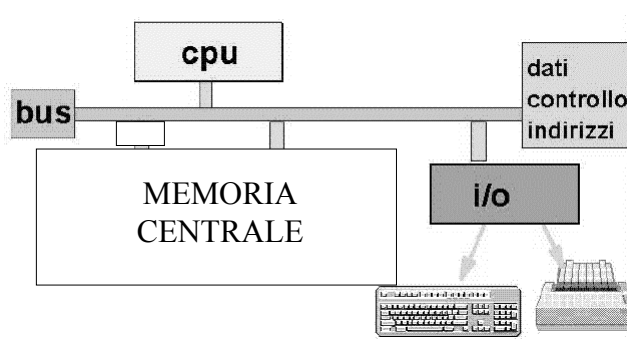
## CENNI STORICI





Il calcolatore ENIAC

### Architettura della macchina di Von Neumann



**CPU (Central Processing Unit), o Processore:** svolge le elaborazioni e il trasferimento dei dati, cioè esegue i programmi

**RAM (Random Access Memory) e ROM (Read Only Memory) :** Insieme formano la **Memoria centrale**.

- Dimensioni relativamente limitate
- Accesso estremamente rapido

**RAM** è volatile (perde il suo contenuto quando si spegne il calcolatore) ed è usata per memorizzare dati e programmi.

**ROM** è persistente (mantiene il suo contenuto quando si spegne il calcolatore) ma il suo contenuto è fisso e immutabile. È usata per memorizzare programmi di sistema.

### **Bus di sistema**

Il **BUS** è il componente del calcolatore dedicato al trasferimento dei dati e delle informazioni di controllo tra le varie parti del calcolatore

Il BUS è l'insieme dei collegamenti su cui vengono trasferiti i dati e i segnali di controllo in un calcolatore

L'idea alla base del BUS è la seguente:

- nel calcolatore è necessario collegare tutti i componenti tra di loro (per permettere lo scambio di dati tra componenti)
- ci sono due modalità per collegare un insieme di componenti collegare ciascun componente con ogni altro componente collegare tutti i componenti a un unico insieme di linee (il BUS, appunto)

**L'uso del BUS favorisce la modularità e l'espandibilità del calcolatore.**

## DISPOSITIVI DI INPUT

Tastiera

Mouse

trackball

Touch pad

Microfono

Tavoletta grafica

Webcam

Penna ottica

Scanner

Fotocam. Dig.

Schermi touch screen

Joystick

## DISPOSITIVI DI OUTPUT

Monitor

Plotter

Stampante

Casse acustiche

Videoproiettore

Cuffie

## TECNOLOGIA DIGITALE

CPU, memoria centrale e dispositivi sono realizzati con **tecnologia elettronica digitale**.

Dati ed operazioni vengono codificati a partire da due valori distinti di grandezze elettriche:

tensione alta e tensione bassa a cui vengono convenzionalmente **associate le due cifre binarie 0 e 1**

**Bit** (*Binary digIT*): due valori possibili:

Dati e operazioni vengono codificati con **sequenze di bit**:  
01000110101 ....

Le operazioni aritmetiche che la CPU è in grado di eseguire operano in aritmetica binaria.

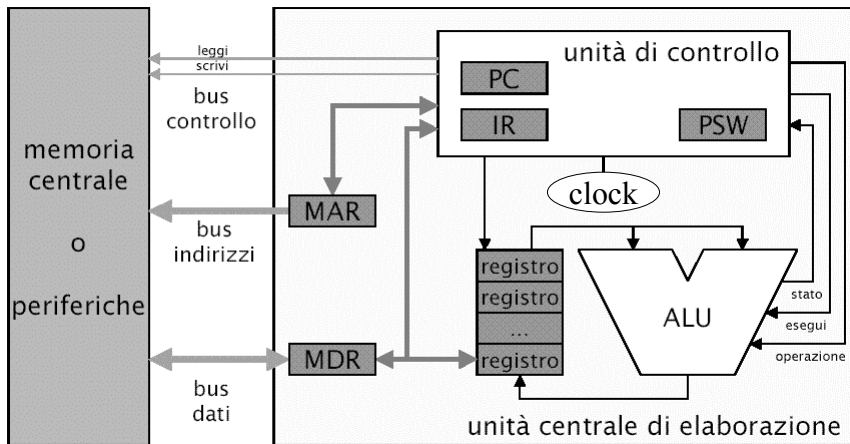
Si tratta di operazioni *elementari*: somma, differenza, scorrimento, ...

◆ Lavorando direttamente sull'hardware, l'utente è forzato a esprimere i propri comandi *al livello della macchina*, tramite *sequenze di bit*.

◆ Una importante unità di informazione è il **byte**: un **byte** è una sequenza di 8 bit

Le possibili combinazioni degli 8 bit in un byte sono  $2^8 = 256$   
Quindi un byte può essere utilizzato per rappresentare un valore tra 256 diversi possibili valori, ad esempio, un numero naturale compreso nell'intervallo da 0 a 255, oppure un numero intero relativo nell'intervallo da -128 a 127

## Unità Centrale di Elaborazione (Central Processing Unit, CPU) e collegamento con le altre unità funzionali.



La CPU è fisicamente realizzata sotto forma di **microprocessore**.

La CPU contiene un numero limitato di celle di memoria (chiamate **registri**) con scopi specifici.

- **PC, Program Counter** : registro contatore delle istruzioni, contiene l'indirizzo della prossima istruzione da eseguire
- **IR, Instruction Register** : registro delle istruzioni, contiene l'istruzione che deve essere eseguita (codificata)
- **MAR** : registro di indirizzamento della memoria, contiene l'indirizzo della cella di memoria che deve essere acceduta o memorizzata
- **MDR** : registro dati di memoria, dato che è stato acceduto o che deve essere memorizzato
- **PSW** : parola di stato del processore contiene informazioni, opportunamente codificate, circa l'esito dell'ultima istruzione che è stata eseguita

## UNITA' DI CONTROLLO

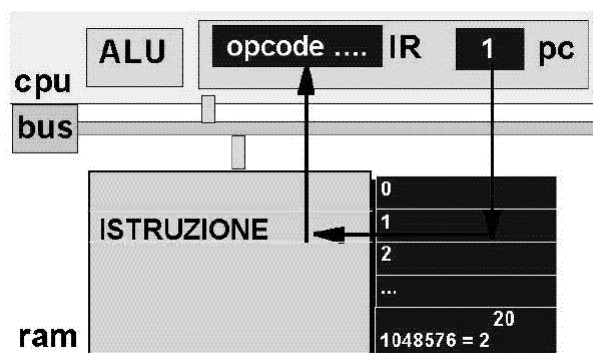


L'unità di controllo fa funzionare l'elaboratore, da quando viene acceso a quando viene spento.

Essa esegue in continuazione il cosiddetto **ciclo di accesso / decodifica / esecuzione** ( *fetch / decode / execute* )

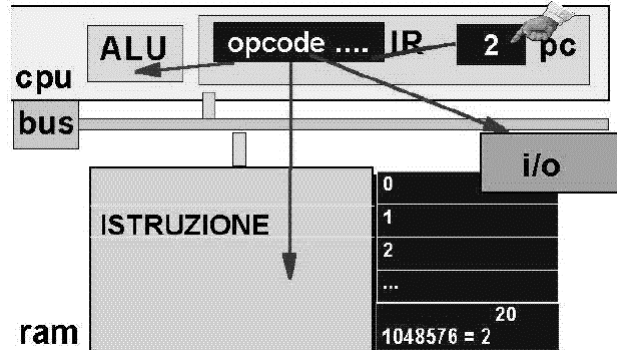
## IL CICLO *fetch/ decode / execute*

1) **FASE DI FETCH** : si accede alla prossima istruzione (puntata da PC) e la si porta dalla memoria centrale al Registro Istruzioni ( IR )



**2) FASE DI DECODE:** si decodifica il tipo dell'istruzione in base al suo *OpCode*, si individuano i dati usati dall'istruzione, e li si trasferisce nei registri opportuni

**3) FASE DI EXECUTE :** si esegue l'istruzione



L'esecuzione di un programma avviene eseguendo ordinatamente le istruzioni del programma, svolgendo ripetutamente il ciclo fetch-decode-execute

**UNITÀ LOGICO-ARITMETICA (ALU):** esegue operazioni aritmetiche e confronti sui dati della memoria centrale o dei registri.

**MEMORIA CENTRALE :** contiene tutta l'informazione che deve essere elaborata "a breve termine". Ogni elemento di informazione, per poter essere elaborato, **deve** prima essere acquisito in memoria centrale.

**RAM è volatile:** se ne perde il contenuto quando la macchina viene spenta.

Ha un **tempo di accesso molto basso** (oggi tipicamente 60-70 ns, dove  $1 \text{ ns} = 10^{-9} \text{ s}$ )

Può essere pensata come una **sequenza di celle** (o "locazioni") di memoria. Ogni cella è identificata da un **indirizzo** e memorizza una sequenza di **cifre binarie (valore)**. Tale sequenza viene chiamata **parola (word)**.



Per ogni elaboratore, **la dimensione della parola** (cioè la “capienza” della cella di memoria) è **fissa**.

Elaboratori diversi possono avere parole di lunghezza differente (8, 16, 32, 64 bit).

**Unità di misura:**

**Byte = 8 bit**

**Multipli:**

**Kilobyte (KB)**  $2^{10}$  byte = 1024 byte

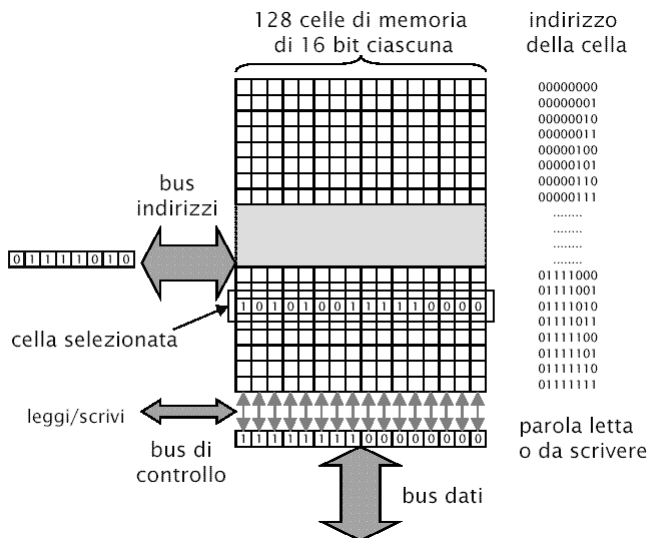
**Megabyte (MB)**  $2^{20}$  byte = 1048576 byte

**Gigabyte (GB)**  $2^{30}$  byte  $\approx 10^9$  byte

**Terabyte (TB)**  $2^{40}$  byte  $\approx 10^{12}$  byte

**Petabyte (PB)**  $2^{50}$  byte  $\approx 10^{15}$  byte

## Struttura di una memoria centrale



## INDIRIZZAMENTO

L'elaboratore seleziona una particolare cella di memoria ponendone l'indirizzo nel **Registro Indirizzi (RI)**.

Se il Registro Indirizzi (RI) è lungo  $N$  bit, si possono indirizzare  $2^N$  celle di memoria, da 0 a  $2^N - 1$  (ad es., se  $N=10$  si possono indirizzare 1024 celle).

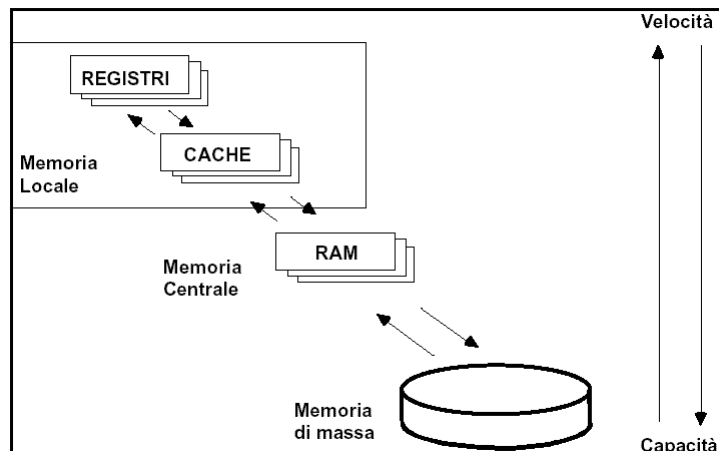
OGGI: RI è almeno di 32 bit

Operazione di **lettura (fetch)** : Copia nel Registro Dati il contenuto della cella di memoria indirizzata dal Registro Indirizzi.

Operazione di **scrittura (store)** : Copia il contenuto del Registro Dati nella cella di memoria indirizzata dal Registro Indirizzi.

Le operazioni avvengono sotto il controllo della CPU.

## GERARCHIA DELLE MEMORIE



## MEMORIA CACHE

Sebbene la RAM sia veloce, non è **abbastanza** veloce da “star dietro” ai moderni processori: il processore perde tempo ad aspettare l’arrivo dei dati dalla RAM.

**SOLUZIONE** : inserire **tra processore e RAM** una **memoria particolarmente veloce** dove tenere i dati usati più spesso (**MEMORIA CACHE**). Così:

“ la prima volta che il microprocessore carica dei dati dalla memoria centrale, tali dati vengono caricati anche sulla cache

“ le volte successive, i dati possono essere letti dalla cache invece che dalla memoria centrale (più lenta)

Se abbiamo memorie veloci, perché non facciamo con esse tutta la RAM? Perché tali memorie veloci sono **molto costose!** **OGGI**, la cache è spesso già presente dentro al microprocessore (**cache di I° livello**) e altra può essere aggiunta (**cache di II° livello**)

## Unità di misura e Abbreviazioni

### TEMPO

- **1 secondo**
- **10<sup>-3</sup> milli m**
- **10<sup>-6</sup> micro  $\mu$**
- **10<sup>-9</sup> nano n**
- **10<sup>-12</sup> pico p**

### FREQUENZA

- **hertz (1 ciclo al sec.) Hz,hz**
- **e quindi : Khz, Mhz, Ghz**
- **rounds per minute rpm**

## TRASMISSIONE

- bits per second bps
- e quindi Kbps, Mbps, Gbps

## VELOCITÀ DEI SISTEMI DI ELABORAZIONE

- instructions per sec. ips
- I/O operations per sec. I/Ops
- transactions per sec. tps

attenzione a:

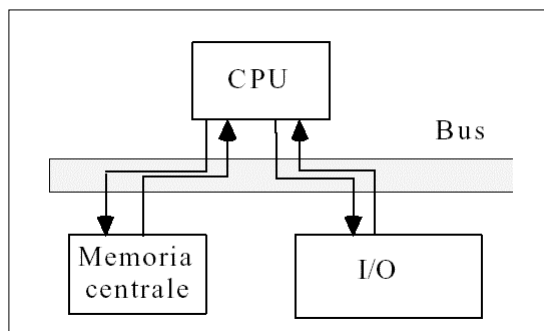
- inch (2.54 cm) 1m = 39,37 inch
- foot (12 inch) 1m = 3,28 feet
- $32\text{ F}^\circ = 0\text{ C}^\circ$     $212\text{ F}^\circ = 100\text{ C}^\circ$

## BUS DI SISTEMA

Interconnette la CPU, la memorie e le interfacce verso dispositivi periferici (I/O, memoria di massa, etc.)

Collega **due unità funzionali alla volta**: una trasmette e l'altra riceve.

Il trasferimento dei dati avviene sotto il controllo della CPU.



Su questo supporto (spesso costituito da più linee) viaggiano dati, indirizzi e comandi. Si distinguono spesso tali linee in:

- **Bus dati (data bus)** : bidirezionale. Serve per trasmettere dati dalla memoria al registro dati o viceversa.
- **Bus indirizzi (address bus)** : unidirezionale. Serve per trasmettere il contenuto del registro indirizzi alla memoria. Viene selezionata una specifica cella per successive operazioni di lettura o scrittura.
- **Bus comandi (command bus)** : unidirezionale. Serve per inviare comandi verso la memoria (es: lettura o scrittura) o verso una periferica (es. stampa verso la stampante / interfaccia).

## **MEMORIA DI MASSA**

**Scopo:** memorizzare *grandi masse* di dati in modo *persistente*

I dati memorizzati su questo tipo di memoria sopravvivono all'esecuzione dei programmi.

### **Altre caratteristiche:**

· **Capacità** (dimensione della memoria)

Varia molto da dispositivo a dispositivo

· **Unità di misura: Byte** (collezione di 8 bit) e multipli:

MegaByte = 1.048.576 Byte

GigaByte = 1.073.741.824 Byte

TeraByte = 1.099.511.627.776 Byte

- **Tempo di accesso**

Anche il tempo di accesso varia da dispositivo a dispositivo, ma è comunque *molto superiore* a quello della memoria centrale.

**Taccesso** a memoria centrale  $\approx 60-70$  nsec ( $10^{-9}$  sec)

**Taccesso** a dischi fissi  $\approx 10-15$  msec ( $10^{-3}$  sec)

**Taccesso** a dischetti (floppy)  $\approx 100$  msec

- Capacità disco fisso (PC)  $\approx 2 - 6$  Gbyte
- Capacità disco floppy  $\approx 1,4$  Mbyte

## **DISPOSITIVI DI MEMORIA DI MASSA**

**Due classi fondamentali** di dispositivi in base al metodo di accesso consentito:

1. **ad accesso sequenziale** (ad esempio, **NASTRI**): per recuperare un dato è necessario accedere prima a tutti quelli che lo precedono sul dispositivo;

2. **ad accesso diretto (DISCHI)**: è possibile recuperare direttamente un qualunque dato memorizzato.

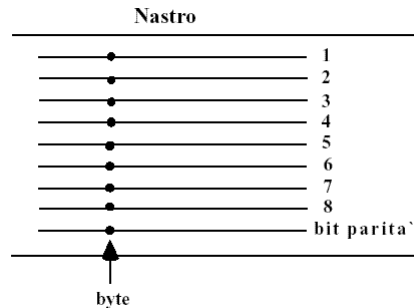
**Dispositivi magnetici** (nastri o dischi): l'area del dispositivo è suddivisa in micro-zone ogni micro-zona memorizza una informazione elementare sotto forma di *stato di magnetizzazione* : area magnetizzata / area non magnetizzata

ai due possibili stati di magnetizzazione vengono **associate le due cifre binarie 0 e 1 (bit** : Binary digIT)

## NASTRI MAGNETICI

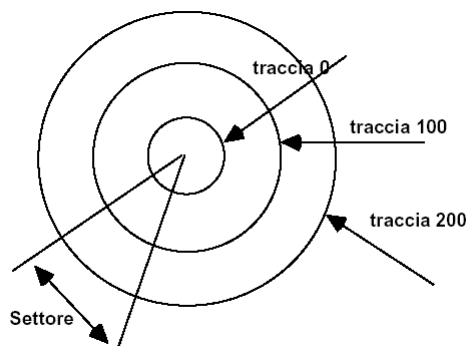
Sono nastri di materiale magnetizzabile raccolti su supporti circolari, o in cassette.

Sul nastro sono tracciate delle piste orizzontali parallele. Di solito, 9 piste parallele di cui 8 corrispondono ad un byte e la nona è il bit di parità.



I dati sul nastro sono organizzati in zone contigue dette *record*, separate da zone prive di informazione (*interrecord gap*).

## DISCHI MAGNETICI



Un disco consiste in un certo numero di *piatti* con due superfici che ruotano attorno ad un perno centrale. Ogni superficie dispone di una propria testina di lettura / scrittura.

Le superfici sono organizzate in *cerchi concentrici* (**tracce**) e in *spicchi* di ugual grandezza (**settori**). Le tracce equidistanti dal centro formano un **cilindro**. I dati sono scritti occupando posizioni successive lungo le tracce, e corrispondono ad uno stato di *polarizzazione* (positiva o negativa) del materiale magnetico che costituisce i dischi.

Ogni *blocco* del disco è identificato con la terna

⟨*superficie, traccia, settore*⟩

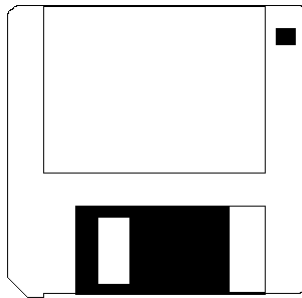
Per effettuare il trasferimento dei dati in memoria centrale occorre disporre di un'area di memoria (*buffer*) di dimensioni pari al blocco.

1. spostamento della testina (*seek*) verso la traccia richiesta;
2. attesa che il settore arrivi sotto la testina;
3. trasferimento dei dati in / da memoria centrale, solitamente eseguito da un processore dedicato (Direct Memory Access, DMA).

**Calcolo del tempo di accesso:**

$$T_{i/o} = T_{seek} + T_{rotazione} / 2 + T_{trasferimento}$$

**DISCHETTI (FLOPPY DISK)**



Sono dischi magnetici di piccola capacità, portatili, usati per trasferire informazioni (file) tra computer diversi. Sono costituiti da un unico disco con due superfici.

Storicamente ne sono stati creati vari tipi, identificati dal loro diametro (3.5, 5.25 e 8 pollici).

OGGI sopravvivono solo dischetti da 3.5" (1.4 Mbyte)



## **DISPOSITIVI OTTICI E MAGNETO-OTTICI**

**1984, CD-ROM (Compact-Disk Read-Only Memory)** Capacità di oltre 600 Mbyte e costo inferiore a \$1

Velocità di trasferimento:

- originariamente 150 Kbyte / secondo ( "1X" )
- OGGI: 12, 16, 24 volte tanto...

### **1984, WORM (Write Once Read Many)**

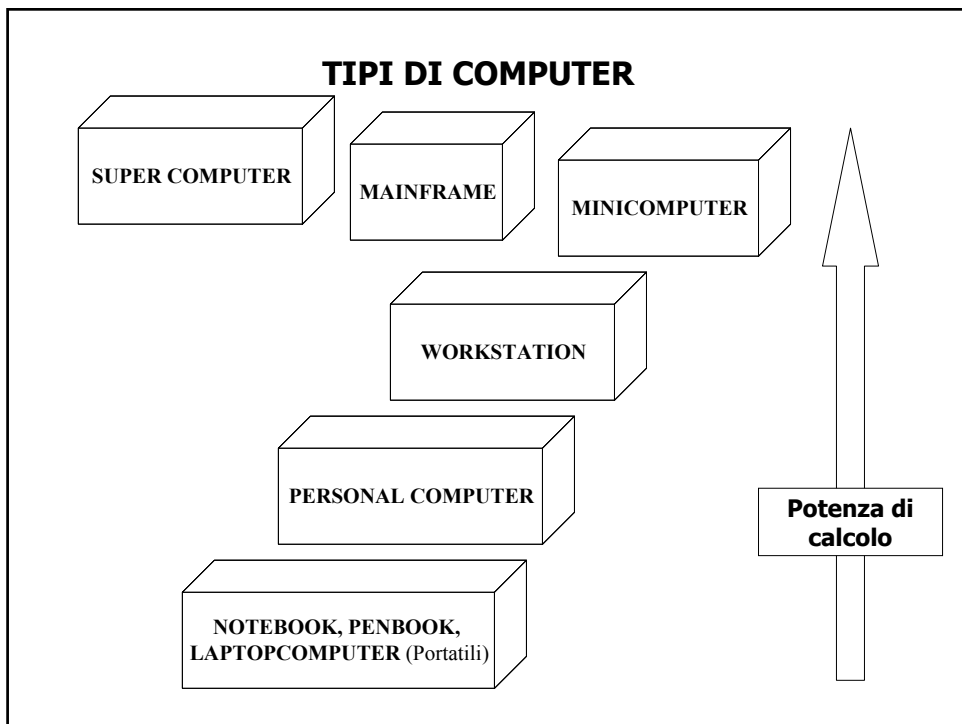
Sono dischi ottici scrivibili (una sola volta) Parenti stretti dei CD audio (CD-DA, 1982) Accesso diretto ai settori (capacità 2.048 Kbyte)

### **1997, DVD (Digital Video Disk)**

Evoluzione del CD-ROM. Capacità fino a 17 Gbyte. Velocità di trasferimento molto elevata

## **CAPACITÀ DELLE MEMORIE**

<b>Tipo di memoria</b>	<b>Capacità</b>
Memoria centrale	8-512 Mbyte
Dischi magnetici	800 Mbyte -10 Gbyte
Dischi floppy	1.4 Mbyte
Nastri (bobina)	20-400 Mbyte
Nastri (cassetta)	200-5000 Mbyte
Dischi ottici	650 Mbyte - 17 GByte



### PERSONAL COMPUTER – PC

Usano processori della famiglia *Intel 80x86* :

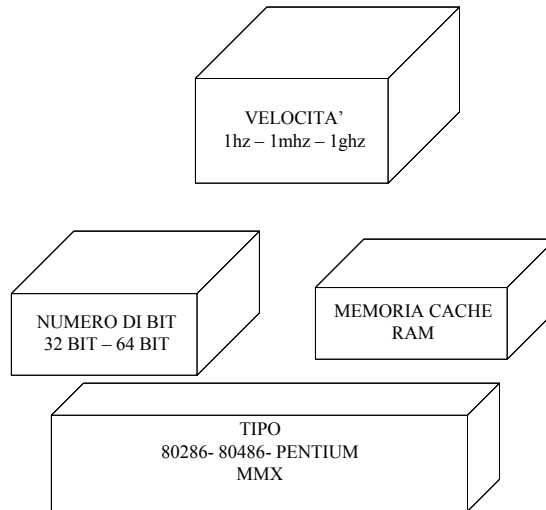
8086		prestazioni
80286		
80386		
80486		
Pentium		
Pentium Pro		
...		

Le prestazioni sono influenzate anche da altri parametri: frequenza dell'orologio di sistema (*clock*), dimensione della memoria centrale (RAM), velocità/parallelismo delle linee dati/comandi (bus)

Unità di misura delle prestazioni: **MIPS** (migliaia di istruzioni per secondo) e **Mflops** (migliaia di operazioni in virgola mobile per secondo)

**Macintosh:** usano processori *Motorola 68xxx* e *PowerPC*

## CLASSIFICAZIONE DEI MICROPROCESSORI



## ALTRI SISTEMI DI CALCOLO

**Workstation:** sistemi con capacità di supportare più attività contemporanee, spesso dedicati a più utenti. Prestazioni normalmente superiori a quello di un tipico Personal Computer.

**Mini-calcolatori :** macchine di medio-alta potenza, capaci di servire fino ad un centinaio di utenti contemporaneamente, collegati tramite terminali.

**Mainframe:** macchine di grandi dimensioni , costosi e sofisticati, che possono essere utilizzati da molti utenti contemporaneamente mediante terminali.

**Super-calcolatori :** hanno molti processori, grandi memorie di massa e servono tipicamente centinaia o migliaia di terminali.

Possibilità di connettere assieme vari calcolatori anche di tipo diverso  
→ **Reti di calcolatori**

## SOFTWARE

Software = insieme (complesso) di programmi.

**Organizzazione a strati**, ciascuno con funzionalità di livello più alto rispetto a quelli sottostanti:

### ***“Macchina Virtuale”***



**Firmware:** confine fra hardware e software. È uno strato di *microprogrammi* scritti dai costruttori, che agiscono direttamente al di sopra dello strato hardware. Sono memorizzati su una speciale *memoria centrale permanente*.

## SISTEMA OPERATIVO

Strato di programmi che opera *al di sopra di hardware e firmware* e **gestisce l'elaboratore**.

**Spesso si può scegliere tra diversi sistemi operativi** per lo stesso elaboratore, con diverse caratteristiche.

Le funzioni messe a disposizione dal S.O. dipendono dalla complessità del sistema di elaborazione:

- gestione delle risorse disponibili
- gestione della memoria centrale
- organizzazione e gestione della memoria di massa
- interpretazione ed esecuzione di comandi elementari
- gestione di un sistema multi-utente

**Un utente "vede" l'elaboratore solo tramite il Sistema Operativo, quindi il S.O. realizza una "macchina virtuale"**

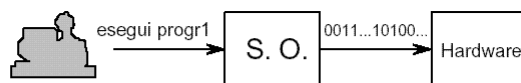
→ Diversi S.O. possono realizzare diverse macchine virtuali *sullo stesso elaboratore fisico*

Attraverso il S.O. il livello di interazione utente / elaboratore viene elevato:

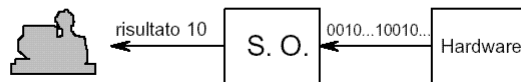
- senza S.O.: sequenze di bit
- con S.O.: comandi, programmi, dati.

Il S.O. **traduce le richieste dell'utente** in opportune **sequenze di valori e comandi** da sottoporre alla macchina fisica.

**Qualsiasi operazione di accesso a risorse** della macchina implicitamente richiesta dal comando di utente **viene esplicitata dal S.O.**



e viceversa:



### **Utente S.O.**

**"esegui progr1"** - input da tastiera

- ricerca codice di "**progr1**" su disco
- carica in memoria centrale codice e dati
- <elaborazione>

**"risultato 10"** - output su video

## CLASSIFICAZIONE dei sistemi operativi

### In base al numero di utenti:

- **Mono-utente (*mono-user*):** un solo utente alla volta può utilizzare il sistema
- **Multi-utente (*multi-user*):** più utenti in contemporanea possono interagire con la macchina.

Nel caso di più utenti contemporanei, *il S.O. deve fornire a ciascuno l'astrazione di un sistema "dedicato"*.

### In base al numero di programmi in esecuzione:

- **Mono-programmato (*mono-task*):** si può eseguire *un solo programma* per volta.
- **Multi-programmato (*multi-task*):** il S.O. è in grado di portare avanti in contemporanea l'esecuzione di più programmi (pur usando una sola CPU). In questo caso il S.O. deve gestire la suddivisione del tempo dell'unità di elaborazione (CPU) fra i vari programmi (*time-sharing*).

## SISTEMI OPERATIVI

DOS	Computer IBM compatibili con architettura 80x86, monoutente, monoprogrammato
Windows 3.1	Per le architetture a partire da 80386, monoutente, multiprogrammato
Windows '9x	Per le architetture a partire da 80486, monoutente, multiprogrammato
Windows NT	Ambiente di rete per le architetture a partire da 80486, multiutente, multiprogrammato
OS/2	Ambiente per le architetture a partire da 80486, monoutente, multiprogrammato
Unix	Ambiente vasto e potente, capace di ospitarne altri, montato su gran parte delle workstation, multiutente, multiprogrammato
MacOS	Noto come Macintosh, dal nome del computer che lo adotta, monoutente, multiprogrammato

## GUI : L'INTERFACCIA GRAFICA

### GUI ( Graphical User Interface) :

- Prima dell'avvento di windows95 i computer all'accensione visualizzavano il prompt del DOS, cioè l'avviso che la macchina era pronta a ricevere i comandi.
- La situazione è cambiata grazie alla nascita delle interfacce grafiche che traducono in termini grafici la stessa situazione.
- Con l'interfaccia grafica tutti i programmi e le funzioni sono mostrate sullo schermo mediante simboli immediatamente comprensibili denominate icone.
- La prima interfaccia grafica è stata realizzata nei computer Apple.

## COMANDI E FUNZIONI DEL S.O. DOS/WIN

- **COMMAND.COM** : interprete dei comandi di sistema, non è modificabile dall'utente;
  - **CONFIG.SYS** : in questo file ci sono le istruzioni per caricare i driver, ossia i file pilota che consentono il riconoscimento delle periferiche;
  - **AUTOEXEC.BAT** : sono contenuti i veri e propri comandi di avvio del computer;
- Persino windows 98 si appoggia sul DOS e questi file.

In questi tre file è racchiusa gran parte delle informazioni per l'avvio e il corretto funzionamento del computer;

**PLUG AND PLAY (PnP)** : L'inserimento di una nuova scheda all'interno del computer è operazione alquanto semplice, ma la sua configurazione per il riconoscimento della periferica da parte del sistema operativo e per l'eliminazione dei conflitti con altre schede installate richiede spesso competenze specifiche. Con windows 98 il problema sembra superato; il S.O. è in grado di riconoscere un gran numero di componenti hardware e di configurarli autonomamente