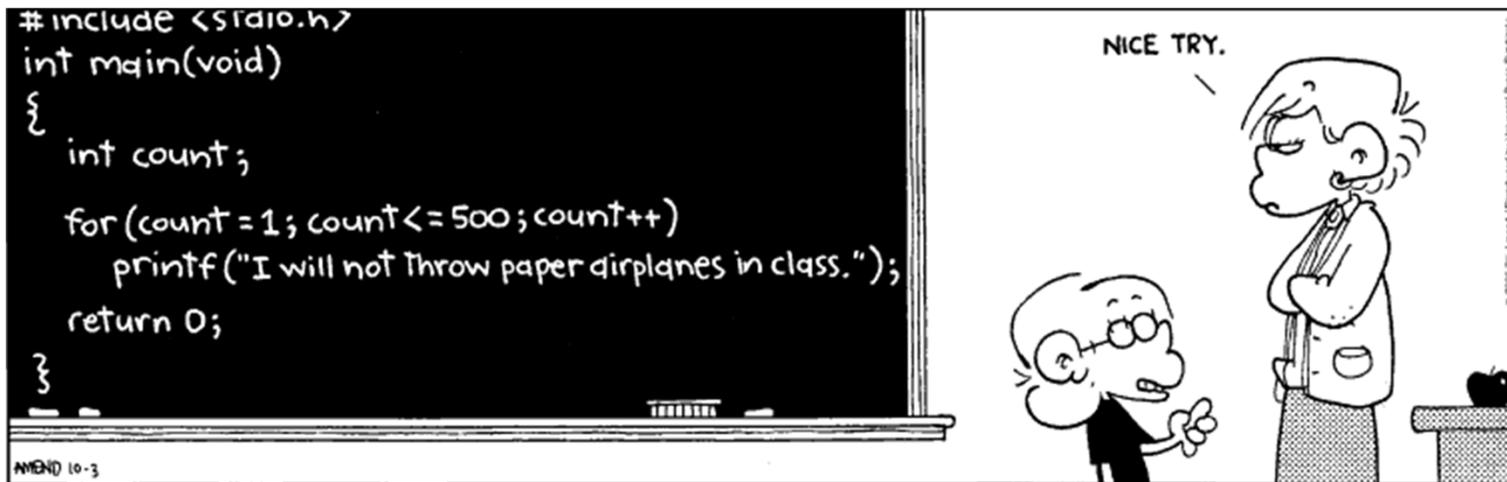


# Piccoli Ingegneri Informatici...



- La disumanità del computer sta nel fatto che una volta programmato e messo in funzione, si comporta in maniera *perfettamente onesta*.(Isac Azimov)
- I computer sono incredibilmente veloci, accurati e stupidi. Gli uomini sono incredibilmente lenti, imprecisi ed intelligenti. L'insieme dei due costituisce una forza incalcolabile.(Albert Einstein)

# INFORMATICA

---

- Varie definizioni:
  - “Scienza degli elaboratori elettronici”  
(*Computer Science*)
  - “Scienza dell’informazione”
- Definizione proposta:
  - ***Scienza della rappresentazione e dell’elaborazione dell’informazione***

# L'informatica comprende:

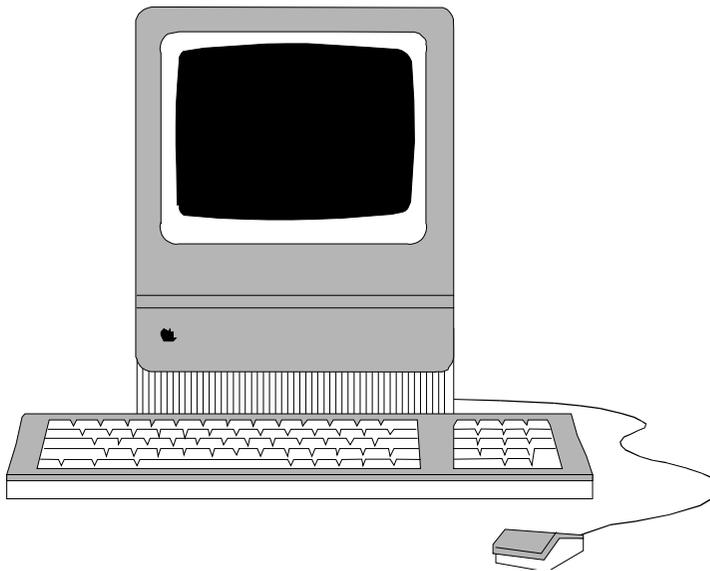
---

- Metodi per la **rappresentazione** delle informazioni
- Metodi per la **rappresentazione** delle soluzioni
- **Linguaggi di programmazione**
- **Architettura** dei calcolatori
- **Sistemi operativi**
- **Reti di calcolatori**
- Sistemi e applicazioni distribuite
- **Tecnologie Web**
- Algoritmi
- .....

# ELABORATORE ELETTRONICO (“*COMPUTER*”)

---

**Strumento** per la rappresentazione e  
l’elaborazione delle informazioni



# L'ELABORATORE

---

## Componenti principali

- Unità centrale
- Video (“monitor”)
- Tastiera e Mouse
- Lettore CD/DVD
- Dischi fissi (“hard disk”)
- Dischetti (“floppy”)/USB stick

## Componenti accessori

- Stampante
- Modem/interfaccia di rete
- Scanner
- Tavolette grafiche

...

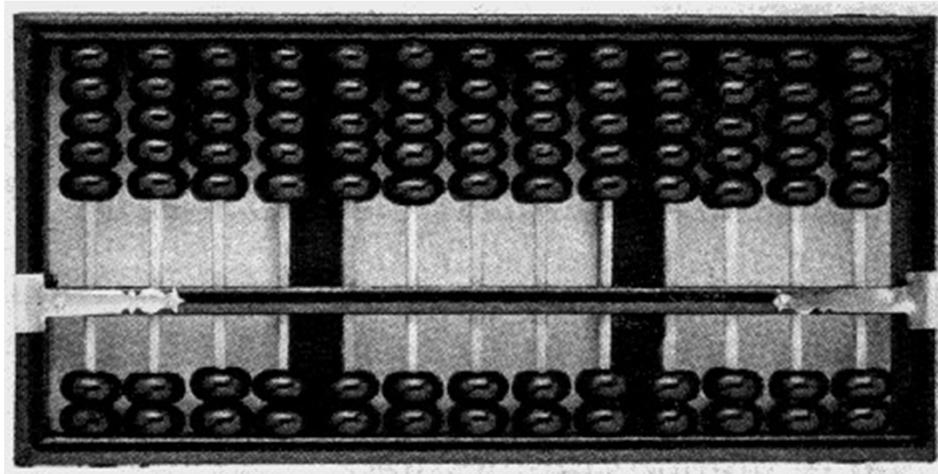


**HARDWARE**

# Brevissima ... storia del calcolatore

---

Nell'VIII secolo a.C. fu inventato in Cina il primo pallottoliere.  
Presso i Romani era diffuso l'abaco.



# Brevissima storia del calcolatore

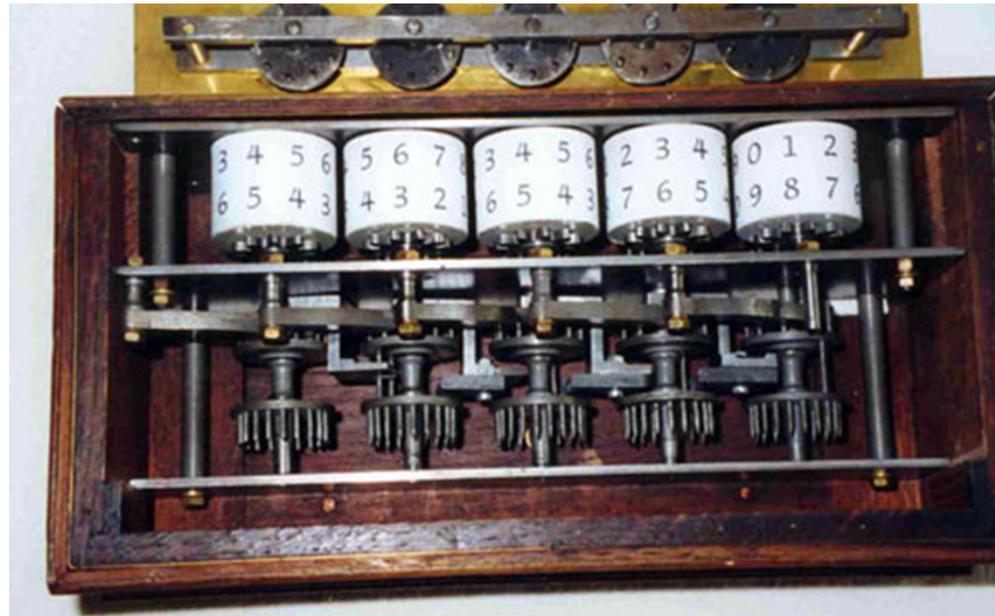
---

La *pascalina* di Blaise Pascal (1642) che meccanizzò addizioni e sottrazioni

La *macchina di Leibniz* (1671) che perfezionò il principio della pascalina, poiché eseguiva anche le moltiplicazioni

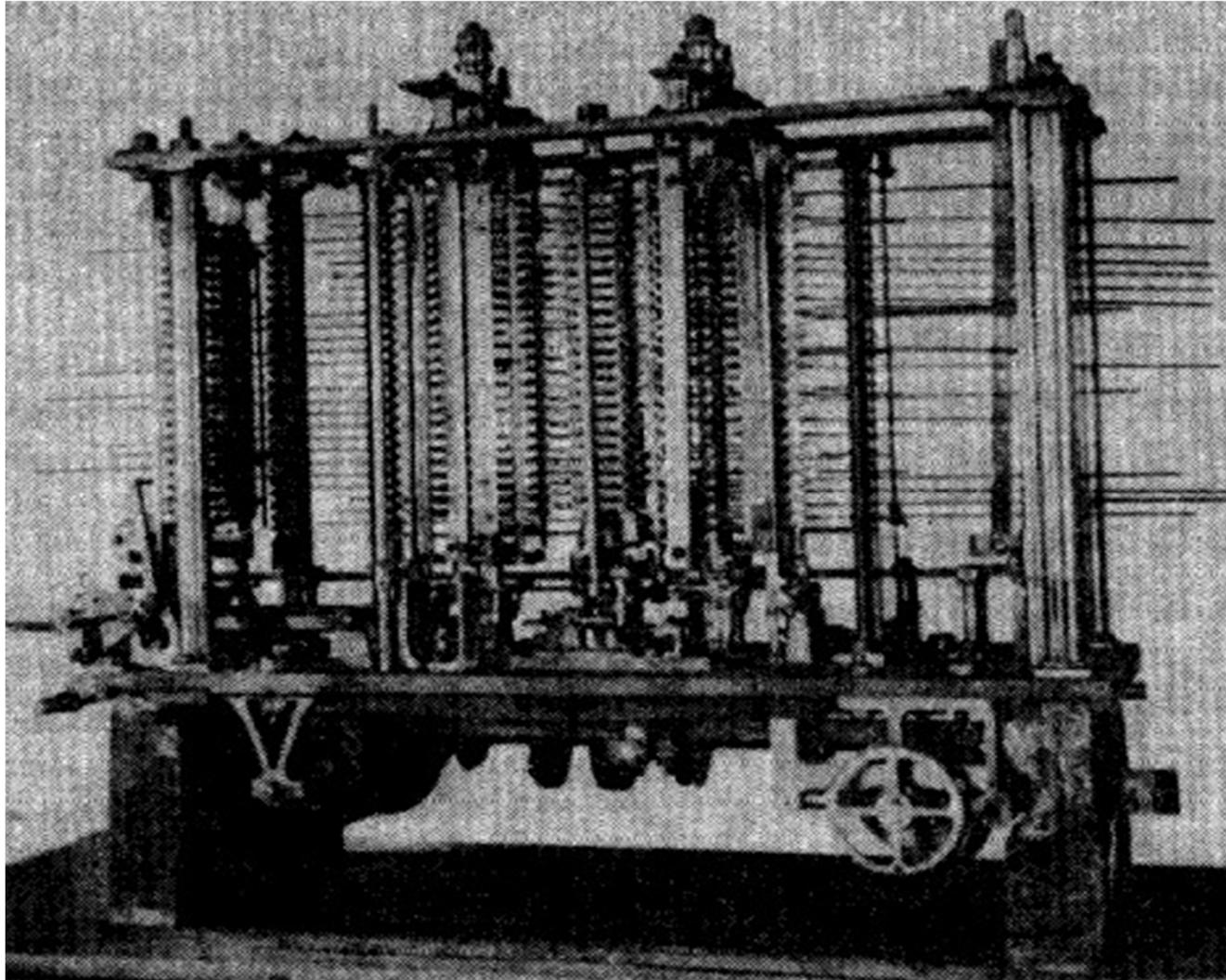
Funzionava con le ruote dentate e poteva eseguire solo le addizioni.

1642



# La macchina analitica di Babbage (1792-1871)

---



# Macchina analitica

- **Calcolatore programmabile.** Possibilità di fornire in ingresso alla macchina non solo i dati su cui operare, ma anche **la sequenza di** operazioni (*istruzioni*) da eseguire sui dati (programma).
- Strumento di calcolo “universale” le cui operazioni possono essere di volta in volta specificate assieme ai dati da elaborare
- un meccanismo di controllo a schede perforate, ripreso dal telaio tessile inventato dal francese Jacquard nel 1801.
- La realizzazione della **macchina analitica non fu mai** portata a termine sia per mancanza di fondi sia per la tecnologia meccanica inadeguata alla realizzazione di una macchina di tali dimensioni.

- **Telaio di Jacquard**  
**(1752-1834)**

- prima applicazione pratica delle schede perforate. Le schede comandano la tessitura di disegni e trame sui tessuti.



“..... E' risaputo infatti, che il telaio di Jacquard può tessere qualsiasi disegno che l'immaginazione umana è capace di concepire.

E' pure noto che è pratica costante dei fabbricanti di rivolgersi ad artisti provetti per **disegnare i modelli** (*scelta e messa a punto dell'algoritmo*).

Questi modelli sono poi affidati ad un artigiano specializzato il quale , per mezzo di una certa macchina, **punziona una serie di schede** (*traduzione dell'algoritmo in programma*).

Dopo averle inserite nel telaio Jacquard, esso **tesserà il disegno** così come lo ha tratteggiato l'artista (*esecuzione del programma da parte della macchina*).”

La macchina analitica era composta da:

- **Store** (memoria) per immagazzinare i dati iniziali, i risultati intermedi ed i valori finali. Costituita da 200 colonne (25 ruote dentate).
- **Mill** (unità aritmetico logica) in grado di svolgere le quattro operazioni aritmetiche con procedimenti meccanici (1 addizione al secondo).
- Un nastro perforato contenente la sequenza delle operazioni richieste per il problema.

# ENIAC

(Electronic Numerical Integrator And Computer)  
Università di Pennsylvania (1946).

---

L'ENIAC, acronimo di *Electronic Numerical Integrator and Computer*.

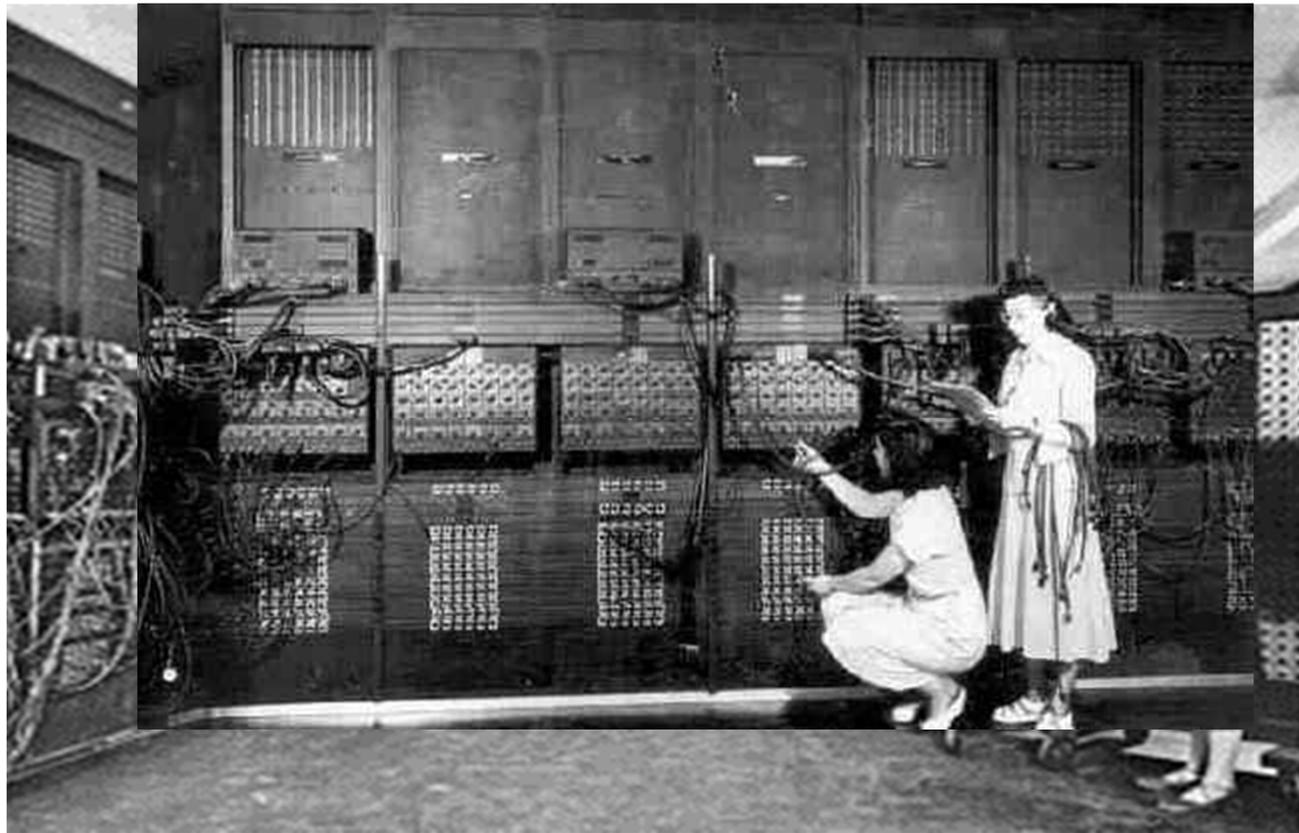
- Primo calcolatore digitale a circuiti elettronici senza parti meccaniche 'General-Purpose'
- programmabile tramite cablaggio interno (fili ed interruttori).

Le sue dimensioni sono impressionanti:

lungo trenta metri, alto tre e largo uno, pesante 27 tonnellate, occupava una sala calcolo di 167 m<sup>2</sup> e richiedeva una potenza di 150 kW.

- Problemi di calcolo balistico per il lancio dei proiettili d'artiglieria (soluzione di un sistema di equazioni differenziali)
- 18.000 valvole termoioniche, collegate da 500.000 contatti saldati manualmente, 1.500 relè.

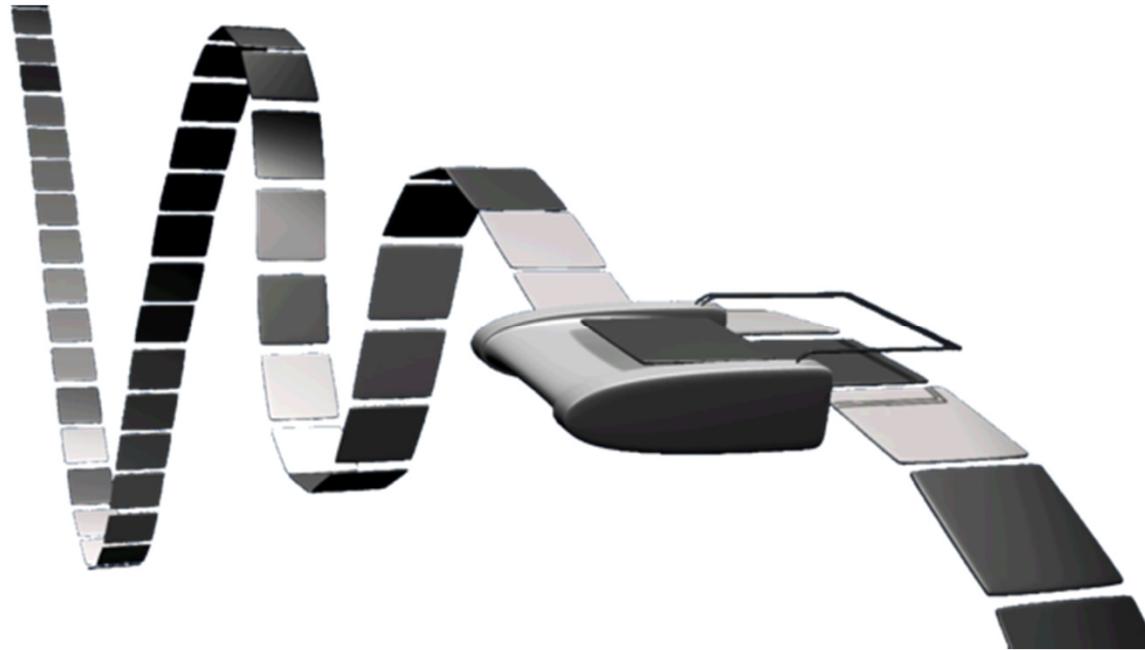
# ENIAC



# Alan Turing (1912-1954)

- E` stato un matematico, logico, crittanalista britannico
- Uno dei padri della informatica e uno dei più grandi matematici del Novecento.
- Egli introdusse la *macchina universale*, il test che portano il suo nome.
- Fu anche uno dei più brillanti decrittatori che operavano in Inghilterra, durante la seconda guerra mondiale per decifrare i messaggi tedeschi.
- Omosessuale, morì suicida a soli 41 anni in seguito a una persecuzione omofobica condotta nei suoi confronti.
- In suo onore l' Association for Computer Machinery (ACM) ha creato nel 1966 il Turing Award Turing, massima riconoscenza nel campo dell'informatica, dei sistemi intelligenti e dell'intelligenza artificiale.

- Macchina di Turing



- La macchina è dotata di un insieme di istruzioni che descrivono le operazioni che la macchina può eseguire in funzione dello stato in cui si trova

# Macchina di Turing Universale

- in grado di emulare il funzionamento di una qualunque altra macchina di Turing creata per risolvere un particolare problema.
- **tesi di Church-Turing** : *se esiste un algoritmo in grado di svolgere un compito di manipolazione dei simboli, allora esiste una macchina di Turing in grado di svolgere lo stesso compito.*

# John von Neumann (1903 - 1957):



Il punto qualificante del progetto architettonico che da lui prende il nome (1945) consiste nel funzionamento in base al programma registrato in memoria (*Stored Program Computer*).

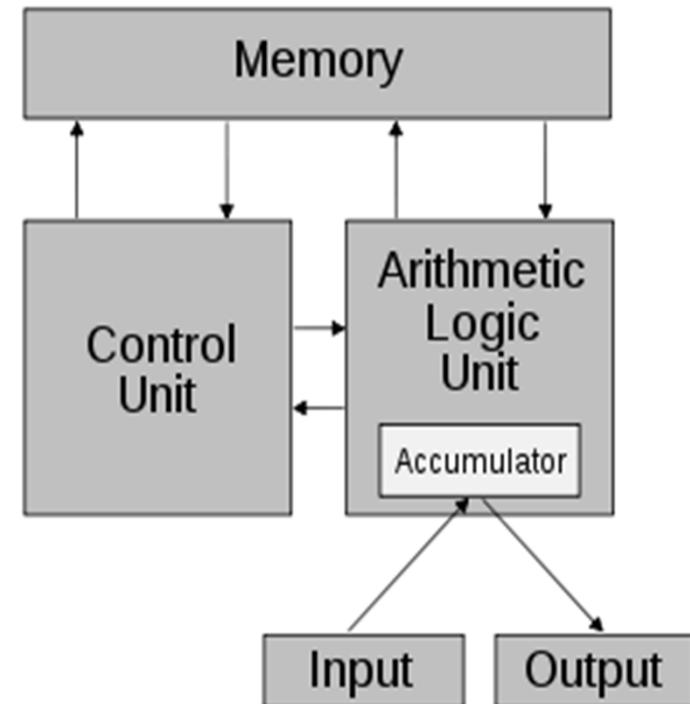
# Von Neumann e Turing

- Tra il '37 ed il '38 Turing trascorse un periodo di ricerca a Princeton.
- Incontrò e lavorò con Von Neumann che venne in quella occasione a conoscenza del lavoro di Turing sulla **macchina universale**.

- **Architettura di Von Neumann**

(1903- 1957)

- Programma e dati contenuti nella stessa memoria.
- Unità di calcolo separata dalla unità di memoria.
- Il calcolatore può modificare il proprio comportamento



# EDVAC

- **1952**
- **EDVAC**- Electronic Discrete Variable Computer sviluppato da **John Von Neumann e H.H. Goldstine**
- Si tratta del primo progetto di **calcolatore elettronico a programma memorizzato**. In altre parole del vero e proprio calcolatore moderno. Derivato dall'**ENIAC**, esso ne perfeziona il concetto di programmabilità, in quanto i **programmi sono incorporati nella memoria della macchina**.

# George Boole (1815-1864)

- 1854- “An investigation of the law of thought, on which are founded the mathematical theories of logic and probabilities”.
- Partendo dalla logica di Aristotele, riprendendo le idee di Leibniz introdusse **l'algebra booleana**.
- Nuovi operatori per rappresentare combinazioni di “vero” e “falso”.
- **1940 Claude Shannon** si laurea con una tesi basata sul lavoro di BOOLE, sostenendo che **l'algebra booleana** poteva essere applicata per studiare e progettare i circuiti a relè necessari tanto in informatica che in telefonia.

## Brevissima .. Storia dei calcolatori

---

- La **prima generazione** di computer (ENIAC) fino circa al 1959 fa uso di tubi catodici, che avevano l'inconveniente di consumare molta energia, di scaldare molto e di essere piuttosto inaffidabili.
- La **seconda generazione**, tra il 1959 e il 1964, introduce i transistor; sono macchine più piccole, più affidabili e più economiche.
- La **terza generazione**, tra il 1965 e il 1970, introduce i circuiti integrati e miniaturizzati che contengono molti transistor.
- La **quarta generazione** fa uso dei circuiti LSI e VLSI. Anche la memoria è molto cambiata e usa la tecnologia dei semiconduttori. È questa l'epoca dei microcomputer.
- Infine la **quinta generazione** dal 1980 introduce nuovi concetti sia nell'hardware (computer RISC, computer paralleli) sia nel software (calcolo parallelo, reti neurali).

# Legge di Moore

---

- Moore (uno dei fondatori di Intel), 1965
- Numero di transistor in un chip raddoppia ogni anno
- Dal 1970, piu' lento: raddoppia ogni 18 mesi
- Conseguenze:
- Costo del chip invariato minor costo totale
- Circuiti piu' vicini maggiore velocita'
- Calcolatori piu' piccoli
- Minori requisiti di raffreddamento e alimentazione

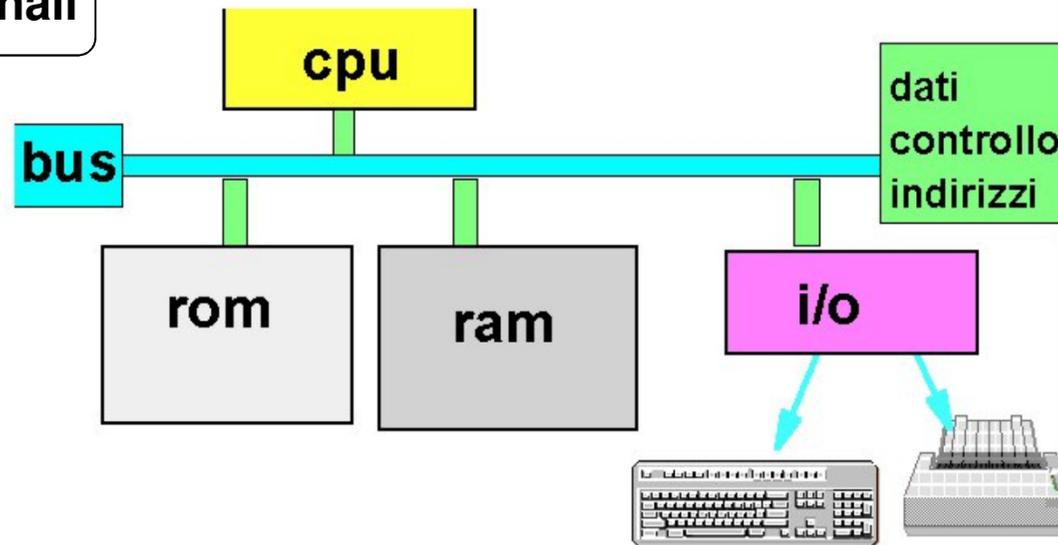
# Generazioni di calcolatori

Tubi a vuoto - 1946-1957

- Transistor - 1958-1964
- Integrazione su piccola scala – dal 1965 - Fino a 100 componenti su un chip
- Integrazione su media scala – fino al 1971 100-3,000 dispositivi su un chip
- Integrazione su larga scala- 1971-1977 3,000 - 100,000 dispositivi su un chip
- Integrazione su grandissima scala – dal 1978 fino a oggi 100,000 - 100,000,000 dispositivi su un chip
- Integrazione su ultra larga scala
- Piu' di 100,000,000 dispositivi su un chip

# ARCHITETTURA DI UN ELABORATORE

Unità funzionali



Ispirata al modello della **Macchina di Von Neumann**  
(Princeton, Institute for Advanced Study, anni '40)

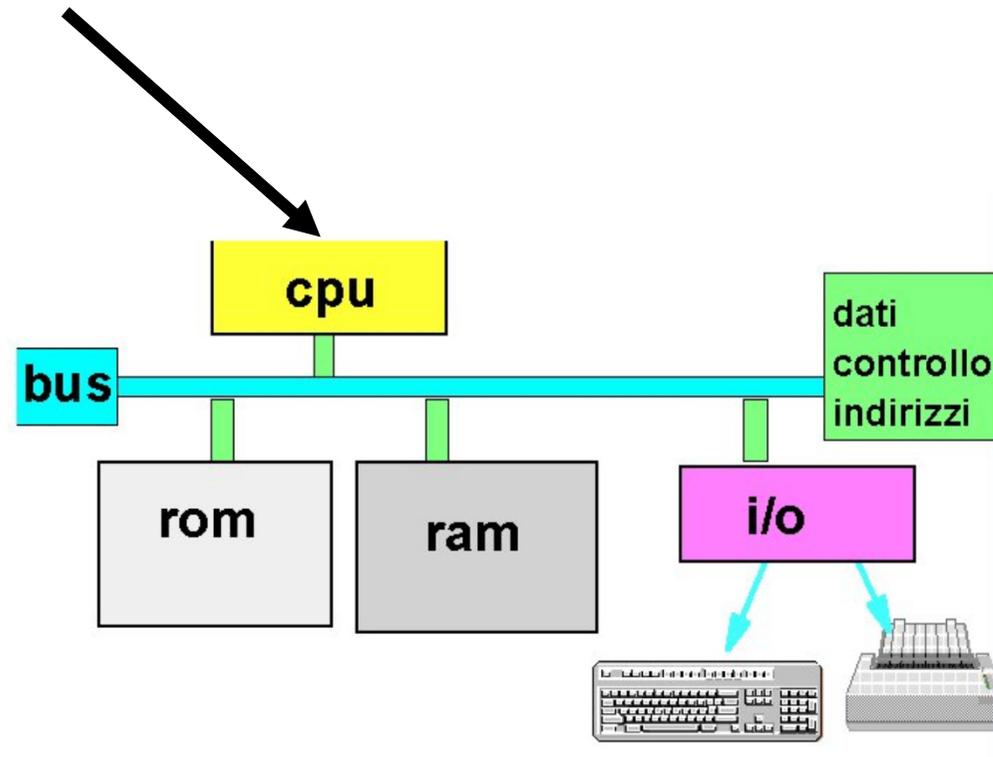
Macchina di Von Neumann:

- Non distingueva fra RAM e ROM
- Non aveva un bus ma collegamenti punto-punto

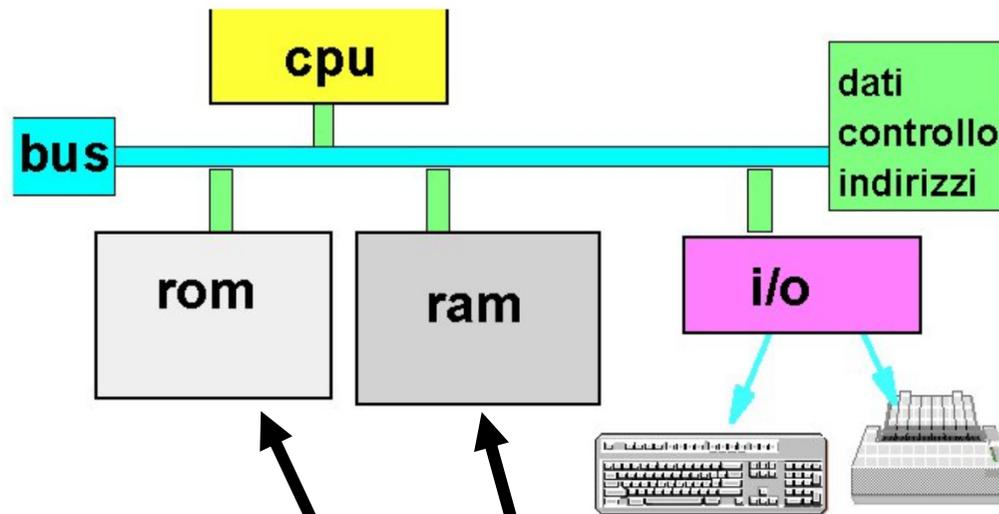
# HARDWARE

## CPU (Central Processing Unit), o Processore

**CPU:** svolge le elaborazioni e il trasferimento dei dati, cioè *esegue i programmi*



# HARDWARE



## RAM & ROM

- Dimensioni relativamente limitate
- Accesso molto rapido

**RAM** (*Random Access Memory*), e  
**ROM** (*Read Only Memory*)

Insieme formano la **Memoria centrale**

# HARDWARE

---

**RAM** è **volatile** (perde il suo contenuto quando si spegne il calcolatore)

- usata per memorizzare dati e programmi

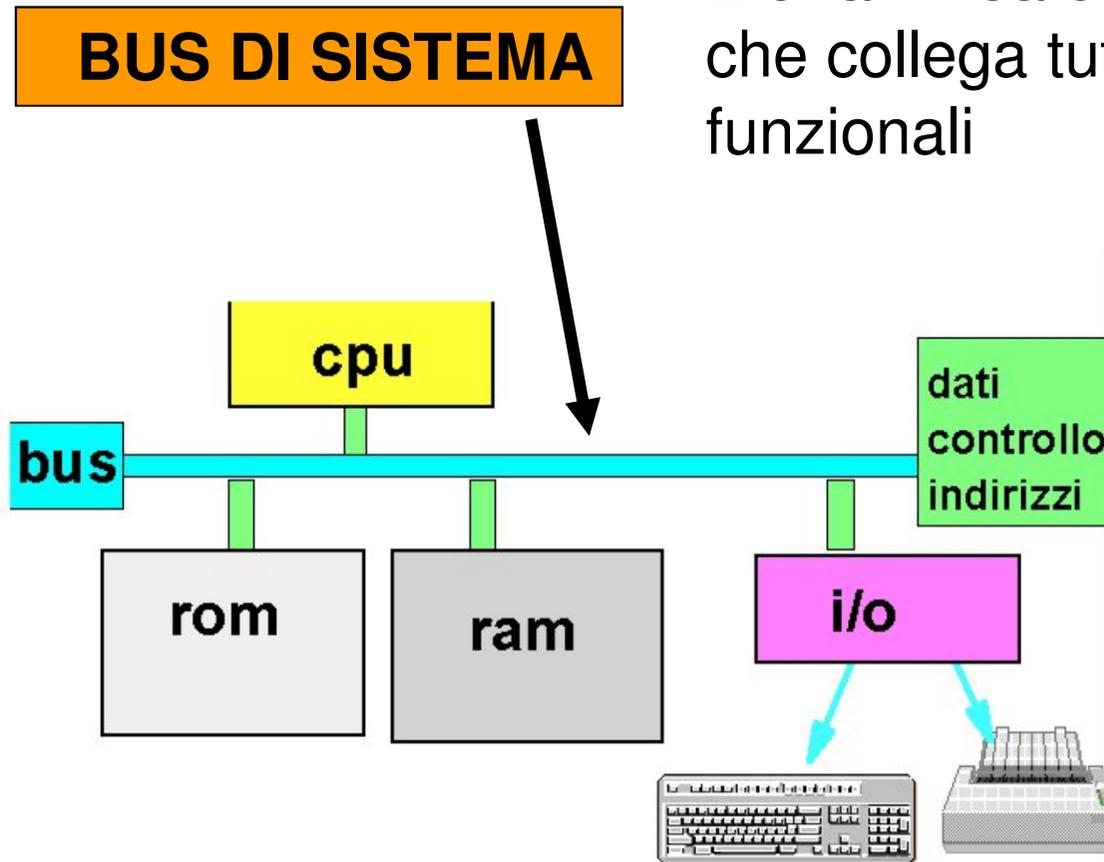
**ATTENZIONE**

**ROM** è **persistente** (mantiene il suo contenuto quando si spegne il calcolatore) ma il suo ***contenuto è fisso e immutabile***

- usata per memorizzare programmi di sistema (tipicamente *firmware*)

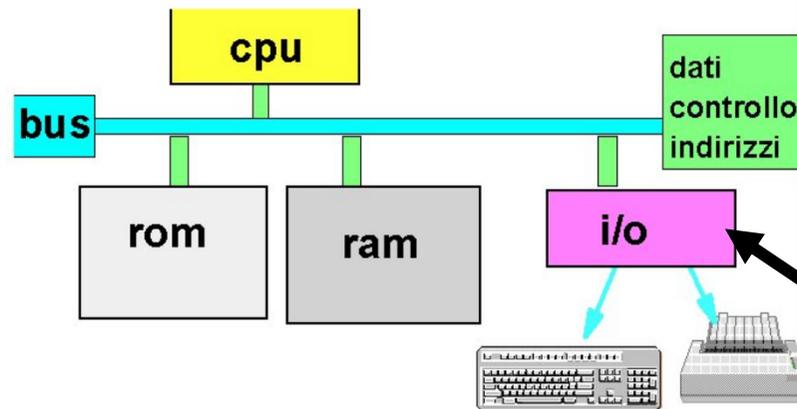
# HARDWARE

---



È una “linea di comunicazione”  
che collega tutti gli elementi  
funzionali

# HARDWARE



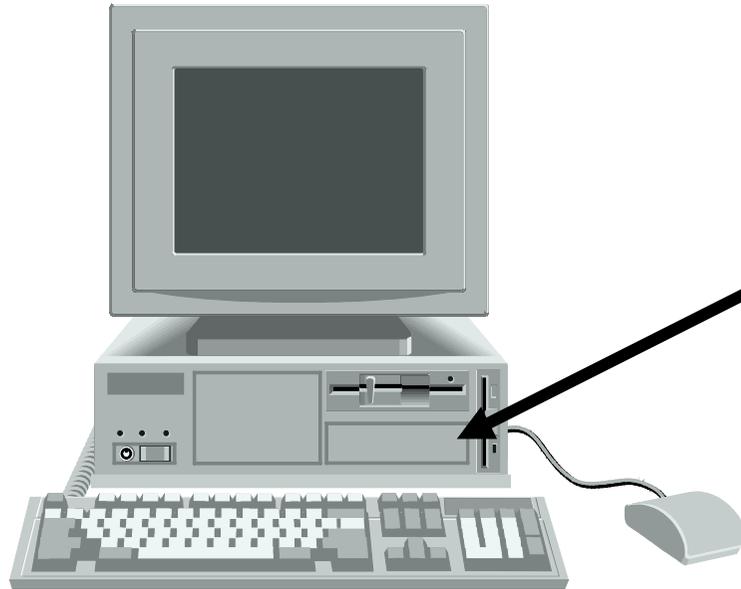
Sono usate per far comunicare il calcolatore con l'esterno (in particolare con l'utente)

## UNITÀ DI INGRESSO/ USCITA (I/O)

- Tastiera e Mouse
- Video e Stampante
- Scanner
- Tavoleta grafica
- **Dispositivi di memoria di massa**
- ...

# HARDWARE

---



## MEMORIA DI MASSA

- HD
- CD
- DVD
- ...
- PenDrive
- ...

- memorizza **grandi quantità** di informazioni
- **persistente** (le informazioni non si perdono spegnendo la macchina)
- **accesso molto meno rapido** della memoria centrale (**millisecondi** contro **nanosecondi**; differenza  $10^6$ )

# TECNOLOGIA DIGITALE

---

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

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

- tensione alta ( $V_H$ , ad es. 5V o 3.3V)
- tensione bassa ( $V_L$ , ad es. 0V)

A tali valori vengono convenzionalmente **associate le due cifre binarie 0 e 1:**

- **logica positiva:**  $1 \leftrightarrow V_H$ ,  $0 \leftrightarrow V_L$
- **logica negativa:**  $0 \leftrightarrow V_H$ ,  $1 \leftrightarrow V_L$

# TECNOLOGIA DIGITALE (segue)

---

Dati e operazioni vengono codificati tramite **sequenze di bit**

**01000110101 ....**

CPU è in grado di operare soltanto in aritmetica binaria, effettuando operazioni *elementari*:

- somma e differenza
- scorrimento (shift)
- ...

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

Calvino  
(Lezioni americane)  
Leggerezza

- ...Poi, l'informatica. E' vero che il software non potrebbe esercitare i poteri della sua *leggerezza* se non mediante la pesantezza dell'hardware; ma è il software che comanda, che agisce sul mondo esterno e sulle macchine, le quali esistono solo in funzione del software, si evolvono in modo da elaborare programmi sempre più complessi.....

## Ada Byron: la prima programmattrice (1815-1852)

**Ada** Augusta Byron, figlia del poeta George Byron, collaborò con Babbage seguendo i progetti della Macchina Analitica e arricchendo con numerosi ed importanti contributi (idea di esecuzione sequenziale e loop) **il concetto di programmazione**



# ADA

- 1833. Incontro con Charles Babbage.
- Primo programma di computer : algoritmo (per il calcolo dei numeri di Bernoulli) codificato per essere elaborato da una macchina (*macchina analitica di Babbage*).
- Parla di variabili, operazioni e stati e si esprime in linguaggio puramente simbolico.
- Strutture cicliche annidate (problemi ricorrenti).
- Uso di un linguaggio molto simile all'attuale assembler

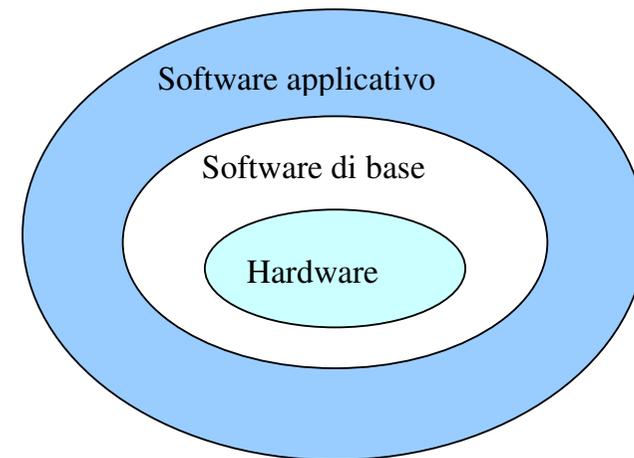
# SOFTWARE

---

**Software: programmi** che vengono eseguiti dal sistema

**Distinzione fra:**

- Software di base (es. Sistema Operativo)
- Software applicativo



# IL SOFTWARE

---

## Software:

insieme (complesso) di programmi

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

Concetto di  
***MACCHINA VIRTUALE***



# IL FIRMWARE

---

## **Firmware:**

il confine fra hardware e software

**È uno strato di *micro-programmi*, scritti dai costruttori**, che agiscono direttamente al di sopra dello strato hardware

Sono memorizzati su una speciale *memoria centrale permanente* (ROM, EPROM, ...)

# IL SISTEMA OPERATIVO

---

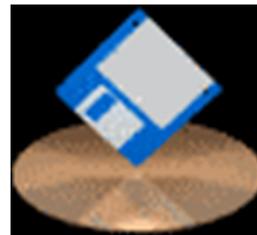
Strato di programmi che opera *al di sopra di hardware e firmware* e **gestisce l'elaboratore**

Spesso è venduto insieme all'elaboratore

**Si può scegliere tra *diversi sistemi operativi*** per lo stesso elaboratore, con diverse caratteristiche

## Esempi:

- Windows 95/98/XP
- Windows NT/2000
- Linux v.2.6
- MacOS X
- Symbian
- Palm OS
- VISTA



# FUNZIONI DEL SISTEMA OPERATIVO

---

Le funzioni messe a disposizione dal SO 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 (SO)**

**→ il SO realizza una “macchina virtuale”**

# FUNZIONI DEL SISTEMA OPERATIVO

---

**Qualsiasi operazione di accesso a risorse** implicitamente richiesta da comando utente **viene esplicitata dal SO**

**Conseguenza:** diversi SO possono realizzare *diverse macchine virtuali* **sullo stesso elaboratore fisico**

Attraverso il SO il livello di interazione fra utente ed elaboratore viene elevato:

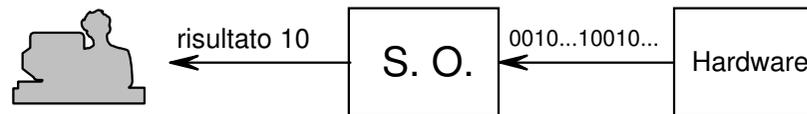
- senza SO:           sequenze di bit
- con SO:             comandi, programmi, dati

I sistemi operativi si sono evoluti nel corso degli anni (interfacce grafiche, Mac, Windows, ...)

# ESEMPIO



e viceversa:



|                 |  |
|-----------------|--|
| <u>Utente:</u>  | <u>Sistema Operativo:</u>  |
| “esegui progr1” | - input da tastiera<br>- ricerca codice di “progr1” su disco<br>- carica in memoria centrale codice e dati<br><elaborazione> |
| <u>Utente:</u>  | <u>Sistema Operativo:</u>  |
| “stampa 10”     | - output su video  |

# PROGRAMMI APPLICATIVI

---

## Risolvono problemi specifici degli utenti:

- *word processor*: elaborazione di testi (es. *MSWord*)
- *fogli elettronici*: gestione di tabelle, calcoli e grafici (es. *MSExcel*)
- *database*: gestione di archivi (es. *MSAccess*)
- *suite* (integrati): collezione di applicativi capaci di funzionare in modo integrato come un'applicazione unica (es. *Open Office*)

- Sono scritti in **linguaggi di programmazione** di alto livello
- Risentono in misura ridotta delle caratteristiche della architettura dell'ambiente sottostante (***portabilità***)

# AMBIENTI DI PROGRAMMAZIONE

---

È l'insieme dei programmi che consentono la scrittura, la verifica e l'esecuzione di nuovi programmi (*fasì di sviluppo*)

## Sviluppo di un programma

- Affinché un programma scritto in un qualsiasi linguaggio di programmazione sia comprensibile (e quindi eseguibile) da un calcolatore, *occorre tradurlo* dal linguaggio originario al linguaggio della macchina
- Questa operazione viene normalmente svolta da speciali programmi, detti *traduttori*

# Crisi del software

- Progetti oltre il budget
- Progetti oltre i limiti di tempo
- Software di scarsa qualità
- Software che spesso non rispettava i requisiti
- Progetti ingestibili e codice difficile da mantenere

“Se il settore dell’automobile si fosse sviluppato come l’industria informatica, oggi avremo veicoli che costano 25 dollari e fanno 500 Km con un litro”. (Bill Gates).

“Se le auto funzionassero come i software, si bloccherebbero due volte al giorno senza motivo e l’unica soluzione sarebbe reinstallare il motore”  
(Dirigente General Motors)

# Nuove metodologie e strumenti

- Programmazione strutturata (Böhm-Jacopini-1966)
- Tecniche di decomposizione (Dijkstra- 1968)
- Verifica formale delle proprietà dei programmi (Floyd, Hoare , fine anni 60)
- Modularizzazione e progettazione per il cambiamento (Parnas, anni 70)
- Programmazione orientata agli oggetti (anni 70)
- Nuovi linguaggi di programmazione:ADA, JAVA(90)
- Service oriented architecture(componenti software)
- .....