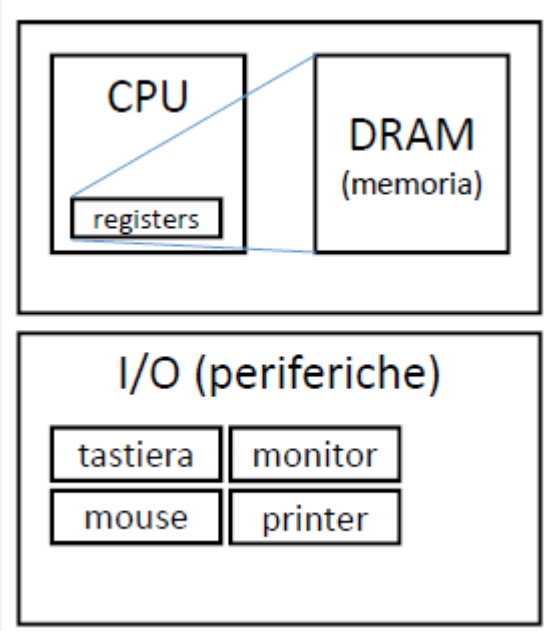
- **ELEMENTI DI INFORMATICA**

# Cenni di architettura dei calcolatori

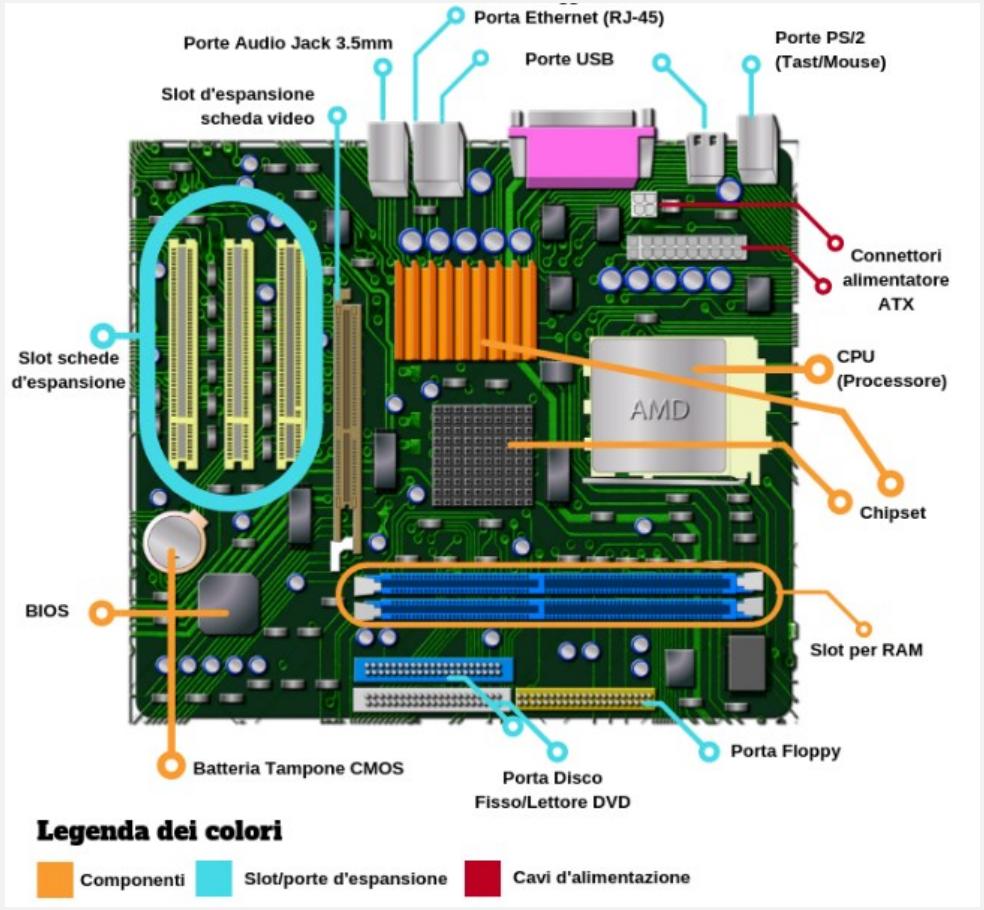
- Partiamo dal disegnare uno schema semplificato dell'architettura di un calcolatore



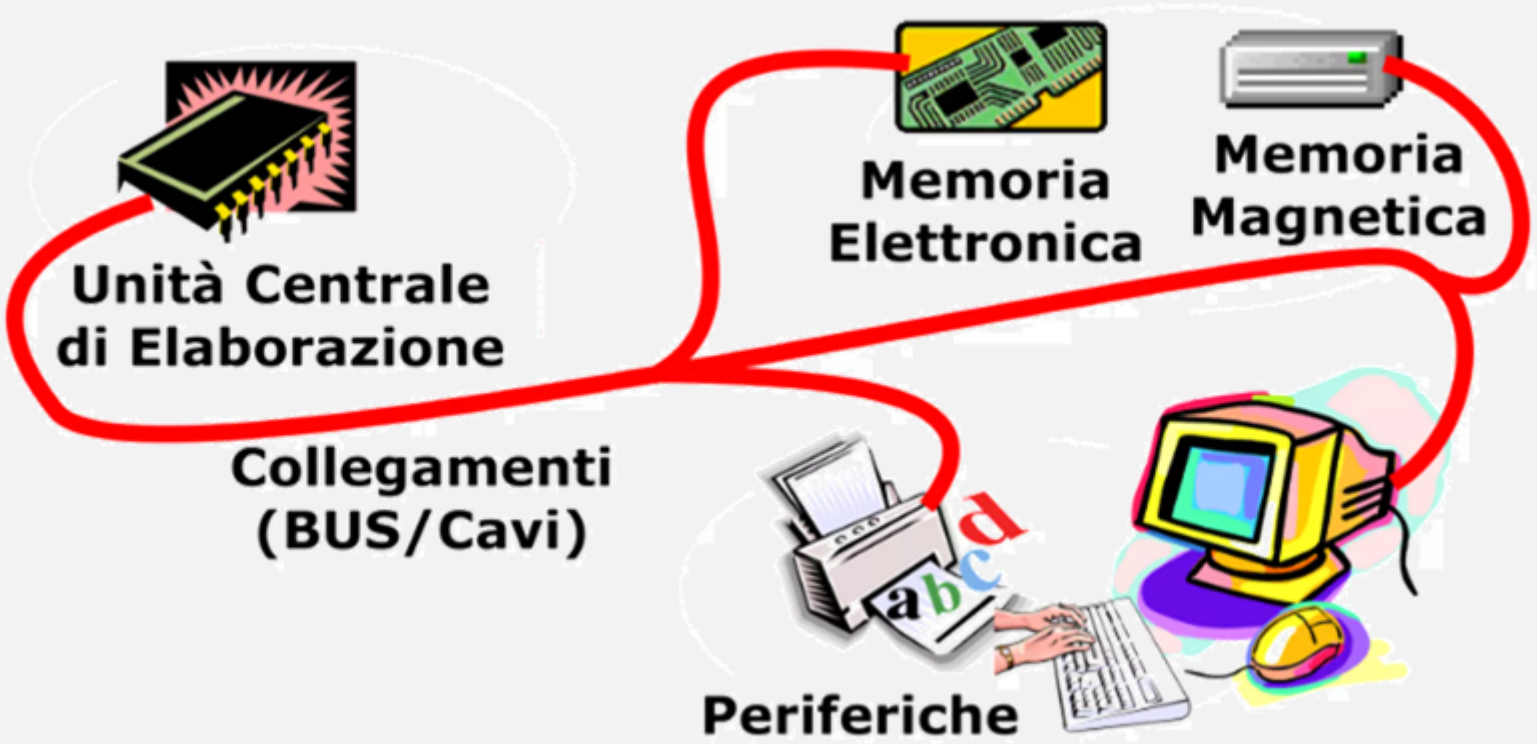

- Considereremo solo
  - CPU (Central Processing Unit)
  - Memoria (gerarchia)
  - Dispositivi di input/output (I/O)

# Motherboard

I vari blocchi di un calcolatore sono fisicamente interconnessi tramite la scheda madre

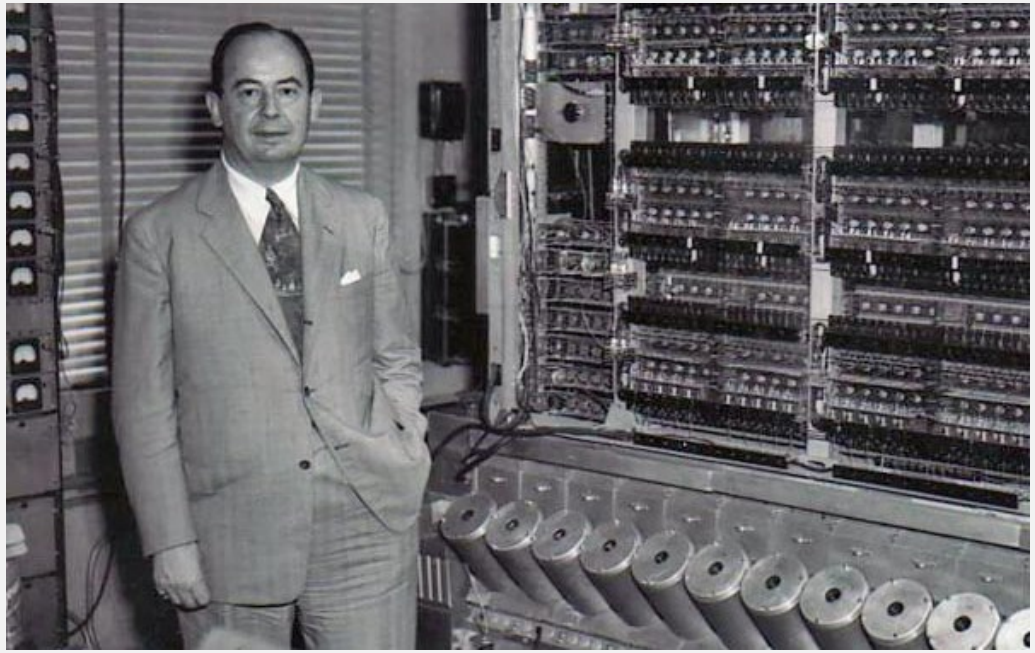


# Cenni di architettura dei calcolatori



# Cenni di architettura dei calcolatori

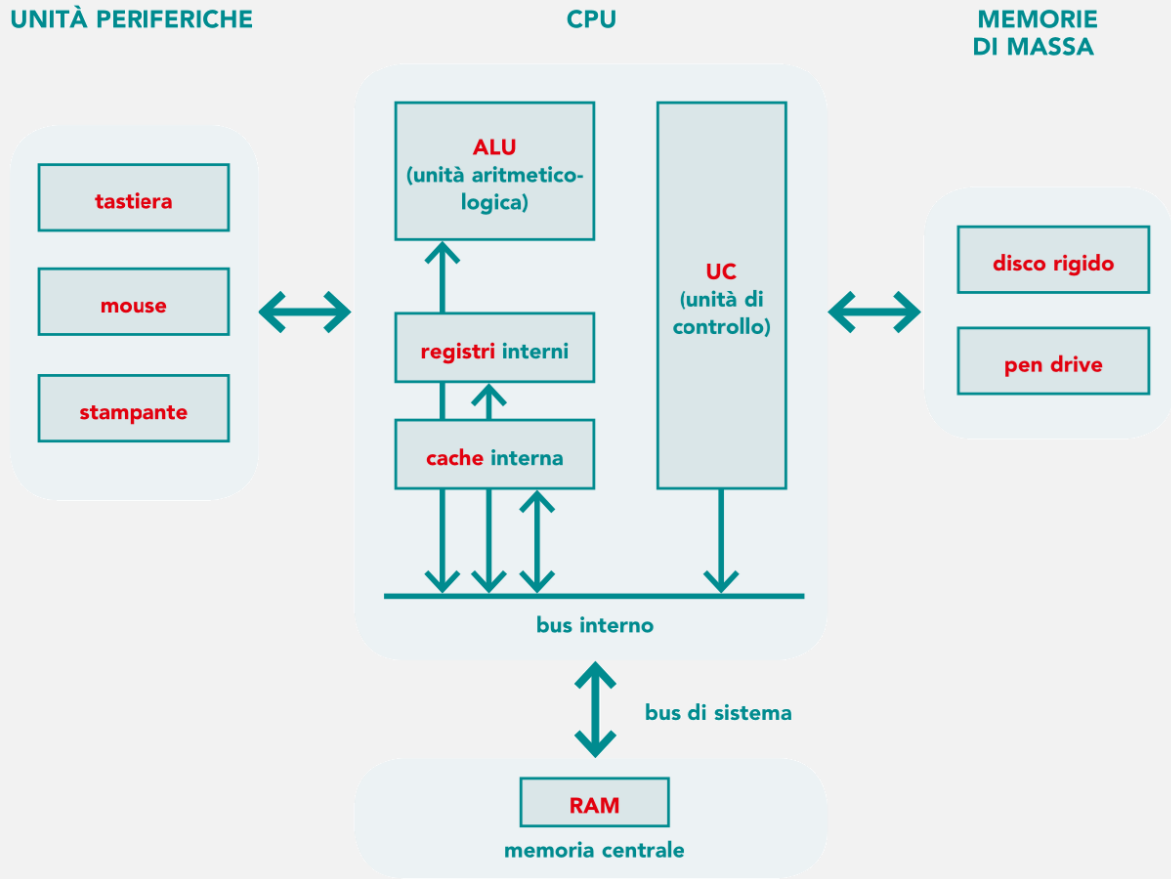
- L'architettura dell'**hardware** di un calcolatore reale è molto complessa
- La **macchina di Von Neumann** è un modello semplificato dei calcolatori moderni
  - **Von Neumann** progettò, verso il 1945, il primo calcolatore con programmi memorizzabili anziché codificati mediante cavi e interruttori



# Macchina di Von Neumann

E' composta da 4 tipologie di componenti funzionali:

- unità centrale di elaborazione (CPU)
  - esegue istruzioni per l'elaborazione dei dati
  - svolge anche funzioni di controllo
- memoria centrale
  - memorizza e fornisce l'accesso a dati e programmi
- interfacce di ingresso e uscita
  - componenti di collegamento con le periferiche del calcolatore
- bus
  - svolge la funzionalità di trasferimento di dati e di informazioni di controllo tra le varie componenti funzionali



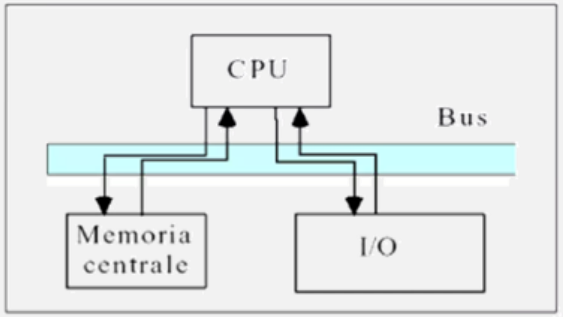
# Caratteristiche del collegamento a BUS

Nel modello della macchina di Von Neumann, le informazioni transitano attraverso dei collegamenti fisici a cui ci si riferisce con il termine **bus di sistema**.

- Semplicità
  - un'unica linea di connessione → costi ridotti di produzione
- Estendibilità
  - aggiunta di nuovi dispositivi molto semplice
- Standardizzabilità
  - regole per la comunicazione da parte di dispositivi diversi
- Lentezza
  - utilizzo in mutua esclusione del bus
- Limitatà capacità
  - al crescere del numero di dispositivi collegati
- Sovraccarico del processore (CPU)
  - perchè funge da *master* sul controllo del bus

# BUS di sistema

- Interconnette CPU, memorie ed interfacce verso dispositivi periferici (I/O, memoria di massa, ...)
- Collega **due unità funzionali alla volta**
  - una trasmette e l'altra riceve
- Il trasferimento dei dati avviene sotto il controllo della CPU



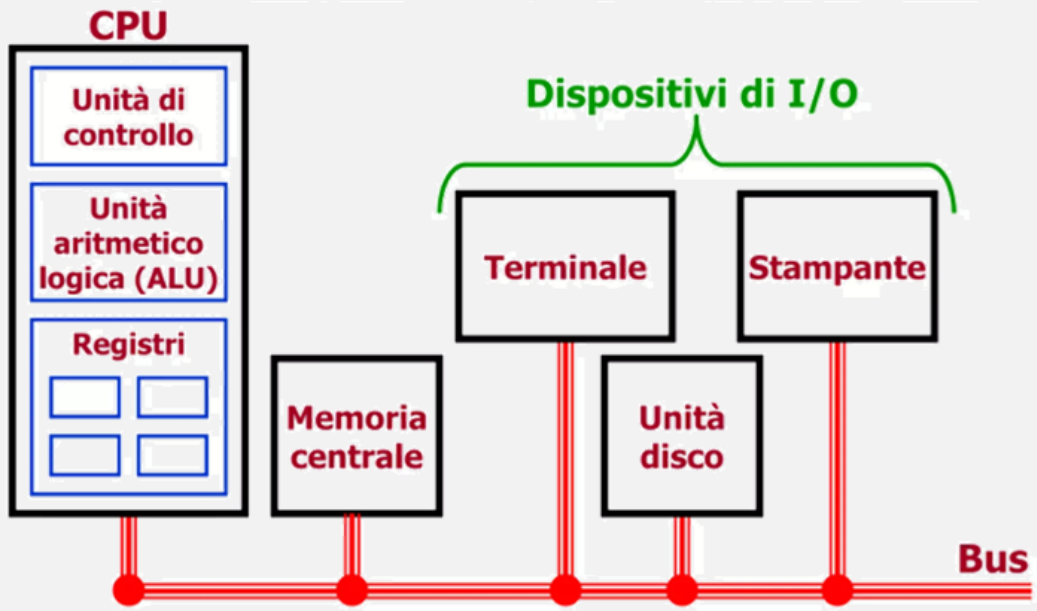

- Il bus trasporta dati, indirizzi e comandi
- Componenti del bus (sottogruppi di linee):
  - **Bus dati (data bus)**
  - **Bus indirizzi (address bus)**
  - **Bus comandi (command bus)**

- **Bus dati (data bus)**
  - Serve per trasferire dati
    - tra la memoria centrale ed il registro dati (MDR) della CPU
    - tra periferiche e CPU (o memoria centrale)
  - Bidirezionale

# BUS di sistema

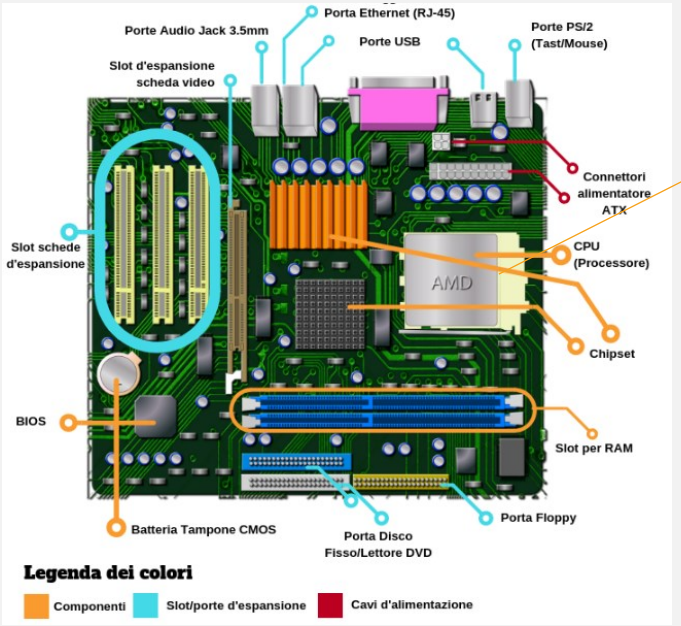
- **Bus indirizzi (address bus)**
  - Serve per trasmettere il contenuto del registro indirizzi (MAR) alla memoria (o ad una periferica)
    - si seleziona una cella per successive operazioni di lettura o scrittura
  - Unidirezionale
- **Bus comandi (command bus)**
  - Serve per inviare comandi
    - verso la memoria (es: lettura o scrittura)
    - o verso una periferica (es. stampa verso la stampante / interfaccia)
  - Unidirezionale

## Organizzazione tipica di un calcolatore «bus oriented»

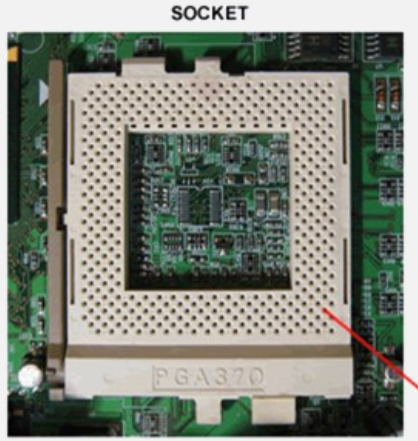


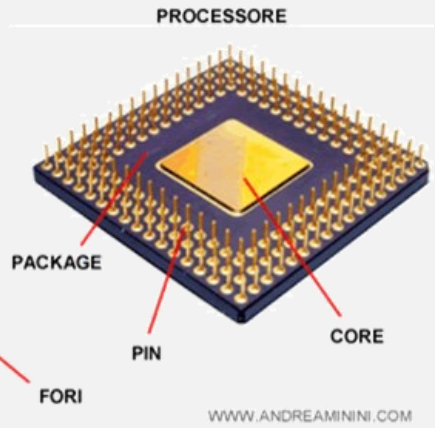
# Processore

Central Processing Unit (CPU)
Tutte le operazioni di elaborazione delle informazioni effettuate da un calcolatore sono svolte direttamente dal processore, oppure da altri componenti dietro comando del processore







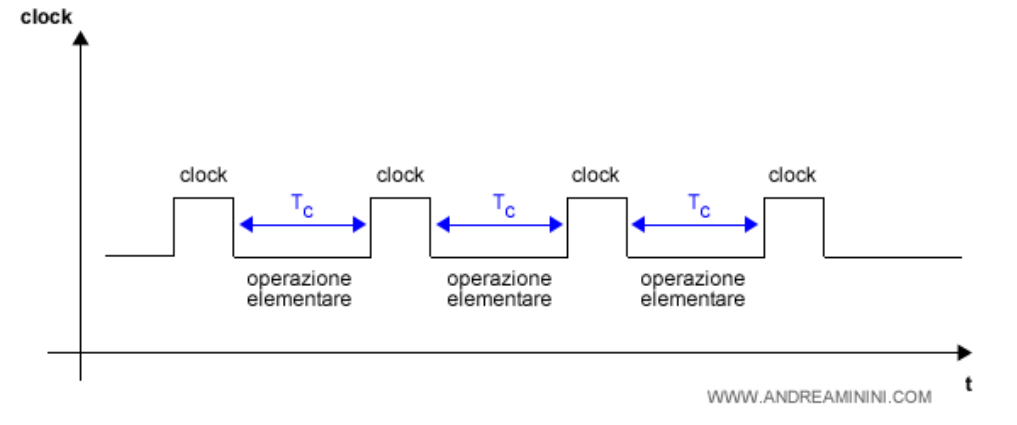
# Elementi di una CPU


- **Unità di controllo**
  - Svolge funzioni di controllo, decide quali istruzioni eseguire.
- **Unità aritmetico–logica**
  - esegue le operazioni aritmetico-logiche (+,-,ecc. , confronto).
- **Registri**
  - **memoria ad alta velocità** usata per risultati temporanei e informazioni di controllo;
  - il **valore massimo** memorizzabile in un registro è determinato dalle **dimensioni** del registro;
  - esistono registri di uso generico e registri specifici:
    - **Program Counter** (**PC**) – qual è l'istruzione successiva;
    - **Instruction Register** (**IR**) – istruzione in corso d'esecuzione;
    - **...**

# Processore


Un processore è in grado di compiere solo operazioni molto semplici:

- lettura/scrittura/copia di una o più celle di memoria
- somma/sottrazione/moltiplicazione/divisione del contenuto di una o più celle di memoria
- lettura/scrittura in zone di memoria 'speciali' per pilotare dispositivi di ingresso/uscita (ad esempio
- schede video)
- altre semplici operazioni sulle celle di memoria




Il processore esegue le operazioni elementari in sequenza, una dopo l'altra.
Il tempo necessario del processore per eseguire un'operazione elementare è delimitato da due segnali detti **clock**.
Ogni clock segna l'inizio di un ciclo.

# Processore


- Ciclo **Fetch–Decode–Execute** (**leggi–decodifica–esegui**)
  1. Prendi l'**istruzione corrente** dalla memoria e mettila nel **registro istruzioni** (**IR**).
  2. **Incrementa** il **program counter** (**PC**) in modo che contenga l'indirizzo dell'istruzione successiva.
  3. Determina il tipo dell'istruzione corrente (**decodifica**).
  4. Se l'istruzione usa una parola in memoria, determina dove si trova.
  5. Carica la parola, se necessario, in un registro della CPU.
  6. **Esegui** l'istruzione.
  7. Torna al punto 1 e inizia a eseguire l'istruzione successiva.

# Processore


- Ogni processore è caratterizzato da un proprio insieme di istruzioni, tramite le quali è possibile fargli svolgere le precedenti operazioni
- L'insieme delle istruzioni di un processore viene chiamato linguaggio macchina di quel processore
- Ogni istruzione è identificata da una certa configurazione di bit

| carattere | cod. binario | carattere | cod. binario | carattere | cod. binario |
|---|---|---|---|---|---|
| 0 | 0011 0000 | M | 0100 1101 | i | 0110 1001 |
| 1 | 0011 0001 | N | 0100 1110 | j | 0110 1010 |
| 2 | 0011 0010 | O | 0100 1111 | k | 0110 1011 |
| 3 | 0011 0011 | P | 0101 0000 | l | 0110 1100 |
| 4 | 0011 0100 | Q | 0101 0001 | m | 0110 1101 |
| 5 | 0011 0101 | R | 0101 0010 | n | 0110 1110 |
| 6 | 0011 0110 | S | 0101 0011 | o | 0110 1111 |
| 7 | 0011 0111 | T | 0101 0100 | p | 0111 0000 |
| 8 | 0011 1000 | U | 0101 0101 | q | 0111 0001 |
| 9 | 0011 1001 | V | 0101 0110 | r | 0111 0010 |
| A | 0100 0001 | W | 0101 0111 | s | 0111 0011 |
| B | 0100 0010 | X | 0101 1000 | t | 0111 0100 |
| C | 0100 0011 | Y | 0101 1001 | u | 0111 0101 |
| D | 0100 0100 | Z | 0101 1010 | v | 0111 0110 |
| E | 0100 0101 | a | 0110 0001 | w | 0111 0111 |
| F | 0100 0110 | b | 0110 0010 | x | 0111 1000 |
| G | 0100 0111 | c | 0110 0011 | y | 0111 1001 |
| H | 0100 1000 | d | 0110 0100 | z | 0111 1010 |
| I | 0100 1001 | e | 0110 0101 | | |
| J | 0100 1010 | f | 0110 0110 | | |
| K | 0100 1011 | g | 0110 0111 | | |
| L | 0100 1100 | h | 0110 1000 | | |

Data la semplicità delle istruzioni e dei dati su cui lavora un processore si ha che scrivere (interamente) in linguaggio macchina un programma che faccia cose complesse diviene un lavoro estremamente impegnativo e costoso

# Astrazione e linguaggi di programmazione

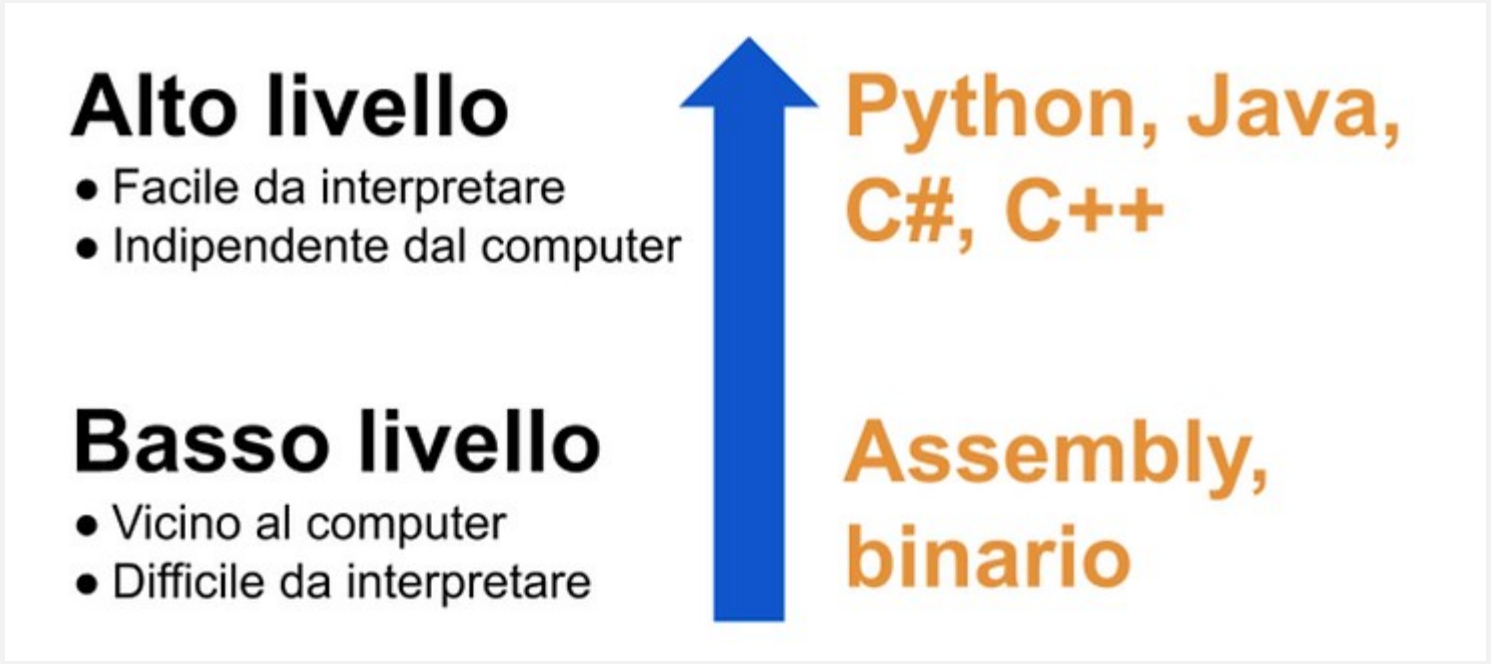

- Questo è fondamentalmente il motivo per cui sono stati inventati moltissimi altri **linguaggi cosiddetti ad alto livello**, che sono molto più 'vicini' al linguaggio umano rispetto al linguaggio macchina
- Tali linguaggi si basano sul concetto di astrazione dalla macchina sottostante: astraggono dai dettagli, cosiddetti di basso livello

# Astrazione e linguaggi di programmazione


Per far eseguire un programma ad un processore, basta:

- memorizzare da qualche parte nella memoria la sequenza di configurazioni di bit relativa alle istruzioni da eseguire
- dire al processore a che indirizzo si trova la prima di tali istruzioni

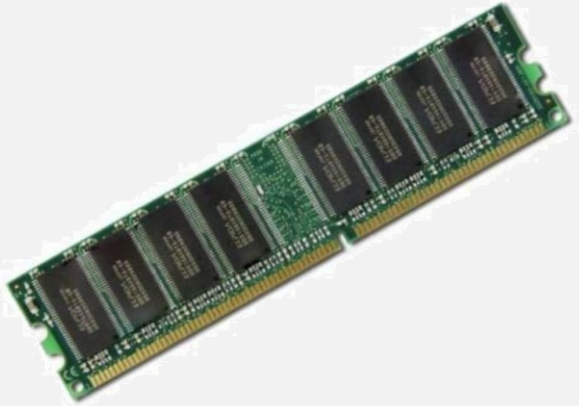
Il processore eseguirà, una dopo l'altra, le istruzioni che trova a partire da tale indirizzo




***Codifica in linguaggio C dell'algoritmo che converte gradi Celsius in Fahrenheit***

```
int main(){
  float c, f; /* Celsius e Fahrenheit */
  printf("Inserisci la temperatura da convertire");
  scanf("%f", &c);
  f = 32 + c * 9/5;
  printf("Temperatura Fahrenheit %f", f);
  }
```

# Memoria principale/centrale



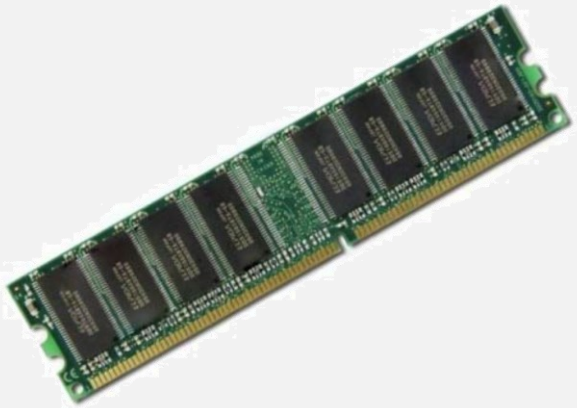
Definiamo memoria (principale) di un elaboratore il contenitore in cui sono memorizzati tutti i dati su cui lavora il processore

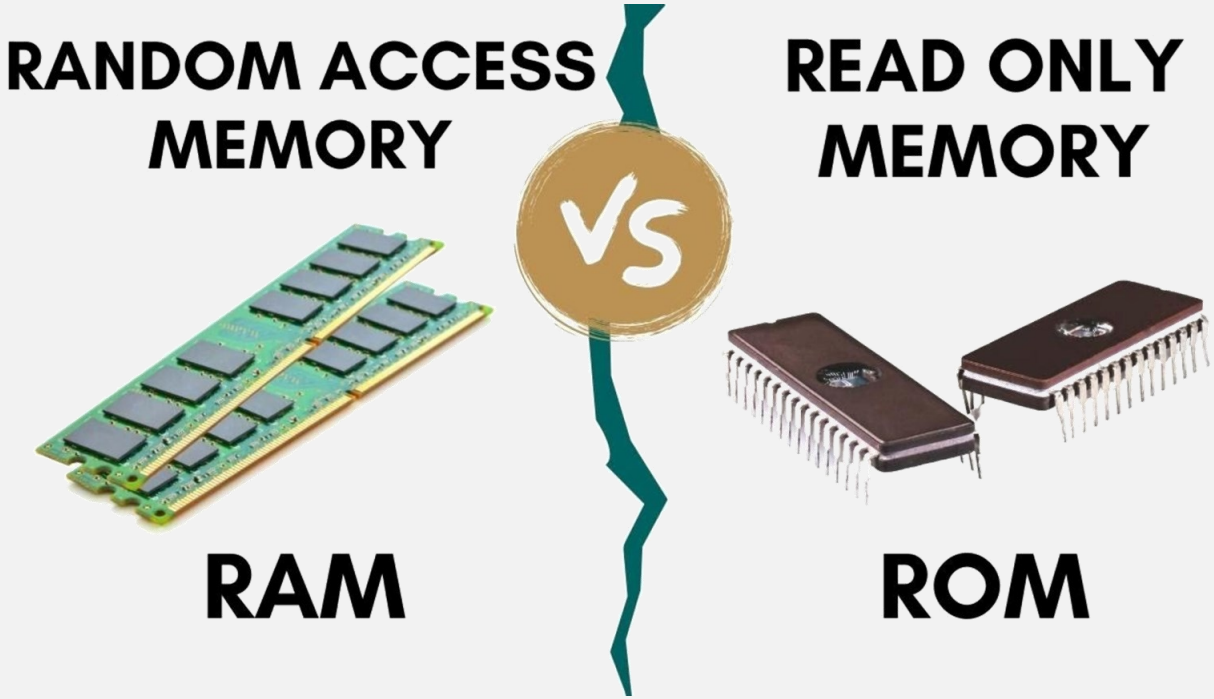
Possiamo schematizzare la memoria come una sequenza contigua di celle (chiamate anche locazioni di memoria)

Ciascuna cella fornisce l'unità minima di memorizzazione, ossia l'elemento più piccolo in cui si può memorizzare un'informazione

# Memoria principale/centrale

- Le memorie **RAM** (random access memory)
  - possono essere accedute sia in lettura che in scrittura
  - sono volatili (i dati memorizzati vengono persi allo spegnimento del calcolatore)
  - sono usate per memorizzare dati e programmi
- La memorie **ROM** (read only memory)
  - permettono solo la lettura dei dati
  - sono persistenti (mantengono il suo contenuto anche quando non c'è alimentazione)
  - sono usate per memorizzare alcuni programmi di sistema (*firmware*)

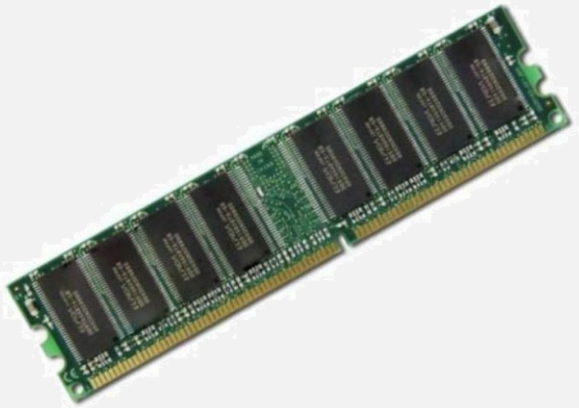


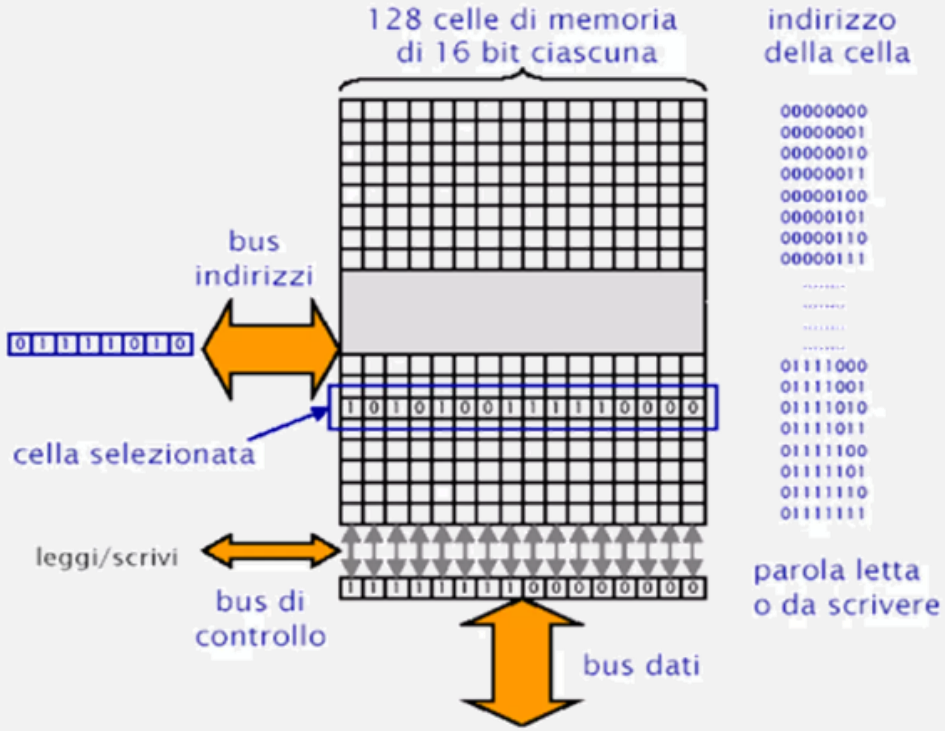


# Celle di memoria

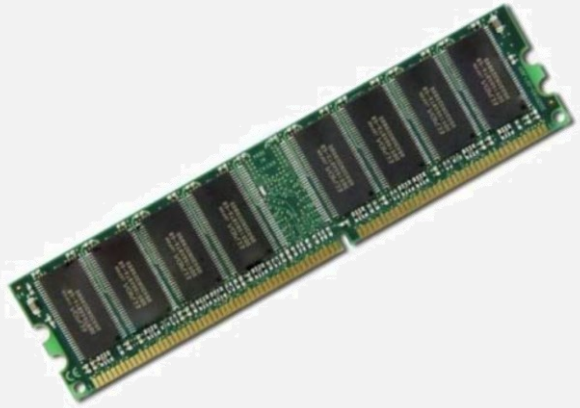
Ogni cella contiene un byte, ossia una sequenza di bit (cifre binarie)

- Tipicamente un byte è costituito da 8 bit
- Tutte le celle hanno quindi la stessa dimensione in termini di numero di bit



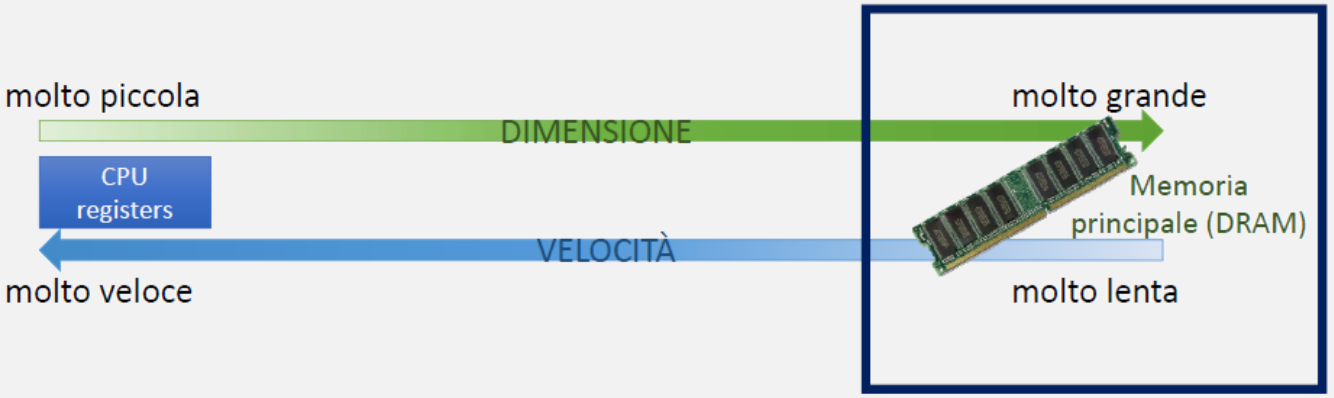
# Gerarchia di memoria



La memoria principale è grande abbastanza da contenere svariati programmi, ma ha due limitazioni:

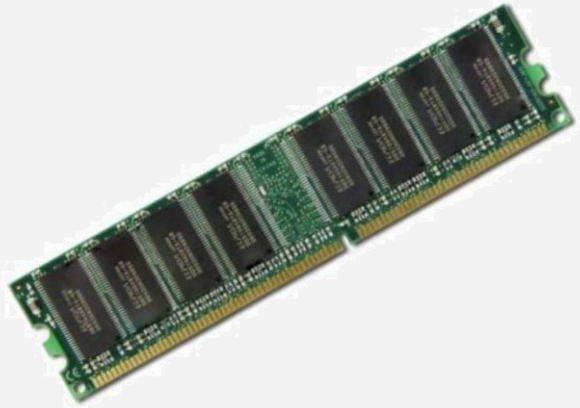
1. È lenta: considerando le attuali tecnologie per lo sviluppo di memorie esiste un compromesso tra la loro dimensione e la loro velocità
2. È volatile: è capace di conservare l'informazione solo finche il computer rimane acceso

Per ovviare alla lentezza della memoria principale, si costruiscono delle gerarchie




La memoria più veloce è usata per realizzare i **registri** dentro la CPU
..ma è piccolissima, occorre aggiornarne in continuazione il contenuto

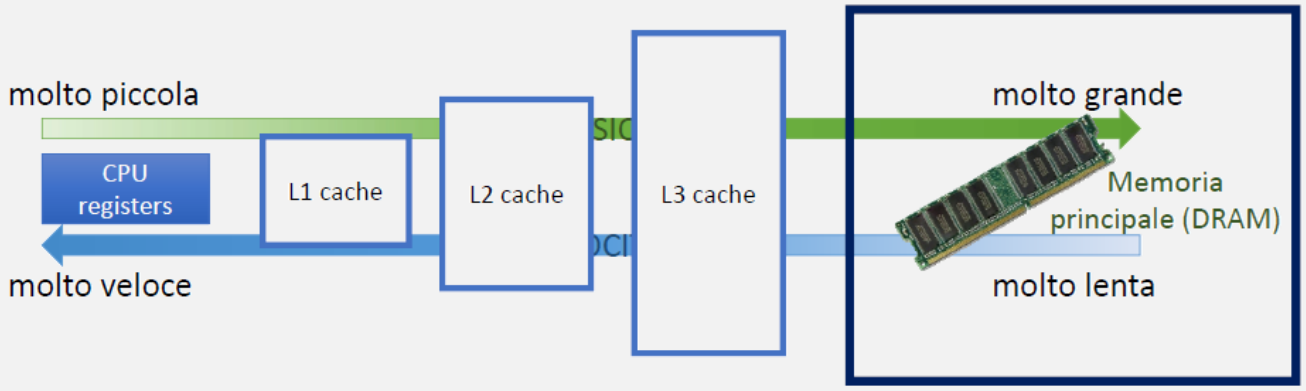
# Gerarchia di memoria



La memoria principale è grande abbastanza da contenere svariati programmi, ma ha due limitazioni:

1. È lenta: considerando le attuali tecnologie per lo sviluppo di memorie esiste un compromesso tra la loro dimensione e la loro velocità
2. È volatile: è capace di conservare l'informazione solo finche il computer rimane acceso

Per ovviare alla lentezza della memoria principale, si costruiscono delle gerarchie




Tra i **registri** e la **memoria principale** ci sono diversi livelli di **CACHE**

- una memoria gestita trasparentemente dal sistema
- Porta vicino ai registri i dati necessari
- Nasconde l'alta **latenza** della memoria principale

# Memorie di massa

Come gestire il problema della volatilità della memoria principale?

• Per memorizzare le informazioni in modo permanente si usano **dispositivi di memorizzazione di massa**

**Memoria secondaria – memoria di massa**

- memorizza grandi masse di dati
- i dati memorizzati sopravvivono all'esecuzione dei programmi
- **non vi si può accedere direttamente** tramite la CPU
  → per essere elaborati dal processore, i dati devono passare dalla memoria centrale

Memoria principale vs memoria secondaria:

- La memoria secondaria memorizza in maniera permanente tutti i programmi e i dati del calcolatore
- La memoria centrale memorizza i programmi in esecuzione e i dati necessari per la loro esecuzione

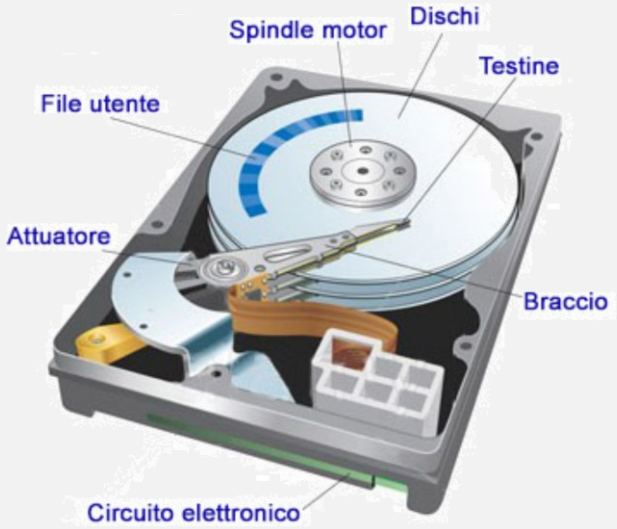
- **non volatilità**
  - i dati memorizzati non si perdono allo spegnimento del calcolatore (perché memorizzati in forma magnetica o ottica anziché elettronica)
- **grande capacità**
  - capacità maggiore (anche di diversi ordini di grandezza) rispetto alla memoria centrale
- **bassi costi**
  - il costo per bit di una memoria secondaria è minore (di diversi ordini di grandezza) rispetto alla memoria centrale
- **bassa velocità di accesso**
  - tempi di accesso maggiori (di qualche ordine di grandezza) rispetto a quelli della memoria principale

# Memorie di massa

Come gestire il problema della volatilità della memoria principale?

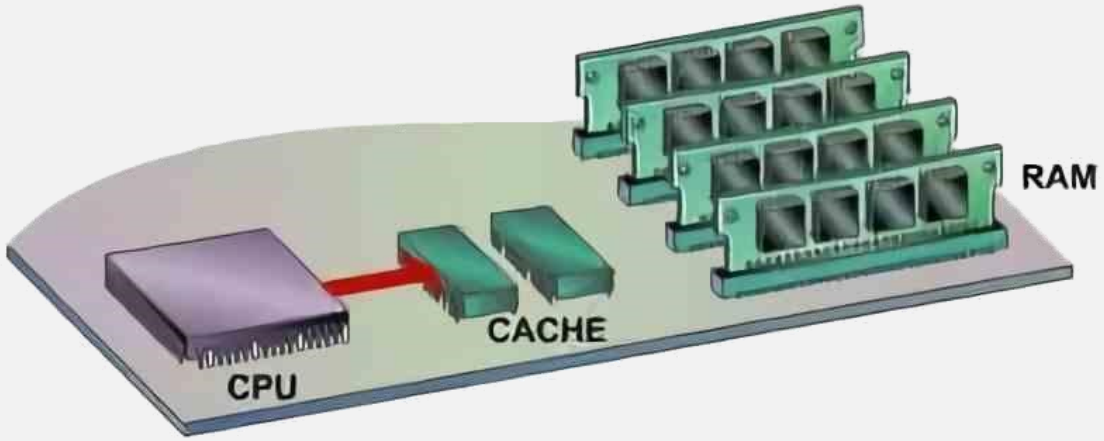
- Per memorizzare le informazioni in modo permanente si usano **dispositivi di memorizzazione di massa**

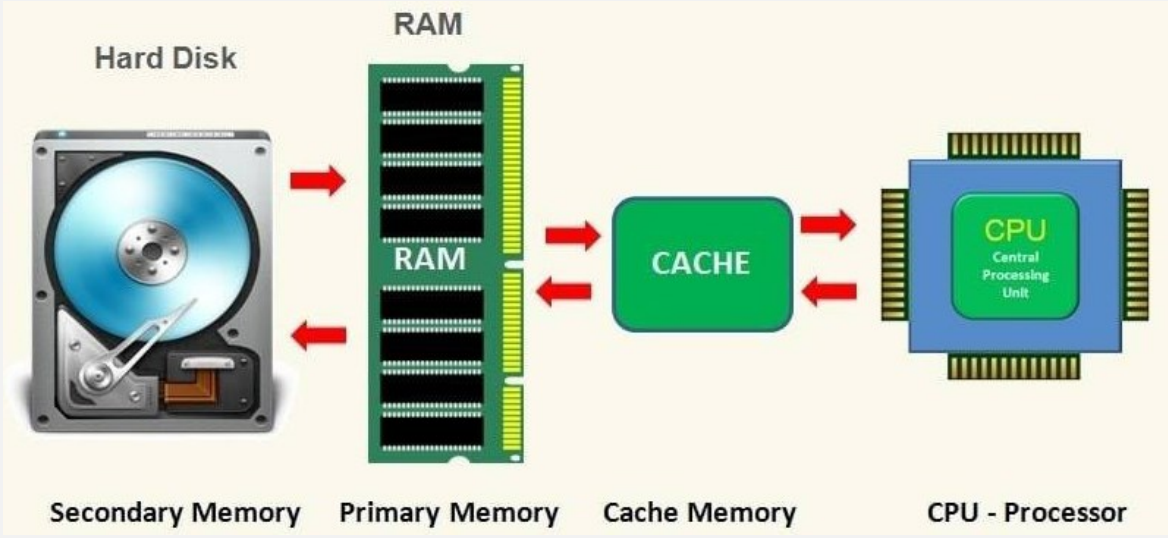
L'esempio più rappresentativo è il **disco fisso** o **hard disk**



# Memorie cache

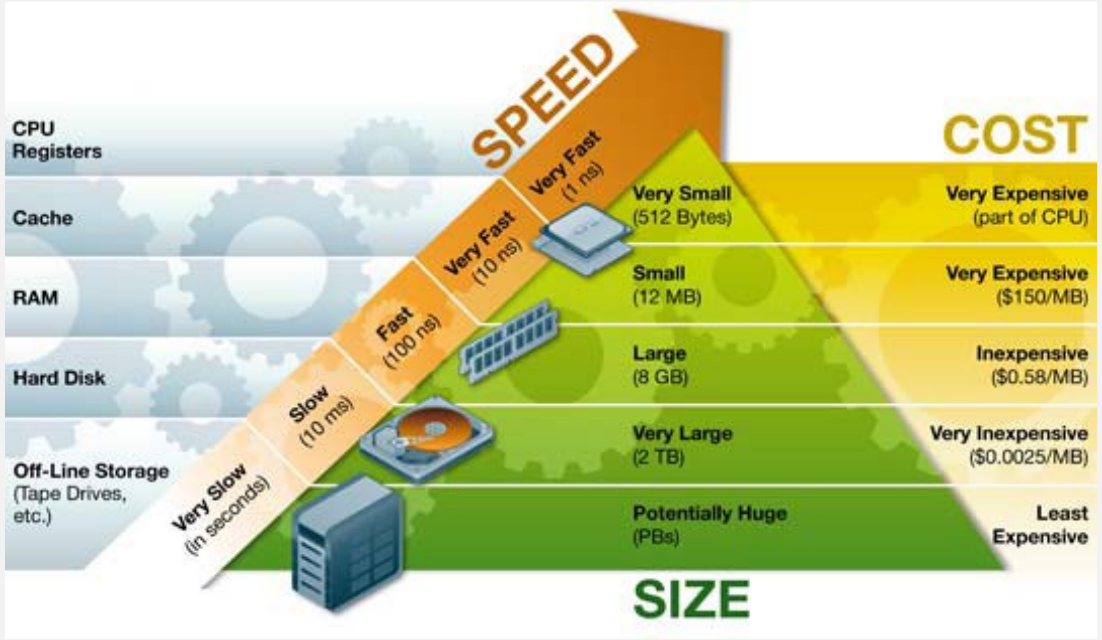
- E' una **memoria particolarmente veloce** usata per memorizzare i dati usati più frequentemente
  - La RAM ha tempi di accesso molto alti rispetto alla velocità dei microprocessori e ne ritarda l'elaborazione
- Strategia di utilizzo:
  - la prima volta che il microprocessore carica dei dati dalla memoria 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)
- La RAM non è realizzata con tale tipo di memoria perché si tratta di dispositivi **molto costosi!**
- Tipi di memoria cache:
  - **cache di I° livello:** contenuta nel microprocessore
  - **cache di II° livello:** aggiungibile successivamente

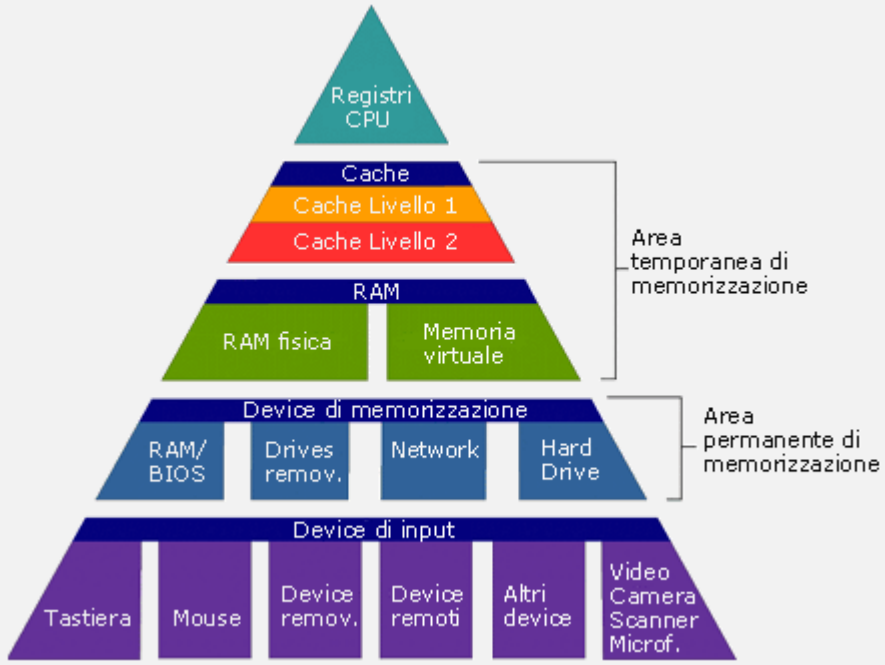


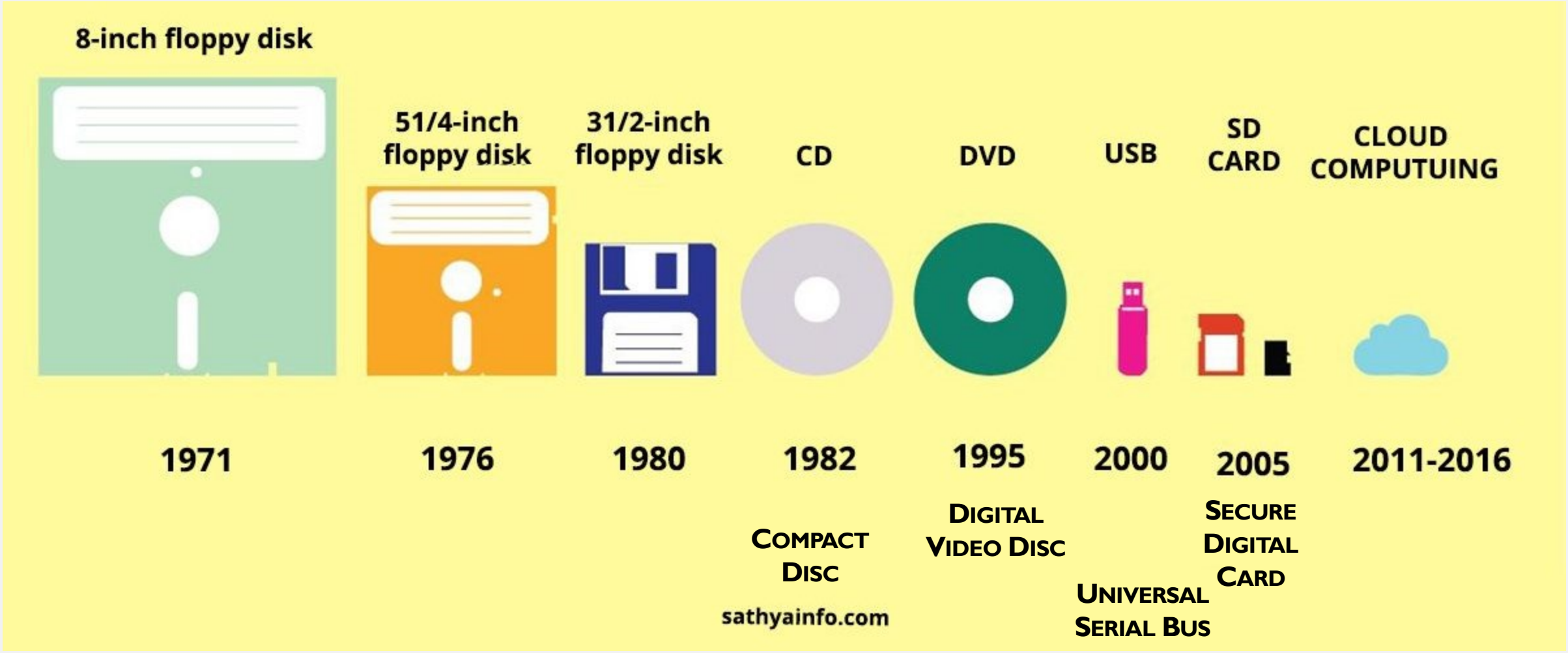
# Memorie di massa

Oltre ad essere non-volatili, le memorie di massa hanno una dimensione molto maggiore rispetto alla memoria principale (RAM)
Ma quanto sono veloci? (o lente)? Di fatto, le memorie di massa costituiscono parte della gerarchia di memoria, conservando il compromesso velocità/dimensione



# EVOLUTION OF STORAGE DEVICES



# Cenni di architettura del software

- Software = insieme (complesso) di programmi.
- Organizzazione a strati, ciascuno con funzionalità di livello più alto rispetto a quelli sottostanti:

*"Macchine Virtuali"*

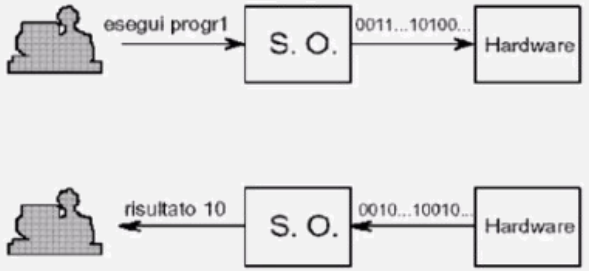

- **Firmware**:
  - strato di (*micro*-)programmi che agiscono direttamente sullo strato hardware
  - memorizzato dal costruttore su una memoria permanente (ROM)

# Sistema operativo (S.O.)

- Strato di programmi che opera *al di sopra di hardware e firmware* e **gestisce l'elaboratore**
- Le funzioni offerte dal S.O. dipendono dalla complessità del sistema di elaborazione:
  - gestione delle varie risorse hardware
  - gestione della memoria centrale
  - organizzazione e gestione della memoria di massa
  - interpretazione ed esecuzione di comandi elementari
  - gestione della multi-utenza e del multi-tasking

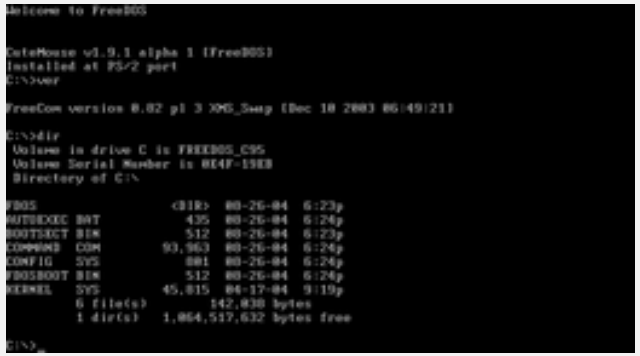
# Sistema operativo (S.O.)

- Un utente "vede" l'elaboratore solo tramite il S.O., che simula una "macchina virtuale"
  - diversi S.O. possono realizzare diverse macchine virtuali *sullo stesso hardware*
  - aumenta l'astrazione nell'interazione utente/elaboratore
    - senza S.O.: sequenze di bit
    - con S.O.: comandi, programmi, dati.
- Il S.O. **traduce le richieste dell'utente** in opportune **sequenze di comandi** da sottoporre alla macchina fisica
  - Il S.O. **esplicita qualsiasi operazione di accesso a risorse** hardware, implicitamente implicata dal comando dell'utente

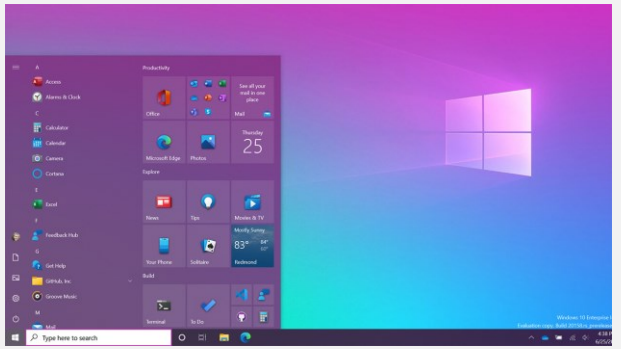



| Utente | S.O. |
|---|---|
| ➢ "*esegui progr1*" | input da tastiera<br>ricerca codice di "*progr1*" su disco<br>carica in RAM codice e dati<br>*<elaborazione>* |
| ➢ "*risultato = 10*" | output su video |

## Interfaccia testuale o a riga di comando




## Interfaccia grafica o GUI



# Reti informatiche

- Una rete informatica consente di mettere in comunicazione due o più elaboratori allo scopo di scambiare informazioni e condividere risorse.
- I computer collegati vengono detti **nodi** della rete

Vantaggi:

- Ogni computer di una rete può accedere alle risorse informative (programmi o dati) residenti su altri computer.
- Può utilizzare alcune periferiche (stampanti, fax) collegate agli altri computer.
- È possibile realizzare «applicazioni distribuite», cioè **programmi modulari**, i cui componenti «girano» su computer diversi.

# Reti informatiche

Una rete telematica ha:

- Componenti fisiche (hardware)
- Componenti logiche (software)

**Componenti hardware**: computer, canali trasmissivi, apparati di trasmissione

**Componenti logiche:**
Per lo scambio di informazioni i diversi nodi della rete utilizzano uno stesso linguaggio di comunicazione detto "**Protocollo**"
**Protocolli**: programmi di gestione del collegamento e del traffico dei dati. Svolgono diverse funzioni: instradamento dei dati tra i vari nodi di una rete, correzione degli errori di trasmissione, coordinazione dei rapporti tra i moduli di una applicazione distribuita

# Classificazione delle reti informatiche

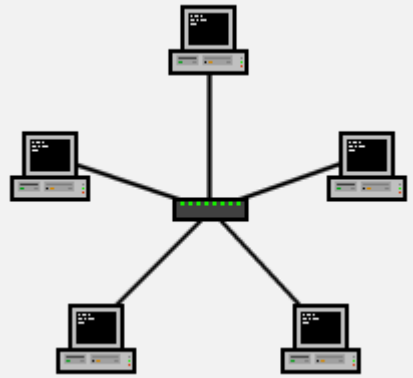
Sono possibile diverse classificazioni sulla base di:

1. Topologia di interconnessione fisica
2. Ampiezza geografica
3. Tipo di organizzazione e gestione dei protocolli di comunicazione

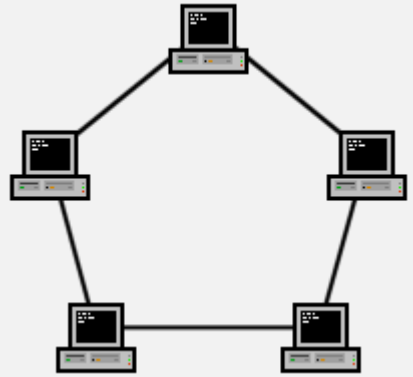
# 1. Topologie di interconnessione fisica

La topologia di interconnessione fisica descrive il grafo dei collegamenti che coinvolgono ciascun nodo della rete

Ogni collegamento tra i **nodi** di elaborazione è rappresentato da un **arco** del grafo



1. **Topologia a stella**: ha un nodo centrale, a cui sono collegati tutti gli altri nodi. In passato molto utilizzata per connessioni tra computer periferici ed un grande elaboratore centralizzato. Problema del collo di bottiglia; scarsamente robusta.

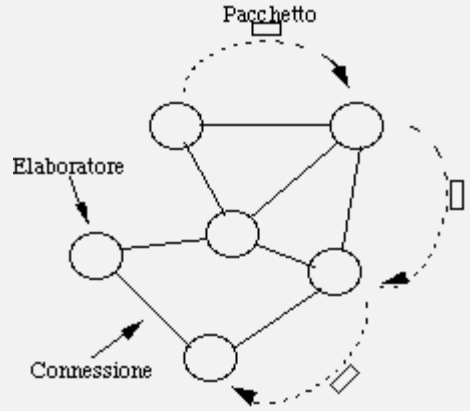


2. **Topologia ad anello (token ring)**: ciascun nodo della rete è collegato al precedente ed al successivo in un grafo che si chiude ad anello. Ogni nodo passa al suo successore un messaggio (token) che circola nella rete. Bassa velocità di comunicazione; scarsamente robusta.

# 1. Topologie di interconnessione fisica

3. **Topologia a Bus**: tutti i nodi di elaborazione interagiscono con un supporto di comunicazione comune detto bus. Tutti i nodi possono scrivere e leggere contemporaneamente dal supporto comune. Facilmente espandibile.

4. **Topologia irregolare**: ciascun nodo ha un numero variabile di collegamenti punto a punto con altri nodi della rete. Molto robusta: esistono più percorsi che collegano coppie di nodi. Espandibile. Caratteristica delle reti internet.



## 2. Ampiezza della distribuzione geografica

È possibile classificare le reti in base alla distribuzione geografica dei nodi.

**LAN (Local Area Network)**
Realizzata all'interno di uno o più edifici contigui, solitamente per esigenze di un singolo ente.

**WAN (Wide Area Network)**
Le distanze fra i nodi di elaborazione sono notevoli (>km).

**Internet**
È il risultato dell'interconnessione tra reti di differenti tipologie. Il protocollo che unifica tutte le componenti presenti è il TCP/IP.

**Intranet**
Reti locali aziendali, che utilizzano tecnologie sviluppate per internet per scambiare dati internamente alla rete locale aziendali (posta interna, relazioni…).

**Extranet**
Diverse reti intranet di una stessa azienda collegate in un'unica rete virtuale con accesso riservato.

# 3. Modalità di comunicazione

Consideriamo una rete con topologia irregolare e supponiamo che si possa individuare univocamente ciascun nodo.
Detto il nodo A mittente e il nodo B destinatario, supponiamo che il nodo A debba mandare un messaggio al nodo B: l'insieme dei nodi attraversati dal messaggio per raggiungere il destinatario è detto cammino/percorso del messaggio.

1. **Commutazione di circuito**: usa una connessione fisica tra mittente e destinatario realizzata attraverso la connessione di nodi intermedi sulla rete; il circuito è occupato per tutta la durata della comunicazione (rete telefonica tradizionale).

2. **Commutazione di messaggio**: ogni comunicazione occupa al più una tratta del circuito alla volta. Ciascun nodo deve decidere dove instradare il messaggio (servizio telegrafico o servizio postale).

3. **Commutazione di pacchetto**: il messaggio viene suddiviso in unità di dimensioni omogenee fissate dette pacchetti. Pacchetti di uno stesso messaggio seguono percorsi diversi; il messaggio è poi riassemblato nella forma originale dal destinatario.

# TCP/IP (Transfer Control Protocol / Internet Protocol)

Il protocollo TCP/IP è un protocollo basato sulla commutazione di pacchetto utilizzato per la comunicazione tra nodi di elaborazione attraverso internet.

Sul TCP/IP sono basati una serie di servizi fondamentali:

- **mail**: permette di inviare/ricevere messaggi di posta elettronica;
- **ftp**, File Transfer Protocol: consente di effettuare il trasferimento di file;
- **telnet**: consente di stabilire una connessione remota con un altro elaboratore, creando un terminale virtuale che consente di utilizzare il sistema remoto come se fosse un terminale direttamente collegato;
- **http, https**: consente di trasferire ipertesti nella rete, è il protocollo che realizza la trasmissione delle pagine web
- **newsgroup**: consente di interagire e scambiare informazioni con i gruppi di discussione costituiti sui diversi argomenti

# Gli indirizzi IP

Allo scopo di instradare correttamente i pacchetti di informazioni e realizzare correttamente le comunicazioni sulla rete, è necessario che venga identificato univocamente ciascun computer sulla rete.

A questo scopo, ciascun nodo collegato ad internet è identificato da un IP Address, costituito da 4 gruppi di cifre (ciascuna da 0 a 255) separati da un punto.

Accanto alla forma numerica di un indirizzo IP, è possibile utilizzare un nome mnemonico.

# Focus: banche dati biologiche

La Bioinformatica è l'applicazione di strumenti propri delle scienze dell'informazione (es. algoritmi, intelligenza artificiale, databases) a problemi di interesse biologico, biotecnologico e biomedico.

**Cos'è una banca dati**? Una banca dati (o database) è un insieme organizzato di dati, memorizzati in un computer, che possono essere facilmente recuperati, aggiornati e interrogati. In altre parole, una banca dati è un sistema di archiviazione e gestione dei dati che consente di organizzare e consultare grandi quantità di informazioni.

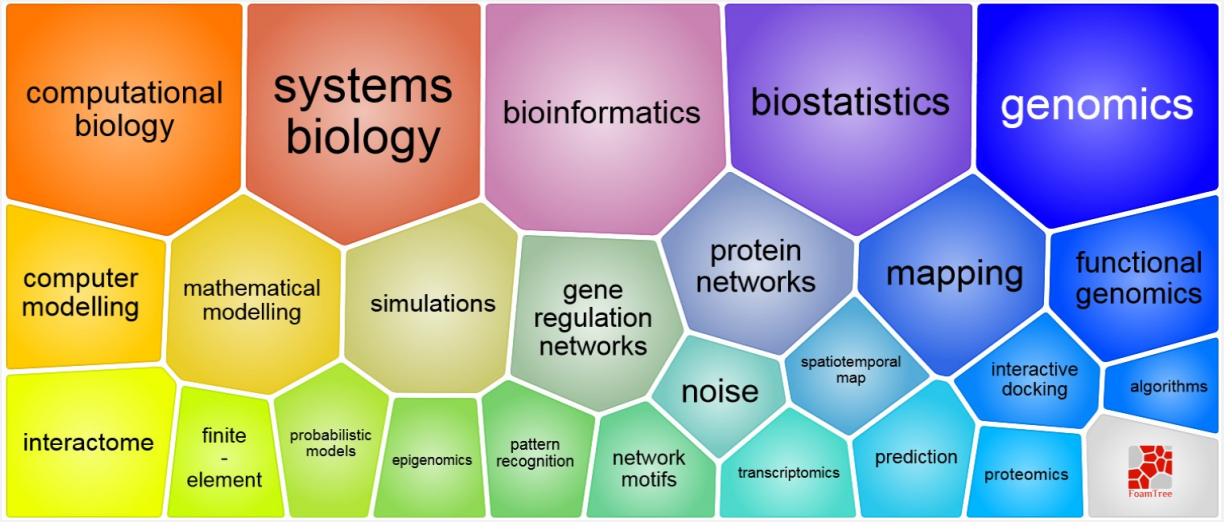
**Cos'è una banca dati biologica?** Una banca dati biologica è una banca dati che contiene informazioni biologiche, come sequenze genomiche, strutture proteiche, funzioni di proteine, espressione genica e molto altro. Le banche dati biologiche sono state create per gestire e archiviare grandi quantità di dati biologici, consentendo ai ricercatori di accedere a queste informazioni in modo efficiente e di utilizzarle per i loro studi scientifici.

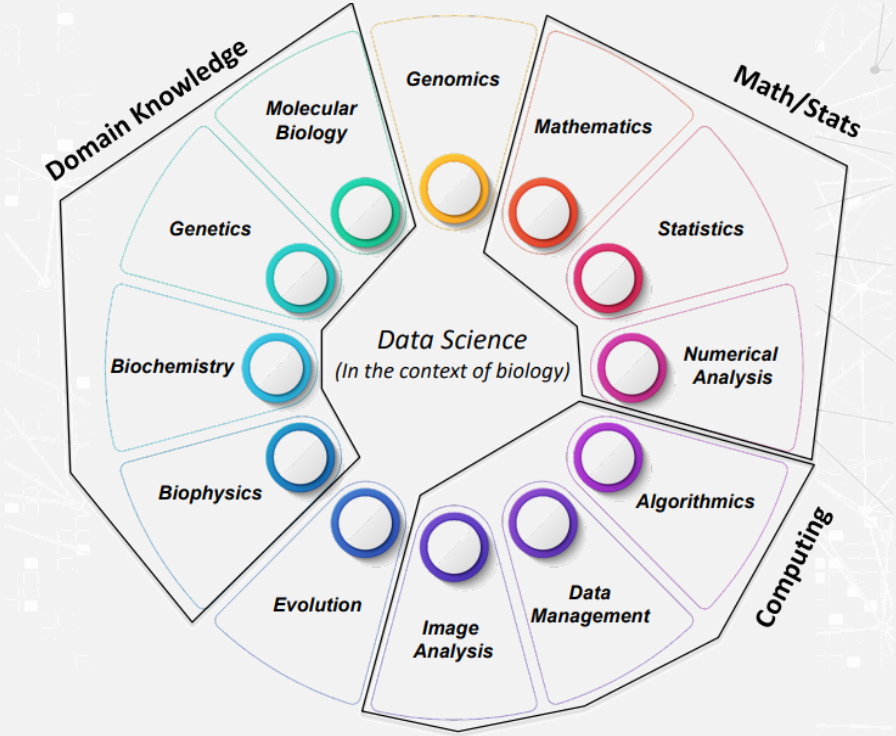
# Focus: banche dati biologiche

La ricerca biologica moderna genera **enormi quantità di dati**, dalle sequenze genomiche ai dati di espressione genica, che possono essere utilizzati per rispondere a importanti domande biologiche.
Tuttavia, per fare ciò in modo efficace, è necessario archiviare e gestire questi dati in modo accurato e organizzato

# Focus: banche dati biologiche

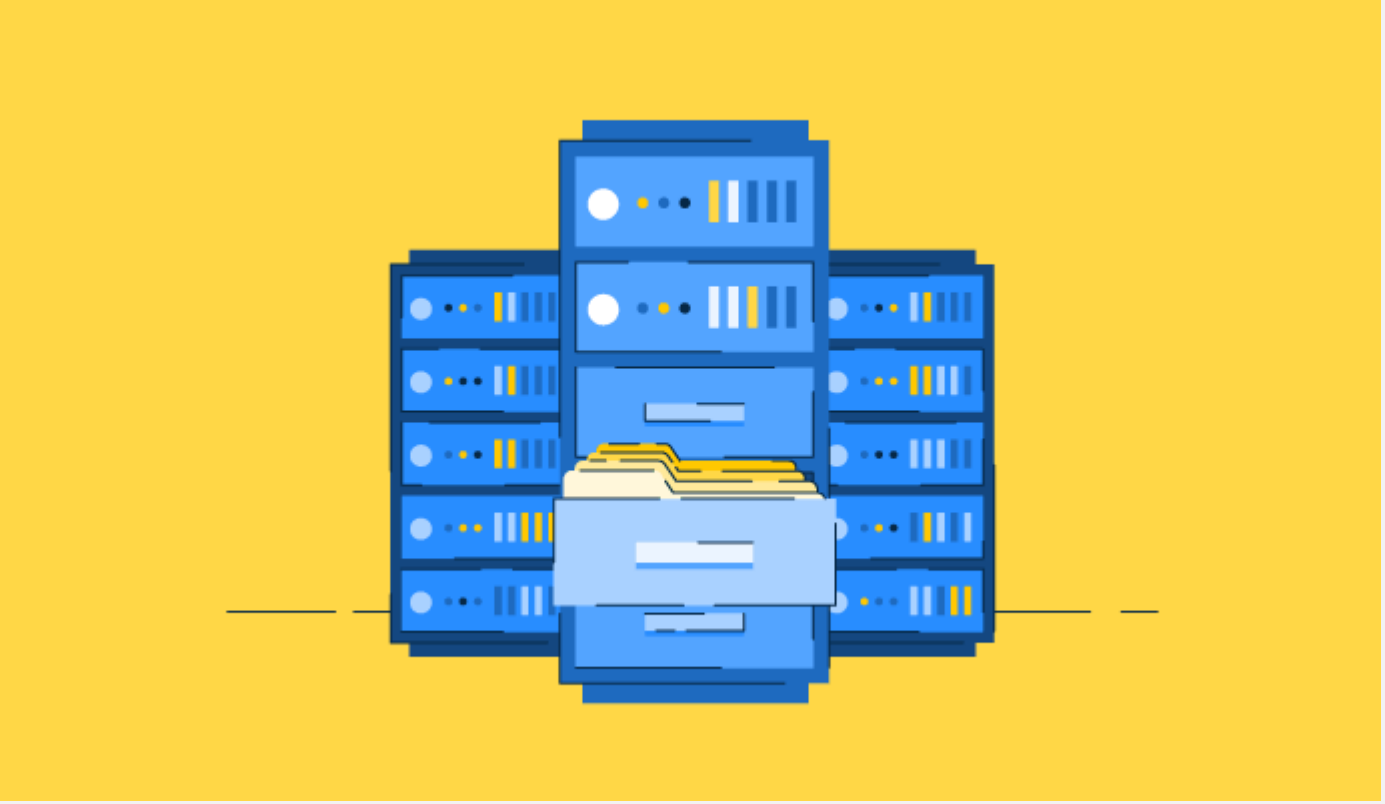
I dati biologici sono spesso molto complessi e richiedono l'uso di tecniche avanzate di analisi dei dati per essere compresi e utilizzati efficacemente. Ad esempio, le sequenze genomiche possono essere lunghe e complesse, con molte regioni non codificanti e regioni codificanti che producono proteine diverse. Inoltre, i dati biologici possono provenire da diverse fonti, come la genomica, la trascrittomica, la proteomica e la metabolomica, ognuna delle quali fornisce informazioni su diversi aspetti del sistema biologico.



# Focus: banche dati biologiche

La **gestione e la condivisione dei dati biologici** sono essenziali per evitare la duplicazione degli sforzi di ricerca e aumentare la produttività scientifica complessiva.
Inoltre, l'archiviazione dei dati biologici consente ai ricercatori di accedere ai dati stessi e di utilizzarli per rispondere alle loro domande di ricerca.

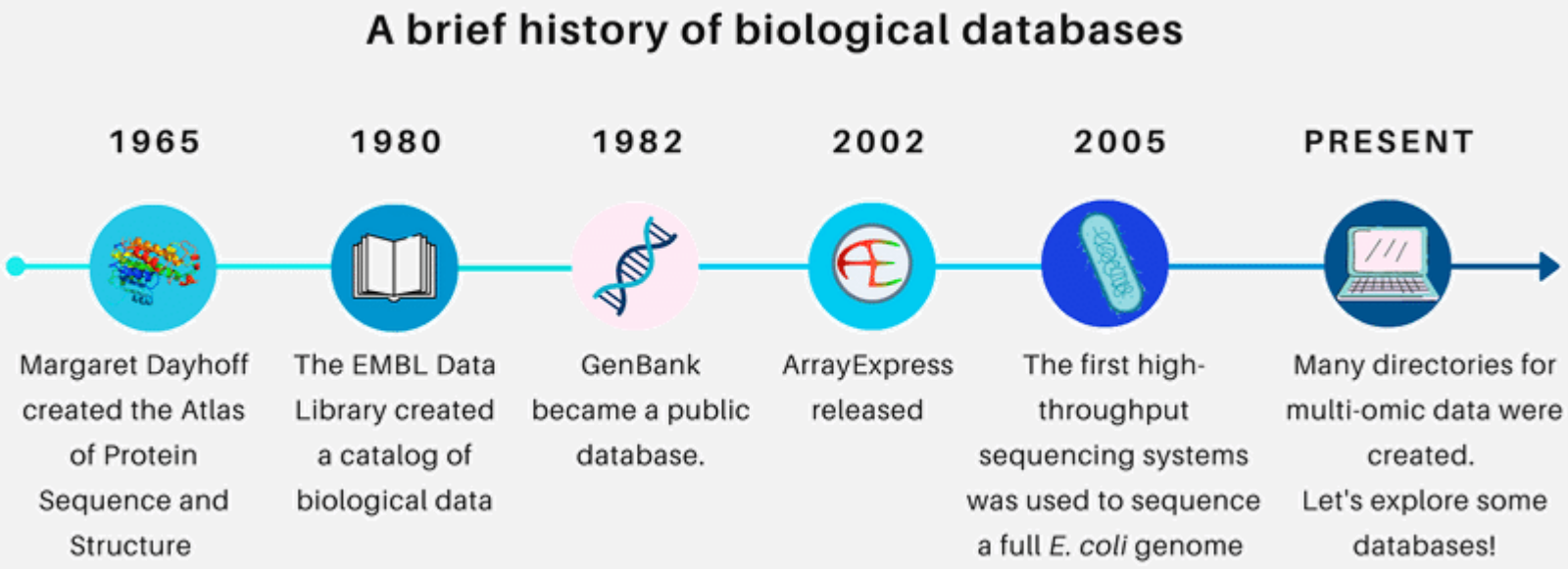


# Focus: banche dati biologiche

1. **Accessibilità**: un database biologico deve essere facilmente accessibile a tutti gli utenti interessati, indipendentemente dal loro livello di competenza informatica.
2. **Aggiornamento regolare**: i dati contenuti nel database biologico devono essere aggiornati regolarmente per includere nuove informazioni e correzioni.
3. **Organizzazione**: un database biologico deve essere organizzato in modo da consentire una facile ricerca e recupero delle informazioni. Ciò può includere l'organizzazione per specie, tipo di molecola biologica o funzione biologica.
4. **Standardizzazione**: i dati contenuti nel database biologico devono essere standardizzati in modo da garantire che possano essere utilizzati in modo coerente da tutti gli utenti
5. **Compatibilità**: un database biologico deve essere compatibile con altri database biologici e software di analisi, in modo che le informazioni possano essere facilmente scambiate e utilizzate in diversi contesti.
6. **Sicurezza**: un database biologico deve essere protetto da accessi non autorizzati e attacchi informatici per garantire la sicurezza delle informazioni sensibili contenute al suo interno
7. **Documentazione**: un database biologico deve essere accompagnato da documentazione chiara e completa che spiega come accedere ai dati e come sono stati raccolti e curati.

# Focus: banche dati biologiche

Le prime banche dati biologiche erano semplici pagine web che fornivano l'accesso a collezioni di file di testo. Ogni file di testo aveva un nome che corrispondeva ad un codice identificativo della molecola descritta al suo interno. Ad esempio se il file conteneva informazioni su una proteina avente codice PR001 allora il file di testo contenente la sua scheda aveva nome PR001.txt
La pratica di assegnare un identificativo ad ogni molecola inserita in una banca dati biologica è utilizzata ancora adesso: l'identificativo viene detto **accession number** ed è una parola contenente lettere, numeri e simboli

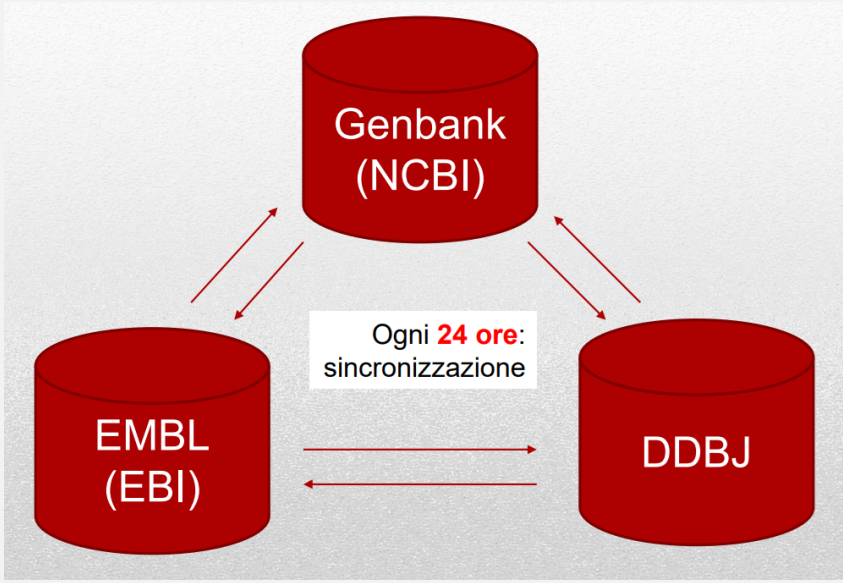


# Focus: banche dati biologiche

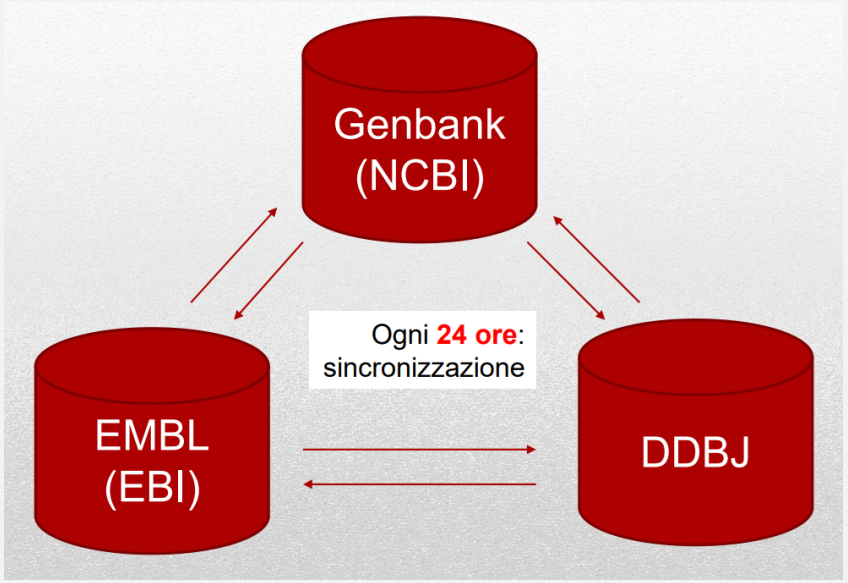
Le banche dati biologiche si possono classificare in base a vari criteri. Uno di questi riguarda la qualità delle informazioni in esse contenute.

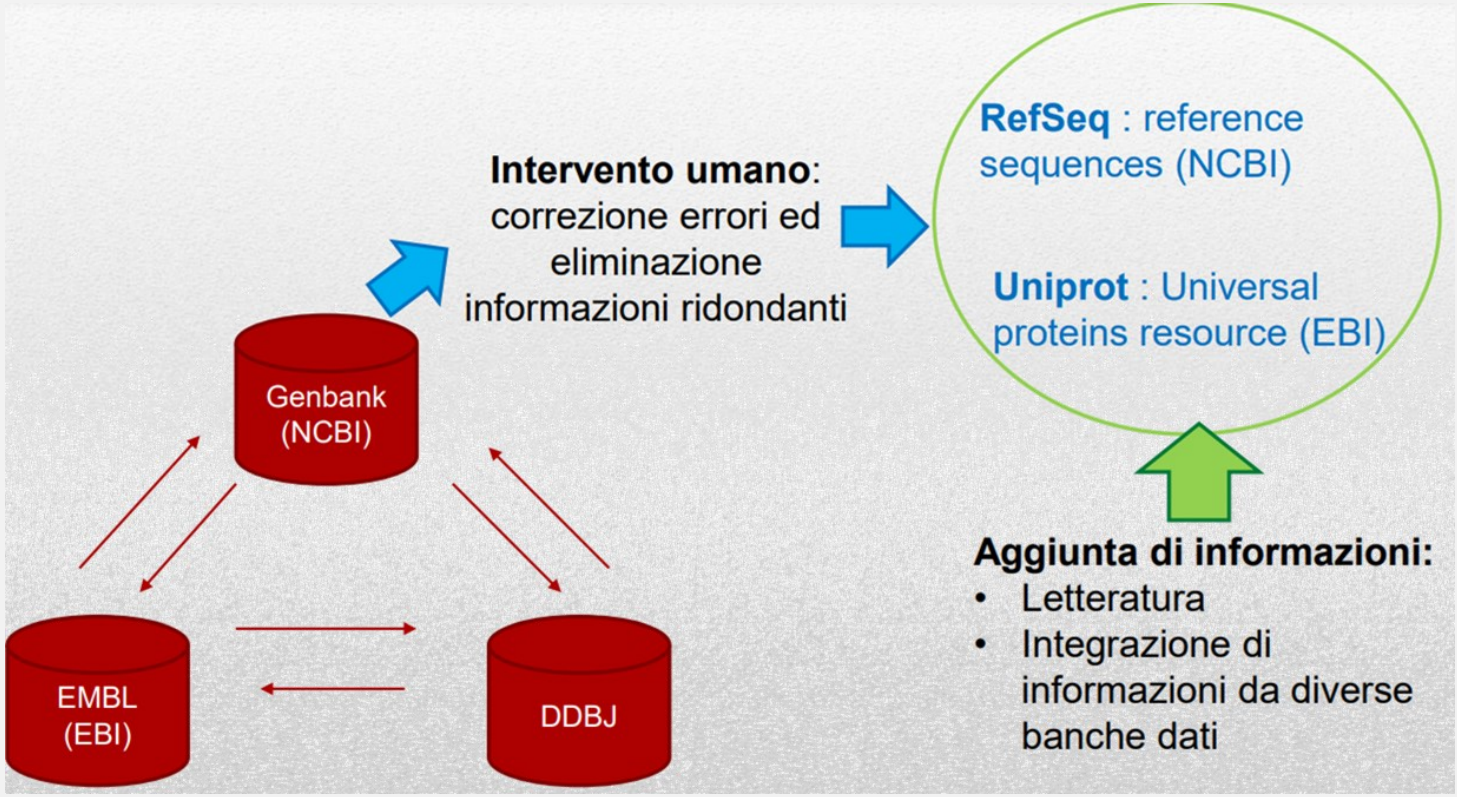
• **Banche dati PRIMARIE** : dette anche collettori primari. Il loro ruolo è quello di raccogliere, giornalmente, tutte le informazioni che riguardano biomolecole prodotte in tutti i laboratori del mondo e renderle disponibili.

• **Banche dati SECONDARIE**: dato che l'informazione contenuta nei collettori primari è sporca e ridondante esse esaminano i dati dei collettori, correggono eventuali errori, includono informazioni aggiuntive e rendono disponibili i risultati di questo processo di raffinamento.

# Focus: banche dati biologiche



# Focus: banche dati biologiche

**Banche dati PRIMARIE**

Queste banche dati contengono, letteralmente, **miliardi di schede**. Sarebbe impossibile trovare quello di cui abbiamo bisogno in assenza di strumenti che permettano di cercare le informazioni a cui siamo interessati.

Per questo, le banche dati biologiche non sono mai costituite solamente dalla collezione di dati che contengono ma anche da un **insieme di strumenti progettati per rendere possibile estrazione e manipolazione delle informazioni** in esse contente.

Wikipedia (lista di banche dati biologiche) https://en.wikipedia.org/wiki/List_of_biological_databases

Nucleic Acids Research Database Listing http://nar.oupjournals.org/cgi/content/full/30/1/1/DC1 (esempio di pubblicazione in cui è presente una lista di database biologici "storica" … articolo del 2002)

• Più di 500 banche dati esistenti sono state catalogate fino ad oggi. E sono costantemente in crescita.

# Focus: banche dati biologiche

## Accediamo ad una banca dati biologica

Supponiamo di conoscere il "nome" di un gene: INDY. E di voler cercare informazioni su di esso in una banca dati. Nel moscerino della frutta (*Drosophila melanogaster*) è stato identificato un gene che, se mutato, raddoppia la durata della vita media dei moscerini. A questo gene è stato dato il nome di INDY: "I'm Not Dead Yet"

Prima prova: usiamo strumenti "classici" per cercare informazioni …

# Focus: banche dati biologiche

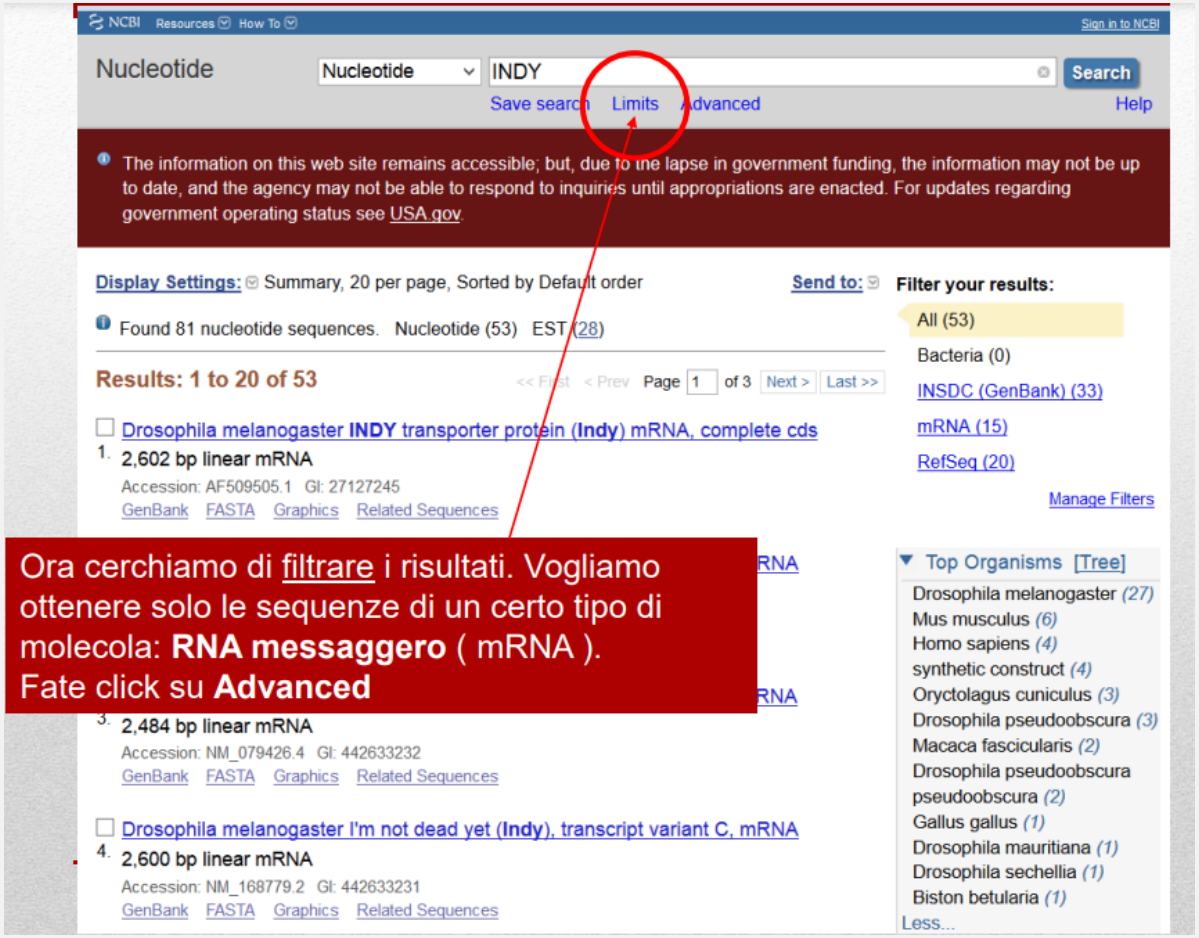
## Accediamo ad una banca dati biologica

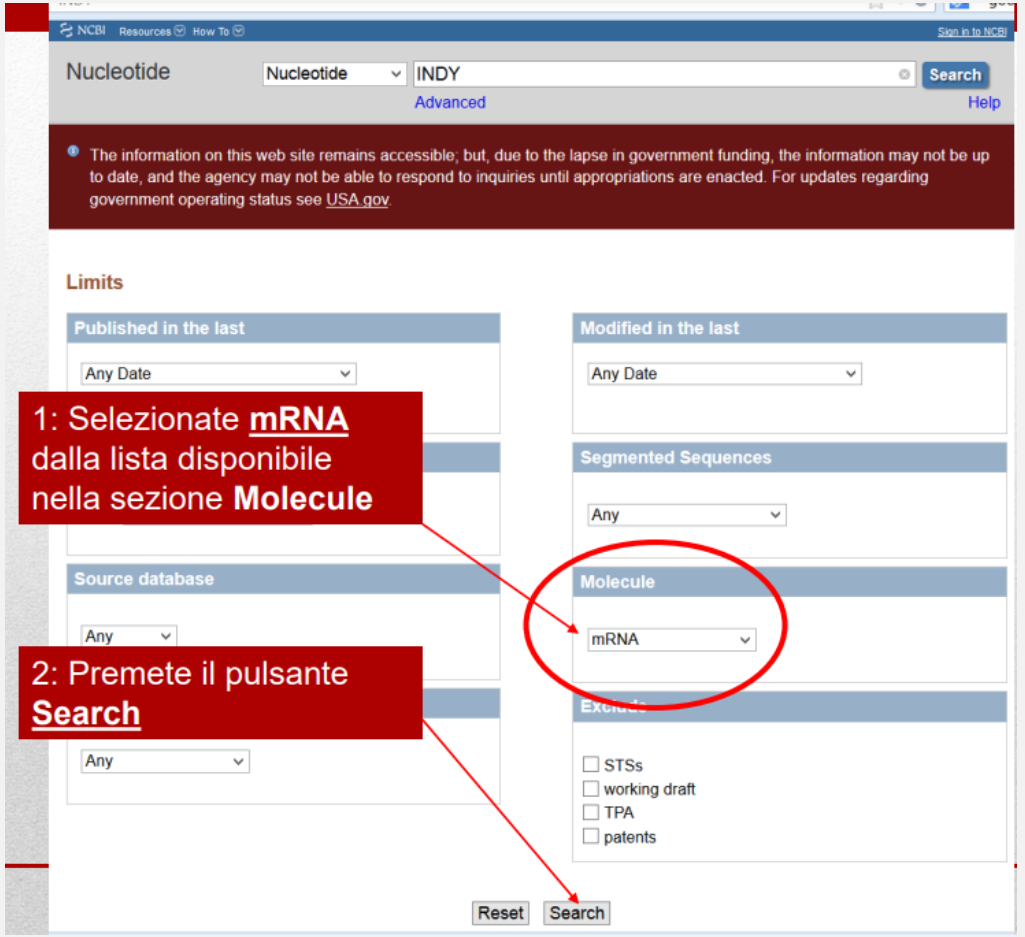
Supponiamo di conoscere il "nome" di un gene: INDY. E di voler cercare informazioni su di esso in una banca dati. Nel moscerino della frutta (*Drosophila melanogaster*) è stato identificato un gene che, se mutato, raddoppia la durata della vita media dei moscerini. A questo gene è stato dato il nome di INDY: "I'm Not Dead Yet"

Apriamo il web browser e colleghiamoci alla divisione NUCLEOTIDE delle banche dati gestite da NCBI:
http://www.ncbi.nlm.nih.gov/nuccore
Scegliendo la "sezione" Nucleotide di Genbank otterremo solo risultati riguardanti molecole composte da nucleotidi (DNA o RNA). Non otterremo risultati riguardanti le schede delle proteine.

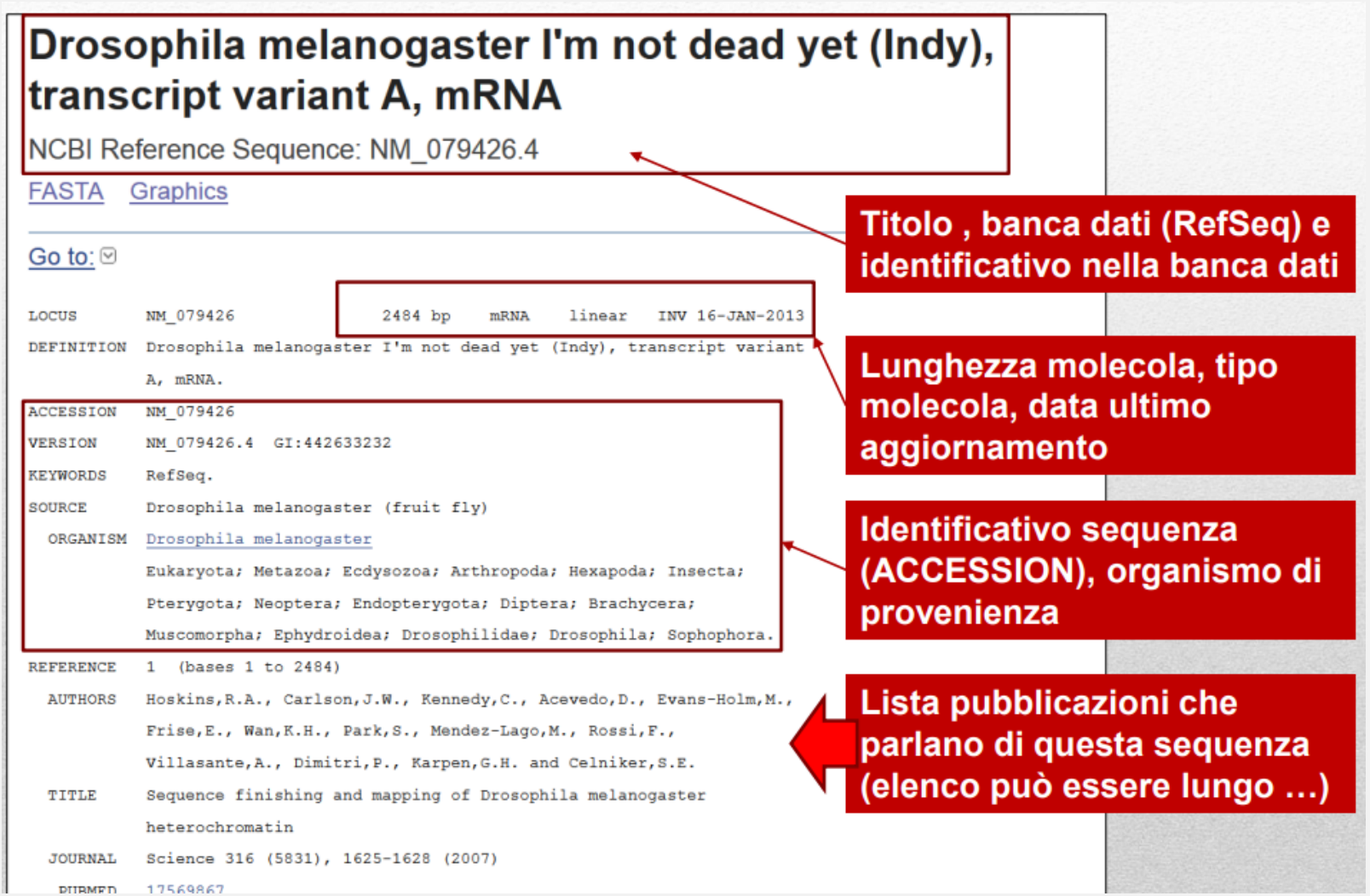
# Focus: banche dati biologiche

# Focus: banche dati biologiche



# Focus: banche dati biologiche



**Drosophila melanogaster I'm not dead yet (Indy), transcript variant A, mRNA**

NCBI Reference Sequence: NM_079426.4

FASTA Graphics

Go to:

```
LOCUS       NM_079426              2484 bp    mRNA    linear   INV 16-JAN-2013
DEFINITION  Drosophila melanogaster I'm not dead yet (Indy), transcript variant
            A, mRNA.
ACCESSION   NM_079426
VERSION     NM_079426.4  GI:442633232
KEYWORDS    RefSeq.
SOURCE      Drosophila melanogaster (fruit fly)
  ORGANISM  Drosophila melanogaster
            Eukaryota; Metazoa; Ecdysozoa; Arthropoda; Hexapoda; Insecta;
            Pterygota; Neoptera; Endopterygota; Diptera; Brachycera;
            Muscomorpha; Ephydroidea; Drosophilidae; Drosophila; Sophophora.
REFERENCE   1  (bases 1 to 2484)
  AUTHORS   Hoskins,R.A., Carlson,J.W., Kennedy,C., Acevedo,D., Evans-Holm,M.,
            Frise,E., Wan,K.H., Park,S., Mendez-Lago,M., Rossi,F.,
            Villasante,A., Dimitri,P., Karpen,G.H. and Celniker,S.E.
  TITLE     Sequence finishing and mapping of Drosophila melanogaster
            heterochromatin
  JOURNAL   Science 316 (5831), 1625-1628 (2007)
   PUBMED   17569867
```

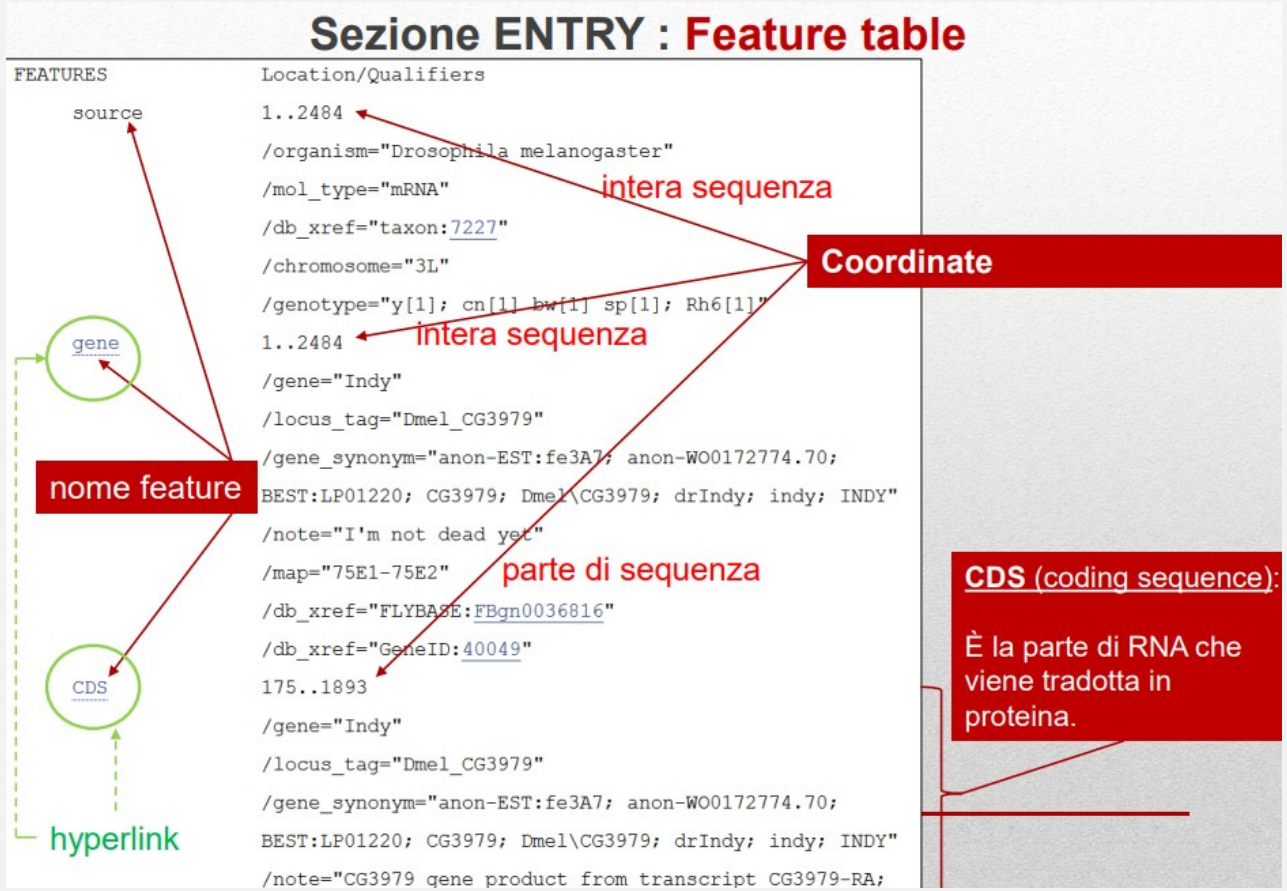
**Titolo , banca dati (RefSeq) e identificativo nella banca dati**

**Lunghezza molecola, tipo molecola, data ultimo aggiornamento**

**Identificativo sequenza (ACCESSION), organismo di provenienza**

**Lista pubblicazioni che parlano di questa sequenza (elenco può essere lungo …)**

# Focus: banche dati biologiche




# Focus: banche dati biologiche

Le banche dati biologiche possono essere ulteriormente suddivise in base al tipo di dati che contengono. Ad esempio, le **banche dati di sequenza** contengono le sequenze di DNA o RNA, mentre le **banche dati di struttura** contengono le informazioni sulla struttura tridimensionale delle proteine. Le **banche dati di funzione** forniscono informazioni sulla funzione biologica delle molecole, come le attività enzimatiche o le vie metaboliche coinvolte nella sintesi di un metabolita. Le **banche dati di espressione** contengono informazioni sulla quantità di mRNA o proteine prodotte da un gene in diverse condizioni o tessuti.

# Focus: banche dati biologiche

## Banche dati di sequenza

**NCBI GenBank** è un database di sequenze di DNA, RNA e proteine gestito dal National Center for Biotechnology Information (NCBI) degli Stati Uniti. GenBank contiene oltre 300 miliardi di basi di dati genetiche, tra cui sequenze di organismi eucarioti, procarioti e virali. GenBank è stato fondato nel 1982 ed è uno dei più antichi e grandi database di sequenze a livello mondiale.

**European Nucleotide Archive (ENA)** è un database di sequenze di DNA, RNA e proteine gestito dal European Bioinformatics Institute (EMBL-EBI). ENA è uno dei più grandi database di sequenze a livello mondiale. È stato fondato nel 2002 e raccoglie, conserva e rende liberamente accessibili i dati di sequenza provenienti da tutte le forme di vita.

# Focus: banche dati biologiche

## Banche dati di sequenza

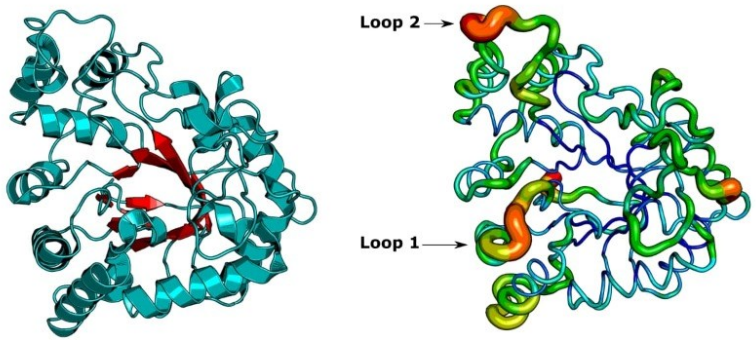
**DNA Data Bank of Japan (DDBJ)** è un database di sequenze di DNA, RNA e proteine gestito dall'Università di Chiba, in Giappone. DDBJ è uno dei tre membri fondatori del International Nucleotide Sequence Database Collaboration (INSDC) insieme a GenBank e ENA. Il database contiene sequenze di organismi eucarioti, procarioti e virali e collabora con GenBank ed ENA per mantenere un unico database di sequenze a livello mondiale.

# Focus: banche dati biologiche

## Banche dati di strutture

**Protein Data Bank (PDB)** contiene informazioni sulla struttura tridimensionale delle proteine, ottenute attraverso tecniche sperimentali come la cristallografia ai raggi X e la risonanza magnetica nucleare (NMR). Le informazioni sulle strutture tridimensionali delle proteine presenti nel PDB includono le coordinate atomiche e le informazioni sulla sequenza di amminoacidi. Queste informazioni possono essere utilizzate per comprendere le proprietà e le funzioni delle proteine, per identificare potenziali bersagli terapeutici e per sviluppare nuovi farmaci.

RCSB PDB
PROTEIN DATA BANK




**Atoms in loop 1**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ATOM | 15 | O | ARG | A | 3 | 12.195 | 41.671 | 55.513 | 1.00 | 28.42 | O |
| ATOM | 16 | CB | AARG | A | 3 | 14.362 | 39.263 | 55.270 | 0.50 | 29.74 | C |
| ATOM | 17 | CB | BARG | A | 3 | 14.355 | 39.274 | 55.278 | 0.50 | 32.13 | C |
| ATOM | 18 | CG | AARG | A | 3 | 15.248 | 38.232 | 54.592 | 0.50 | 31.80 | C |
| ATOM | 19 | CG | BARG | A | 3 | 15.294 | 38.303 | 54.582 | 0.50 | 36.13 | C |
| ATOM | 20 | CD | AARG | A | 3 | 16.718 | 38.490 | 54.881 | 0.50 | 35.67 | C |
| ATOM | 21 | CD | BARG | A | 3 | 16.760 | 38.590 | 54.904 | 0.50 | 41.65 | C |
| ATOM | 22 | NE | AARG | A | 3 | 17.200 | 37.726 | 56.028 | 0.50 | 36.32 | N |
| ATOM | 23 | NE | BARG | A | 3 | 16.971 | 39.913 | 55.495 | 0.50 | 43.59 | N |
| ATOM | 24 | CZ | AARG | A | 3 | 18.076 | 38.184 | 56.917 | 0.50 | 37.14 | C |
| ATOM | 25 | CZ | BARG | A | 3 | 17.056 | 41.047 | 54.806 | 0.50 | 45.42 | C |
| ATOM | 26 | NH1 | AARG | A | 3 | 18.458 | 37.415 | 57.927 | 0.50 | 39.10 | N |
| ATOM | 27 | NH1 | BARG | A | 3 | 17.227 | 42.196 | 55.439 | 0.50 | 45.37 | N |
| ATOM | 28 | NH2 | AARG | A | 3 | 18.561 | 39.414 | 56.802 | 0.50 | 35.79 | N |
| ATOM | 29 | NH2 | BARG | A | 3 | 16.988 | 41.032 | 53.487 | 0.50 | 47.91 | N |
| ATOM | 30 | N | LEU | A | 4 | 11.199 | 39.664 | 55.731 | 1.00 | 25.95 | N |

High B factors

**Atoms in loop 2**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ATOM | 210 | C | GLU | A | 29 | 17.341 | 22.439 | 38.728 | 1.00 | 21.43 | C |
| ATOM | 211 | O | GLU | A | 29 | 18.105 | 23.405 | 38.733 | 1.00 | 23.36 | O |
| ATOM | 212 | CB | GLU | A | 29 | 17.821 | 20.622 | 37.080 | 1.00 | 34.02 | C |
| ATOM | 213 | CG | GLU | A | 29 | 18.838 | 21.329 | 36.196 | 1.00 | 47.94 | C |
| ATOM | 214 | CD | GLU | A | 29 | 18.870 | 20.777 | 34.776 | 1.00 | 58.18 | C |
| ATOM | 215 | OE1 | GLU | A | 29 | 19.733 | 19.912 | 34.496 | 1.00 | 63.56 | O |
| ATOM | 216 | OE2 | GLU | A | 29 | 18.058 | 21.232 | 33.935 | 1.00 | 61.24 | O |
| ATOM | 217 | N | ALA | A | 30 | 16.025 | 22.552 | 38.879 | 1.00 | 18.10 | N |

High B factors

# Focus: banche dati biologiche

**Banche dati di strutture**

**Altre (per RNA):**
RNAcentral https://rnacentral.org/
PseudoBase++ https://rnavlab.utep.edu/database
NONCODE v5.noncode.org

# Focus: banche dati biologiche

## Banche dati di funzione

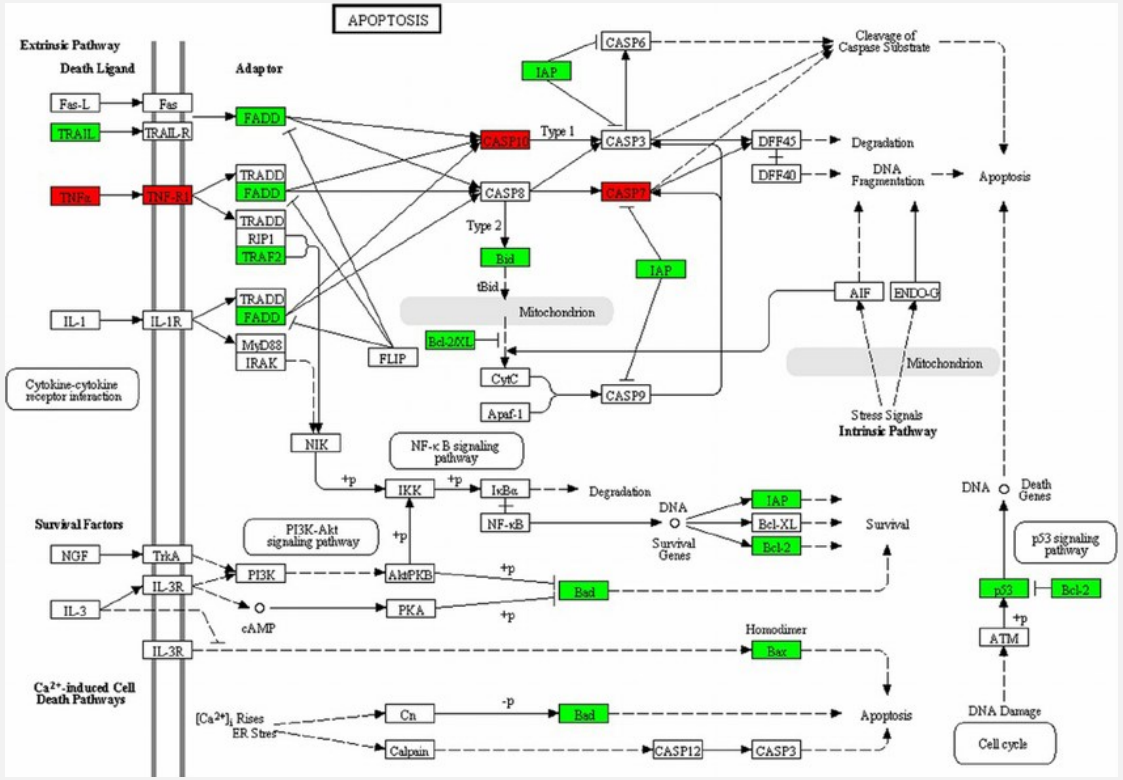
Queste banche dati sono utilizzate per annotare le sequenze e identificare le loro **funzioni biologiche**, come ad esempio la catalisi di una reazione chimica, l'interazione con altre proteine o la regolazione dell'espressione genica. Le informazioni contenute nelle banche dati di funzione sono spesso ottenute attraverso esperimenti di laboratorio o analisi bioinformatiche.

**UniProt** è un database di sequenze proteiche annotate, che fornisce informazioni dettagliate sulla funzione, la struttura e le interazioni di proteine provenienti da una vasta gamma di organismi. Le informazioni contenute in UniProt sono ottenute da una varietà di fonti, tra cui la letteratura scientifica, i depositi di sequenze, i laboratori di ricerca e la cura manuale. UniProt integra informazioni provenienti da altre banche dati, come Pfam, InterPro, GO e OMIM, per fornire una visione completa della funzione delle proteine

# Focus: banche dati biologiche

## Banche dati di funzione

**KEGG** (Kyoto Encyclopedia of Genes and Genomes) è un database di pathway metabolici e genetici, che fornisce informazioni sulle funzioni biologiche dei geni e delle proteine, nonché sui loro ruoli nei pathway metabolici e delle malattie. KEGG integra informazioni provenienti da diverse fonti, tra cui sequenze genomiche, pathway, funzioni biologiche, espressione genica, malattie e droghe, per fornire una visione integrata della biologia dei sistemi.




# Focus: banche dati biologiche

## Banche dati di espressione

**Gene Expression Omnibus (GEO)** è un database curato dal National Center for Biotechnology Information (NCBI) che archivia dati di espressione genica provenienti da diversi tipi di esperimenti, come sequenziamento dell'RNA. GEO è uno dei principali repository di dati di espressione genica e permette agli utenti di accedere e scaricare facilmente i dati.

**ArrayExpress** è un altro database di espressione genica curato dal European Bioinformatics Institute (EBI) che archivia dati di espressione genica provenienti da diversi organismi e tipi di esperimenti. ArrayExpress offre strumenti di analisi dei dati e visualizzazioni per aiutare gli utenti a comprendere e interpretare i dati di espressione genica.