MEMORIA

- Caratterizzazione
  - Capacità
  - Costo per bit
  - Velocità di accesso
    - Velocità di trasferimento (bit/s o Byte/s, dati trasferiti/secondo)
  - Volatilità

- Categorie
  - Memoria centrale
  - Memoria di massa

MEMORIA

- Tipo di accesso
  - Sequenziale (nastro)
  - Casuale (memoria centrale)
  - Misto (disco)

- RAM (*Random Access Memory*)
  - Memoria centrale, Volatile, Lettura/scrittura, Accesso casuale
- ROM (*Read Only Memory*)
  EPROM (*Erasable Programmable ROM*)
  - Permanente, Sola lettura (ROM) / Cancellabili e riscrivibili (EPROM)
  - BIOS

PRINCIPIO DI LOCALITA'

- Località spaziale:
  quando si accede all'indirizzo A, è molto probabile che gli accessi successivi richiedano celle vicine ad A.
  - le istruzioni del codice vengono in genere lette da locazioni consecutive della memoria
  - Gli accessi a strutture dati sono "vicini"

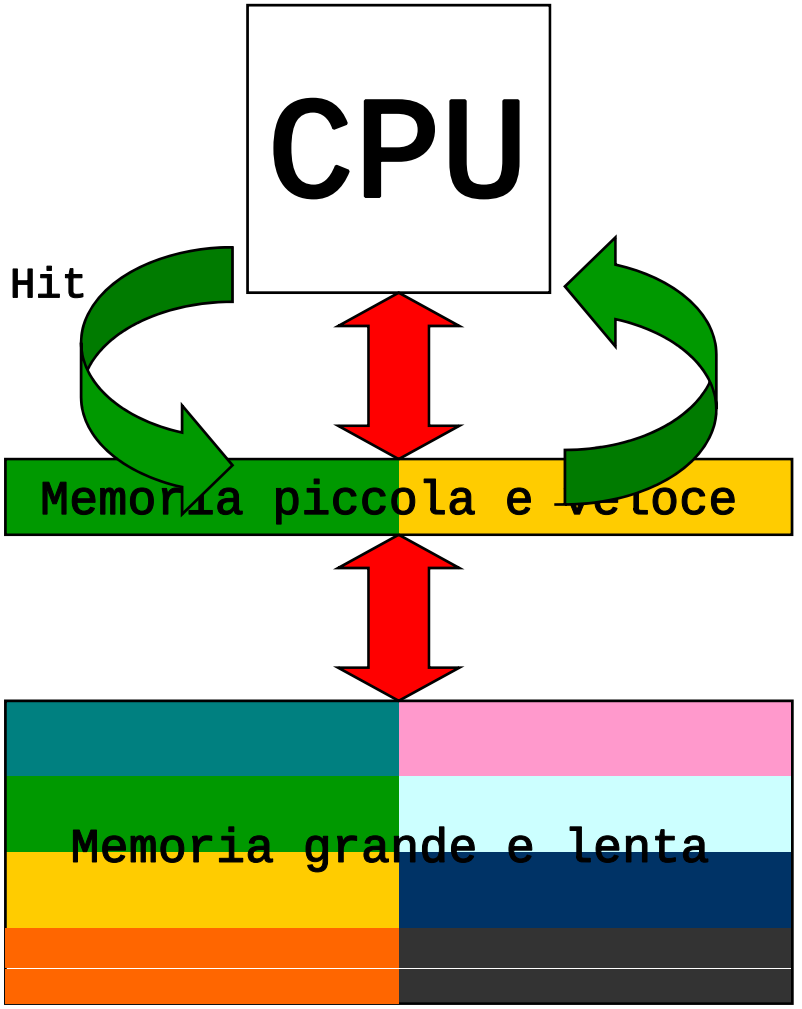
- Località temporale:
  quando si accede all'indirizzo A, è molto probabile che negli accessi successivi si richieda di nuovo la cella A.
  - cicli di istruzioni accedono ripetutamente alle stesse locazioni di memoria
  - istruzioni vicine tendono ad utilizzare le stesse variabili

GERARCHIA DI MEMORIA

- Obiettivi
  - Dimensione elevata
  - Velocità elevata
  - Costo limitato

- I dati prelevati dalla MGL vengono conservati il più a lungo possibile
- Quando si copia un dato dalla MGL si copiano anche i dati vicini

- *h*, hit ratio (frequenza di successo) pari al 99%.
  1 – h, miss ratio

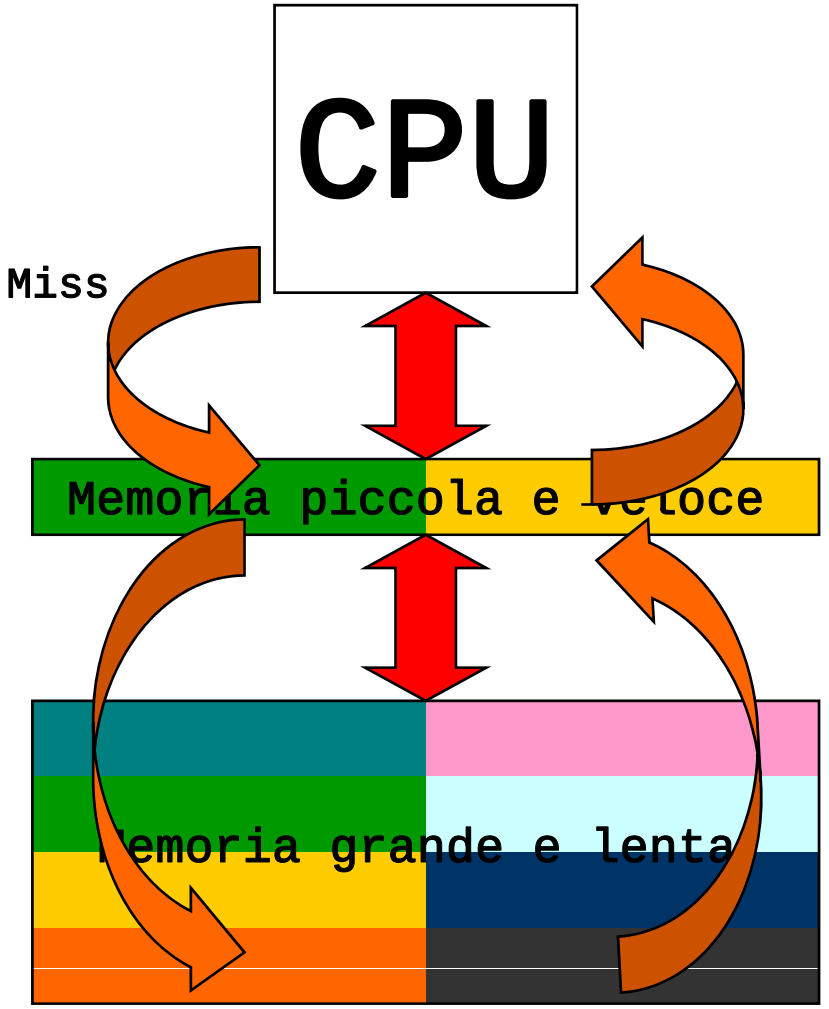


GERARCHIA DI MEMORIA

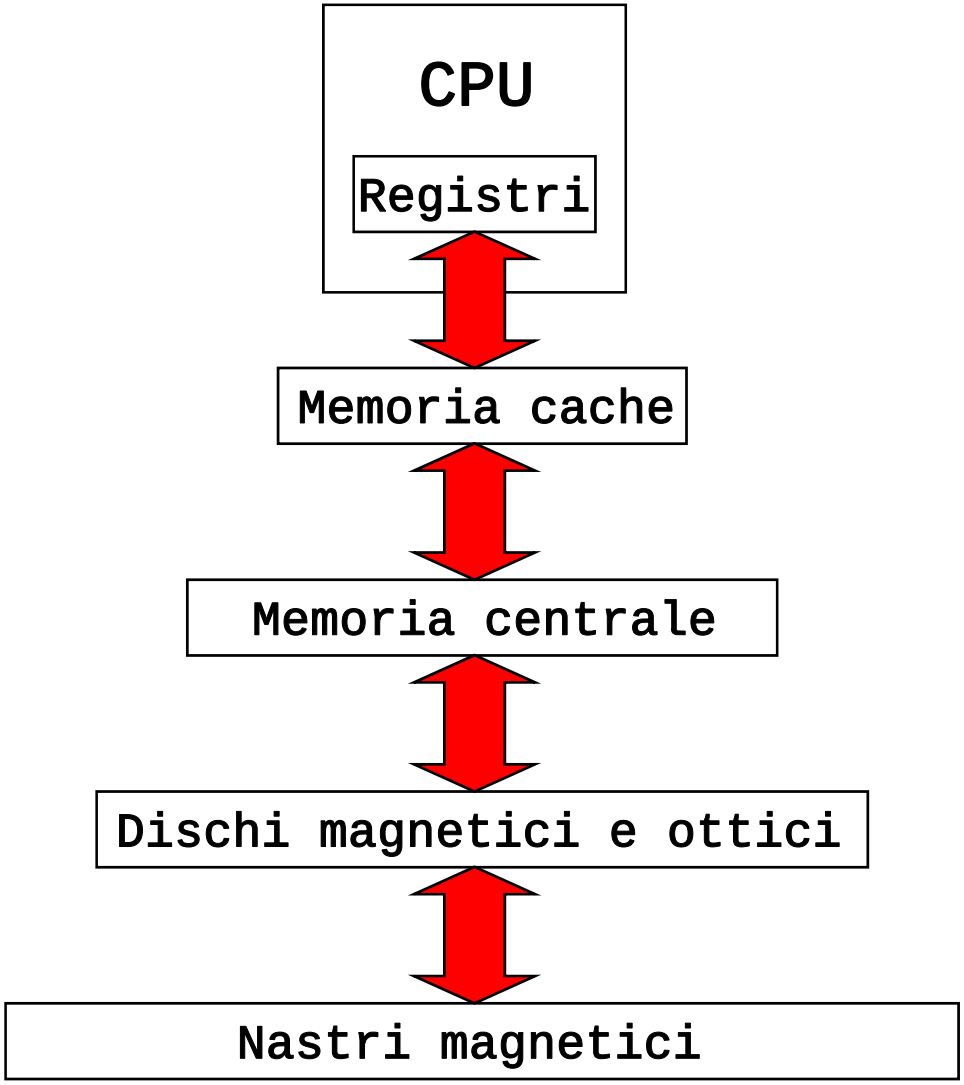
- Obiettivi
  - Dimensione elevata
  - Velocità elevata
  - Costo limitato

- I dati prelevati dalla MGL vengono conservati il più a lungo possibile
- Quando si copia un dato dalla MGL si copiano anche i dati vicini

- $h = 0.99$, $T_1 = 1$, $T_2 = 10$
  $T_{medio} = T_1\ h + T_2\ (1 - h) = 1.09!$



GERARCHIA DI MEMORIA

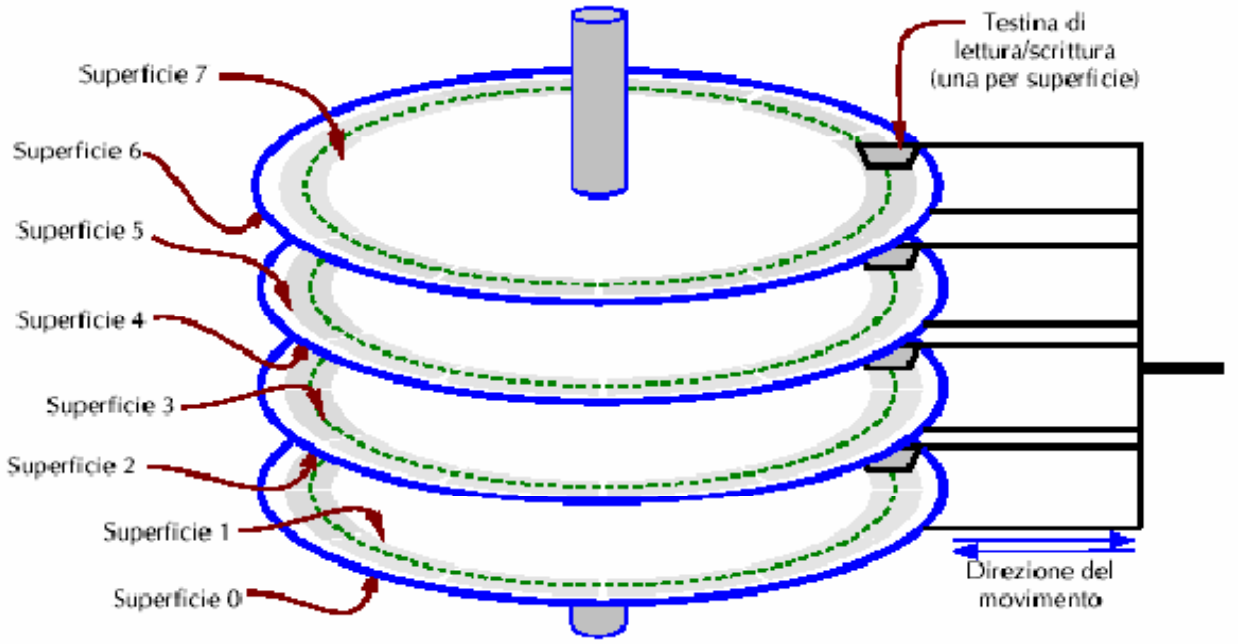


GERARCHIA DI MEMORIA

	Capacità	Velocità (TA)	€/MByte
registri	~1KB	~1ns	NA
cache	64 ÷ 1024 KB	~10ns	300
RAM	64 ÷ 2048 MB	~100ns	2
HD	8 ÷ 100 GB	~10ms	0.005
nastri/CD	~GB per unità	~100ms	0.005

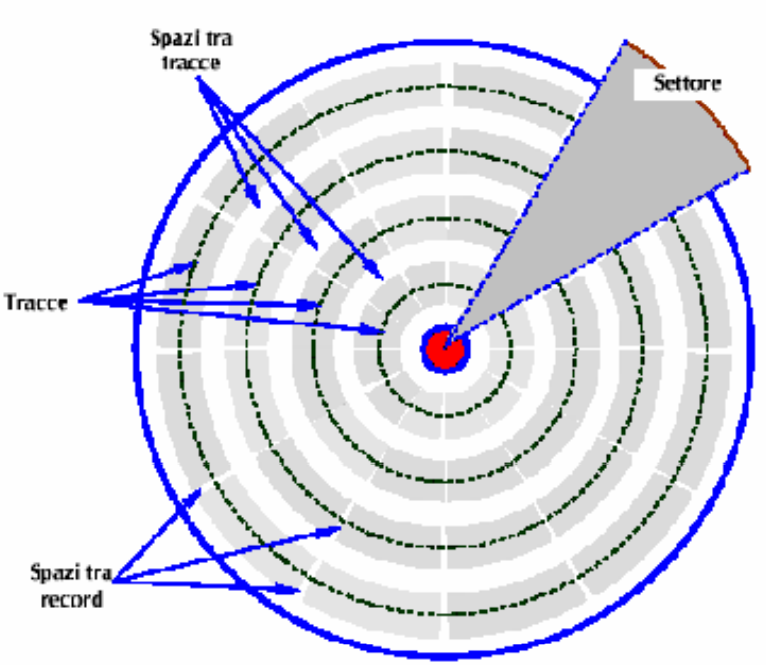
DISCHI MAGNETICI: HARD DISK

- Un hard disk consiste di un insieme di piatti con due superfici magnetizzabili
  - ogni superficie ha una propria testina di lettura/scrittura
  - i dischi ruotano attorno ad un perno centrale



DISCHI MAGNETICI: HARD DISK

- Organizzazione fisica dei record
  - le superfici sono organizzate in cerchi concentrici (tracce) e in spicchi di pari grandezza (settori) separati da parti vuote (gap)
  - tutte le tracce equidistanti dal centro (su più piatti) forma un cilindro
  - ogni traccia contiene lo stesso numero di bit (densità di memorizzazione variabile dalla periferia verso il centro)

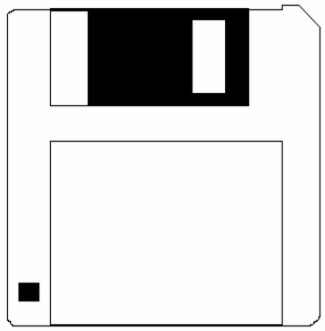


PRESTAZIONI DEGLI HARD DISK

- Tempo di accesso
  - *Seek Time*: la testina deve arrivare alla traccia giusta
    - dipende dalla meccanica
    - misurato in millisecondi (10-100ms)
  - *Latency Time*: il disco deve ruotare fino a portare il record nella posizione giusta
    - dipende dalla velocità di rotazione, misurata in giri/min (RPM)
- Transfer Rate
  - Velocità di trasferimento del disco
    - dipende dalla densità e dalla velocità di rotazione
    - misurata in MB per secondo (MBps)
    - valore tipico: 5-80 MBps

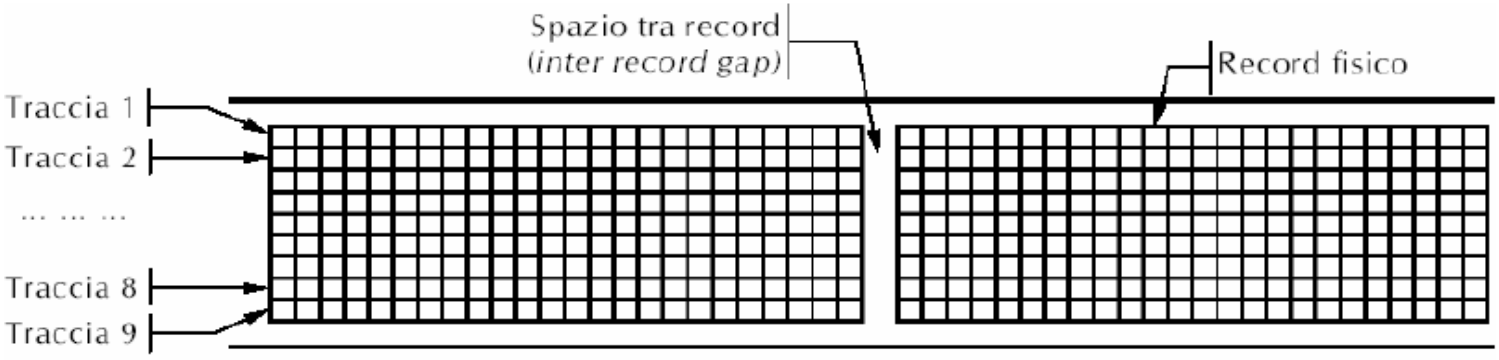
DISCHI MAGNETICI: FLOPPY DISK

- Sono dischi magnetici:
  - di piccola capacità
  - portatili
  - usati per trasferire informazioni (file) tra computer diversi
- Sono costituiti da un unico disco con due superfici
- Si ferma quando non è acceduto (ritardo per avvio della rotazione 0.5sec)
- Storicamente ne sono stati creati vari tipi identificati dal loro diametro (3.5, 5.25 e 8 pollici).
  - oggi sopravvivono solo da 3.5” (1.4 Mbyte)



NASTRI MAGNETICI

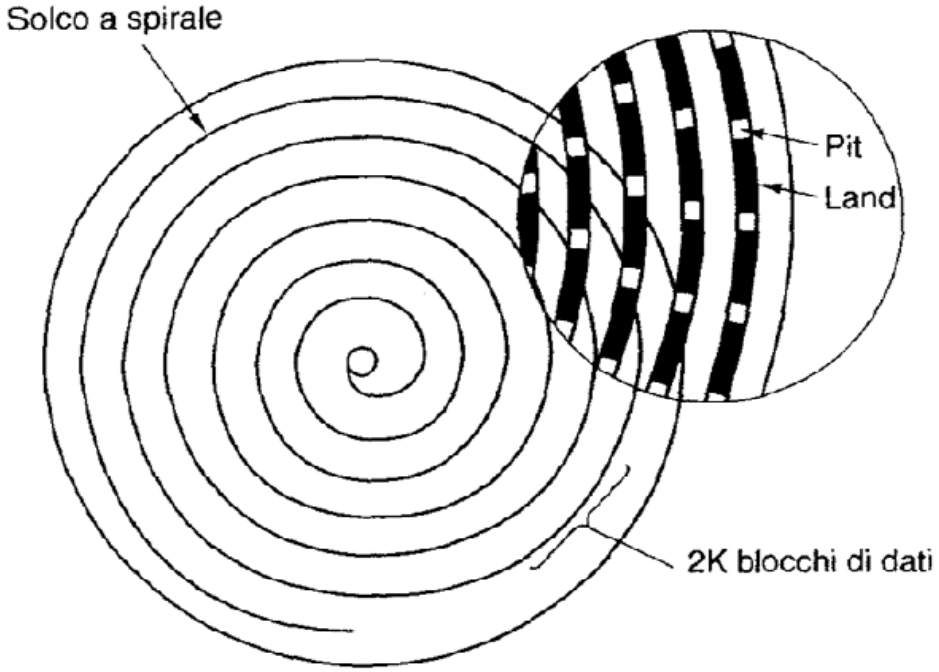
- Sono nastri di materiale magnetizzabile raccolti su supporti circolari, o in cassette (es.: DAT, *Digital Audio Tape*)
- Sul nastro sono tracciate piste orizzontali parallele (tracce)
  - di solito 9: un byte di dati + il bit di parità
- I dati sul nastro sono organizzati in zone contigue (record), separate da zone prive di informazione (inter record gap)
- Accesso sequenziale



DISCHI OTTICI

- La superficie di un disco presenta una successione di tratti disposti secondo un'unica traccia a spirale
  - pit: tratto di superficie avvallata
  - land: tratto di superficie liscia

  } **riflettono raggi luminosi in modo diverso**
- Il passaggio da pit a land (e viceversa) rappresenta 1 mentre l'assenza di variazione rappresenta 0



DISPOSITIVI OTTICI

- 1984, CD-ROM (Compact-Disc Read-Only Memory)
  - Disco 12cm di diametro di alluminio e materiale plastico
  - Sola lettura
  - Capacità di oltre 650 Mbyte e costo inferiore a €1
  - Velocità di trasferimento:
    - originariamente 150 KByte / secondo (“1X”)
    - OGGI: 12, 16, 24, 48 volte tanto…
  - CD-R (CD Recordable)
  - CD-RW (CD Rewritable)
- 1997, DVD (Digital Versatile Disc)
  - Evoluzione del CD-ROM
  - Capacità 4.7 - 17 GByte
  - Velocità di trasferimento paragonabile a quella dei CD-ROM

FINE

Introduzione all'Informatica

Francesco Folino

Sistema Operativo

FIRMWARE: IL BIOS

- BIOS = Basic Input-Output System
  - gestisce direttamente le risorse hardware e offre delle funzionalità standard di accesso
- risiede su un chip di memoria permenente
  - ROM, RAM + batteria di alimentazione
- gestisce la procedura di di avviamento del calcolatore, consistente delle seguenti fasi
  - diagnostica
  - inizializzazione delle risorse hardware (setup)
  - caricamento (da disco rigido in RAM) ed esecuzione della routine di bootstrap, che provvede quindi a caricare il sistema operativo

SISTEMA OPERATIVO (S.O.)

- Strato di programmi che opera al di sopra di hardware e firmware e gestisce l'elaboratore
- Come Gestore delle Risorse
  - controlla e gestisce tutte le funzioni del calcolatore in modo efficiente
  - accetta e soddisfa le richieste dell'utente
  - funziona come mediatore tra risorse in conflitto
  - tiene traccia di chi utilizza le risorse
- Come Macchina estesa
  - costituisce una base sulla quale è possibile scrivere programmi applicativi
  - rappresenta all'utente una macchina estesa più facile da programmare

FUNZIONI DI UN S.O.

- Esecuzione di applicazioni
  - Caricamento dei programmi (istruzioni e dati) nella memoria centrale
- Accesso ai dispositivi di I/O
  - Gestione dei segnali per il trasferimento dei dati
  - Operazioni astratte di lettura/scrittura
- Archiviazione di dati e di programmi
  - Organizzazione logica dei dati (directory, file)
- Controllo di accesso
  - Condivisione di risorse da più utenti o applicazioni
  - Meccanismi di protezione
- Contabilizzazione
  - Monitoraggio dell'uso delle risorse da parte di utenti e/o applicazioni (ottimizzazione/fatturazione)
- Gestione dei malfunzionamenti
  - rilevare e risolvere guasti hardware e operazioni scorrette del software

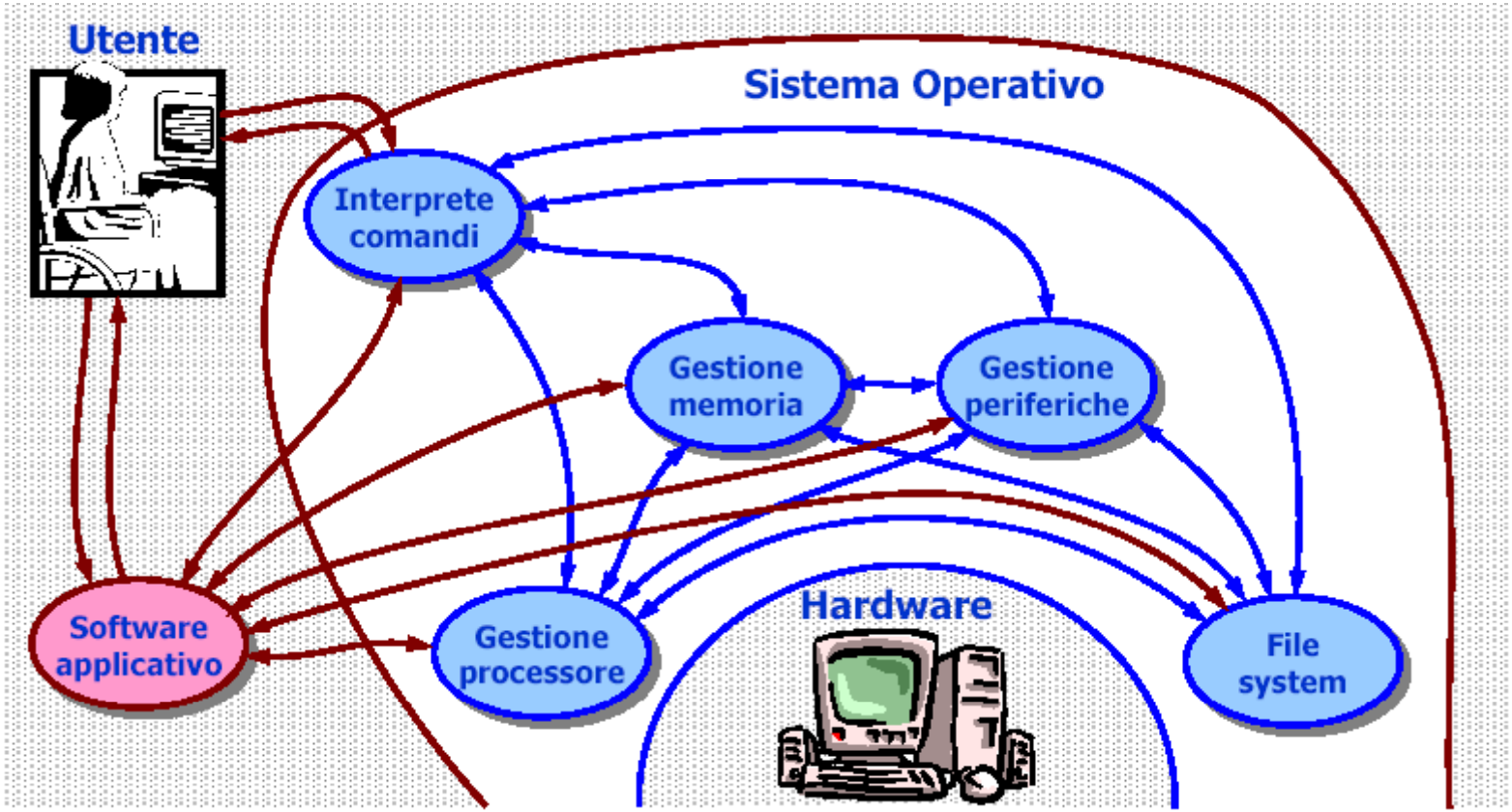
PROGRAMMI APPLICATIVI VS S.O.

- Programmi applicativi
  - hanno accesso a un insieme ridotto di risorse;
  - possono utilizzare solo un sottoinsieme delle istruzioni del processore (esecuzione in **modalità utente**);
  - non possono decidere autonomamente quando e come avere accesso alle risorse del sistema (richiedono al sistema operativo l'esecuzione di alcuni servizi);
- Sistema operativo
  - ha accesso a tutte le risorse;
  - può utilizzare tutte le istruzioni del processore (esecuzione in **modalità supervisore**);
  - stabilisce in che ordine e come le richieste che riceve devono essere soddisfatte;
  - ...

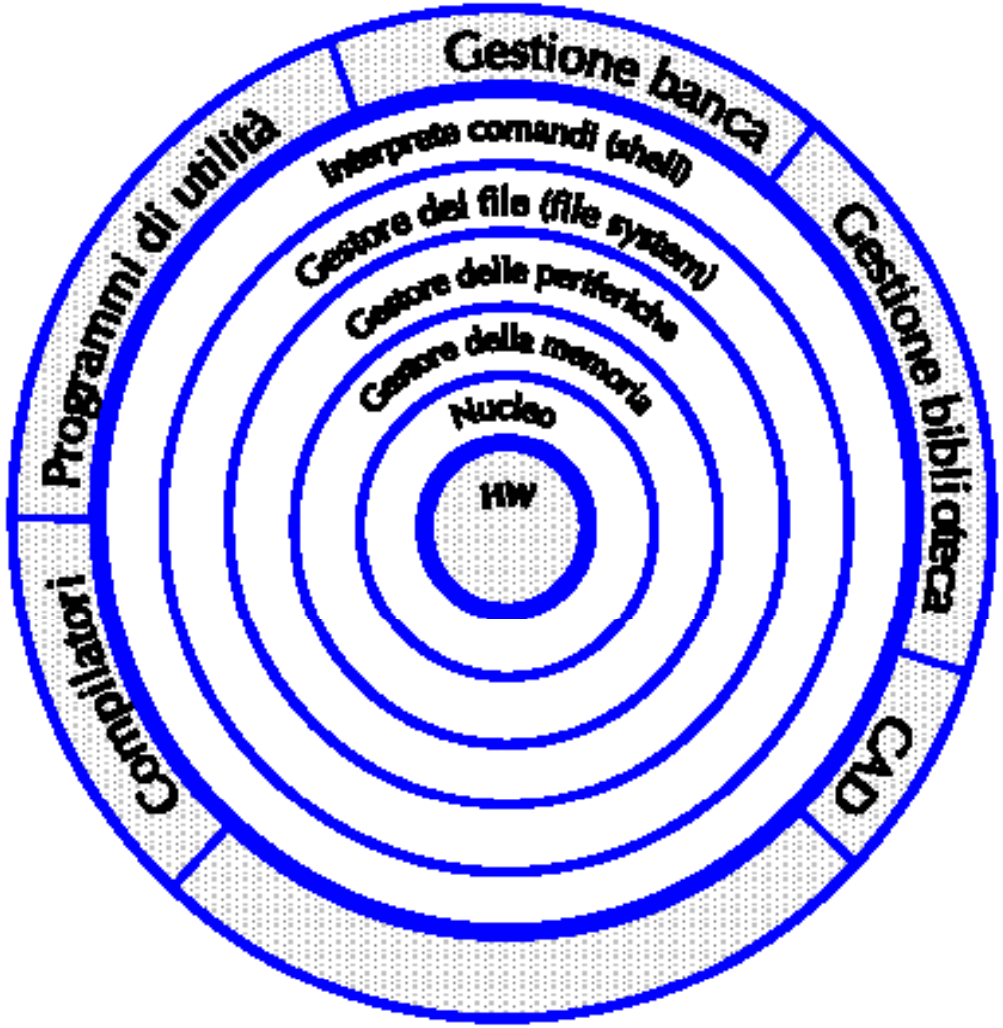
ELEMENTI DI UN S.O.

- Sistema di **gestione del processore**,
  - controlla l'unità centrale di elaborazione (CPU);
  - definisce quali programmi sono da eseguire e quali compiti sono da assegnare alla CPU;
- Sistema di **gestione della memoria**,
  - controlla l'allocazione della memoria di lavoro ai diversi programmi che possono essere contemporaneamente in esecuzione;
- Sistema di **gestione delle periferiche**,
  - garantisce l'accesso ai dispositivi di ingresso/uscita,
  - maschera i dettagli di basso livello e gli eventuali conflitti che possono insorgere nel caso che diverse richieste arrivino contemporaneamente a uno stesso dispositivo;
- Sistema di **gestione dei file** (**file system**)
  - consente l'archiviazione e il reperimento dei dati sfruttando le periferiche che costituiscono la memoria di massa;
- Sistema di **gestione degli utenti e dei relativi comandi** (**interprete comandi**),
  - interfaccia diretta con gli utenti,
  - permette agli utenti di accedere in maniera semplice e intuitiva alle funzionalità disponibili.

ELEMENTI DI UN S.O.



MODELLO A STRATI DI UN S.O.



CLASSIFICAZIONE DEI S.O.

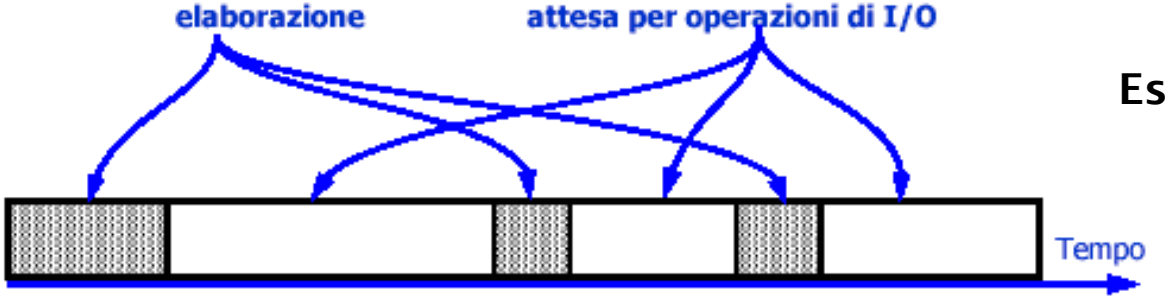
- In base al numero di utenti:
  - mono-utente (mono-user)
    - un solo utente alla volta può utilizzare il sistema
  - multi-utente (multi-user)
    - più utenti in contemporanea interagiscono con la macchina
    - il S.O. fornisce a ciascuno l'astrazione di un sistema "dedicato"
- In base al numero di processi:
  - Mono-programmato (mono-task)
    - si può eseguire un solo programma per volta
  - Multi-programmato (multi-task)
    - il SO permette di eseguire più programmi in contemporanea
    - il SO gestisce la suddivisione del tempo della CPU fra i vari processi (*time-sharing*)

LIMITI DEI S.O. MONOPROGRAMMATI

- Qualunque programma alterna fasi di esecuzione a fasi in cui è bloccato in attesa di qualche evento esterno
  - attesa che sia terminata un'operazione di input
  - attesa per usare una risorsa al momento occupata
- Sotto-utilizzo del processore
  - mentre il programma è bloccato in attesa di eventi esterni, il processore rimane inattivo (*idle*)
  - i tempi di lavoro delle periferiche di input/output, o addirittura i tempi di reazione umani sono maggiori di molti ordini di grandezza della velocità del processore

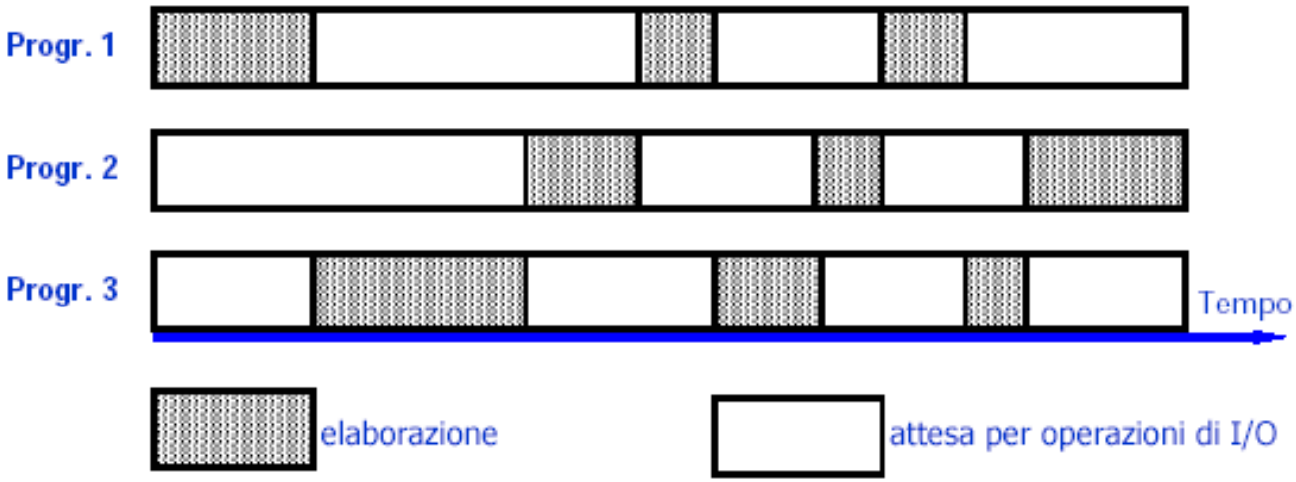
MULTIPROGRAMMAZIONE

Sistema monoprogrammato: mono-tasking



Es. Programma videoscrittura
Utilizzo CPU 2-5%

Sistema multiprogrammato: multi-tasking e time-sharing



MULTIPROGRAMMAZIONE

- Il tempo di lavoro della CPU è diviso tra i vari programmi
  - Ad ogni istante vi è un solo programma attivo
  - Il processore alterna l'esecuzione dei vari programmi
- Se l'alternanza tra i programmi è frequente (es.10/100 ms), si ha l'impressione di un'esecuzione simultanea
  - a livello macroscopico si ha quindi l'impressione della contemporaneità, mentre a livello microscopico si ha una semplice alternanza sequenziale molto veloce
- Il tempo totale di esecuzione di un singolo programma aumenta rispetto al caso mono-tasking
  - a causa dell'alternanza con gli altri programmi

MULTIPROGRAMMAZIONE

- Nel sistema sono presenti diversi programmi, ognuno con un proprio tempo di elaborazione e propri tempi di attesa per le operazioni di ingresso/uscita.
- Per evitare che la CPU venga utilizzata in modo esclusivo (o per troppo tempo) da parte di un solo programma, il tempo viene idealmente suddiviso in unità elementari, dette **quanti**, da assegnare secondo opportune politiche a tutti i programmi.
- **Round-robin**: assegnare a rotazione la disponibilità di un quanto di tempo della CPU ai vari programmi presenti contemporaneamente in memoria.
- La durata del quanto di tempo incide significativamente sia sulle prestazioni del sistema che sull'efficacia del quasi parallelismo, che tende a scomparire se la durata diviene eccessiva e degrada nella sequenzializzazione dei programmi. D'altra parte, pur migliorando in generale le proprietà di parallelismo la scelta di un valore molto piccolo può comportare un degrado delle prestazioni complessive del sistema, qualora il tempo di commutazione fra programmi sia dello stesso ordine della durata del quanto di tempo (un valore tipico per il sistema operativo Unix è 100 ms).

MULTIPROGRAMMAZIONE

- **Programma**:
  entità statica composta dal codice eseguibile dal processore.
- **Processo**:
  entità dinamica che corrisponde al programma in esecuzione, composto da:
  - codice (il programma);
  - dati (quelli che servono per l'esecuzione del programma);
  - stato (a che punto dell'esecuzione ci si trova, cosa c'è nei registri, …).

CONTEXT SWAPPING

- Il processo non si rende conto delle interruzioni:
  - il nucleo maschera al processo come effettivamente la sua elaborazione evolve nel tempo;
  - il nucleo rende trasparente la presenza delle operazioni di interruzione e di riassegnamento del processore a un processo.
- Contesto di un processo
  - insieme dei dati che rappresentano lo "stato" del processo: situazione della memoria, contenuto dei registri, livello di priorità, ...
  - quando un processo viene interrotto (esce dallo stato di esecuzione) il nucleo provvede a **salvare** del suo contesto (in una struttura dati chiamata **descrittore del processo**);
  - quando un processo torna nello stato di esecuzione il nucleo provvede a **ripristinare** il suo contesto (recuperando i dati precedentemente salvati nel descrittore).
- Cambio di contesto (context swapping)
  - si verifica quando un processo (e.g. P1) in esecuzione viene sostituito da un altro processo P2
  - il nucleo provvede a
    1. salvare il contesto di P1 e gestirne l'evoluzione (pronto vs attesa);
    2. ripristinare il contesto di P2 per consentirgli una corretta evoluzione.

STATI DI UN PROCESSO