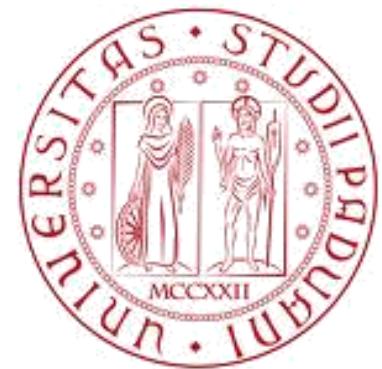


# Nozioni introduttive sui sistemi operativi

Prof. Luca Trevisan

Dipartimento di Ingegneria  
dell'Informazione

Università degli Studi di Padova  
A.A. 2025/2026





# Nozioni introduttive sui sistemi operativi

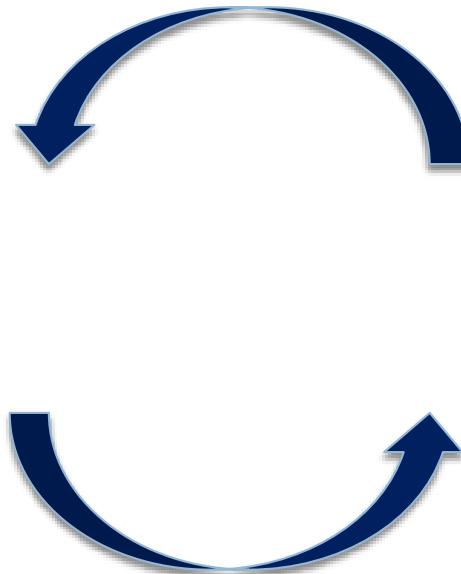
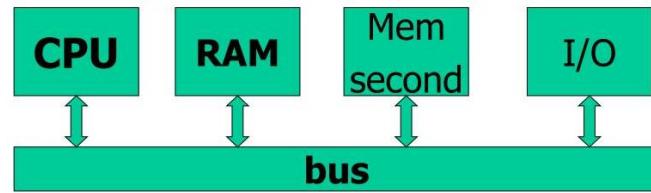


# Cos'è un sistema operativo?

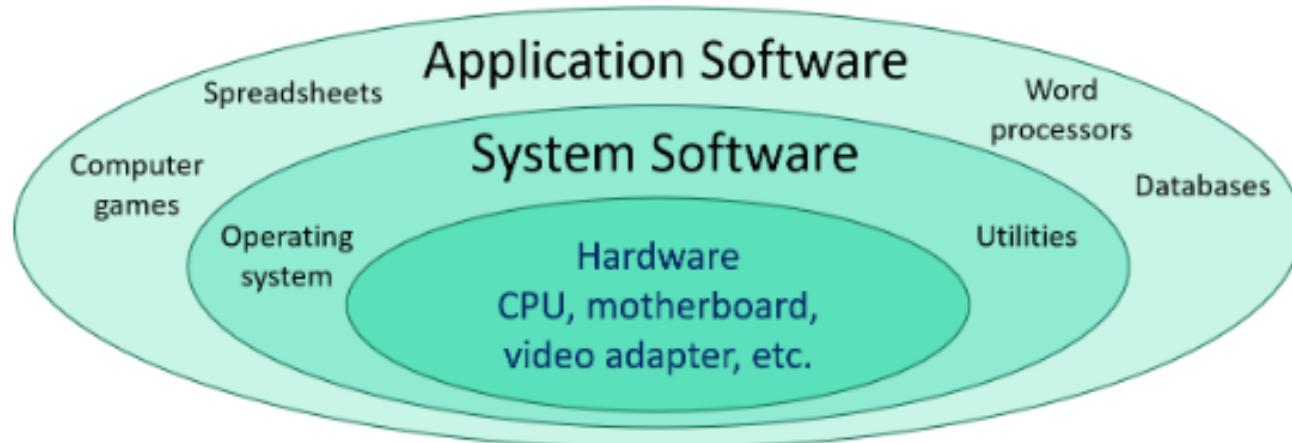
- Un sistema operativo è una **collezione di programmi** che consente di eseguire le operazioni fondamentali del computer, inizializza il sistema e assicura il corretto funzionamento della macchina.
- Quindi, per poter eseguire qualsiasi altro software, nella memoria del computer deve essere **prima** caricato il **sistema operativo**.
- I sistemi operativi attualmente più usati al mondo sono **Microsoft Windows, Linux, MAC OS e Android**.
- Un sistema operativo è un insieme di **software** che fornisce all'utente una serie di comandi e servizi per usufruire al meglio della potenza di calcolo di un qualsivoglia **elaboratore elettronico**, spaziando dal più piccolo dei **palmari** al più potente tra i **mainframe**.

# A cosa serve un sistema operativo?

L'architettura di Von Neumann



# A cosa serve un sistema operativo?



- I sistemi operativi **nascondono** tutti i dettagli tecnici legati alla specifica **architettura hardware**, rappresentando le informazioni ad un alto livello, meglio comprensibile dall'uomo.
- Essi garantiscono l'operatività di base di un **elaboratore**, coordinando e gestendo le **risorse** hardware di **elaborazione** e **memorizzazione**, le **periferiche**, le risorse/attività **software** e facendo da **interfaccia** con l'utente, senza il quale quindi non sarebbe possibile l'utilizzo del computer stesso e dei **programmi**.

# A cosa serve un sistema operativo?



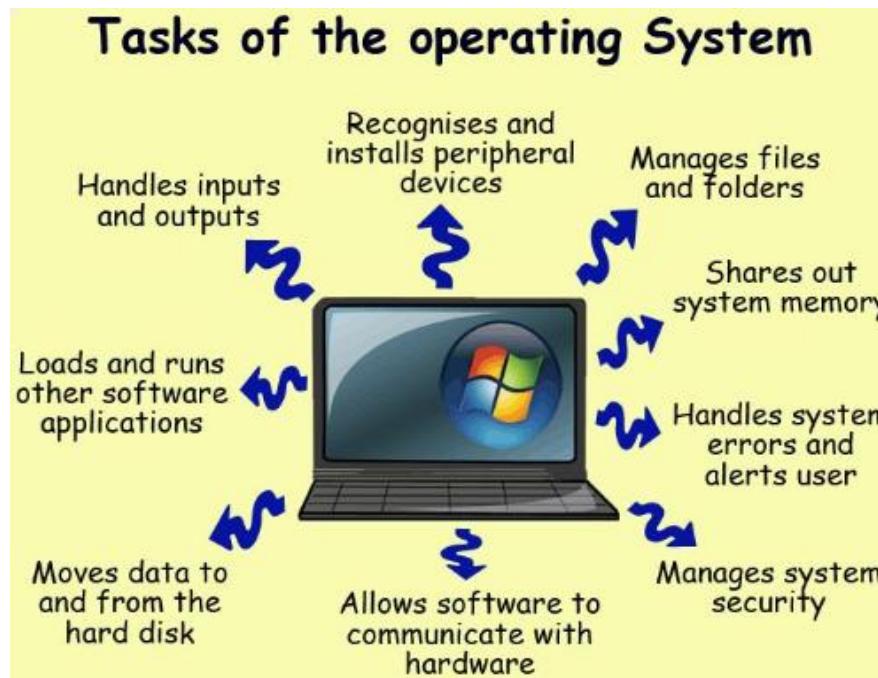
- È dunque un componente essenziale del sistema di elaborazione che **funge da interfaccia tra l'utente e la macchina** ed inoltre è una base alla quale si appoggiano gli altri software, che dunque dovranno essere **progettati e realizzati** in modo da essere riconosciuti e supportati da quel particolare sistema operativo.
- Assieme al **processore**, con cui spesso è strettamente legato, costituisce la cosiddetta **piattaforma** del sistema di elaborazione.

# L'avvio (bootstrap)



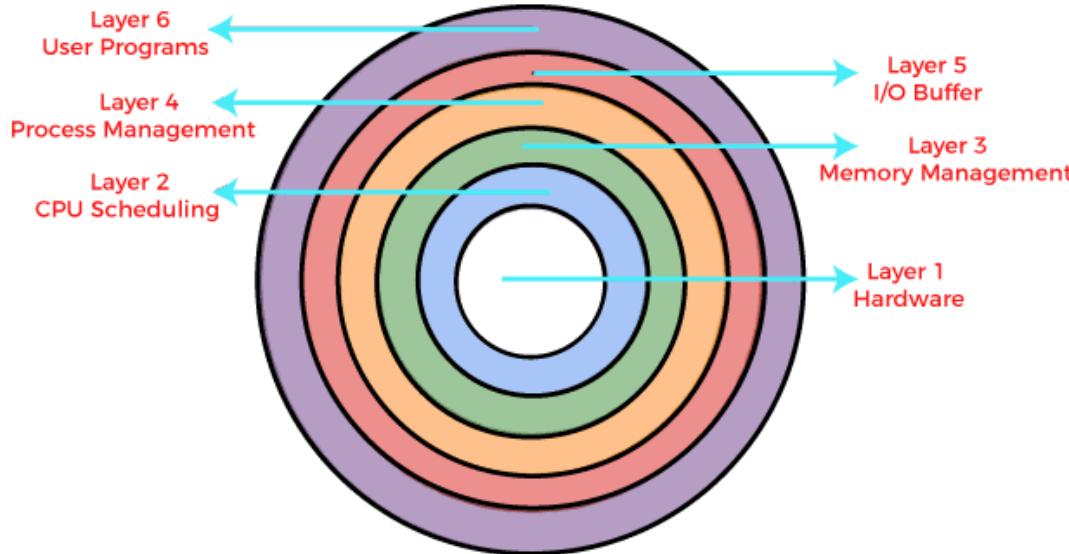
- La procedura di **avvio (bootstrap)** è memorizzata nella memoria **ROM** del computer (**ROM** = **Read Only Memory**).
- Quando si accende un computer, la sua **memoria di lavoro** o **RAM** (**RAM** = **Random Access Memory**) è completamente vuota.
- La **CPU** (**CPU** = **Central Processing Unit**) legge ed esegue l'elenco di istruzioni di bootstrap nella ROM per **inizializzare** la macchina.

# L'avvio (bootstrap)



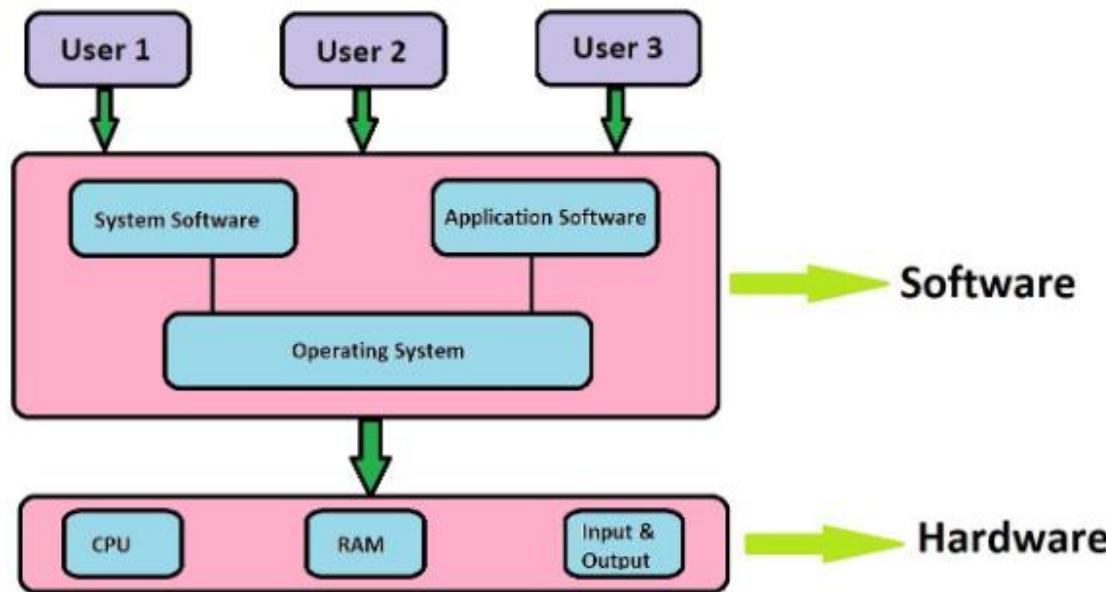
- Poi cerca e carica il sistema operativo dall'hard disk del computer.
- Il sistema operativo viene **caricato** nella **RAM** ed **eseguito**.
- Una volta in esecuzione si può accedere al pannello di controllo del sistema operativo (ad es. il desktop di Windows) e avviare gli altri programmi che ci interessano (ad es. browser, foglio di calcolo, programma di videoscrittura, ecc.).

# I livelli del sistema operativo



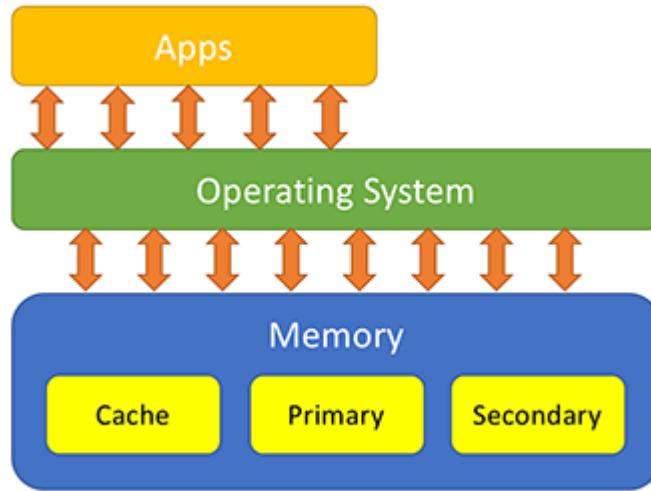
- I **programmi** del sistema operativo sono classificati in base alla **vicinanza ai componenti hardware** del sistema. Esistono diversi **livelli**:
- **Hardware Abstraction Layer (HAL)**: interagisce direttamente con l'**hardware**, fornendo un'**astrazione** per i driver di dispositivo. Permette al **kernel** di comunicare con diversi componenti hardware (come CPU, memoria e dispositivi I/O) **senza** la necessità di conoscere **dettagli** specifici su di essi.

# I livelli del sistema operativo



- **Driver di Dispositivo e Gestione I/O:** Questi moduli gestiscono specifici **dispositivi hardware** (ad es. unità disco, interfacce di rete, dispositivi USB). Si occupano della comunicazione tra il kernel e l'hardware, fornendo un'interfaccia standard da utilizzare.
- **Gestione dei Processi:** è responsabile della gestione dei processi, compresa la loro **creazione, pianificazione e terminazione**. Gestisce il multitasking, dando **priorità** ai processi e gestendo il **tempo** della **CPU**.

# I livelli del sistema operativo



- **Gestione della Memoria:** gestisce la **memoria di sistema**, compresa la memoria fisica e virtuale. Si occupa dell'**allocazione** e **deallocazione** della memoria, del paging e dello swapping per garantire un **uso efficiente** delle risorse di memoria.
- **Gestione del File System:** fornisce la funzionalità per **leggere e scrivere file**. Gestisce i file system, assicurando che i file siano memorizzati, accessibili e organizzati correttamente.

# I livelli del sistema operativo



- **Networking:** gestisce tutti gli aspetti della **rete**, inclusa l'implementazione dei **protocolli**, le **interfacce di rete** e la trasmissione dei dati. Supporta vari protocolli come **TCP/IP**, consentendo la comunicazione attraverso le reti.
- **Comunicazione Inter-Processo (IPC):** fornisce meccanismi per la comunicazione tra i processi, come pipe, code di messaggi, semafori e memoria condivisa. L'IPC è cruciale per **coordinare i compiti** tra i **processi**.

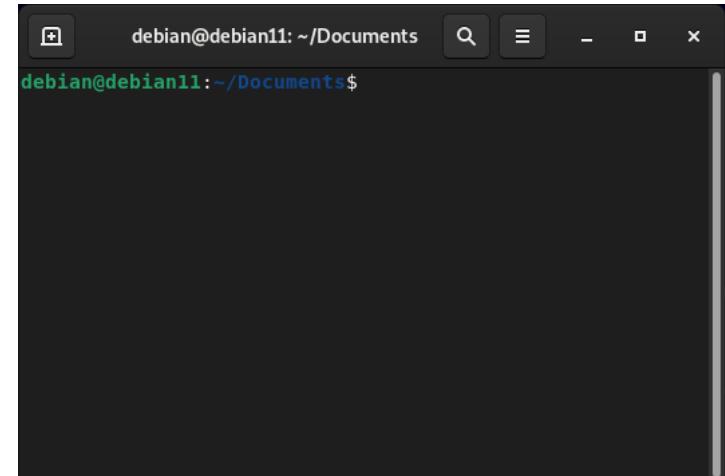
# I livelli del sistema operativo



- **Interfaccia delle Chiamate di Sistema:** funge da ponte tra le **applicazioni utente** e il **kernel**. Fornisce le chiamate di sistema necessarie che le applicazioni utilizzano per **richiedere servizi** dal kernel, come operazioni su file, gestione dei processi e allocazione della memoria.
- **L'interfaccia utente:** costituita dai programmi che comunicano direttamente con l'utente. Ad esempio, la **shell** del “vecchio” MS-DOS o di UNIX (Linux), oppure l'**interfaccia grafica** (GUI) di Windows e Android. Costituisce lo strato più “alto” del sistema operativo, quello più vicino all'uomo.

# I linguaggi di controllo del S.O.

- Ogni sistema operativo ha un proprio **linguaggio di controllo**.
- Il linguaggio di controllo è composto da un elenco di comandi utili per eseguire o controllare le operazioni e le attività del sistema.
- **Linguaggio formale**: ogni comando ha una propria sintassi e una “grammatica” molto rigida.
- I comandi sono impartiti dall’utente tramite la **riga di comando**.
  
- **Osservazione**: nei sistemi operativi con interfaccia grafica i comandi sono richiamati dagli utenti tramite le icone del desktop o dei menu.



# Take Home Message

