

---

ANNO ACCADEMICO 2024/2025

---

# Sistemi Operativi

---

Teoria

Dione's Notes



---

DIPARTIMENTO DI INFORMATICA

---

1.1	Prima Lezione	5
	Architetture Single/Multi-Core — 5 • Gestione delle Interruzioni — 6 • Memoria e Cache — 6	
1.2	Multitasking e Time-sharing	6
1.3	Protezione della Memoria	6

---



# Premessa

## Licenza

Questi appunti sono rilasciati sotto licenza Creative Commons Attribuzione 4.0 Internazionale (per maggiori informazioni consultare il link: <https://creativecommons.org/version4/>).



## Formato utilizzato

Box di "Concetto sbagliato":

### Concetto sbagliato 0.1: Testo del concetto sbagliato

Testo contenente il concetto giusto.

Box di "Corollario":

### Corollario 0.0.1 Nome del corollario

Testo del corollario. Per corollario si intende una definizione minore, legata a un'altra definizione.

Box di "Definizione":

### Definizione 0.0.1: Nome delle definizioni

Testo della definizione.

Box di "Domanda":

### Domanda 0.1

Testo della domanda. Le domande sono spesso utilizzate per far riflettere sulle definizioni o sui concetti.

Box di "Esempio":

### Esempio 0.0.1 (Nome dell'esempio)

Testo dell'esempio. Gli esempi sono tratti dalle slides del corso.

**Box di "Note":**

**Note:-**

Testo della nota. Le note sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive.

**Box di "Osservazioni":**

**Osservazioni 0.0.1**

Testo delle osservazioni. Le osservazioni sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive. A differenza delle note le osservazioni sono più specifiche.



# 1

## Introduzione

### 1.1 Prima Lezione

Un Sistema Operativo (SO) agisce come intermediario tra l'utente e l'hardware, fornendo gli strumenti per un uso corretto delle risorse della macchina (CPU, memoria, periferiche). Ha due obiettivi principali:

- Dal punto di vista dell'utente: rendere il sistema facile da usare.
- Dal punto di vista della macchina: ottimizzare l'uso delle risorse in modo sicuro ed efficiente.

#### 1.1.1 Architetture Single/Multi-Core

Negli anni 2000 si è passati da processori single-core a multi-core, con CPU dotate di più core in grado di eseguire istruzioni di programmi diversi simultaneamente. La figura 1.1 illustra una tipica architettura dual-core.

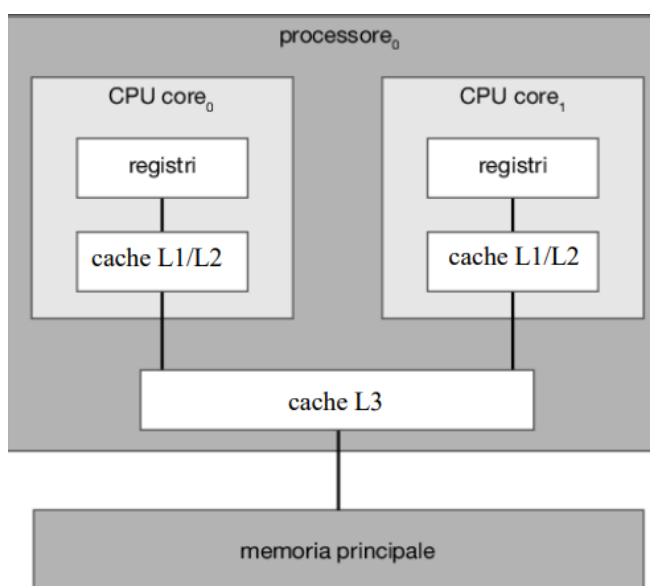


Figure 1.1: Architettura di un processore dual-core



### 1.1.2 Gestione delle Interruzioni

Il SO è un sistema “event-driven”, ossia viene attivato quando si verificano eventi come *interrupt* (hardware) o *eccezioni* (software). Ogni interruzione attiva una porzione specifica di codice del SO. La figura ?? rappresenta un esempio di gestione delle interruzioni.

### 1.1.3 Memoria e Cache

La gerarchia delle memorie include registri, cache, RAM e memoria secondaria. I dati più frequentemente utilizzati vengono spostati in memorie più veloci, come la cache. In figura 1.2 si può osservare la gerarchia delle memorie.

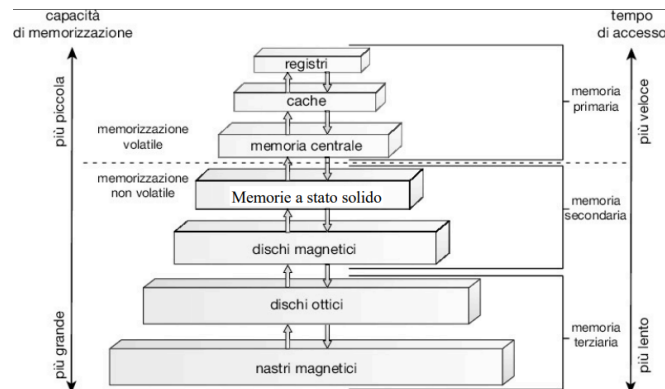


Figure 1.2: Gerarchia delle memorie

## 1.2 Multitasking e Time-sharing

Il SO gestisce più programmi contemporaneamente, assegnando la CPU a ciascuno quando disponibile. Il time-sharing permette di distribuire il tempo della CPU tra più utenti, dando l'impressione di simultaneità.

## 1.3 Protezione della Memoria

Il SO protegge la memoria primaria da accessi non autorizzati attraverso il meccanismo di registri base e limite, come mostrato in figura ??.

