

# Processi

---

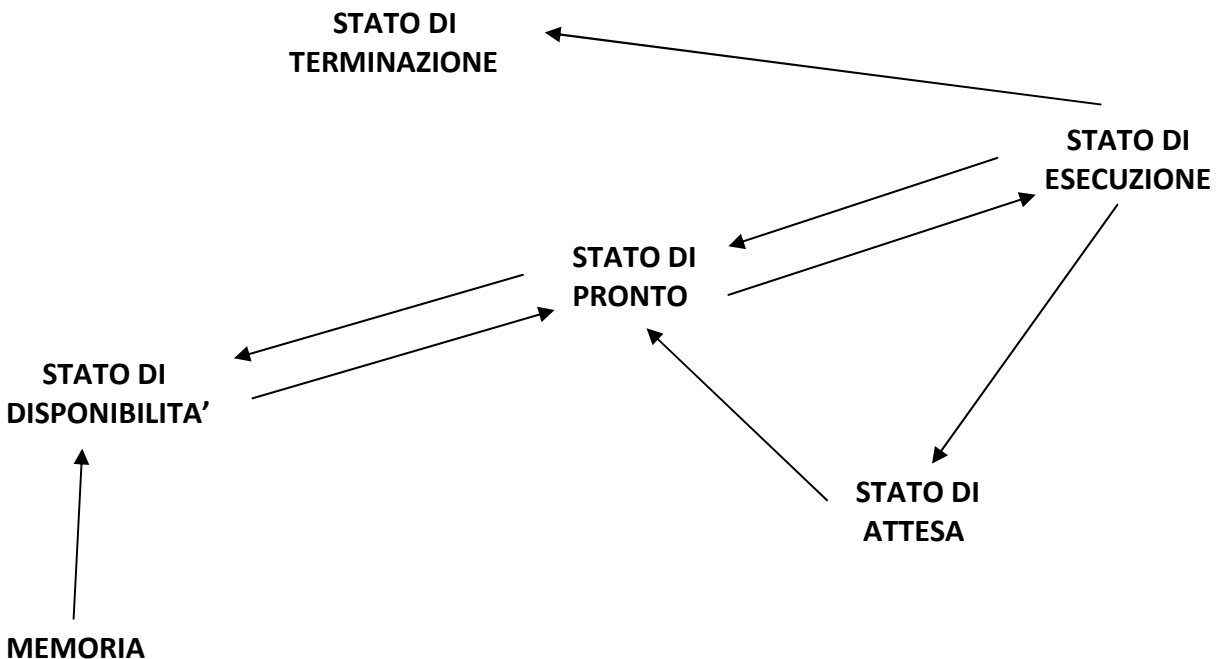
Un **Sistema Operativo (SO)** è un insieme di programmi che gestiscono le risorse sia hardware che software del sistema di elaborazione rendendone più semplice l'uso (supporto per l'utente) ed ottimizzano l'utilizzazione delle stesse (**gestore di risorse**).

Esaminiamo alcune caratteristiche di un generico SO in multiprogrammazione e multiutente. In questo contesto si hanno più programmi che sembra vengano eseguiti "in parallelo", contemporaneamente, anche se in effetti l'unica CPU li avanza uno alla volta, "a turno".

L'idea di base della multiprogrammazione è quella di sfruttare i "tempi morti" delle lentissime operazioni di Input/Output non costringendo la CPU ad attendere il loro completamento, ma impiegandola nell'avanzamento di altri processi. Perché sia possibile la multiprogrammazione occorre quindi che la CPU sia svincolata dall'esecuzione delle operazioni di I/O, delle quali si occuperanno appositi processori di I/O (canali, DMA) che lavorano in parallelismo effettivo con la CPU. La CPU (processore centrale) può così occuparsi esclusivamente dell'**avanzamento dei processi**.

**Un processo è l'insieme delle azioni compiute dal processore per eseguire un programma. Ogni processo, per l'avanzamento necessita di risorse HW e SW. È compito del SO, oltre all'avanzamento dei processi, la gestione di tali risorse.**

## Stati di un processo

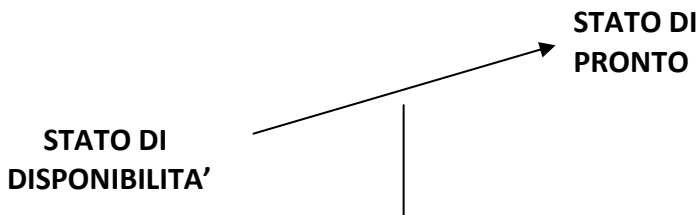


STATO DI  
DISPONIBILITA'

**DESCRITTORI DEI LAVORI (organizzati in lista):**

- nome del lavoro, data e ora di accettazione, priorità, tempo CPU
- risorse statiche
- risorse dinamiche
- indicatore stato di esecuzione

MEMORIA

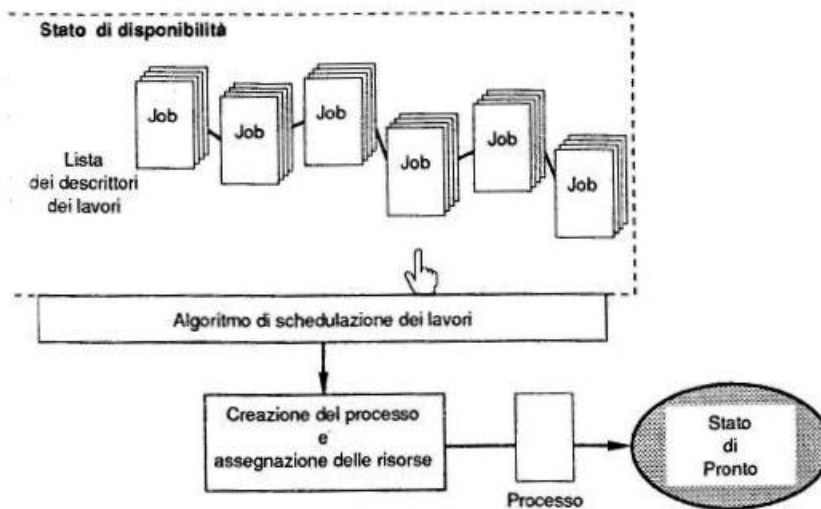


**Schedulatore dei lavori** che sceglie quale lavoro portare in memoria centrale e genera il **Descrittore dei processi** (gestore della memoria e gestore delle periferiche)

**SCHEDULATORE DEI LAVORI (alto livello)**

Funzioni:

- aggiornare le informazioni
- selezionare quali lavori devono acquisire risorse per avanzare
- generare processi assegnando le opportune risorse

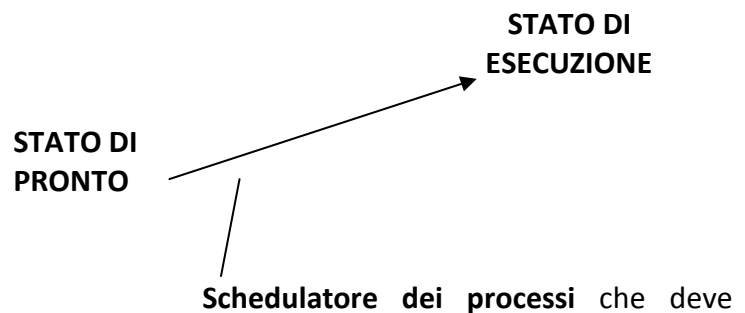
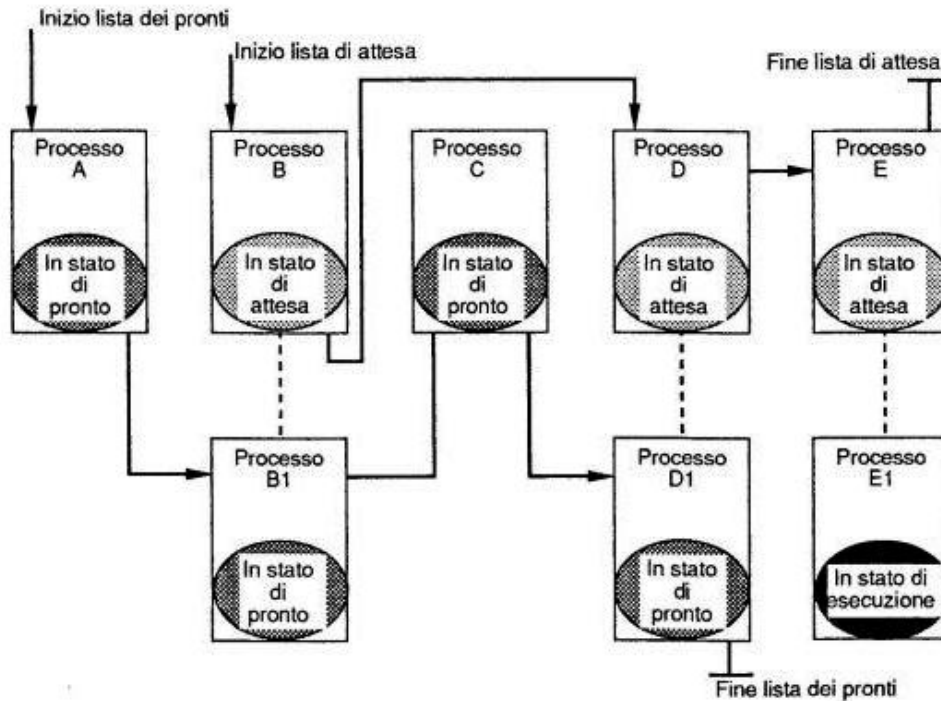


**POLITICHE DI SCHEDULAZIONE DEI LAVORI:**

- **FIFO** (First In First Out)
- Assegnando una **priorità statica**
- Assegnando una **priorità dinamica**
- Combinazione di lavori con **Elevata richiesta di CPU** ed **Elevata richiesta di I/O**

## DESCRITTORE DEI PROCESSI - PCB (Process Control Block) - basso livello (liste bidirezionali multiple):

- Nome processo
- Indirizzo descrittore processo
- Stato di avanzamento
- Informazioni aree di memoria e altre risorse
- Contabilizzazione CPU



tenere conto di:

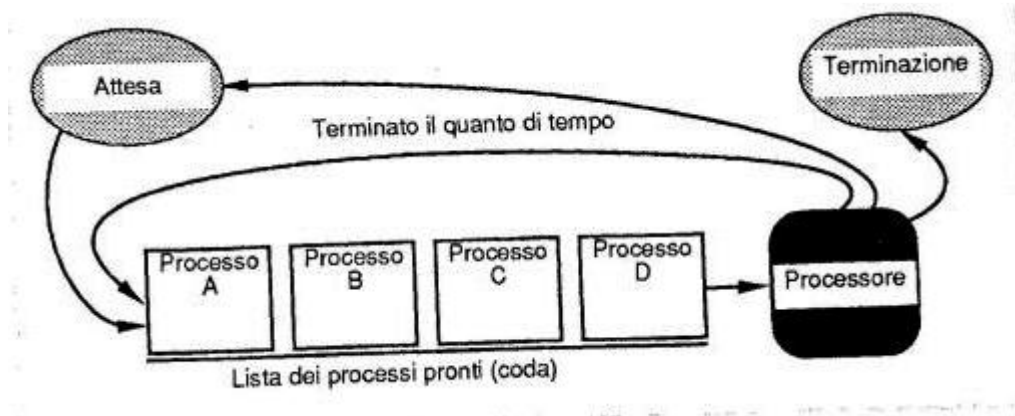
- rendere **massimo** il tempo di **utilizzo della CPU**
- servire il **maggior numero di processi** per unità di tempo
- rendere **minimo il tempo di attesa**
- minimizzare l'**overhead**

## POLITICHE DI SCHEDULAZIONE DEI PROCESSI:

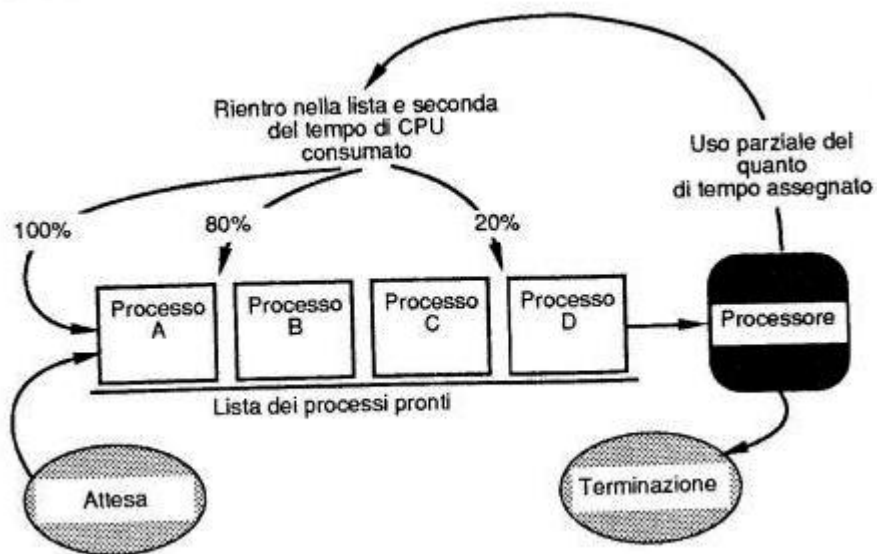
- **NON PREEMPTIVE**
  - **FIFO** (First In First Out)
  - **Shortest job first**
  - **Higher response ratio next:**  
$$\text{Priorità} = (\text{tempo attesa} + \text{tempo esecuzione}) / \text{tempo esecuzione}$$

- **PREEMPTIVE**

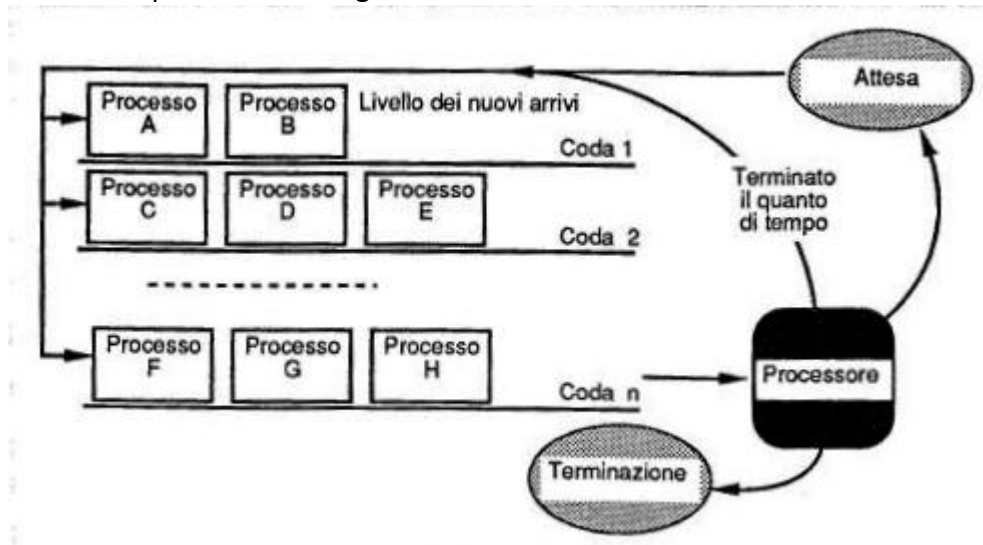
- Round robin



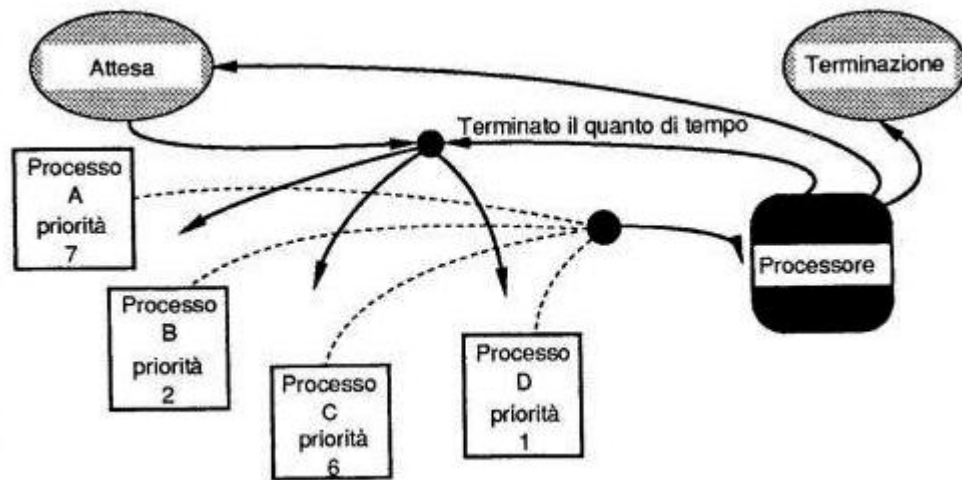
- Inverse of remainder of quantum



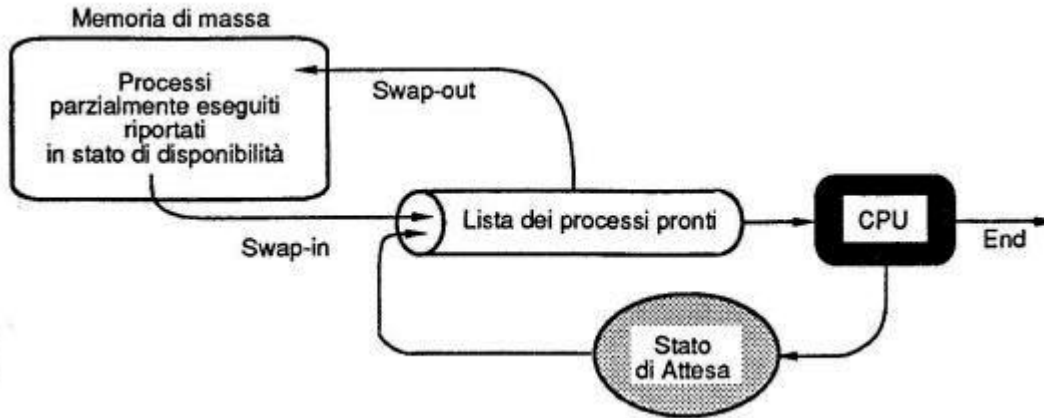
- Multilevel queue scheduling



- o Priorità con preemptive

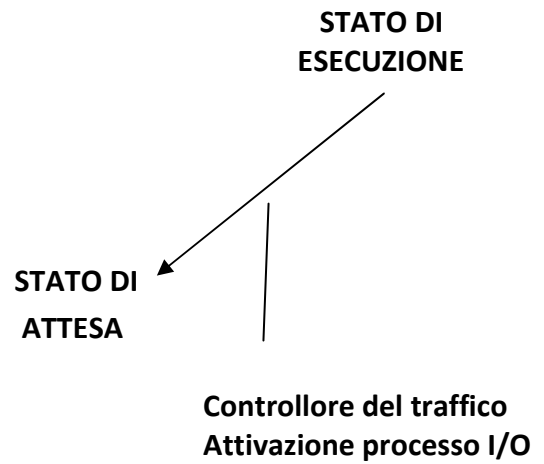


### SCHEDULATORE MEDIO LIVELLO



*Dallo stato di esecuzione può succedere:*

**1** Processo I/O



**2** Interruzione (viene attivata routine di risposta)

**3** Superamento tempo assegnato

**4** Termine processo (rilascio di tutte le risorse)

