

<b>INTRODUZIONE AI SISTEMI OPERATIVI</b>	<b>1</b>
Alcune definizioni	1
<b>Sistema dedicato:</b>	<b>1</b>
<b>Sistema batch o a lotti:</b>	<b>2</b>
<b>Sistemi time sharing:</b>	<b>2</b>
<b>Sistema multiprogrammato:</b>	<b>3</b>
<b>Processo e programma</b>	<b>3</b>
<b>Risorse:</b>	<b>3</b>
<b>Spazio degli indirizzi:</b>	<b>4</b>
Gestione risorse e politiche di schedulazione	4
Stato dei processi	6
Il nucleo	7
Schedulatore dei lavori detto anche schedulatore di alto livello	8
Schedulatore dei processi	8
Schedulatore di medio livello	10

## Introduzione ai sistemi operativi

Un sistema operativo è un insieme di programmi che consentono ad un utente di utilizzare un computer in maniera da

- Nascondergli tutti i dettagli tecnici della macchina
- Offrirgli un'interfaccia semplice e intuitiva
- Ottimizzare l'uso delle risorse della macchina

## Alcune definizioni

### ***Sistema dedicato:***

è la tipologia dei primi computer. Essi potevano svolgere un solo programma per volta. In memoria risiedeva un solo programma per volta e aveva a disposizione tutte

le risorse della macchina. Il sistema operativo si doveva limitare a gestire le operazioni di ingresso/uscita.

### ***Sistema batch o a lotti:***

in questi sistemi tipici dei grandi centri di calcolo i vari programmi venivano caricati in lotti nel computer e venivano eseguiti l'uno dopo l'altro in sequenza. Alla fine gli utenti ricevevano stampe con risultati delle elaborazioni. Non era consentita alcuna forma di interattività con l'utente.

### ***Sistemi time sharing:***

per consentire l'interazione fra i vari utenti e la macchina un sistema time sharing o a divisione di tempo, suddivide il tempo in intervalli detti timeslice e assegna a turno il processore ai vari lavori attivi. Il singolo utente ha l'impressione di avere la macchina a sua completa disposizione.

Il time sharing comporta il concetto di context switch: ogni volta che si deve passare da un processo all'altro bisogna salvare su disco lo stato del programma che si stava elaborando (ad esempio il contenuto del program counter e degli altri registri interni del microprocessore). Il sistema operativo comincia ad essere complesso e per svolgere le sue funzioni deve competere con gli altri programmi di utente per ottenere le risorse della macchina.

Per definire l'efficienza di un sistema operativo si utilizza allora un parametro definito overhead che è la percentuale di tempo in cui la CPU è utilizzata da parte del sistema operativo rispetto al tempo totale.

### ***Sistema multiprogrammato:***

si parla di sistema multiprogrammato se i vari programmi che stanno avanzando risiedono contemporaneamente nella memoria centrale. Ne derivano problemi di

- Protezione poiché occorre proteggere l'area di memoria assegnata ad ogni programma per impedire sovrascritture da parte degli altri programmi
- Condivisione, cioè la giusta tecnica di ripartizione delle risorse della macchina (processori, unità di IO, ecc) fra i vari processi

### ***Processo e programma***

Non sono termini equivalenti. il programma è la sequenza di istruzioni che il programmatore ha ideato per far svolgere un particolare compito al computer. Il processo è invece la sequenza di transizioni di stato che subisce il computer mentre esegue un programma

### ***Risorse:***

è l'insieme di tutte le entità che servono per l'esecuzione di un programma. Esse possono essere hardware (processore, dischi, stampanti, ecc.) o software (utility del sistema operativo, file di dati, ecc.)

### **Spazio degli indirizzi:**

nei sistemi multiprogrammati più programmi possono risiedere in memoria contemporaneamente. Le aree di memoria assegnate ad ogni programma prende il nome di spazio degli indirizzi. Per proteggere gli spazi degli indirizzi dei vari processi di utente da sovrascritture operate da altri programmi vi sono diverse soluzioni sia hardware che software. In generale si dividono l istruzioni di un microprocessore in due gruppi:

- Istruzioni standard che possono essere seguite da ogni utente
- Istruzioni privilegiate riservate al sistema operativo.

Il microprocessore si dice in stato utente se sta eseguendo istruzioni standard e in stato supervisore se sta eseguendo istruzioni privilegiate.

### **Gestione risorse e politiche di schedulazione**

In un sistema multiprogrammato il sistema operativo deve decidere come ripartire risorse come

- Il microprocessore
- La memoria
- Le periferiche

fa i vari processi.

Esso deve garantire

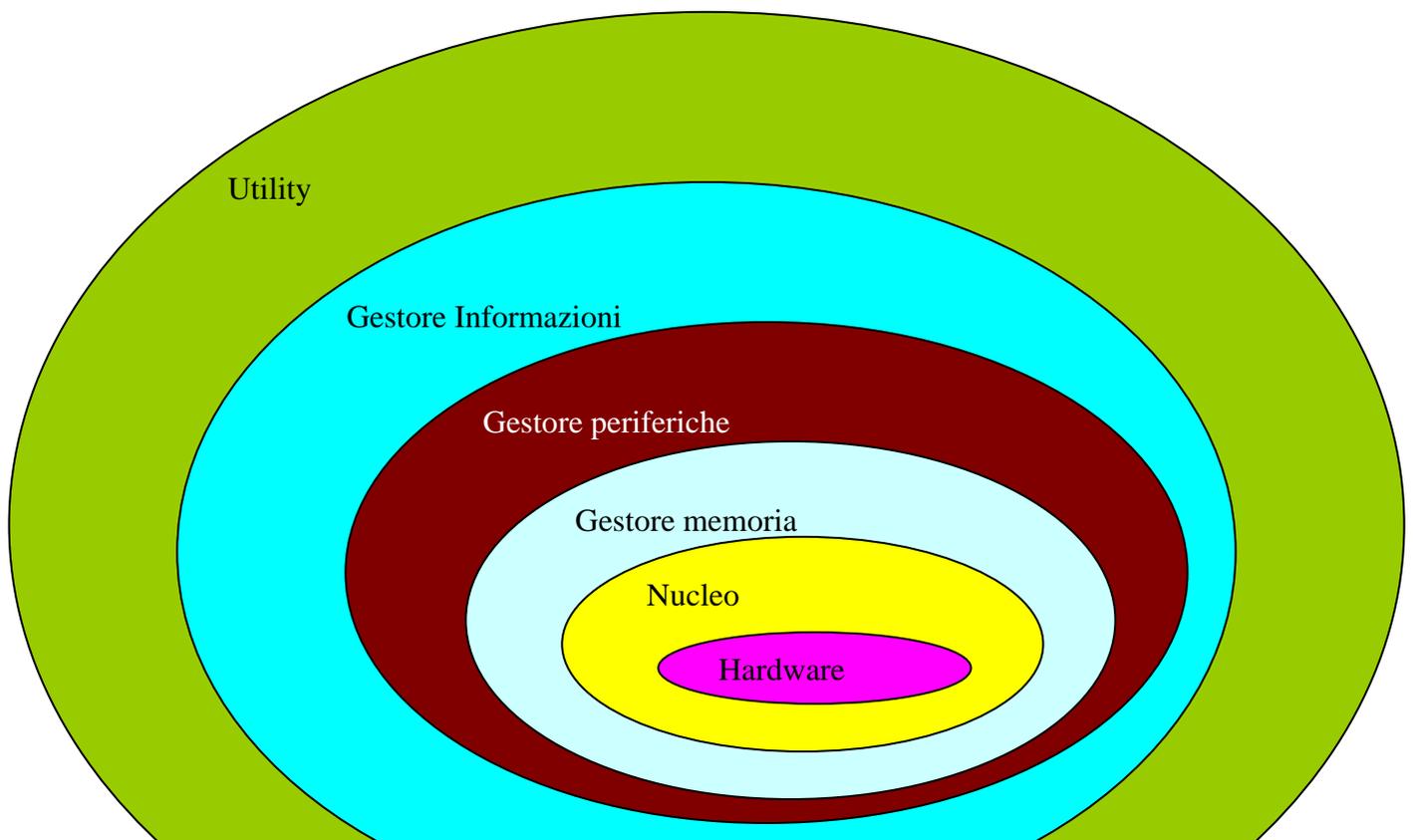
- Che ogni processo riesca prima o poi a ricevere le risorse che richiede
- Arbitrare conflitti fra i processi che cercano di accedere alla stessa risorsa

- Ottimizzare l'uso delle risorse
- Limitare i tempi di attesa delle risorse da parte dei processi

Vari moduli del sistema operativo gestiscono le varie risorse

- Gestore dei processori
- Gestore della memoria
- Gestore delle periferiche
- Gestore delle informazioni (gestione del file system, apertura e chiusura dei file, cancellazione dei file, decisione su quali utenti possono accedere ad un file e quali operazioni essi possono svolgere, ecc.)

Un sistema operativo si può immaginare come organizzato a strati in cui ogni livello offre i suoi servizi al livello superiore

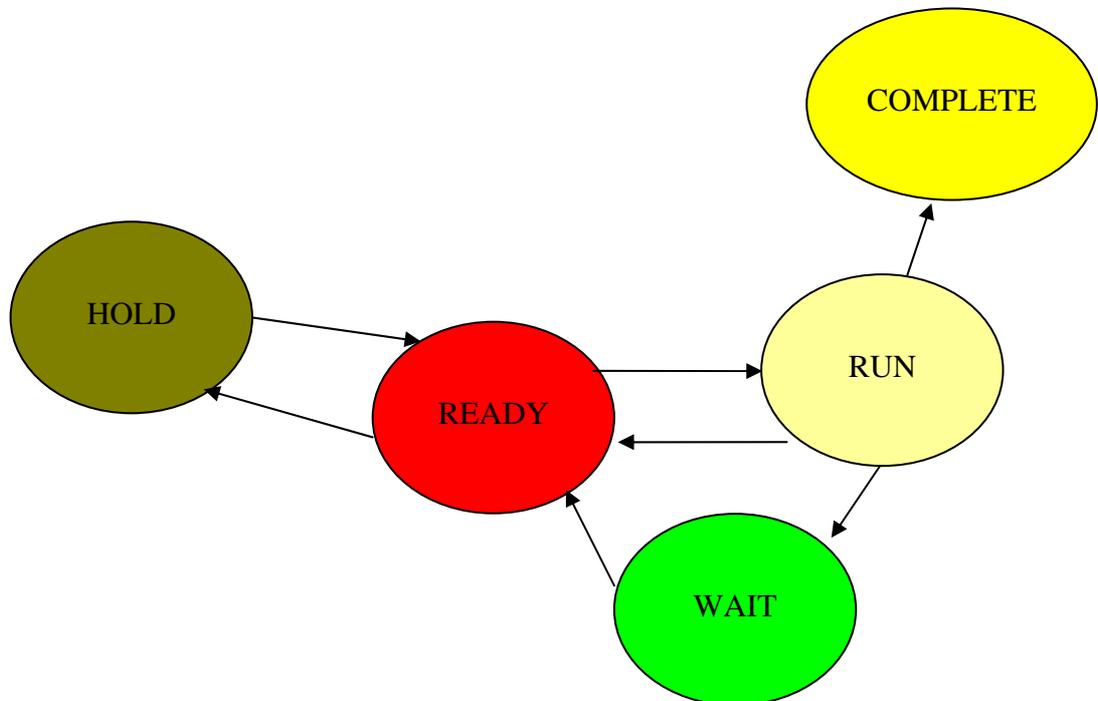


## **Stato dei processi**

Un processore può trovare in più stati diversi

- Hold o Disponibilità – Il processo è in attesa di esecuzione ma non è caricato in memoria
- Pronto o Ready – al processo sono state assegnate alcune risorse ed è stato caricato in memoria ma è in attesa di poter utilizzare il microprocessore
- Esecuzione o Run
- Attesa o Wait – un processo in esecuzione è costretto a fermarsi perché ha la necessità di attendere una risorsa, ad esempio la conclusione di una operazione di IO

- Terminazione – il processo è terminato



## Il nucleo

Si compone dello

- Scheduler dei lavori (job scheduler o admission scheduler) che seleziona i lavori che devono passare dallo stato di hold a quello di ready
- Scheduler dei processi che decide quali passano da stato di ready a stato di run
- Controllore del traffico che controlla lo stato di avanzamento dei singoli processi e la transizione da uno stato all'altro

## **Schedulatore dei lavori detto anche schedulatore di alto livello**

Assegna le risorse ai lavori che decide di caricare in memoria e recupera le risorse dei processi che sono terminati.

La più semplice politica di schedulazione possibile potrebbe essere quella di tipo FIFO.

Un metodo alternativo potrebbe essere quello di assegnare una priorità statica predefinita ai vari lavori da attivare e di assegnare risorse e memoria ad essi in ordine di priorità decrescente. Questa priorità viene determinata una volta per tutte e può essere calcolata in base a vari criteri quali l'importanza del lavoro oppure una stima dei tempi di esecuzione (ad esempio si dà priorità maggiore ai lavori più veloci in modo da aumentare il throughput del sistema).

Il problema della priorità statica è che un processo a priorità bassa potrebbe teoricamente attendere all'infinito di avere il permesso di passare in stato di ready perché continuano ad arrivare richieste di lavori a priorità superiore.

Una possibile soluzione è quella dell'aging cioè di rendere dinamica la priorità assegnata ai vari lavori facendola incrementare via via che passa il tempo.

## **Schedulatore dei processi**

Lo schedulatore dei processi deve attribuire o togliere la risorsa microprocessore ai vari processi cercando di contemperare varie esigenze

- Massimizzare il tempo di utilizzo della CPU
- Massimizzare il throughput del sistema cioè il numero di processi completati nell'unità di tempo

- Minimizzare il tempo di attesa dei processi in stato di pronto
- Minimizzare l'overhead. Se si passa troppo frequentemente da un processo all'altro aumentano i tempi necessari al Sistema Operativo per effettuare il context switch per cui aumenta l'overhead cioè il tempo impiegato dal SO a discapito degli altri processi.

Le politiche di scheduling possono essere

- Preemptive (cioè una volta che la CPU è assegnata ad un processo non gli viene tolta fino alla fine del processo)
- Non preemptive (in cui il Sistema Operativo può decidere di togliere la risorsa processore ad un processo prima del suo completamento).

Esempio di politiche non preemptive sono

- Politica FIFO (viene assegnato il processore ai processi nell'ordine con cui sono stati generati)
- Shortest job first in cui vengono privilegiati i lavori con tempo di esecuzione più rapido. Il problema è che non è detto che si conoscano a priori i tempi di esecuzione di singoli processi né che questi siano sempre gli stessi
- Higher response ratio next: si assegna una priorità ai vari processi calcolata come  $P = \frac{\text{Tempodiattesa} + \text{tempodiesecuzione}}{\text{Tempodiesecuzione}}$ . Poiché il tempo di attesa è al numeratore la priorità crescerà per i processi più vecchi.

Esempio di politiche preemptive sono

- Round Robin: il processo viene assegnato a turno ad ogni processo per un certo intervallo di tempo, e se alla fine del turno il processo non è terminato, esso ritorna nella coda dei processi pronti
- Inverse of Remainder of Quantum: è simile al Round Robin, la differenza sta nel fatto che quando un processo torna nello stato di pronto, al sua posizione nella coda dipende dalla frazione dell'intervallo di tempo a sua disposizione che ha effettivamente consumato. Minore sarà tale percentuale di tempo e migliore sarà la sua posizione nella coda.
- A priorità preemptive: ad ogni istante viene assegnato il processore al processo che ha priorità superiore. Per evitare il problema della starvation , cioè attesa infinita di processi a bassa priorità si ricorre alla tecnica dell'aging con incremento della priorità con il passar del tempo

### **Schedulatore di medio livello**

Se un processo non può passare per molto tempo dallo stato di Ready allo stato di Run, diventa oneroso tenerlo in memoria, impedendo ad altri processi di avanzare. Lo schedulatore di medio livello è il modulo del Sistema Operativo che si occupa di scaricare tali processi dalla memoria per assegnarla ad altri processi