

# Sistemi operativi

## 1. Introduzione ai sistemi operativi

Il sistema operativo è un *software* che permette di interfacciare, controllare e coordinare l'uso dell'hardware fra i differenti programmi applicativi dei vari utenti.

Il compito principale del sistema operativo include anche la gestione delle **risorse hardware** e quelle **software**, ovvero risorse astratte assegnate e/o utilizzate dai vari processi.

Nella gestione di queste risorse il sistema operativo:

- traccia le risorse;
- adotta strategie di assegnazione di risorse (dove, quando e per quanto assegnarle, contingentando tempo e richieste);
- alloca le risorse;
- recupera e ripristina le risorse;

Le risorse vengono utilizzate in modo **mutuamente esclusivo**.

In tutto questo, il SO deve garantire la risoluzione di possibili conflitti, mantenendo l'integrità del sistema e ottimizzando le prestazioni globali.

Le aree del controllo del sistema operativo sono:

- CPU
- RAM
- Dispositivi I/O

I sistemi operativi **multiutente** dovranno essere in grado di proteggere i singoli utenti nelle singole operazioni che svolgono.

## 2. Sistemi operativi multiprogrammati

Fra le categorie più importanti di sistemi operativi ci sono i sistemi **multiprogrammati** (UNIX/Linux, ad esempio).

La multiprogrammazione è un termine che descrive un sistema operativo che supporta più programmi caricati in RAM, che vengono gestiti concorrentemente.

Se un sistema è **monoprogrammato** i programmi vengono eseguiti in **sequenza**, piuttosto che concorrentemente.

Se il sistema è multiprogrammato i programmi vengono eseguiti concorrentemente e le risorse non vengono assegnate completamente a uno dei programmi per volta.

Ovviamente il sistema multiprogrammato è un modello molto più complicato, ma che è in grado di ottimizzare gli evidenti problemi di un sistema monoprogrammato.

L'obiettivo principale della multiprogrammazione è quello di utilizzare le risorse nel modo più ottimizzato possibile, riducendo l'overhead e massimizzando le prestazioni (throughput).

## 3. Processi

La differenza fra **programma** e **processo** è delineata dal fatto che il programma sia un'entità statica (linee di codice scritte in un file). Il processo è l'entità che viene caricata nel sistema operativo quando il programma viene eseguito.

Si possono distinguere:

- processi I/O bound;
- processi CPU bound;

Si può inoltre aggiungere la definizione di **multiutenza**: più utenti accedono **contemporaneamente** (reale, non concorrente) allo stesso sistema operativo.

## 4. S.O time sharing

Un sistema operativo T-S fornisce a ciascun utente una porzione di un sistema di calcolo creando “un’illusione” del fatto che abbia il sistema (compreso l’hardware) tutto per sé.

Ciò che accade è che il sistema operativo attribuisce porzioni temporali nelle quali il sistema operativo è di un determinato utente. Tutto questo è più *seamless* all’aumentare della potenza delle sue componenti.

L’elemento che garantisce questa “illusione” è lo **scheduling** (alternanza) dei processi, suddividendo il tempo attribuito a ciascun processo.

I processi di ciascun utente vengono, come al solito, eseguiti concorrentemente.