

Sistemi Operativi

Docente: Ugo Erra
ugoerr+so@dia.unisa.it



1° LEZIONE

INTRODUZIONE AI SISTEMI OPERATIVI

***CORSO DI LAUREA TRIENNALE IN INFORMATICA
UNIVERSITA' DEGLI STUDI DELLA BASILICATA***



Informazioni sul corso - 1



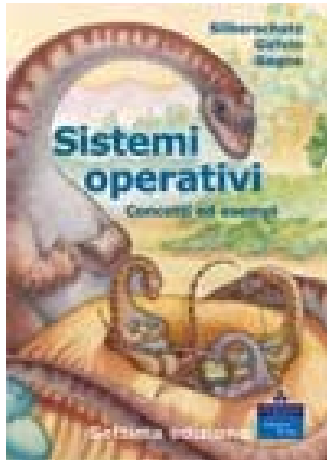
- **Sistemi Operativi**
 - Per il corso di laurea triennale in Informatica
 - 6 CFU
- **Organizzazione temporale**
 - 1° periodo didattico: dal 1 Ottobre al 7 Dicembre
 - Pausa intercorso: dal 10 dicembre al 21 dicembre
 - 2° periodo didattico: dal 7 gennaio al 1 febbraio
 - Prove finali: dal 4 febbraio al 29 febbraio
- **Modalità di esame**
 - Una prova scritta ed una prova orale (no prove intermedie)

Informazioni sul corso - 2



- **Lezioni**
 - Martedì ore 11:30-14:30 Aula A1
 - Mercoledì ore 08:30-11:30 Aula A1
- **Orario di ricevimento**
 - Mercoledì ore 11:30-12:30
- **Materiale del corso**
 - Slide, esercizi ed appunti sul portale del corso
 - ✦ <http://scienze.unibas.it/progettoIns/index.jsp>
- **Per qualunque comunicazione**
 - *ugoerr+so@dia.unisa.it*

Informazioni sul corso - 3



- Testo principale di riferimento:
- Sistemi operativi - Concetti ed esempi - 7a Edizione - A. Silberschatz - P. Galvin - G. Gagne - Prentice Hall

Sommario della lezione

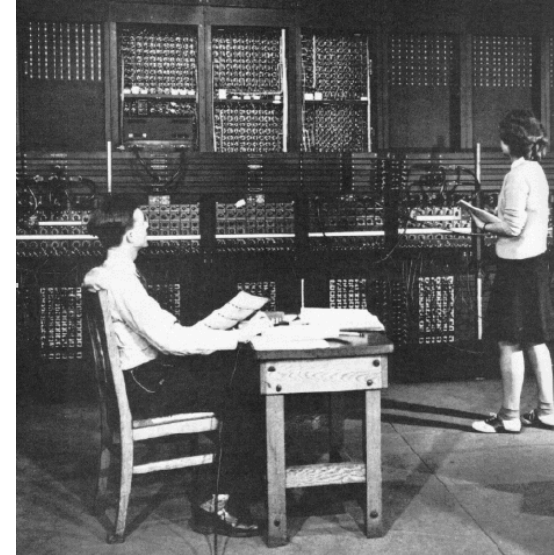


- Una breve storia dei Sistemi Operativi
- Che cosa è un Sistema Operativo
- Organizzazione di un sistema di calcolo
- Architettura degli elaboratori
- Attività di un Sistema Operativo
 - Gestione dei processi
 - Gestione della memoria
 - Gestione della memoria di massa
 - Sistema di I/O
 - Protezione e Sicurezza

Prima dei Sistemi Operativi



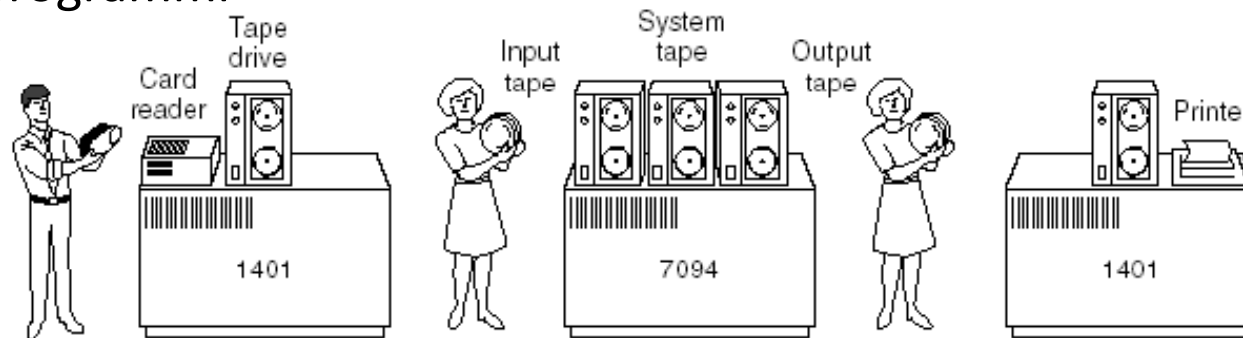
- Tra il 1945 e il 1955 gli elaboratori elettronici erano progettati con valvole termoioniche
- Occupavano intere stanze, erano lentissimi e costosi che potevano permetterseli soltanto grossi centri di calcolo o Università
- Inoltre questi calcolatori erano molto inaffidabili, in quanto le valvole che li componevano si rompevano spesso
- In questo periodo non esisteva ancora il concetto di Sistema Operativo
 - Il programma da eseguire veniva inserito ad ogni esecuzione in codice binario attraverso dei primitivi lettori di schede perforate e dopo alcune ore il risultato veniva inviato ad una stampante



I primi Sistemi Operativi (anni '50)



- Tra il 1955 e il 1965, grazie ai transistor gli elaboratori (mainframe) divennero abbastanza affidabili da poter essere costruiti e venduti in serie, anche se erano comunque macchine grosse e costosissime
- Per eseguire dei programmi (o come venivano chiamati job), un programmatore doveva:
 - Scrivere il proprio programma su carta
 - Trasferirlo su schede
 - Caricarlo nel computer
 - Attendere il termine dell'esecuzione e la stampa del risultato
- Tale operazione era molto dispendiosa in termini di tempo e non permetteva di sfruttare la macchina durante le lunghe fasi di caricamento di dati e programmi

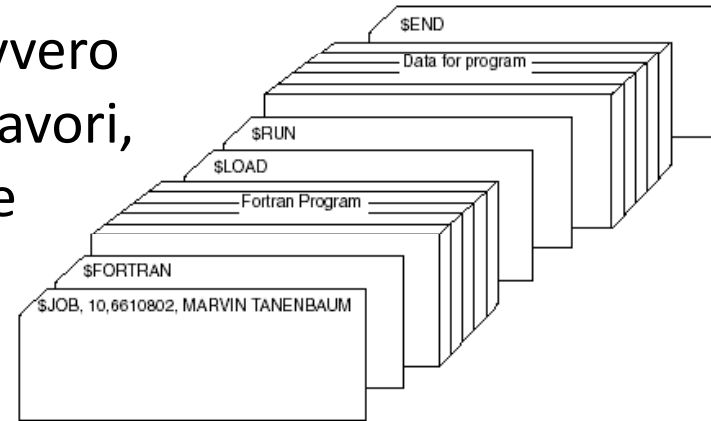


I primi Sistemi Operativi (anni '50)



- Nasce l'idea dei sistema batch (a *lotti*) ovvero l'idea di base era quella di dividere i tre lavori, ovvero il caricamento dei dati, il calcolo e la stampa su macchine distinte

- L'accesso diretto alla memoria non era stato ancora concepito quindi durante le fasi di input/output il processore era totalmente utilizzato per il controllo di queste operazioni



- I sistemi operativi tipici per questi elaboratori, per lo più programmati in FORTRAN e in Assembler erano il FMS (*Fortran Monitor System*) e IBSYS
- Vengono introdotte *politiche di ordinamento dei job*. Essi venivano caricati ed eseguiti in maniera sequenziale, ma l'ordine di esecuzione dei programmi era gestito da opportune politiche implementate nel Sistema Operativo

La multiprogrammazione (anni '60)



- Nel 1964 IBM presenta una famiglia di computer chiamata IBM System/360, prima realizzazione di una netta distinzione tra *architettura hardware e implementazione*
- Per la prima volta è introdotta la multiprogrammazione, che rendeva possibile la presenza di più programmi in memoria contemporaneamente
 - Questo rendeva necessaria la presenza di hardware specializzato per proteggere i programmi dalle reciproche interferenze
- Quando un job in esecuzione aveva bisogno di effettuare una operazione di I/O il Sistema Operativo lo sospendeva e al suo posto faceva girare un altro job

La multiprogrammazione (anni '60)



- In questi anni si introduce anche il timesharing
 - Ogni utente dispone di un dispositivo di ingresso (la tastiera) e un dispositivo di uscita (un monitor o stampante)
- Nel 1962 venne realizzato al MIT il primo sistema di timesharing su un IBM 7094 ma la vera rivoluzione di ebbe con il MULTICS
 - Sviluppato congiuntamente dal MIT, dalla General Electric e dai Bell Labs
- Negli stessi anni fu sviluppato il minielaboratore PDP-1 (costava solo 120.000\$) che ebbe un gran successo
- Per questi sistemi vennero progettati appositi Sistemi Operativi, il più famoso dei quali fu UNIX



La multiprogrammazione (anni '60)



- UNIX fu progettato a partire dal 1969 da un gruppo di ricercatori della AT&T presso i Bell Labs, tra cui erano presenti Ken Thompson, Dennis Ritchie e Douglas McIlroy
- Si ispirò a Multics, e divenne un sistema molto interattivo, affidabile e ricco di funzionalità, tanto che tuttora domina il mercato delle workstation
- Furono sviluppate diverse varianti di UNIX, come il System V, BSD (*Berkley Software Distribution*), Minix (usato in ambito didattico) e successivamente (sulla base di MINIX e UNIX) l'ormai famosissimo Linux sviluppato dallo studente finlandese Linux Torvalds

I personal computer (anni '80)

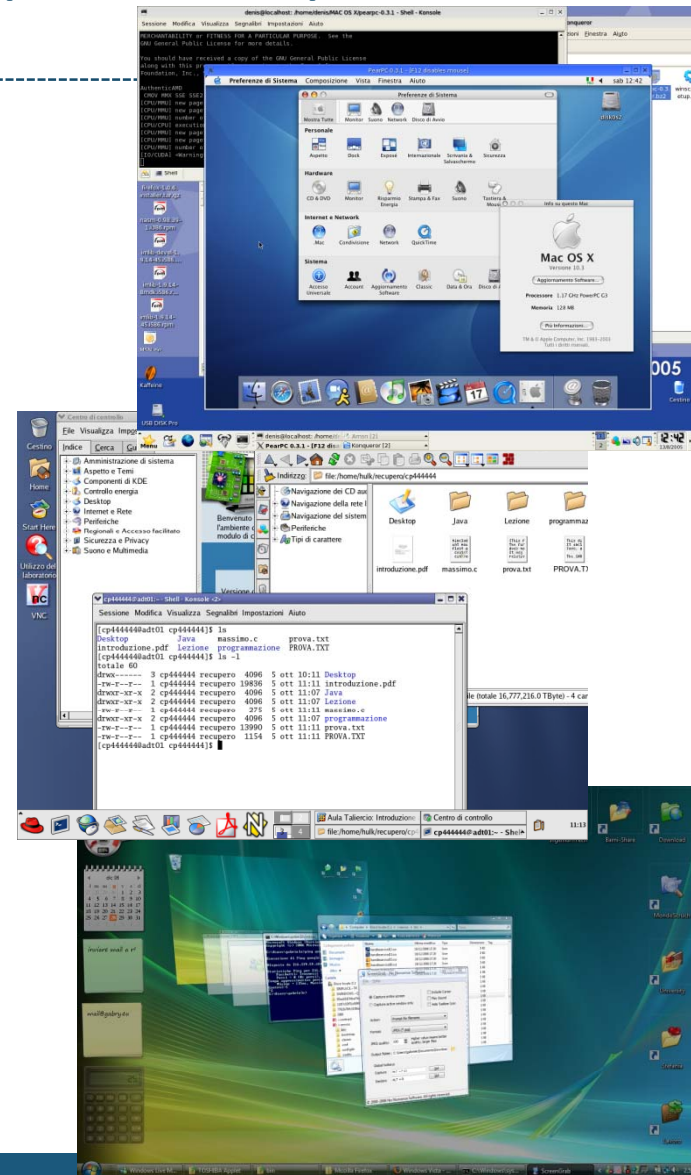


- Negli anni '80 la tecnologia LSI porta alla costruzione di chip integrati ed all'abbattimento dei prezzi dell'hardware, facendo sorgere l'era del personal computer
- I primi modelli erano dotati di Sistemi Operativi monoutente con accesso interattivo e supporto al timesharing
- Il più importante tra i primi Sistemi Operativi per Personal computer era il CP/M-80 della Digital Research per le CPU 8080 / 8085 / Z-80
 - Era basato sui Sistemi Operativi della specialmente quelli per l'architettura PDP-1.
- MS-DOS (o *PC-DOS* da IBM) era originariamente basato proprio sul CP/M-80

I personal computer (anni '80)



- Negli anni '80 la Apple era uno dei pochi che credeva nell'idea Personal Computer. All'epoca era difficile immaginare cosa potesse farsene una persona di un computer in casa.
- La Xerox lancia il primo Sistema Operativo con interfaccia grafica.
- La Apple prende in "prestito" questa idea e lancia nel 1976 il Mac OS, primo sistema operativo per personal computer.
- Successivamente la Microsoft avrebbe commercializzato Windows (20 novembre 1985).



I Sistemi Operativi oggi



- Oggigiorno è disponibile una grande varietà di sistemi di elaborazione dalle più disparate dimensioni e performance a costi contenuti
- Tutte queste innovazioni hanno portato allo sviluppo di sistemi operativi per le più svariate architetture, in particolare per *dispositivi handheld* come cellulari (tra i quali il Symbian OS) e PDA (con Windows Mobile e Palm OS).
- Per qualunque architettura venga sviluppato un Sistema Operativo moderno esso deve fornire il supporto, oltre a quanto visto sinora, a molteplici esigenze quali:
 - Streaming audio/video (trasmissione ed elaborazione continua di dati multimediali) supporto alle più diverse tecnologie di interconnessione (ad esempio Ethernet, Bluetooth e Wireless LAN)
 - Integrazione di tecnologie per la fruizione di contenuti su Internet
 - Gestione efficiente dell'energia

Che cosa è un Sistema Operativo?



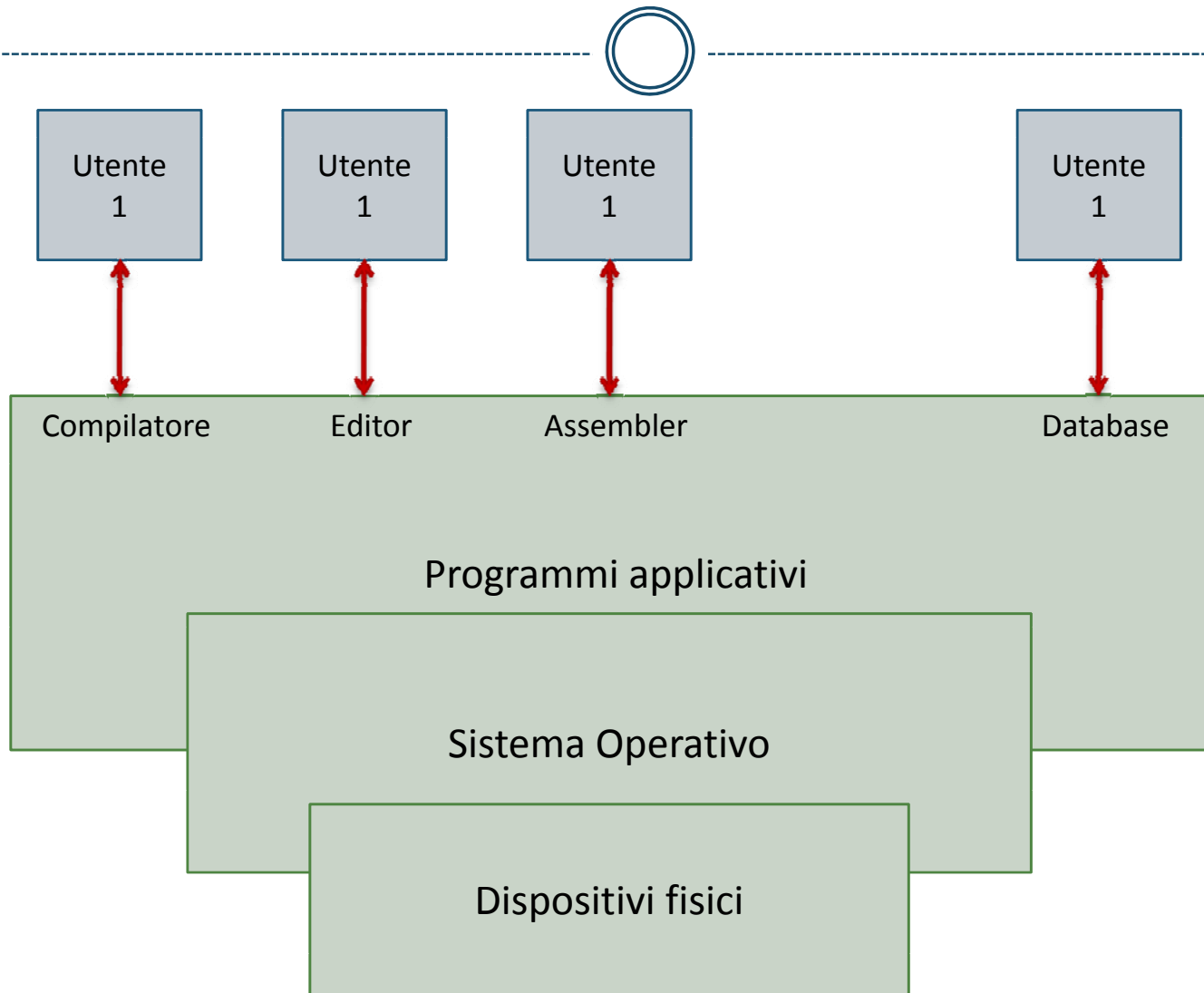
- Un **Sistema Operativo** è un insieme di programmi che interagiscono da intermediari tra l'utente e le risorse di un *sistema di calcolo* (o computer)
- Gli scopi di un sistema operativo sono:
 - Eseguire i programmi degli utenti
 - Rendere un computer semplice da utilizzare
 - Sfruttare in maniera efficiente le risorse hardware
- Un Sistema Operativo fornisce un ambiente nel quale i programmi possono essere eseguiti in modo utile e produttivo

Componenti di un sistema di calcolo



- Un sistema di calcolo può essere diviso in quattro componenti:
 - Hardware
 - ✦ CPU, memoria, dispositivi di I/O
 - Sistema Operativo
 - ✦ Controlla e coordina l'uso dell'hardware tra le diverse applicazioni e tra gli utenti
 - Programmi Applicativi
 - ✦ Rappresentano il modo con i quali gli utenti utilizzando le risorse del sistema per risolvere alcuni problemi o soddisfare le proprie esigenze
 - ✦ Word Processor, Compilatori, Browser, Database, Videogame
 - Gli Utenti
 - ✦ Persone, macchine o altri computer

Componenti di un sistema di calcolo



Definizioni di Sistema Operativo



- Il Sistema Operativo è un **assegnatore di risorse**
 - Gestisce tutte le risorse
 - Decide la risoluzione dei conflitti per un uso efficiente e sicuro delle risorse di calcolo
- Il Sistema Operativo è un **programma di controllo**
 - Controlla l'esecuzione dei programmi per prevenire possibili errori o evitare l'uso improprio delle risorse di calcolo
- Sono definizioni non universalmente accettate
 - Un'altra possibile definizione potrebbe essere tutto quello che il vostro rivenditore vi infila dentro la scatola
- Un'altra definizione è che il Sistema Operativo è il solo programma in esecuzione continuamente detto **kernel**
 - Il resto sono solo programmi applicativi

Funzionamento di un sistema di calcolo



- Un sistema di calcolo può essere descritto nelle sue funzioni essenziali attraverso i seguenti elementi
 - Modello o funzionamento del sistema di calcolo
 - Struttura della memoria
 - Struttura di I/O

Funzionamento di un sistema di calcolo

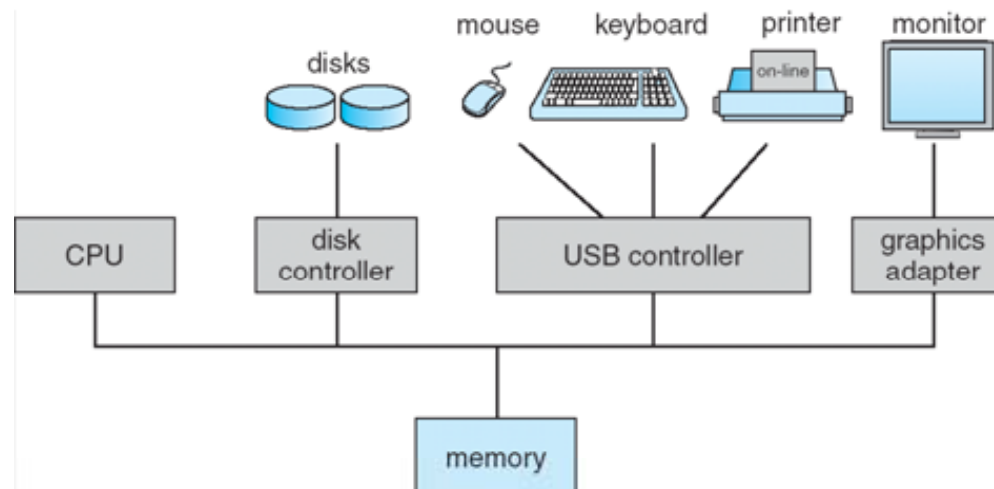


- Normalmente in un sistema di calcolo dopo l'accensione la prima fase è il **bootstrap program** (programma di avviamento)
 - Tipicamente il firmware memorizzato in una ROM o in una EPROM viene caricato
 - Il suo scopo è inizializzare ogni aspetto del sistema
 - ✦ Registri della CPU, controllori dei diversi dispositivi
 - Il passo cruciale è caricare nella memoria il kernel del sistema operativo
 - ✦ Il sistema operativo avvierà normalmente il primo processo (ad esempio sotto Linux è init)
 - ✦ Questo processo una volta avviato si metterà in attesa di un evento o di ulteriori istruzioni

Architetture di sistema di calcolo



- Un sistema di calcolo è composto da una o più CPU, alcuni controllori di *dispositivi di I/O (device)* connessi attraverso un bus alla memoria
- Questi dispositivi operano in maniera concorrente per ottenere cicli di accesso alla memoria



Gestione dei dispositivi di I/O



- I dispositivi di I/O e la CPU sono sempre in funzione in maniera concorrente
- Ogni dispositivo di I/O ha un controllore che gestisce le operazioni del dispositivo attraverso un buffer locale
- La CPU per usare questi dispositivi di I/O sposta dati dalla memoria di sistema nel buffer locale e viceversa
- Il dispositivo di I/O fornisce normalmente nuovi dati all'interno di questo buffer locale
- Ogni qualvolta che un dispositivo ha nuovi dati causa un **interrupt per richiamare l'attenzione della CPU**

Esecuzione di un interrupt



- Un interrupt generalmente trasferisce il controllo ad una routine di servizio
 - Normalmente esiste una sequenza di indirizzi, detta **vettore delle interruzioni**, che punta alle diverse routine di servizio
- Il Sistema Operativo deve salvare l'indirizzo dell'istruzione appena interrotta per poterla continuare ad eseguire successivamente
- Ulteriori interrupt sono normalmente disabilitati per non causare incoerenze nella normale esecuzione dell'interrupt corrente
- Una **eccezione** o **trap** è normalmente un interrupt causata dal software da un errore oppure dall'utente
 - Ad esempio una divisione per zero
- Un sistema operativo è quindi anche un gestore di interrupt

Gestione dell'I/O

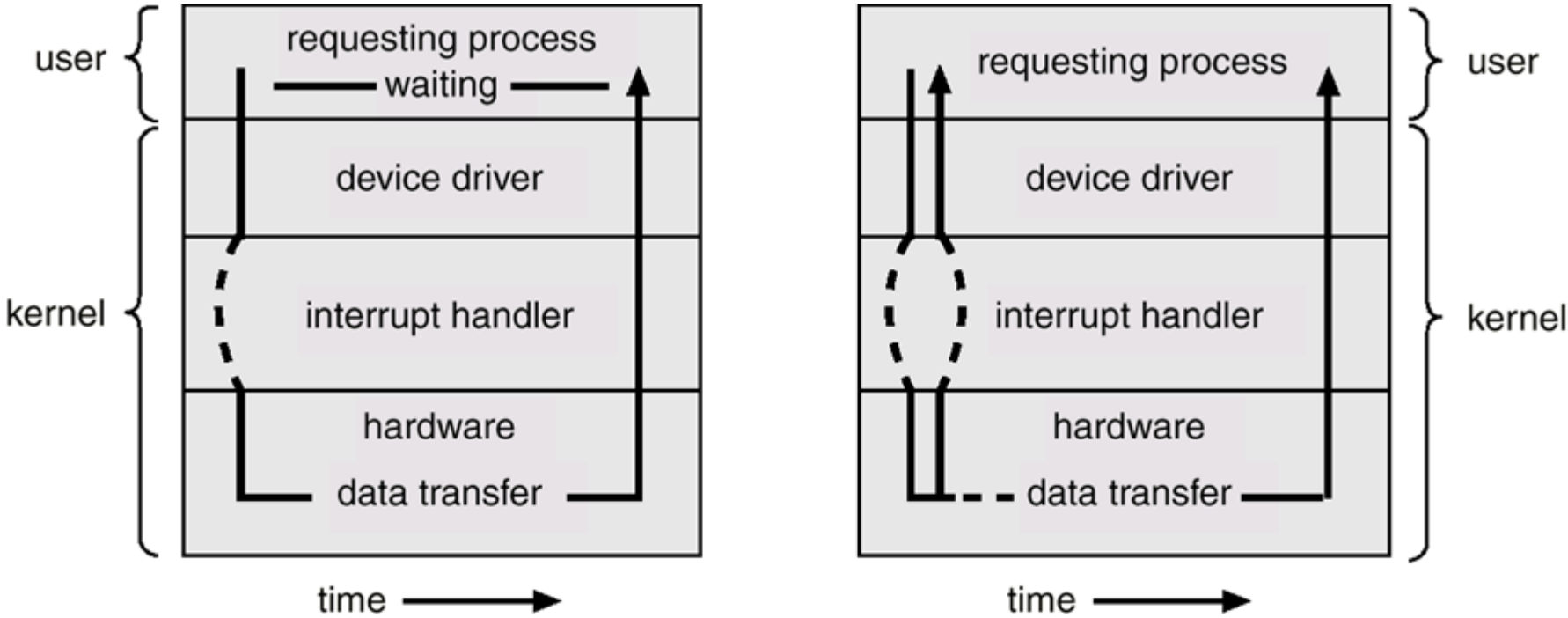
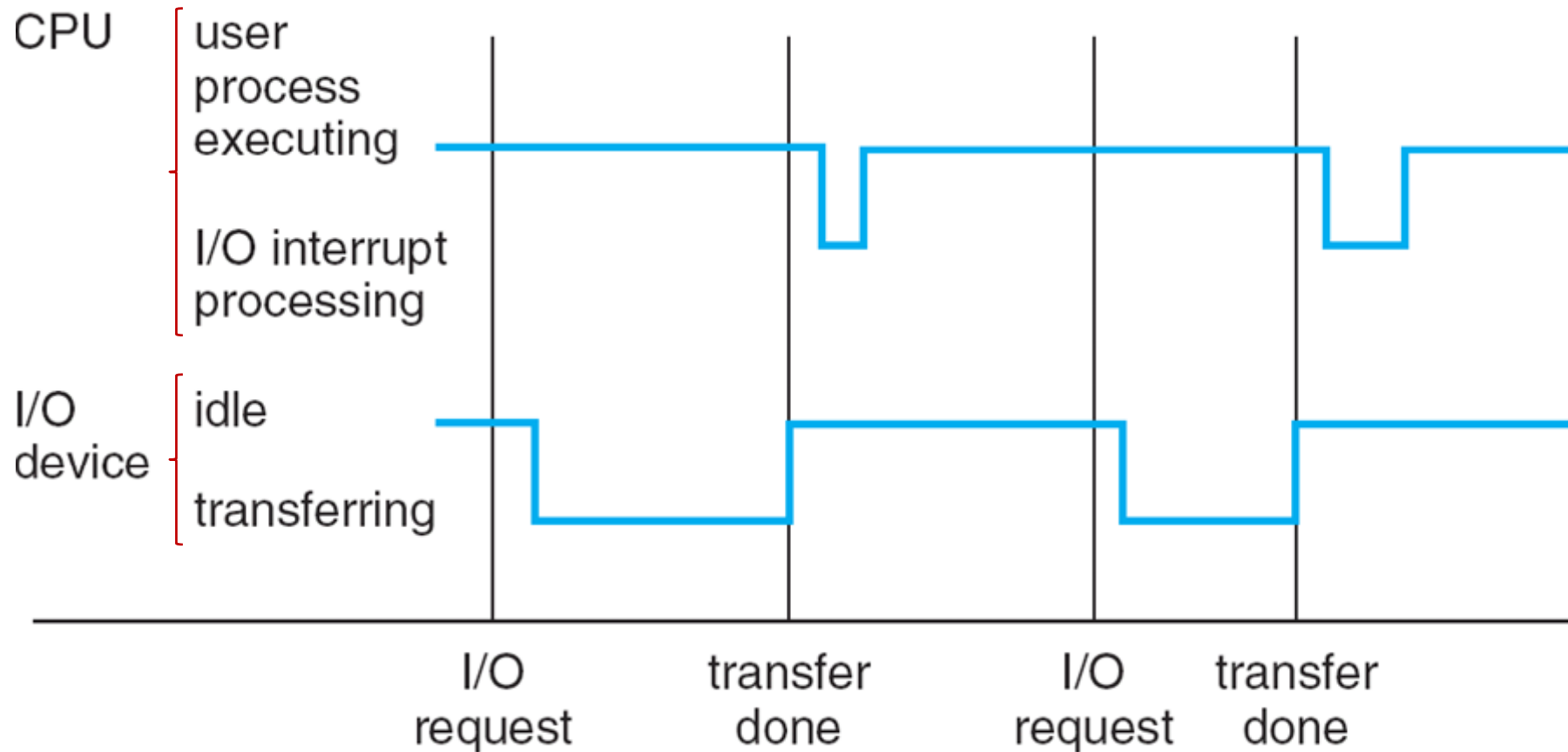


Diagramma temporale degli interrupt



DMA – Accesso Diretto alla Memoria



- L'accesso diretto alla memoria o DMA permette ad un dispositivo di trasmettere i dati alla memoria ad una velocità prossima a quella della memoria
- Il controllore del dispositivo trasferisce blocchi di dati dal proprio buffer locale alla memoria principale senza nessun intervento della CPU
- La CPU reagisce ad un solo interrupt per blocco, piuttosto che ad un interrupt per ogni byte

Struttura della memoria



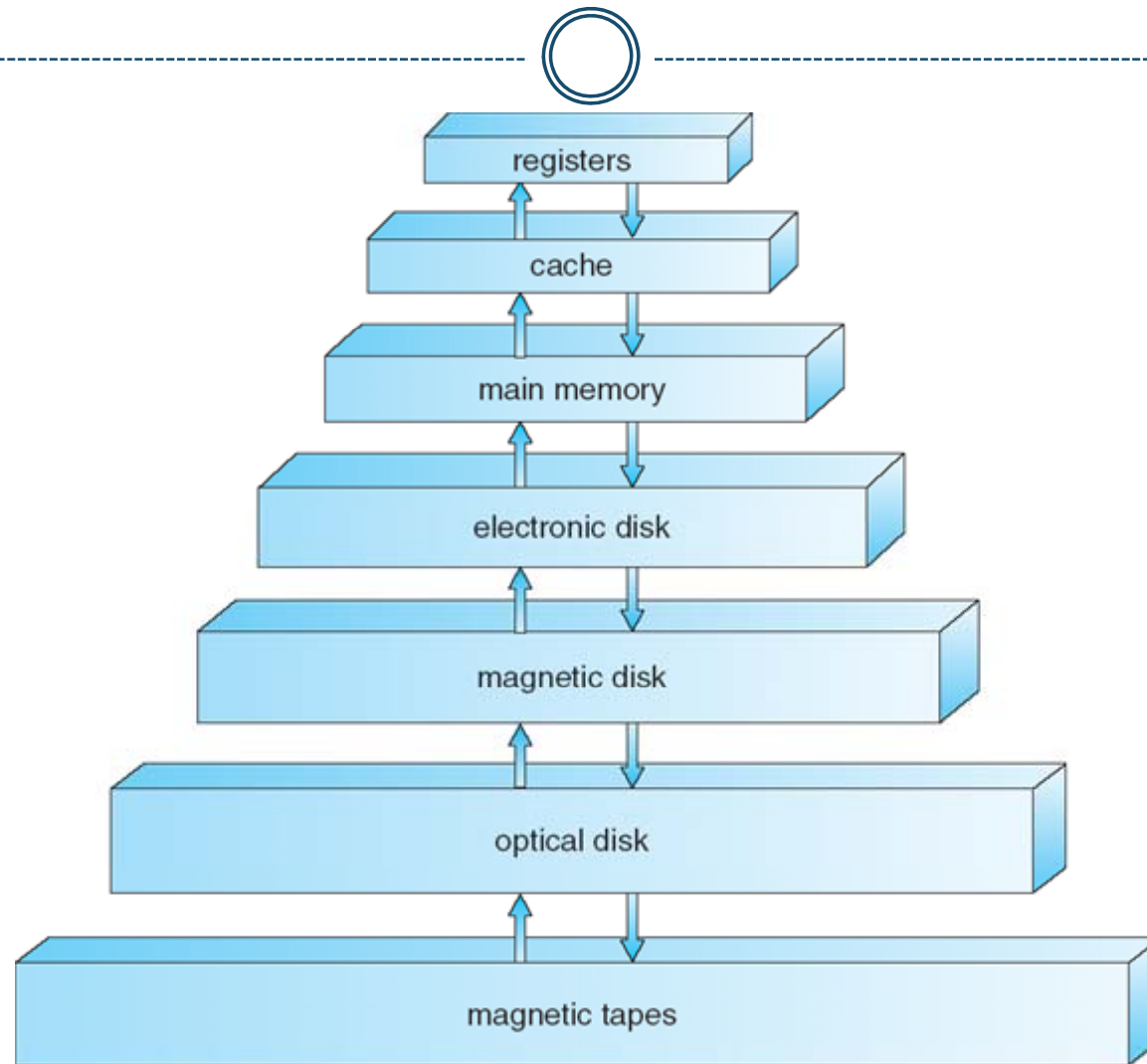
- **Memoria principale**
 - Utilizzata prevalentemente dalla CPU tramite accesso diretto per le istruzioni e i dati
- **Memoria secondaria**
 - Estensione della memoria principale per la memorizzazione di grandi quantità di dati in maniera non volatile
- **Dischi magnetici**
 - Provvedono una memoria non-volatile di grandissime dimensioni
 - Il controllore dei dischi determina la logica di interazione tra il dispositivo ed il computer

Gerarchia dei dispositivi di memoria



- I dispositivi di memorizzazione sono organizzate in gerarchie basate sui seguenti criteri:
 - Velocità
 - Costi
 - Volatile / Non-volatile
- La *cache* è una memoria intermedia veloce che permette di aumentare l'efficienza negli accessi ai dati
 - Le cache sono presenti solitamente su tutti i livelli
 - La memoria principale può essere vista come una enorme cache per la memoria secondaria

Gerarchia dei dispositivi di memoria



Caching



- Il caching è un meccanismo presente in diversi livelli di un sistema di calcolo
 - In hardware, nel sistema operativo, nel software
- Le informazioni di uso più frequente sono copiate da una memoria lenta ad una memoria ad accesso più veloce
- Normalmente si controlla se nella memoria veloce queste informazioni non sia già presenti
 - Se sono presenti le informazioni vengono utilizzate direttamente dalla cache
 - Se non sono presenti, i dati sono copiati nella cache e utilizzati dalla cache
- Solitamente la dimensione delle cache sono minori della memoria vera e propria
 - La gestione della cache rappresenta un problema da affrontare
 - Sono necessarie politiche di rimpiazzamento dei dati

Prestazioni dei diversi livelli di memoria



Level	1	2	3	4
Name	registers	cache	main memory	disk storage
Typical size	< 1 KB	> 16 MB	> 16 GB	> 100 GB
Implementation technology	custom memory with multiple ports, CMOS	on-chip or off-chip CMOS SRAM	CMOS DRAM	magnetic disk
Access time (ns)	0.25 – 0.5	0.5 – 25	80 – 250	5,000.000
Bandwidth (MB/sec)	20,000 – 100,000	5000 – 10,000	1000 – 5000	20 – 150
Managed by	compiler	hardware	operating system	operating system
Backed by	cache	main memory	disk	CD or tape

Architettura degli elaboratori



- Il mercato oggi offre principalmente due differenti sistemi di calcolo
 - Sistemi monoprocessoore
 - Sistemi multiprocessori (o multicore)

Sistemi monoprocesso



- Tipici sistemi ad un solo processore
 - Mainframe
 - PDA
- Solitamente un solo processore governa l'esecuzione delle applicazioni degli utenti
- Altri processori di potenza computazionale diversa possono essere adoperati per compiti specifici
- Controllo dei dischi, scheda grafica, gestione della tastiera

Sistemi multiprocessore



- Nei sistemi multiprocessore più CPU sono collegate tra di loro attraverso un bus e possono eseguire in parallelo più applicazioni
- Vantaggi:
 - Maggiore produttività (throughput)
 - Economia di scala
 - Incremento ed affidabilità
- Inoltre la capacità di poter continuare ad offrire un servizio anche se una delle unità di calcolo è fuori uso permette una maggiore **tolleranza ai guasti**
- I sistemi di elaborazione multiprocessore si distinguono anche in:
 - Asimmetrici – ogni unità elabora un compito specifico
 - Simmetrici – ogni unità ha le stessa capacità degli altri ed è abilitato a svolgere tutte le operazioni del sistema

Struttura del Sistema Operativo



- I Sistemi Operativi possono essere progettati utilizzando
- diversi criteri, i più comuni sono:
 - Multiprogrammazione
 - Time-Sharing (o multitasking)

Multiprogrammazione



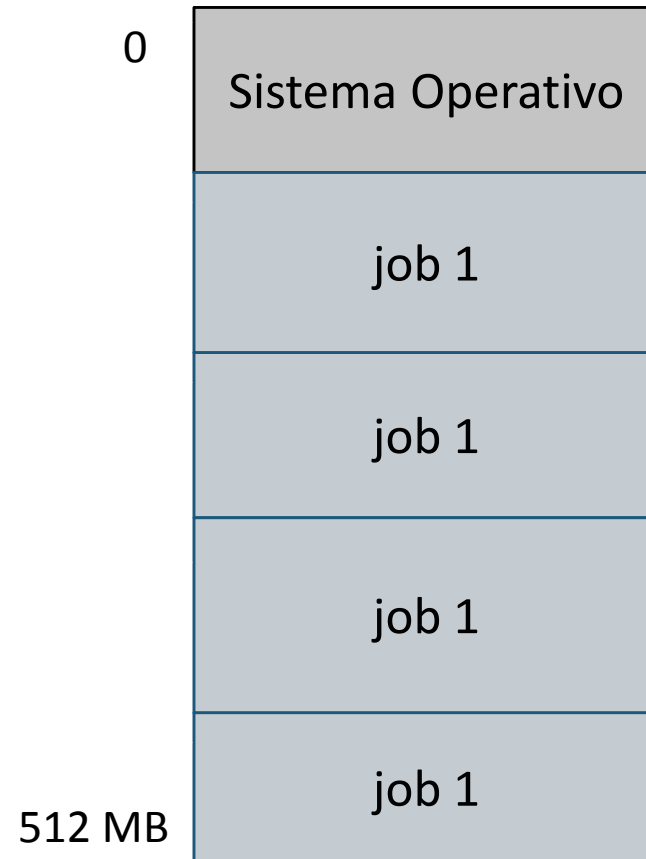
- La **Multiprogrammazione consente di aumentare** l'efficienza della CPU e dei dispositivi di I/O, infatti:
 - Un singolo utente non può tenere occupato la CPU e i dispositivi di I/O per tutto il tempo
 - La multiprogrammazione organizza i **lavori (o jobs)** in modo tale da tenere sempre la CPU occupata
 - Un sottoinsieme dei job totali del sistema risiedono in memoria
 - Un solo job alla volta è selezionato per essere mandato in esecuzione attraverso un algoritmo di scheduling
 - Quando un job è in attesa di un I/O il Sistema Operativo può passare all'esecuzione di un altro job

Timesharing



- Il **Timesharing (o multitasking)** è una estensione logica della multiprogrammazione
- La CPU esegue per un istante un job per poi passare all'esecuzione di un altro job e così via
 - La sensazione è che la CPU sta eseguendo diversi job in parallelo
 - Il **tempo di risposta** deve essere inferiore di 1 secondo
 - Ogni utente ha almeno un programma in esecuzione in memoria chiamato **processo**
 - Se diversi jobs sono in esecuzione nello stesso istante la CPU decide attraverso degli **algoritmi di scheduling** chi deve essere mandato in esecuzione
 - Se un processo non risiede completamente in memoria, lo **swapping** lo trasferisce da o verso una memoria secondaria
 - La **memoria virtuale** permette l'esecuzione di processi che non completamente in memoria

Multiprogrammazione e memoria



Attività del Sistema Operativo



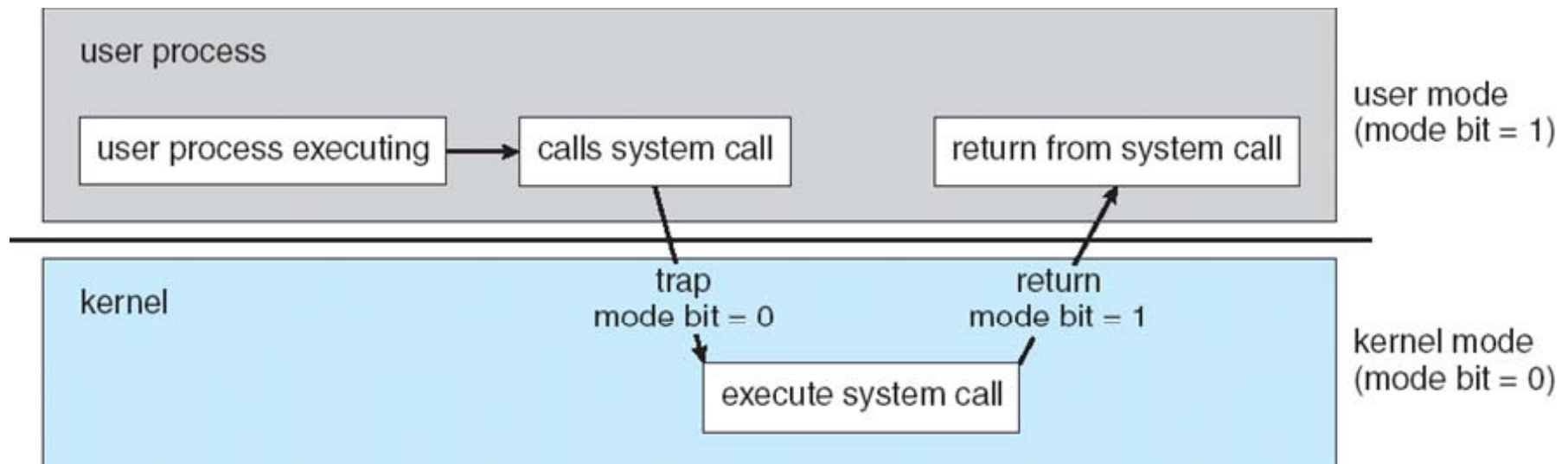
- Doppia modalità di funzionamento
- Gestione dei processi
- Gestione della memoria
- Gestione della memoria di massa
- Sottosistema I/O
- Protezione e Sicurezza

Doppia modalità di funzionamento



- Gli errori generati dal software o da richieste di intervento generano delle **eccezioni o trap**
 - Divisioni per zero o richiesta di intervento da parte di un servizio del Sistema Operativo
- Alcuni problemi possono essere processi che vanno in un ciclo infinito oppure processi che modificano altri processi o il Sistema Operativo
- Il Sistema Operativo gestisce queste situazioni in due modalità
 - **Modalità utente e Modalità di sistema (o kernel)**
 - Un **bit di modalità** fornito dall'hardware permette
 - ✦ Quando il sistema operativo è in esecuzione in una delle due modalità
 - ✦ Per marcare che alcune istruzioni sono privilegiate
 - ✦ Le *system call* al sistema operativo sono eseguite in modalità di sistema

Transizione dalla modalità utente a kernel



Gestione dei processi



- Un processo è un programma in esecuzione e corrisponde ad una unità di lavoro del sistema. Un programma è una **entità passiva**, un processo è una **entità attiva**
- Un processo necessita di risorse per poter compiere il proprio lavoro
 - CPU, memoria, I/O, files
 - Inizializzazione dei dati
- Al termine dell'esecuzione il processo rilascia tutte le risorse utilizzate
- Nei processi single-threaded il **contatore di programma** tiene traccia della prossima istruzione da eseguire
 - Il processo esegui una singola istruzione per volta sino al suo termine
- Nei processi multi-threaded ogni thread ha un suo contatore di programma
- Un sistema solitamente ha diversi processi assegnati a diversi utenti e verranno eseguiti in maniera concorrente su una o più CPU
 - Le CPU sono risorse da utilizzare per eseguire i processi/threads

Attività del gestore dei processi



- Il Sistema Operativo attraverso il **gestore dei processi** ha il compito di:
 - Create e rimuovere i processi degli utenti e di sistema
 - Sospendere e riprendere l'esecuzione dei processi
 - Fornire dei meccanismo per la sincronizzazione dei processi
 - Fornire dei meccanismo per la comunicazione tra processi
 - Fornire dei meccanismo per la gestione dei deadlock

Gestore della memoria



- Il **gestore della memoria** si occupa della gestione dei dati in memoria prima e dopo l'esecuzione di un processo
- In particolare si deve occupare di caricare in memoria le istruzioni del processo per poter avviare la sua esecuzione
- Il gestore della memoria si occupa in particolare di cosa tenere allocato in memoria in ogni istante
 - E' un importante modulo della Sistema Operativo in quanto rende più efficiente l'utilizzo della CPU
- Alcune attività del gestore della memoria sono:
 - Tenere traccia di quale parte della memoria è occupata e da chi è usata
 - Decidere quali processi e dati spostare dentro o fuori la memoria
 - Allocare e deallocare blocchi di memoria quando è richiesto

Gestione dei file



- Un Sistema Operativo deve fornire una visione uniforme e logica delle informazioni memorizzate sui dischi
 - Il Sistema Operativo usa il concetto di **file** per astrarre le proprietà fisiche dell'unità di memorizzazione
 - Ogni unità di massa è diversa dalle altre (ad esempio disk drive, unità di backup)
 - ✦ Ogni device ha caratteristiche diverse come ad esempio velocità, capacità, velocità di trasferimento, metodi accessi(sequenziali o random)
- La gestione della memoria comprende
 - Organizzare i file in directory
 - Controllare gli accessi per verificare chi può accedere a determinati file
- Altre attività includono:
 - Creare e cancellare file e directory
 - Fornire primitive per la gestione dei file e delle directory

Gestione della memoria di massa



- Normalmente la memoria di massa è adoperata per memorizzare informazioni che non possono essere allocate in memoria o che devono essere mantenute per lunghi periodi di tempo
- La gestione dei dati sulla memoria di massa diventa quindi cruciale
- La velocità di un Sistema Operativo è caratterizzate anche da come è gestita la memoria di massa, alcune operazioni cruciali sono:
 - Gestione dello spazio libero
 - Allocazione dello spazio
 - Scheduling del disco
- Altre operazioni non devono essere necessariamente veloci
 - Gli accessi alla memoria terziaria come unità di backup
 - Lettura su dispositivi come memoria di massa RW (read and write)

Sistemi di I/O



- Uno degli scopi di un Sistema Operativo è nascondere la complessità dell'hardware agli utenti
- Il **sottosistema di I/O** ha lo scopo di:
 - Gestire la memoria riservata ai trasferimenti I/O verso i buffer dei dispositivi
 - La gestione della cache e la gestione asincrona delle operazioni di I/O
 - Una interfaccia generale per i driver dei dispositivi
 - I driver per gli specifici dispositivi

Protezione e Sicurezza



- In un Sistema Operativo sono necessari dei meccanismi di **protezione** per controllare l'accesso alle risorse da parte dei processi e degli utenti
- La **sicurezza** è un'altra aspetto che permette ad un Sistema Operativo di fronteggiare eventuali attacchi interni ed esterni
 - Denial-of-service, worms, virus, etc...
- Un sistema operativo normalmente distingue i diversi utenti in modo da poter determinare chi può fare cosa
 - Normalmente ad ogni utente corrisponde un **user ID**
 - Ogni user ID ha quindi i permessi su quali file accedere e su quali file mandare in esecuzione
 - Il **group ID** identifica un insieme di utenti che hanno ulteriori permessi su alcune risorse

Sistemi ad orientamento specifico



- **Sistemi integrati real-time**
 - Sono progettati imponendo dei forti vincoli sui tempi di risposta a carico del processore e del flusso dei dati
- **Sistemi multimediali**
 - Progettati tenendo conto flussi di dati video o audio che devono essere disponibili a velocità costanti (ad esempio un film deve essere riprodotto a 30 fotogrammi al secondo)
- **Sistemi palmari**
 - Sono progettati tenendo in considerazione le scarse risorse
 - Le limitazioni delle periferiche di I/O
 - ...e del consumo della batteria!