

# Calcolo numerico e **programmazione** Sistemi operativi

Tullio Facchinetti  
<tullio.facchinetti@unipv.it>

25 maggio 2012

13:47

<http://robot.unipv.it/toolleeo>

## Sistemi operativi

insieme di programmi che rendono facilmente disponibile all'utente le potenzialità offerte dalla macchina (hardware)

sistema operativo = nucleo (kernel) + software di base

il sistema operativo può essere:

- a pagamento, venduto o meno unitamente alla macchina (es. Microsoft Windows, Apple OS X, UNIX)
- gratuito, liberamente disponibile e installabile (Linux, sistemi BSD)

## Sistemi operativi

- UNIX: AIX (IBM), XENIX (IBM), HP-UX (HP), Ultrix (Digital), Solaris (SUN), SCO (PC), IRIX (Silicon Graphics)
- Linux (PC)
- Mac OS (MAC)
- Windows (95/98/ME, NT, 2000, XP) (PC)
- Windows CE (Palmari e dispositivi portatili)

## Funzioni del sistema operativo

### principali funzionalità:

- gestione dell'esecuzione dei programmi
- gestione delle risorse del computer (tempo di calcolo, memoria, spazio su disco, connessione alla rete)
- ausili per la realizzazione e messa a punto dei programmi

### funzioni di:

- contabilità (conteggio del tempo di elaborazione utilizzato da ciascun utente)
- sicurezza (es. nell'accesso al sistema)

## Il kernel

il kernel (nucleo) del sistema operativo è un programma che va in esecuzione all'accensione della macchina (bootstrapping)

è composto da:

- firmware per l'avvio delle funzionalità di base della macchina
- device driver per la gestione delle periferiche
- il kernel vero e proprio

## Funzioni del kernel

### controllo dell'esecuzione dei programmi dell'utente

- es., controllo dei permessi per l'esecuzione di programmi e l'accesso alle risorse

### gestione del multitasking

- possibilità di avere in memoria più di un programma contemporaneamente
- i programmi vengono eseguiti **a turno** dal processore (scheduling)
- il kernel gestisce il **context switch** (cambio di contesto) tra un programma e l'altro

## Funzioni del kernel

### gestione delle periferiche

- un opportuno software, i device driver, gestiscono l'interazione delle periferiche
- astrazione della periferica (lettura/scrittura/impostazione dei parametri)

### gestione degli interrupt

- sono segnali inviati dalle periferiche alla CPU per segnalare un evento
- es., segnalano l'arrivo di un messaggio dalla rete

## Funzioni del kernel

### amministrazione della memoria

- alloca memoria per le variabili utilizzate dai programmi
- gestisce la protezione della memoria
- gestisce la memoria virtuale (estende la memoria RAM disponibile salvando dati su disco)

### accesso al disco

- gestione del file system (directory e file)
- gestione degli attributi (data/ora di modifica, proprietario, ecc.)
- gestione dei permessi di accesso (lettura/scrittura/esecuzione)



## Overhead introdotto dal kernel

- il kernel, essendo esso stesso un programma, impiega del tempo di CPU per la sua esecuzione
- questo tempo viene sottratto all'esecuzione dei programmi utente
- non può mai essere utilizzato il 100% del tempo di calcolo per i programmi utente
- il tempo di calcolo usato dal kernel è conteggiato come overhead (sovraccarico)

## Funzioni del software di base

il software di base è un insieme di programmi utili a facilitare la creazione di programmi da parte degli utenti della macchina

l'utente si serve del software di base per:

- editare un file (insieme ordinato di dati)
- compilare un programma
- duplicare un file
- cancellare un file
- visualizzare un file
- trasferire un file

## Funzioni del software di base

- utilizzare le periferiche (nastri, stampanti, ...)
- colloquiare con altri utenti
- utilizzare la rete locale
- inviare e ricevere posta
- giocare

## Sistemi operativi: networking

i computer sono collegati in rete e si scambiano informazioni

- i computer interconnessi possono condividere risorse quali tempo di calcolo, file, periferiche (stampanti, scanner, ecc.)
- indispensabile l'uso di protocolli di comunicazione standard
- architettura client/server: un computer server fornisce un servizio al computer client che lo utilizza

## Sistemi operativi: sicurezza

la sicurezza di un computer prevede la protezione dei dati da furti, manomissione, o perdita (per cancellazione intenzionale fino a disastri naturali)

le minacce alla sicurezza di un computer derivano da:

- virus o altre forme di “malware” (trojan, backdoor, rootkit, keylogger)
- attacchi informatici (es. Denial of Service, DoS)
- social engineering (l’anello debole della catena è spesso l’uomo)

## Sistemi operativi: interfaccia utente

è l'interfaccia che permette all'utente di richiedere servizi al sistema operativo

- non è formalmente parte del sistema operativo
- tipica interfaccia è la shell dei sistemi UNIX
- interazione con l'utente mediante tastiera e terminale (video)

## Sistemi operativi: interfaccia grafica

i sistemi moderni facilitano l'interazione con l'utente  
mediante la GUI (Graphical User Interface)

- interfaccia composta da finestre, bottoni cliccabili, e altri elementi grafici
- prima implementazione all'interno del sistema operativo Mac OS
- evoluzione continua delle interfacce

## Classificazione dei sistemi operativi

- dedicati
- a lotti (batch)
- multiprogrammazione
- interattivi (time-sharing)
- real-time
- distribuiti
- embedded



## Sistemi operativi dedicati

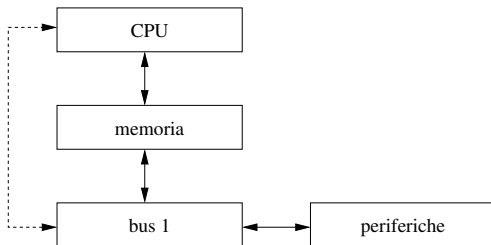
- i primi sistemi operativi
- la macchina viene utilizzata da un utente per volta che può eseguire un solo programma per volta (applicativo o di base)
- kernel molto semplice

## Sistemi operativi per gestione a lotti (batch)

- nati per meglio sfruttare la velocità crescente delle macchine: eliminano i tempi morti tra programmi successivi di utenti diversi
- un insieme di lavori (jobs) viene accorpato in un lotto (batch) e trasferito su una unità di ingresso veloce (nastro o disco)
- ogni lavoro viene caricato ed eseguito in sequenza senza interruzioni fino al termine
- la CPU viene comunque sottoutilizzata perché, durante le operazioni di I/O, deve adeguarsi alla bassa velocità delle periferiche

## Multiprogrammazione

- più di un programma è caricato in memoria contemporaneamente
- le operazioni di I/O di un programma sono sovrapposte temporalmente all'esecuzione delle istruzioni di un altro programma
- ciò è possibile grazie al bus condiviso che gestisce l'I/O



## Multiprogrammazione

- quando un processo in esecuzione (running) chiede un servizio di I/O al sistema operativo si blocca in attesa del risultato
- il sistema operativo avvia l'operazione di I/O e manda in esecuzione un altro processo che si trova in stato di pronto (ready)
- il sistema operativo mantiene una o più code di processi gestite secondo la priorità assegnata a ciascuno di essi (scheduling)

## Vantaggi e svantaggi

### vantaggi

- CPU ben sfruttata

### svantaggi

- incapacità di differenziare fra programmi che richiedono un uso frequente delle periferiche, da quelli che richiedono l'utilizzo per lunghi tempi della CPU (es. programmi di elaborazione matematica su dati ricevuti inizialmente)
- quando uno di questi programmi conquista la CPU non la rilascia, bloccando gli altri programmi

## Sistemi operativi interattivi (time sharing)

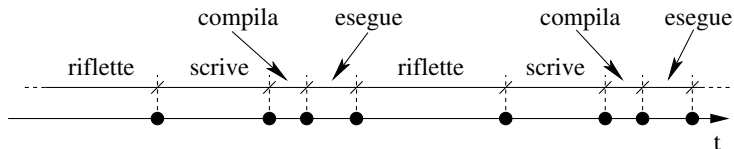
- il tempo di utilizzo della CPU viene suddiviso dal sistema operativo in fette (time slice con durata di 100-800 ms)
- ogni processo in memoria riceve a turno l'uso della CPU per una fetta di tempo
- usati soprattutto per lo sviluppo di software in ambiente multiutente

## Sistemi operativi interattivi (time sharing)

- più utenti possono collegarsi contemporaneamente alla stessa macchina
- c'è un solo computer e molti terminali (tastiere e monitor), uno per ciascun utente
- ciascun utente che lavora al terminale e ha l'impressione di avere la macchina a sua completa disposizione
- ogni processo ottiene ciclicamente l'utilizzo della CPU
- elevato overhead per la gestione della CPU con molti processi attivi

una unica macchina può essere usata da più utenti contemporaneamente grazie all'assegnamento a turno della CPU durante i "tempi morti"

## Sistemi operativi interattivi (time sharing)



- mentre riflette l'utente non usa il computer, la cui CPU può essere assegnata ad un altro utente
- la scrittura usa poco la CPU, visto che l'utente è molto più lento a battere i tasti della macchina: il tempo tra una pressione di tasti e l'altra può essere assegnato ad altri utenti
- la compilazione e l'esecuzione richiedono un uso intensivo della CPU, ma hanno una durata relativamente breve rispetto al totale



## Il processo

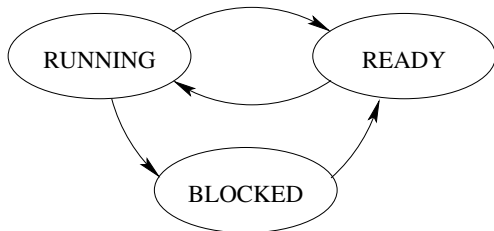
un processo è una sequenza di istruzioni eseguite dal processore per eseguire un programma

- ogni programma è costituito da uno o più processi
- la CPU viene assegnata a turno ai vari processi attivi
- lo **scheduler** è il componente del kernel che si occupa di assegnare la CPU al processo avente la priorità maggiore

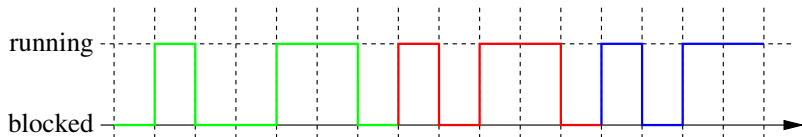
## Stati di un processo

un processo è caratterizzato dal suo **stato**, che può essere:

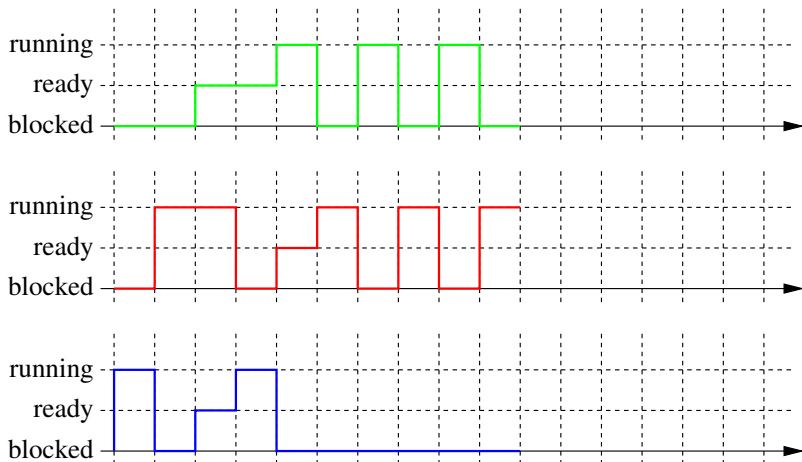
- **RUNNING**: il processo è attualmente in esecuzione sulla CPU
- **READY**: il programma è pronto per l'esecuzione ma sta attendendo che un processo a priorità maggiore della sua liberi la CPU
- **BLOCKED**: il processo è bloccato in attesa che si verifichi un evento



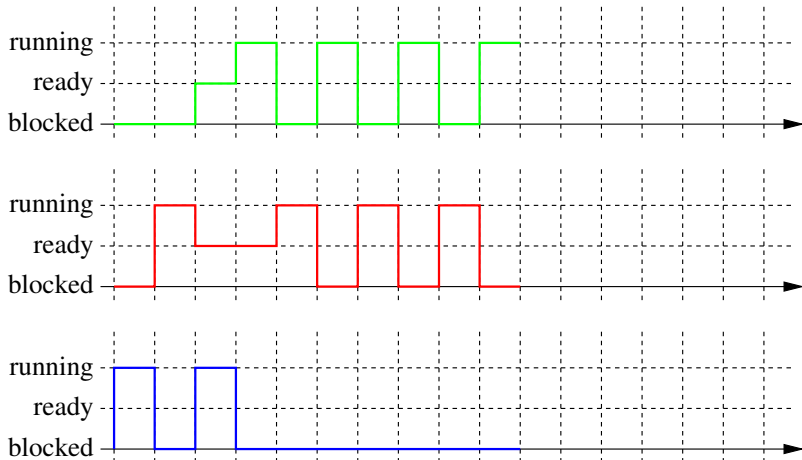
## Stato dei programmi: OS dedicato



## Stato dei programmi: OS multiprogrammato



## Stato dei programmi: OS timesharing



## Sistemi operativi real-time

- sistemi operativi al servizio di una specifica applicazione che ha dei vincoli precisi nei tempi di risposta
- il sistema operativo deve garantire un tempo massimo entro il quale mandare in esecuzione un programma a seguito di una richiesta in tal senso
- in generale si ha un sistema real time quando il tempo che passa dalla richiesta di esecuzione di un processo al completamento della stessa è minore del tempo fissato

### impieghi:

- gestione di strumentazione
- controllo di processo (acquisizione di sensori e pilotaggio di attuatori)
- gestione di allarmi
- sistemi transazionali (banche, prenotazione)

## Sistemi operativi distribuiti

- gestisce un insieme di computer indipendenti
- permette all'utente di usare l'insieme di computer come fossero solo uno
- richiedono la interconnessione tra le macchine e lo scambio di informazioni
- il calcolo distribuito avviene in parallelo su più di una macchina contemporaneamente
- formano un cosiddetto “sistema distribuito”

## Cosa è un sistema embedded

- un sistema embedded è un apparato computerizzato per la gestione di specifiche funzioni in un sistema complesso
- **embedded** significa che il processore è integrato nel sistema, insieme a componenti hardware, elettriche e meccaniche



## Sistemi operativi embedded

- sviluppati per gestire un **sistema embedded**
- pensati per macchine piccole e spesso funzionanti a batteria
- devono gestire nel migliore dei modi risorse (potenza di calcolo, memoria, energia) molto limitate