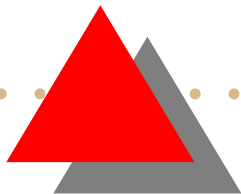




# Sistemi in tempo reale: applicazioni alla robotica



# *Sistemi operativi*

Hardware

---

Firmware

---

Kernel

---

Driver

---

Applicazioni

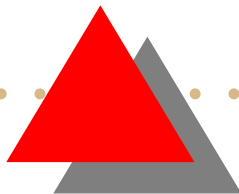
---

Interfacce



# *Compiti del sistema operativo*

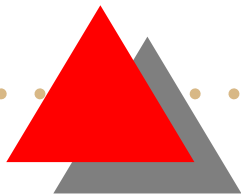
- gestisce le risorse hw/sw del sistema di calcolo
  - processore, memoria, spazio disco, periferiche
- fornisce alle applicazioni un modo stabile e consistente per interagire con le risorse senza dover conoscere tutti i dettagli implementativi
  - anche all'utente!





# *Tipologie di OS*

- Single-user, single task
- Single-user, multi-tasking
- Multi-user
- Real-time



# *Gestione del processore*

- programma, processo, thread, task
- scheduler
  - batch
  - priorità
  - time-sharing
- interrupt
  - mascherabilità
  - stack
  - vettore degli interrupt



# *Gestione delle risorse*

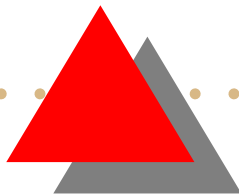
Alcune risorse sono riutilizzabili, ma vi sono di natura seriale

Bisogna assicurare:

- mutua esclusione
- assenza di deadlock

Meccanismo:

- semaforo

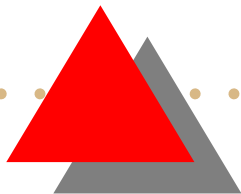




# Concorrenza

Collaborazione tra processi che accedono alle stesse risorse per la realizzazione di compiti complessi.

- Esempio: produttore/consumatore
- sincronizzazione
- azioni asincrone / non determinismo
- competizione per le risorse comuni
- cooperazione



# *Concorrenza* (2)

Meccanismi di comunicazione:

- Memoria condivisa
- Scambio di messaggi



# *Sistemi in tempo reale*

Esecuzione di un compito in tempo reale

- per tale compito è ben determinata un istante di tempo entro il quale il compito deve essere eseguito (*deadline*)
- il sistema che lo esegue è in grado di assicurare il rispetto di questo vincolo
  - correttezza dei calcoli
  - rispetto dei vincoli temporali



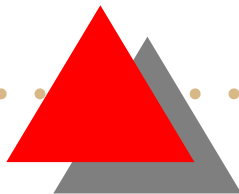
# *Sistemi in tempo reale* (2)

- velocità vs. prevedibilità
- soft real-time vs. hard real-time

Esempi:

**soft real-time** Videoregistratore, distribuzione di messaggi, sistema di progettazione;


**hard real-time** Sistema di pilotaggio automatico, sistema di controllo di un impianto nucleare, sistema di controllo di un'apparecchiatura medica.





# *Prevedibilità*

La prevedibilità del sistema è influenzata da diversi fattori:

- Caratteristiche architettureali della macchina fisica: interrupt, DMA, cache.
    - migliori prestazioni medie del processore
    - peggiore prevedibilità
  - Meccanismi del kernel: algoritmo di scheduling, semafori, gestione delle risorse (memoria e periferiche).
  - Linguaggio di programmazione: costrutti per la gestione esplicita dei vincoli temporali.
- 

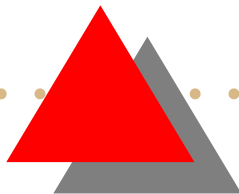


# *Scheduler*

Il meccanismo di scheduling è quello che maggiormente influisce sulla prevedibilità del sistema.

I processi possono essere divisi in due categorie:

- processi periodici
- processi aperiodici





# *Processi periodici*

Alcuni processi devono essere periodicamente attivati.

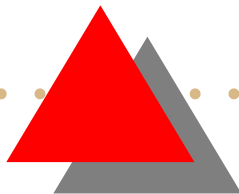
Vari algoritmi di scheduling per trattare i processi periodici:

**Timeline scheduling**

**Rate monotonic**

**Earliest deadline first**

**Deadline monotonic**



# *Processi aperiodici*

**gestione immediata**

**gestione in background**

**server**

**a deadline**



# *Risorse condivise*

- problema: dell'inversione di priorità
- soluzione: ereditarietà della priorità

