

# Informatica Generale

## Andrea Corradini

---

### **08 - Competizione su risorse e deadlock**

# Comunicazione/sincronizzazione di processi

- Quando ci sono più processi in esecuzione, essi possono:
  - **cooperare** per raggiungere un obiettivo, e/o
  - **competere** per l'uso di risorse
- Per **cooperare** i processi devono scambiarsi informazioni. Questo può avvenire in due modi principali:
  - tramite memoria condivisa
  - con scambio di messaggi
- Queste problematiche vengono affrontate nella **programmazione concorrente**

# Gestione della competizione per risorse

- Il Sistema Operativo deve gestire le situazioni in cui più processi vogliono accedere alla stessa risorsa, fisica o logica (es: un file, una stampante, ...)
- Una semplice tecnica: associare un **flag (bit)** alla risorsa
  - se il **flag** è nello stato **set (1)** la risorsa è **occupata**
  - se il **flag** è nello stato **clear (0)** la risorsa è **libera**
- Quando un processo vuole accedere alla risorsa:
  - se il **flag** è **clear**,
    - il **flag** viene messo a **set**
    - la risorsa viene assegnata
    - alla fine del suo uso il **flag** viene messo a **clear**
  - altrimenti il processo va in attesa finché il **flag** non è **clear**

# Gestione della competizione per risorse

- Il Sistema Operativo deve gestire le situazioni in cui più processi vogliono accedere alla stessa risorsa, fisica o logica (es: un file, una stampante, ...)
- Una semplice tecnica: associare un **flag (bit)** alla risorsa
  - se il **flag** è nello stato **set (1)** la risorsa è **occupata**
  - se il **flag** è nello stato **clear (0)** la risorsa è **libera**
- Quando un processo vuole accedere alla risorsa:
  - se il **flag** è **clear**,
    - il **flag** viene messo a **set**
    - la risorsa viene assegnata
    - alla fine del suo uso il **flag** viene messo a **clear**
  - altrimenti il processo va in attesa finché il **flag** non è **clear**

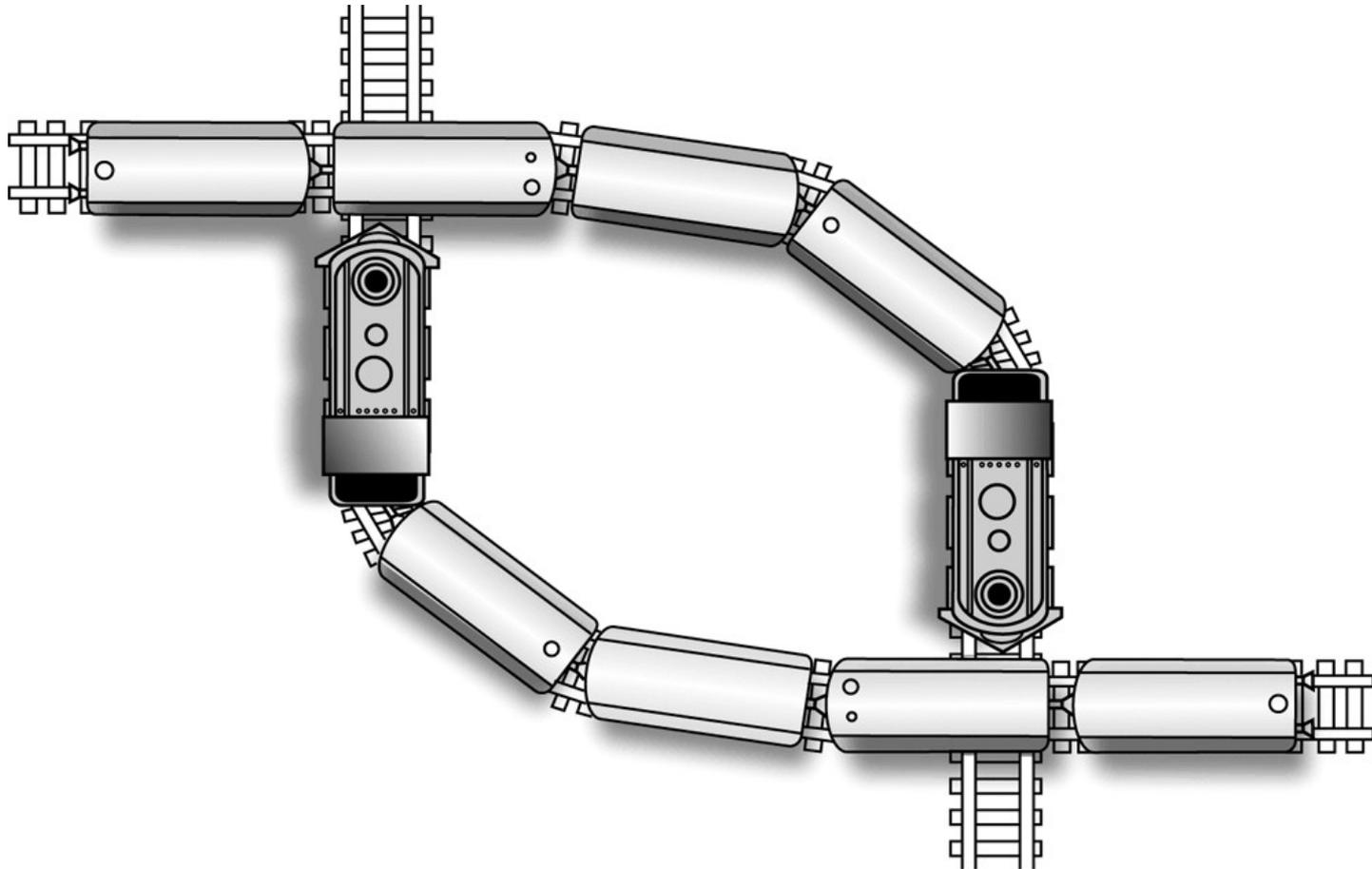
**Attenzione: funziona solo se queste due istruzioni non possono essere interrotte**

# Deadlock (stallo)

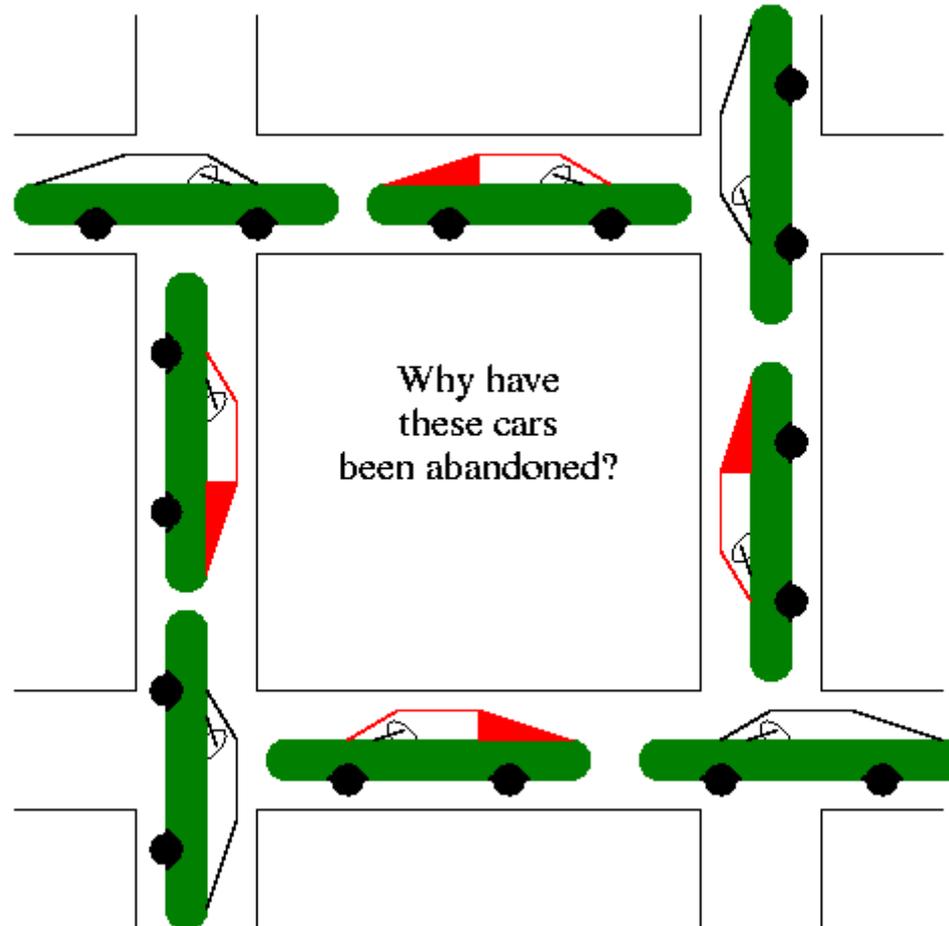
- Il deadlock si verifica quando due processi sono in attesa l'uno dell'altro
  - Es: uno accede alla stampante, ma è in attesa del lettore CD, l'altro accede al CD e aspetta la stampante
- **Condizioni necessarie** perché si verifichi un deadlock
  1. Le risorse per cui si compete non sono condivisibili (ognuna può essere allocata ad un solo processo alla volta)
  2. Le risorse sono richieste “un po' alla volta”
  3. Una risorsa allocata non può essere liberata con la forza
- Se una di queste condizioni è falsa, non c'è deadlock.

# Deadlock: un esempio

Quali sono le risorse? Come si può sbloccare?



# Deadlock: un altro esempio





# Esercizio

---

- Quale condizione viene meno con le seguenti proposte per evitare il deadlock su un ponte a una corsia?
  - Non lasciare entrare una macchina sul ponte finché non è vuoto;
  - Se due macchine si incontrano, una delle due deve fare retromarcia;
  - Aggiungere una nuova corsia al ponte.