

# Deadlock

Arresto di un programma dovuto ad uso improprio dei lock. È dato da situazioni di dipendenza circolare.

Approcci risolutivi:

**prevention** controlli statici sul corretto utilizzo dei mutex;

**avoidance** capacità del RTS di stabilire se sta per verificarsi un deadlock e modificare l'ordine di esecuzione di conseguenza;

**recovery** il RTS ripristina uno stato senza deadlock quando se ne verifica uno.

## 2-phase locking

Disciplina di uso dei lock che previene deadlock:

- stabilire un ordinamento dei mutex;
- acquisire sempre in ordine crescente;
- rilasciare in ordine decrescente;
- una singola fase di lock in e una di unlock in blocco (non si intrecciano lock e unlock).

## Esempio

Programma che può causare deadlock:

```
(lock m; lock n; l := 0; unlock m; unlock n)
| (lock n; lock m; l := 1; unlock n; unlock m)
```