



ISTITUTO DI ISTRUZIONE SUPERIORE
"Buonarroti – Volta"

Tecnologie e Progettazione di Sistemi Informatici e Telecomunicazioni

4P

Programma svolto al 08/06/2026

1 PROCESSI SEQUENZIALI E PARALLELI

1.1 Il modello a processi

- Il modello a processi
- Stato dei processi
- Sospensione per interrupt
- Comandi per la creazione, sospensione e terminazione dei processi
- PCB (Process Control Block)

1.2 Risorse e condivisione

- Generalità
- Condivisione e gestione
- Classificazioni
- Classificazione delle richieste
- Classificazione dell'assegnazione
- Classificazione delle risorse
- Grafo di Holt
- Riducibilità di un grafo di Holt

1.3 I thread o "processi leggeri"

- Generalità
- Processi "pesanti" e "processi leggeri"
- Processi pesanti
- Processi leggeri
- Soluzioni adottate: single threading vs multithreading
- Realizzazione di sistemi multithreading
- User-Level
- Kernel-Level
- Soluzione mista
- Stati di un thread
- Utilizzo dei thread

1.4 L'elaborazione concorrente

- Generalità
- Processi non sequenziali e grafo delle precedenze
- Scomposizione di un processo non sequenziale
- Processi indipendenti
- Processi interagenti

1.5 La descrizione della concorrenza

- Esecuzione parallela
- Elaboratore non sequenziale
- Linguaggi non sequenziali
- Fork-join
- Fork
- Join
- Join (count)
- Cobegin-coend
- Equivalenza tra fork-join e cobegin-coend

2 COMUNICAZIONE E SINCRONIZZAZIONE

2.1 La comunicazione tra processi

- Introduzione
- Modello a memoria comune (ambiente globale)
- Allocazione delle risorse ai processi
- Tipologie di allocazione delle risorse nel modello ad ambiente globale
- Competizione
- Cooperazione
- Interferenza
- Modello a scambio di messaggi (ambiente locale)
- Modello client-server

2.2 La sincronizzazione tra processi

- Errori nei programmi concorrenti
- Definizioni e proprietà
- Overlapping e interleaving
- Condizioni di Bernstein
- Mutua esclusione e sezione critica
- Starvation e deadlock

2.3 I semafori

- Quando è necessario sincronizzare

- Semafori di basso livello e spinlock
- Allocazione di una risorsa: lock()
- Rilascio di una risorsa: unlock()
- Problema della indivisibilità
- Semafori di Dijkstra
- Semafori binari vs semafori di Dijkstra
- Molteplicità di una risorsa

2.4 Applicazione dei semafori

- Semafori e mutua esclusione
- Mutua esclusione tra gruppi di processi
- Semafori come vincoli di precedenza
- Problema del rendez-vous

2.5 Il problema dei produttori/consumatori

- Generalità
- Problema dei produttori/consumatori
- Un produttore, un consumatore e una singola cella di memoria
- Un produttore e un consumatore e buffer circolare

2.6 Il problema dei lettori/scrittori

- Problema dei lettori e degli scrittori
- Prima soluzione

2.7 Il problema del deadlock

- Perché si genera un deadlock
- Individuazione dello stallo
- Come affrontare lo stallo
- Individuare ed eliminare lo stallo: detection e recovery
- Evitare lo stallo: avoidance
- Prevenire lo stallo: prevention
- Ignorare il problema
- Esempio classico: problema dei filosofi a cena

Laboratorio

1 LINUX

- Virtualizzazione di una macchina con sistema operativo Windows
- Ambiente Linux e principali comandi

2 PROGRAMMAZIONE PARALLELA

- L'esecuzione di processi
- L'istruzione fork-join
- Le funzioni wait() e waitpid()

Gli alunni

I docenti

Deiana Mauro

Zanda Vincenzo
