

Programma di Tecnologie e Progettazione di Sistemi Informatici e di Telecomunicazioni

Classe 4P Anno scolastico 2018/2019

Docenti: Eleonora Minelli, Donato Tartaglione

1. Processi Sequenziali e Paralleli
 - 1.1 Il modello a processi
 - 1.2 Stato dei processi
 - 1.3 Process Control Block (PCB)

2. Risorse e condivisione
 - 2.1 Aspetti generali
 - 2.2 Classificazioni delle richieste e dell'assegnazione e delle risorse
 - 2.3 Grafi di Holt

3. I Thread o processi leggeri
 - 3.1 Definizione di un Thread
 - 3.2 Diagramma di stato di un Thread
 - 3.3 Single threading e multithreading
 - 3.4 Realizzazione di un Thread : user-level e kernel-level
 - 3.5 Utilizzo dei Thread

4. Elaborazione sequenziale e concorrente
 - 4.1 Definizione di elaborazione sequenziale e programmazione concorrente
 - 4.2 Processi non sequenziali e grafo di precedenza
 - 4.3 Scomposizione di un processo non sequenziale
 - 4.4.1 Definizione di scomposizione sequenziale e processi indipendenti
 - 4.4.2 Processi indipendenti
 - 4.4.3 Processi interagenti

5. Comunicazione tra processi
 - 5.1 Modello a memoria comune
 - 5.1.1 Allocazione delle risorse ai processi
 - 5.1.2 Tipologie di allocazione delle risorse nel modello ad ambiente globale
 - 5.1.3 Mutua esclusione
 - 5.1.4 Competizione
 - 5.1.5 Cooperazione
 - 5.2 Modello a scambio di messaggi
 - 5.2.1 Modello client server

6. La sincronizzazione tra processi
 - 6.1 Differenza tra programma sequenziale e programma concorrente
 - 6.2 Errori dipendenti dal tempo
 - 6.3 Interleaving e overlapping
 - 6.4 Condizioni di Bernstein: definizione ed esempi
 - 6.5 Mutua esclusione e sezione critica
 - 6.6 Starvation e deadlock

7. Sincronizzazione tra processi
 - 7.1 Meccanismi per l'accesso alla sezione critica

7.1.1 Spin lock

7.1.1.1 Lock()

7.1.1.2 Unlock()

7.1.2 Semafori di Dijkstra

7.1.2.1 Definizione

Le esercitazioni pratiche sono state svolte in linguaggio Java in ambiente NetBeans affrontando con numerosi esercizi i seguenti punti:

- I thread in java: concetti base
- Thread come sottoclasse della classe Thread e il metodo run()
- Interfaccia Runnable
- Metodi dei Thread: start(), stop(), join(), yield(), sleep(), suspend(), resume().
- Synchronized

Modalità di recupero: Prova scritta.

Roma, 04/06/19

Prof.ssa Eleonora Minelli

Prof. Donato Tartaglione