



UNIONE EUROPEA

FONDI
STRUTTURALI
EUROPEI

pon
2014-2020



MIUR

Ministero dell'Istruzione, dell'Università e della Ricerca
 Dipartimento per la Programmazione
 Direzione Generale per interventi in materia di edilizia
 scolastica, per la gestione dei fondi strutturali per
 l'istruzione e per l'innovazione digitale
 Ufficio IV

PER LA SCUOLA - COMPETENZE E AMBIENTI PER L'APPRENDIMENTO (FSE-FESR)

PROGRAMMA CONSUNTIVO

Codice Mod. **RQ 23.2** Pag. 1 / 3

A.S.	2019-2020		
DOCENTE	ROSARIO FRASCA CACCIA – DAVIDE BARBARIA		
DISCIPLINA	TPSIT		
CLASSE	4H	INDIRIZZO	INFORMATICA E TELECOMUNICAZIONE ART. INFORMATICA

	TITOLO DEL MODULO	Contenuti Svolti
1	Il modello a processi	Contenuti teorici - Scheduling e multiprogrammazione. - Processi cooperanti e processi in competizione. - Stato di un processo e diagramma degli stati. - Il PID. Il context switching. Il Process Control Block (PCB). - reazione, sospensione e terminazione dei processi nei sistemi *NIX like (fork, exit, wait). - Algoritmi di Scheduling: FCFS, SJF, con Priorità, Round Robin, con code multiple a retroazione. Attività di Laboratorio - Esercitazioni pratiche volte a calcolare il tempo medio di accesso dei processi con i principali algoritmi di scheduling. - Introduzione al linguaggio Java. - Test pc java virtual machine
2	Risorse e condivisione	Contenuti teorici - Definizione di risorsa Classi e istanze di risorse. Condivisione e gestione. - Classificazioni in base al tipo di richiesta, alla modalità di assegnazione, alla tipologia delle risorse. - Grafo di Holt o grafo di allocazione delle risorse, rappresentazione delle risorse e dei processi. Significato del verso degli archi che connettono processi e risorse. Grafi riducibili. Attività di Laboratorio - Esercitazioni pratiche volte a rappresentare processi e risorse tramite grafi di Holt. - Esercitazioni pratiche volte a ridurre un grafo di Holt. - Le classi in Java. Metodi e attributi.
3	I thread o "processi leggeri"	Contenuti teorici <i>Processi pesanti e processi leggeri</i>, loro pregi e loro difetti. - Definizione di thread. Thread safety. - Realizzazione dei thread user-level kernel-level. Thread POSIX. Stati di un thread. - Utilizzo dei thread.

		Attività di Laboratorio • Test Machina Java da linea di comando, impostazione var di ambiente. • Primo prog. in java. • Fork annidate ed esecuzione non deterministica.
4	Elaborazione sequenziale e concorrente	Contenuti teorici • Processi non sequenziali e grafo delle precedenze. • Scomposizione di un processo non sequenziale. • Processi indipendenti e processi interagenti. • Processi in competizione. Attività di Laboratorio • Interfaccia grafica Eclipse, Intro ai tipi di variabile in Java. • Esercitazioni pratiche sui grafi delle precedenze. • Vettori e matrici in Java. • Le funzioni wait() e waitpid().
5	La descrizione della concorrenza	Contenuti teorici • Esecuzione parallela. • Fork-join e Cobegin-coend. Equivalenza di fork-join e cobegin-coend. • Semplificazione delle precedenze Attività di Laboratorio • Esercizi guidati in Java con l'uso di classi. • Fork-join e cobegin-coend.
6	La comunicazione tra processi	Contenuti teorici • La comunicazione: modelli software e hardware. • Il modello a memoria condivisa allocazione delle risorse ai processi e loro tipologia competizione e cooperazione Attività di Laboratorio • Il metodo costruttore in Java • La classe Cerchio
7	La sincronizzazione tra processi ed i semafori*	Contenuti teorici • Errori nei programmi concorrenti. • Condizioni per avere la concorrenza: interleaving e overlapping. • Condizioni di Bernstein, Definizione di dominio e rango di una istruzione o procedura. • Mutua esclusione e sezione critica. • Starvation e deadlock. • Semafori di basso livello e spin lock. Allocazione di una risorsa: lock(). Rilascio di una risorsa: unlock(). Problema della indivisibilità. • Semafori di Dijkstra. Le primitive P e V. • Semafori e loro applicazioni. • Mutua esclusione tra gruppi di processi. • Semafori come vincoli di precedenza. Attività di Laboratorio • Utilizzo di software open-source per realizzare diagrammi UML (Whitestar) • I thread in Java. Priorità e parametri nei thread Java. • I thread in Java: sleep, yield e join.

8	Problemi classici della programmazione concorrente: produttori/consumatori, lettori/scrittori*	Contenuti teorici • Problema produttore/consumatore. • Risoluzione del problema in pseudocodifica nel caso in cui il dato prodotto sia "singolo" i semafori "pieno" e "vuoto" e loro inizializzazione. • Risoluzione del problema in pseudocodifica nel caso in cui la memoria condivisa possa contenere più dati (buffer circolare). Attività di Laboratorio • La comunicazione tra processi mediante segnali asincroni. • La soluzione del problema produttore consumatore con i semafori classici in java
---	--	---

STRUMENTI/SUSSIDI DIDATTICI

• Libro di testo: Camagni Paolo / Nikolassy, Tecnologie e Progettazione di Sistemi Informativi e di telecomunicazione - HOEPLI • Presentazioni e dispense • Laboratorio di informatica: eclipse • Piattaforma e-learning di istituto: condivisione dei materiali aggiuntivi, assegnazione e salvataggio delle esercitazioni di laboratorio e degli esercizi per lo studio individuale • Piattaforma di videoconferenze: WeBex, Zoom

VERIFICHE E VALUTAZIONI	ATTIVITÀ DI RECUPERO	NOTE
• Verifiche scritte cartacee • Verifiche orali • Attività di laboratorio	• Studio individuale • Correzione attività svolte • Lezioni di recupero • Verifica di recupero	Dal 24 febbraio tutte le lezioni sono state svolte in modalità di "Didattica a Distanza" a causa delle misure prese dal governo per contrastare l'emergenza sanitaria dovuta al coronavirus

BOLLATE, Lì 06/06/2020

STUDENTI

DOCENTI

ROSARIO FRASCA CACCIA

DAVIDE BARBARIA