



PROGRAMMAZIONE DIDATTICA

Mod. MQ08A03 Rev. 1

A.S. 2025/26

Pag. 1/2

CONTENUTI

Anno Scolastico	2025/26	Classe	4	Sez.	F
Disciplina	Tecnologie e Progettazione di Sistemi Informatici e di Telecomunicazioni				
Docenti	Marco Mastantuono				
	Alessia Troiano				



PROGRAMMAZIONE DIDATTICA

Mod. MQ08A03 Rev. 1

A.S. 2025/26

Pag. 2/2

SEZ. 5 – CONTENUTI SVOLTI

Processi sequenziali e paralleli

Evoluzione delle CPU. Elaborazione sequenziale e parallela. Throughput. Gestione del processore. Modello a processi. Programma. Processo. Multiprogrammazione. Memoria del processo. Parte statica e parte dinamica. Codice, dati globali, stack e heap. Interazione fra i processi. Processi cooperanti e in competizione. Stato di un processo. Assegnazione di un processo alla CPU. Grafo degli stati di un processo. Casi di terminazione di un processo. Sospensione per Interrupt. Contesto del processo. Context Switching. Process Control Block (PCB). PID. Process table. Gerarchia dei processi. Processi padre e figlio. Processo init. Creazione di processi figlio. Flusso nell'esecuzione parallela del codice. Fork annidate. Processi zombi e orfani. Albero dei processi. Esecuzione non deterministica. Risorsa. Condivisione delle risorse. Classi di risorse. Istanza e molteplicità di una risorsa. Richiesta e assegnazione di una risorsa. Pianificazione e controllo dell'accesso alle risorse e possibili problematiche. Classificazione delle risorse. Grafo di Holt. Riducibilità di un grafo di Holt. Processi pesanti e processi leggeri. Thread. Thread Control Block (TCB). Thread safety. Single threading e multithreading. Thread User-Level e Kernel-Level. Thread POSIX. Stati di un thread. Utilizzo dei thread. Passaggio di dati mediante condivisione di memoria. Problemi di errato utilizzo di variabili globali. Elaborazione sequenziale. Programmazione concorrente. Grafo delle precedenze. Processi indipendenti e interagenti. Costrutti dei programmi non sequenziali. Fork-Join. Cobegin-coend. Semplificazione delle precedenze. Emulatore Cygwin. Installazione e configurazione di Cygwin. Principali comandi della shell Linux per la gestione di file e directory. GCC e comandi/opzioni di compilazione. Memoria dinamica in C (stack e heap). Funzioni malloc e free. API C/UNIX per la creazione e la gestione di processi. Funzione fork. Funzioni wait/waitpid, exit e sleep. Funzioni per ottenere il PID di un processo. I thread in C. Libreria pthread. Funzioni per la creazione, terminazione, cancellazione e attesa di un thread. Passaggio di parametri a un thread. Parametro di ritorno da thread a chiamante. Esercitazioni sulla creazione e la gestione di processi e thread con API C/Unix.

Comunicazione e sincronizzazione tra processi

La comunicazione tra processi. Modello a memoria comune, a scambio di messaggi, client-server. Competizione, cooperazione e interferenza. Errori e problematiche nei programmi concorrenti. Race condition. Interleaving e overlapping. Mutua esclusione. Starvation e deadlock. Safety e liveness. Sezione critica. La sincronizzazione tra processi. I semafori. Semafori binari (spinlock). Lock e unlock. Problema dell'indivisibilità. Applicazione dei semafori. Produttore/consumatore. Individuare uno stallo. Affrontare uno stallo. Problema dei filosofi a cena. Scambio di messaggi fra processi. Comunicazione tramite pipe. Comunicazione asincrona fra processi. I segnali. Libreria signal. Handler del segnale. I semafori binari in C. Mutex: creazione/distruzione, lock e unlock. Esercitazioni sull'utilizzo di risorse condivise nei programmi concorrenti. Esercitazioni sull'utilizzo dei mutex in linguaggio C. Esercitazione sull'invio e la gestione di segnali nella comunicazione fra processi.

Ciclo di vita di un progetto software

Ciclo di vita e modelli per lo sviluppo software. Ambiente di sviluppo integrato (IDE). Visual Studio Code. Raccolta, analisi e specifica dei requisiti. Classificazione dei requisiti. Requisiti utente e di sistema. Requisiti funzionali, non funzionali e di dominio. Problemi nella definizione dei requisiti. Attori, casi d'uso e scenari. Descrizione dei casi d'uso. Relazioni fra casi d'uso. Il testing del software. Verifica e validazione. Debugging. Documentazione del software. Strumenti per la documentazione automatica. Il controllo di versione. VCS centralizzati a distribuiti. Git. Repository. Commit. Storico delle versioni. Esercitazioni su definizione e analisi dei requisiti software. Esercitazioni sulla realizzazione di diagrammi UML per la descrizione dei casi d'uso e le loro relazioni.

SEZ. 11 – TESTI E MATERIALI UTILIZZATI ¹

Libro di testo in adozione: Camagni, Nikolassy – "TECNOLOGIE E PROGETTAZIONE DI SISTEMI INFORMATICI E DI TELECOMUNICAZIONI 2" - Hoepli
Materiali d'approfondimento pubblicati sulla piattaforma di e-learning.

¹ Libro di testo in adozione, materiali didattici (anche multimediali) utilizzati