



A.S. 2020/2021

PERCORSO FORMATIVO DISCIPLINARE

Classe	4 DS A
Disciplina	Tecnologie e Progettazione di Sistemi Informatici e di Telecomunicazioni (TPSIT)
Docenti	Prof.ssa Maria Rita Di Berardini, Prof. Luigi Marano.

**OBIETTIVI DIDATTICI DISCIPLINARI**

La disciplina "TPSIT" concorre a far conseguire allo studente al termine del percorso quinquennale i seguenti risultati di apprendimento relativi al profilo educativo, culturale e

- Processi e Thread;
- Concorrenza e problematiche connesse alla sincronizzazione dei processi e thread;
- Implementazione di processi in C;
- Thread in Java e problemi classici della concorrenza;
- La documentazione del software e Javadocs.

**METODOLOGIA DI LAVORO**

La metodologia didattica ha previsto l'approccio ai diversi contenuti partendo dall'illustrazione di problemi connessi con la realtà, analizzati criticamente e alla luce delle conoscenze attuali. Gli alunni sono stato stimolati ad esprimere le proprie opinioni e a confrontarsi (tra di loro e con i docenti) in maniera serena e costruttiva. Si è fatto ricorso alla metodologia del "problem solving" e alla "didattica laboratoriale" e alla "didattica per problemi", abituando gli alunni a utilizzare e riorganizzare le conoscenze già acquisite per formulare ipotesi di risoluzione di un problema, in modo da fare acquisire loro una metodologia scientifica.

**CONTENUTI (ORGANIZZATI IN MODULI DIDATTICI)**

**MODULO 1: PROCESSI E CONCORRENZA**

**Processi e risorse:**

- I processi, gli stati di un processo e il process control block (PCB).
- Le risorse e l'assegnazione di risorse condivise ai processi.
- Il grafo di Holt e identificazione di deadlock tramite grafo di Holt.

**Elaborazione concorrente:**

- Programmazione sequenziale e programmazione concorrente; processi non sequenziali e grafo delle precedenze.
- Processi concorrenti in C/C++: le istruzioni *fork()*, *getpid()*, *getppid()* e *wait()*.

- Realizzazione di applicazioni basate sulla cooperazione e la competizione di processi concorrenti.

#### **I thread:**

- Il concetto di thread, sistemi single-threaded e threaded, gli stati dei thread.
- Implementazione di thread in Java: la classe *Thread* e l'interfaccia *Runnable*; gestione di thread: i metodi *join()*, *suspend()*, *resume()*, *sleep()*, *wait()*, *join()*; passaggio di parametri ad un thread.

### **MODULO 2: Sincronizzazione tra processi**

#### **Problemi connessi alla programmazione concorrente:**

- Comunicazione tra processi tramite memoria condivisa.
- Mutua esclusione e sezione critica; starvation e deadlock.

#### **Implementazione della mutua esclusione:**

- Il concetto di semaforo per proteggere l'accesso a risorse condivise.
- Semafori e mutua esclusione.
- I semafori in Java: il qualificatore *synchronized*, *wait()* e *notify()* e *notifyAll()*.

#### **Problemi classici della concorrenza:**

- Il problema dei produttori/consumatori e sua implementazione in Java.
- Il problema dei lettori/scrittori e sua implementazione in Java.

### **MODULO 3: La documentazione del software**

#### **La documentazione del codice:**

- Documentazione di un progetto software.
- Generazione automatica della documentazione tecnica con JAVADOC: tag per la documentazione di classi e metodi, generazione della documentazione.
- Uso di tag HTML nella documentazione con JavaDoc.

Fermo, 05 Giugno 2021

Gli insegnanti:  
Maria Rita Di Berardini,  
Luigi Marano