

**TAVOLA DI PROGRAMMAZIONE ANNUALE**  
**A.S. 2025/2026**

**Disciplina: Tecnologie e Progettazione di sistemi informatici e di telecomunicazioni**

**Docenti: Moroni – Morandi – Della Gatta**

**Classi: 4                      Sezioni: A-B-C                      Corso: INFORMATICA**

**TEORIA:**

Unità Didattica	ABILITA'	CONOSCENZE	Tipologia Verifiche	Tempi di Realizzazione entro il
TEORIA 1	Progettare e realizzare applicazioni in modalità concorrente. <u>Conoscere gli aspetti dell'esecuzione di processi concorrenti</u>	Casi significativi di funzionalità programmabili di un sistema operativo <u>Concetto di processo. Stati di un processo. Concorrenza, competizione e cooperazione tra processi. Mutua esclusione e sincronizzazione.</u>	Orale/Scritta	OTTOBRE
TEORIA 2	Progettare e realizzare applicazioni in modalità concorrente. <u>Conoscere gli strumenti per gestire la mutua esclusione e la sincronizzazione tra processi</u>	Tecniche e tecnologie per la programmazione concorrente e la sincronizzazione dell'accesso a risorse condivise.  <u>Soluzioni hardware e software per la mutua esclusione.</u> I semafori. Monitor e scambio messaggi. Lo stallo	Orale/Scritta	FEBBRAIO

TEORIA 3	Progettare e realizzare applicazioni in modalità concorrente. <u>Saper progettare algoritmi per gestire i piu' comuni problemi di sincronizzazione e mutua esclusione</u> tra processi, individuando gli strumenti e le strategie piu' efficaci	<u>Descrizione e soluzione teorica di problemi caratteristici con i processi concorrenti.</u>	Orale/Scritta	MARZO/APRILE
TEORIA 4	Conoscere i principali aspetti della comunicazioni in rete tra processi/thread	Introduzione agli Stream di Java e all'utilizzo degli Stream per la comunicazione attraverso Socket	Orale/Scritta	APRILE/MAGGIO
TEORIA 5	Identificare le fasi di un progetto nel contesto del ciclo di sviluppo. Documentare i requisiti e gli aspetti architetturali di un prodotto/servizio, anche in riferimento a standard di settore Conoscere le tecniche di validazione del software, le diverse tipologie dei requisiti software, le modalità di raccolta dei requisiti, la documentazione del software	Fasi e modelli di gestione di un ciclo di sviluppo. Tecniche e strumenti per la gestione delle specifiche e dei requisiti di un progetto. Verifica statica e dinamica del software. Raccolta e analisi dei requisiti. Documentazione dei requisiti. Documentazione del progetto.	Orale/Scritta MAGGIO/ GIUGNO	Orale/Scritta MAGGIO/GIUGNO

## LABORATORIO

LABORATORIO 0	Ripasso programmazione C e <u>utilizzo delle funzioni</u> . Strutture dati	Array, Cicli, Stringhe, Struct. Dichiarazione e chiamata di funzione: <u>passaggio di parametri per valore e per riferimento</u>	PRATICA	OTTOBRE
LABORATORIO 1	Progettare e realizzare applicazioni in modalità concorrente. <u>Conoscere i principali concetti riguardanti la creazione di processi in ambiente Linux</u>	Tecniche e tecnologie per la programmazione concorrente e la sincronizzazione dell'accesso a risorse condivise. Processi in Linux: immagine di un processo; <u>le system call fork(), getpid(), getppid(), execl(), wait(), exit();</u>	PRATICA	NOVEMBRE
LABORATORIO 2	Progettare e realizzare applicazioni in modalità concorrente. <u>Conoscere i principali concetti per la gestione dei processi in ambiente Linux</u> e per la programmazione concorrente e la comunicazione tra processi	Tecniche e tecnologie per la programmazione concorrente e la sincronizzazione dell'accesso a risorse condivise. <u>I comandi shell per la gestione dei processi: ps, top, nice, bg, fg, kill; le macro WIFEXITED e WEXITSTATUS.</u> Cenni ai meccanismi IPC: pipeline e memoria condivisa	PRATICA	DICEMBRE

LABORATORIO 3	Progettare e realizzare applicazioni in modalità concorrente. <u>Saper progettare algoritmi in linguaggio JAVA per gestire i piu' comuni problemi di sincronizzazione e mutua esclusione tra thread</u> , individuando gli strumenti e le strategie piu' efficaci	Tecniche e tecnologie per la programmazione concorrente e la sincronizzazione dell'accesso a risorse condivise. La programmazione concorrente in JAVA: <u>Concetto di Thread, creazione di un Thread, stati di un Thread, accesso a variabili condivise, mutua esclusione, sezioni critiche, sincronizzazione, comunicazione fra thread</u>	PRATICA	PENTAMESTRE
LABORATORIO 4	Saper utilizzare gli stream in linguaggio JAVA per la comunicazione via socket tra processi in esecuzione su host diversi	I file in Java. Le classi di Java per gli Stream. Cenni alle classi Java per Socket TCP. Utilizzo degli stream per la comunicazione in rete via Socket	(eventualmente PRATICA)	Fine pentamestre

- Si considerano obiettivi minimi irrinunciabili, il cui conseguimento comporta la sufficienza, quelli sottolineati.
- Sono previste verifiche per il primo quadrimestre in numero non inferiore a 3 di cui 1 scritte/orali e 2 scritte/pratiche
- Sono previste verifiche per il secondo quadrimestre in numero non inferiore a 4 di cui 2 scritte/orali e 2 scritte/pratiche