



# I.I.S. “Guglielmo Marconi” di Civitavecchia

Classe 4C – INF Anno scolastico 2023/2024

PROGRAMMA SVOLTO DI

## Tecnologie e Progettazione di Sistemi Informatici e Telecomunicazioni (T.P.S.I.T.)

Docenti: Curricolare: **Alessio Benedetti**  
Insegnante tecnico-pratico: **Santo Dell’Omo**

### TEORIA

**Il sistema operativo:** cenni storici sugli OS moderni più diffusi; il BIOS, il boot loader, il Kernel; i moduli Kernel interni ed il Management System; Kernel, Shell ed Applicazioni gli i-node nell’OS Unix/Linux; livello applicativo e Kernel; principi e caratteristiche della memoria virtuale; page table e page faults; analisi strutturale del processo e dei threads in memoria; definizioni delle unità di processamento principali: job, task, process, thread; classificazione e tipologie fondamentali degli algoritmi di scheduling per l’esecuzione di processi.

**Il versioning software e la programmazione professionale:** il triangolo contrattuale costi-risorse-obiettivi; la gestione del lavoro aziendale e le forme contrattuali di sviluppo dei prodotti; parallelismo tra gestione contrattuale e cicli di vita del software; modelli classici di sviluppo: Waterfall, Iterative Incremental, Rapid Prototyping e Spiral; modelli moderni: il manifesto Agile ed il modello Agile-Scrum; figure professionali ed elementi primari in Scrum. Il versioning condiviso multi-utente Git: gestione dei files e cartelle, modalità di codifica, indicizzazione con descrittori in Staging Area, formalizzazione delle modifiche con azioni di Commit nel Repository Level; il portale Git Hub come circuito collettivo di condivisione; repository private, pubbliche e condivise.

**Programmazione Concorrente:** definizioni dell’esecuzione Multi-Tasking, Multi-Processing, Multi-Threading e Parallel Multi-Threading/Thread-Level Multi-Processing in architetture con CPU provviste di Hyper-Threading Technology (HT); lo scheduling dei processi a livello applicativo come modello di interazione tra processi a livello hardware e di comunicazione tra host in una rete; i Semafori come strumento risolutivo per i conflitti da richiesta di risorse da parte di processi; concorrenza di 1° specie e 2° specie; il grafo di Holt e la molteplicità delle risorse; la concorrenza di 2° specie nel paradigma Producer-Consumer ed il caso applicativo dello Scrittore-Lettore; le code e le variabili di stato nel semaforo Mutex e Dijkstra; il formalismo Proberen/Wait e Verhogen/Notify/Signal definito da Dijkstra; l’approccio tensoriale per il paradigma Producer-Consumer in presenza di molteplici Scrittori e Lettori; utilizzo di variabili multi-strutturate per lo scheduling di processi in concorrenza di 1° specie; problemi di Starvation e Deadlock; il Problema dei Cinque Filosofi a Cena.

**Calcolo avanzato in Multi-Threading:** finalità e caratteristiche dell’esecuzione in Multi-Threading; threads comuni e thread Java; thread detached e joinable; Fork e Join di risorse e la pulizia dei threads dopo l’esecuzione su molteplici threads.

**La comunicazione web dinamica:** inclusione di codice JavaScript in pagine HTML; il processo di caricamento sequenziale della pagina web. L’XML come linguaggio personalizzabile per la gestione di dati. la tecnologia AJAX per la gestione asincrona di comunicazioni client-server.



**Il job-planning in Linux:** la daemon utility cronetab e come attivare cronjobs nei sistemi Linux.

**Il versioning software e la programmazione professionale:** i terminali Git Bash e Putty per accedere alle risorse scolastiche remote; Git Bash per creare versioning mediante la daemon utility Git; branching e merging di commit branches; merging diretto e con risoluzione dei conflitti; fetch, pull e clone di repository su GitHub; gestione della repository di destinazione con la variabile origin; connessione autenticata al proprio account GitHub mediante generazione di chiavi crittografiche con ssh-keygen; pull delle commit di modifiche su repository locale.

**Programmazione Concorrente:** realizzazione di Semafori in ambiente C++ per i conflitti da richiesta di risorse da parte di processi in concorrenza di II specie (asincrona); processi in comunicazione diretta mediante files e buffer di memoria condivisa con la libreria SHM; modellazione in C++ del problema dello Scrittore-Lettore; modello di Wait ciclico realizzato con Spin Lock; ottimizzazione dello Spin Lock con variante Test-and-Set; l'approccio tensoriale per il paradigma Producer-Consumer in presenza di molteplici Scrittori e Lettori; utilizzo di variabili multi-strutturate per lo scheduling di processi in concorrenza di I specie (sincrona); modellazione del Problema dei Cinque Filosofi a Cena. Monitors e programmazione concorrente asincrona in JAVA.

**Calcolo avanzato in Multi-Threading:** codifica in multi-threading in linguaggio C++ con librerie Posix pthread in ambiente Linux e librerie Thread in ambiente Windows; brevi esempi di programmazione in Java e Python; esempi di semafori Mutex e Dijkstra per la programmazione concorrente sincrona e asincrona in multi-threading in linguaggio C++ e Java.

**Il File System:** la gestione di files e directories in ambiente Linux con Terminal ed il motore BASH; la gestione in ambiente Microsoft Windows su Prompt di MS-DOS, Powershell e Powershell ISE; i terminali Git Bash e Putty per l'emulazione dei comandi Linux in ambiente Microsoft; il comando SSH per connettersi alle risorse scolastiche remote; comandi base, controllo dei processi e dei threads, uso delle variabili e uso degli alias per la composizione facilitata di comandi.

**Gestione dei sistemi a controllo remoto:** installazione del sistema operativo Ubuntu Raspberry Pi in piattaforma; controllo e gestione remota del sistema preparato.

**La comunicazione web dinamica:** inclusione di codice JavaScript in pagine HTML; costruzione statica delle pagine, modifica di attributi HTML da funzione, aggiunta e cancellazione dinamica di elementi HTML, controllo di temporizzazione, uso del mouse per eventi globali o su elementi HTML specifici; oggetti JS e metodi JSON per conversione stringa-oggetto. L'XML come linguaggio personalizzabile per la gestione e scambio di dati; lettura file XML mediante server locale. Il metodo XMLHttpRequest per la creazione di oggetti AJAX e aggiornamento asincrono delle pagine web mediante comunicazioni client-server in background.

## EDUCAZIONE CIVICA

**Gestione responsabile della rete informatica:** selezione consapevole e critica dei dati ricercati in rete; opportunità e rischi dell'Intelligenza Artificiale; sostenibilità ambientale dell'Internet economy.