

PIANO ANNUALE DELLE ATTIVITÀ– A.S. 2023-2024

Disciplina: TPSIT

PERIODO DIDATTICO: Secondo Periodo

Docente: Lo Turco Davide Co-Docente: Marra Daniele

STRATEGIE DIDATTICHE

Metodologie didattiche (lezioni frontali, didattica laboratoriale, problem solving,...) e strumenti utilizzati

Lezione frontale e FAD, Lezioni a distanza, Lezione metodologica, Lezione dialogica, Ricerca guidata.

PIANO DISCIPLINARE PER L'ORGANIZZAZIONE DELLE ATTIVITÀ DI DIDATTICA DIGITALE INTEGRATA

Partecipazione sincrona alle lezioni e alla pre-ora tramite Google Meet.

MATERIALE DIDATTICO

Uso di materiale didattico multimediale di presentazione; Uso della piattaforma Google Classroom di condivisione e collaborazione a distanza;

CRITERI E STRUMENTI DI VALUTAZIONE

Verifica scritta e pratica delle competenze acquisite al termine di ogni modulo didattico.

ORGANIZZAZIONE MODULARE (UdA) DELLA PROGRAMMAZIONE RELATIVA AL PERIODO DIDATTICO

MODULI	PERIODO	CONOSCENZE	ABILITÀ	COMPETENZE	OBIETTIVI MINIMI
Modulo 1	11/09/2023 10/11/2023	<p>TEORIA:</p> <ul style="list-style-type: none"> • La rappresentazione delle informazioni; • Sistemi di numerazione; • Trasformazione tra sistemi diversi; • Operazioni con i numeri binari; • Rappresentazione dei dati numerici (interi, numeri positivi e negativi -Ca1 e Ca2-) • Operazioni con numeri binari; • Standard IEEE-754; Ascii, Unicode. <p>LABORATORIO:</p> <ul style="list-style-type: none"> • Architettura del calcolatore 	<p>Codificare e utilizzare i numeri in rappresentazione binaria; Codificare i dati alfanumerici; Codificare i dati multimediali. Sapere l'Architettura di Von Neumann, i componenti hardware del sistema di elaborazione, le differenti memorie, individuare il processore e la sua struttura e i principali dispositivi periferici</p>	<p>Comprendere la rappresentazione interna delle informazioni, le operazioni di conversione relativa ai sistemi di numerazione.</p> <p>Descrivere con uno schema gerarchico le diverse tipologie di memorie utilizzate nei computer moderni. Identificare i principali dispositivi periferici e la corretta configurazione di un sistema</p>	<p>Conoscere e saper utilizzare i comandi principali per interagire con il sistema operativo.</p> <p>Saper riconoscere la struttura, l'architettura e i componenti dei sistemi di elaborazione</p>
Modulo 2	13/11/2023 02/02/2024	<p>TEORIA:</p> <ul style="list-style-type: none"> • Funzioni e struttura di un sistema operativo; • I sistemi operativi più diffusi; • Gestione dei file (Il concetto di file, operazione sui file, metodi di accesso, struttura della directory, file nei sistemi multiutente) <p>LABORATORIO:</p> <ul style="list-style-type: none"> • Linguaggio C (printf(), scanf(), variabili, stringhe, cicli ecc). 	<p>Utilizzare in modo appropriato la terminologia tecnica; Identificare e analizzare gli aspetti funzionali dei principali componenti di un sistema operativo; Applicare i concetti del file system. Utilizzare delle funzioni di I/O, delle variabili e i cicli iterativi in programmi scritti in linguaggio C.</p>	<p>Conoscere i compiti del sistema operativo; Classificare i sistemi operativi.</p> <p>Comprendere il funzionamento ed effettuare delle correzioni di programmi semplici scritti in linguaggio C.</p>	<p>Descrivere le tecniche di realizzazione del file system; Conoscenza e progettazione attraverso l'utilizzo del linguaggio C.</p>

Modulo 3	05/02/2024 05/04/2024	<p>TEORIA: Gestione dei processi:</p> <ul style="list-style-type: none"> • programma/processo • stati • interruzioni • strutture per gestire i processi • schedulazione dei processi • algoritmi di schedulazione <p>LABORATORIO:</p> <ul style="list-style-type: none"> • Linguaggio C (Puntatori, vettori, funzioni). 	<p>Applicare politiche di gestione dei processi; Individuare le problematiche per la cooperazione tra processi. Utilizzare e applicare i concetti degli array monodimensionali in linguaggio C. Utilizzare le funzioni in linguaggio C Utilizzare e applicare i concetti degli array monodimensionali in linguaggio C.</p>	<p>Cosa si intende per processo; Descrivere il ciclo di vita di un processo. Comprendere e suddividere un programma in funzioni (anche in presenza di valori di ritorno). Comprendere il funzionamento dei puntatori per scorrere l'array e come passaggio per riferimento a funzioni. Comprendere la necessità di suddividere il programma in funzioni.</p>	<p>Saper scrivere un programma con il linguaggio C utilizzando puntatori, vettori e funzioni. Differenza tra passaggio per riferimento e per valore.</p>
Modulo 4	08/04/2024 11/06/2024	<p>TEORIA: Gestione della memoria:</p> <ul style="list-style-type: none"> • caricamento del programma • binding/linking • allocazione della memoria: partizionamento (fisso e variabile) • memoria virtuale: paginazione e segmentazione • Accenni fasi e modelli di gestione di un ciclo di sviluppo. <p>LABORATORIO:</p> <ul style="list-style-type: none"> • Linguaggio C (strutture selettive e iterative, record). 	<p>Applicare politiche di gestione delle risorse; Individuare le problematiche per la cooperazione tra processi; Saper individuare i tipi di dato e le strutture di dati più idonee alla soluzione di problemi assegnati.</p>	<p>Riconoscere i meccanismi di caricamento del programma in memoria; Conoscere le tecniche di virtualizzazione della memoria.</p>	<p>Saper scrivere un programma più complesso con il linguaggio C con l'utilizzo di strutture.</p>

Modulo 5	11/09/2023 10/11/2023	<p>TEORIA</p> <ul style="list-style-type: none"> • Vari tipi di virtualizzazione. • Installazione sistema operativo Linux come macchina virtuale. • Conoscere il significato di alcune variabili di sistema. <p>LABORATORIO:</p> <ul style="list-style-type: none"> • Programmazione C: (LISTE) 	<p>Saper installare un sistema operativo.</p> <p>Saper utilizzare l'interfaccia grafica di un sistema operativo e i tool di manutenzione.</p> <p>Utilizzare le strutture dati per creare e manipolare liste in linguaggio C.</p>	<p>Competenze teoriche di base.</p> <p>Scegliere dispositivi e strumenti in base alle loro caratteristiche funzionali.</p> <p>Saper creare e scorrere una lista. Essere in grado di aggiungere un elemento, eliminare un elemento da una lista.</p>	<p>Saper utilizzare un software di virtualizzazione.</p> <p>Saper installare un sistema operativo.</p> <p>Conoscere i comandi più utilizzati della shell Linux.</p> <p>Saper creare e scorrere una lista di interi.</p>
Modulo 6	13/11/2023 02/02/2024	<p>TEORIA e LABORATORIO</p> <ul style="list-style-type: none"> • I comandi della shell Linux • Conoscere il linguaggio di script. • Conoscere gli elementi di base degli script. 	<p>Saper scrivere in modo autonomo semplici script di manutenzione.</p>	<p>Competenze teoriche di base.</p> <p>Scegliere dispositivi e strumenti in base alle loro caratteristiche funzionali.</p>	<p>Creare dei semplici script per la manipolazione di file e directory</p>
Modulo 7	05/02/2024 05/04/2024	<p>TEORIA: Processi sequenziali e paralleli:</p> <ul style="list-style-type: none"> • il modello a processi • risorse e condivisione • i thread o processi leggeri • elaborazione sequenziale e concorrente • la descrizione della concorrenza <p>LABORATORIO:</p> <ul style="list-style-type: none"> • thread in Java. 	<p>Distinguere le richieste e le modalità di accesso alle risorse;</p> <p>Conoscere le caratteristiche di un linguaggio concorrente.</p>	<p>Descrivere l'interazione processi-risorse col grafo di Holt;</p> <p>Acquisire il concetto di risorsa condivisa;</p> <p>Conoscere la differenza tra processi e thread.</p>	<p>Acquisire il concetto di programmazione concorrente;</p> <p>Conoscere le caratteristiche di un linguaggio concorrente;</p> <p>Utilizzare i thread e processi in linguaggio Java.</p>

Modulo 8	08/04/2024 11/06/2024	<p>TEORIA:</p> <ul style="list-style-type: none"> • La comunicazione tra processi; La sincronizzazione tra processi; I semafori; • Applicazione dei semafori; • Il problema dei produttori/consumatori; • Il problema dei lettori/scrittori. • Il problema del deadlock: • banchiere e filosofi a cena; <p>LABORATORIO:</p> <ul style="list-style-type: none"> • Monitor e semafori in Java. 	<p>Risolvere i problemi produttore/consumatore in Java. Risolvere le situazioni di deadlock;</p> <p>Conoscere il modello ad ambiente globale e locale; Individuare le tipologie di errori nei processi paralleli.</p> <p>Utilizzare i Monitor e i semafori per la risoluzione di semplici problemi nell'ambito della programmazione concorrente</p>	<p>Utilizzare gli strumenti di sincronizzazione; Conoscere il concetto di regione critica e di mutua esclusione.</p>	<p>Saper scrivere un programma che risolvi i problemi con l'utilizzo dei semafori e monitor in Java.</p>
-----------------	--	---	---	--	--