

UNITÀ DI APPRENDIMENTO	COMPETENZE	ABILITÀ	CONTENUTI	ESERCITAZIONI DI LABORATORIO	COLLEGAMENTI INTERDISCIPLINARI	METODOLOGIE
Linguaggio e logica dei calcolatori ed elementi di base di networking	<p>Analizzare dati e interpretarli sviluppando deduzioni e ragionamenti sugli stessi anche con l'ausilio di rappresentazioni grafiche, usando consapevolmente gli strumenti di calcolo e le potenzialità offerte da applicazioni specifiche di tipo informatico.</p> <p>Essere consapevole delle potenzialità e dei limiti delle tecnologie nel contesto culturale e sociale in cui vengono applicate</p> <p>Realizzare piccoli e esempi di locale con simulatore (come Cisco Packet Tracer) e dispositivi fisici di switching e routing (come i network-device Cisco)</p>	<p>Saper utilizzare sistemi di numerazione posizionali diversi da quello decimale ed effettuare operazioni di conversione ed aritmetiche in un sistema di base qualsiasi</p> <p>Saper rappresentare ed utilizzare operatori logici e tabelle di verità</p> <p>Saper distinguere ruoli e funzioni di hardware e software.</p> <p>Classificare i diversi tipi di software utilizzati da un computer.</p> <p>Saper utilizzare le funzioni del sistema operativo</p> <p>Utilizzare i principali network-device in rete per la condivisione delle risorse</p>	<p>Sistemi di numerazione posizionali e conversioni notevoli</p> <p>Rappresentazione dei caratteri e dei numeri sul calcolatore</p> <p>Linguaggi di programmazione e i traduttori</p> <p>Software di base e applicativo</p> <p>Logica proposizionale (elementi base)</p> <p>Elementi principali dei network-device e della comunicazione tra gli host</p> <p>Struttura generale di una URL e di un indirizzo e-mail</p>	<p>piccole esperienze con i codici di rappresentazione dei caratteri (codice ASCII) e con simulatori di rete (come Cisco Packet Tracer, ad esempio)</p>	<p>area ICT in generale come ORIENTAMENTO</p>	<p>lezione frontale partecipata, problem-solving, didattica laboratoriale</p>
Elementi base della programmazione: problemi, algoritmi, programmi	<p>Individuare le strategie appropriate per la soluzione di problemi</p>	<p>Distinguere tra i concetti di istruzione, algoritmo, programma e processo</p> <p>Saper utilizzare i formalismi per la rappresentazioni degli algoritmi (Flow-chart PDL), essere in grado di darne la specifica</p> <p>Saper verificare le</p>	<p>algoritmi e relative tecniche di analisi ((Flow-chart PDL)</p> <p>Specifica degli algoritmi, tipi di dati ed istruzioni (programmazione imperativa</p>	<p>Esercitazioni possibili con ambienti di sviluppo tipo Scratch [ambiente didattico] e/o Jeany (per C/C++ o altri linguaggi di programmazione [ambiente professionale] per lo sviluppo di semplici programmi secondo il paradigma</p>	<p>area ICT in generale come ORIENTAMENTO</p>	<p>lezione frontale partecipata, problem-solving, didattica laboratoriale</p>

		soluzioni trovate, Saper individuare e correggere gli errori a tempo di compilazione e di esecuzione		imperativo		
Sviluppo di pagine Web client-side ed elementi server-side	<p>Analizzare dati e interpretarli sviluppando deduzioni e ragionamenti sugli stessi anche con l'ausilio di rappresentazioni grafiche, usando consapevolmente gli strumenti di calcolo e le potenzialità offerte da applicazioni specifiche di tipo informatico.</p> <p>Individuare le strategie appropriate per la soluzione di problemi</p>	<p>Realizzare pagine web attraverso l'utilizzo del linguaggio html sfruttando la relativa architettura client-server</p> <p>Riconoscere ed implementare la struttura di base di una pagina web, attraverso la costruzione di iper-link ed oggetti multimediali vari</p> <p>Saper utilizzare gli elementi avanzati del linguaggio html per poter gestire pagine web complete e complesse</p> <p>saper utilizzare i principali elementi di JavaScript per la gestione degli eventi</p> <p>Essere in grado di utilizzare dei linguaggi di script html-embedded client-side e server-side(solo cenni) per lo sviluppo di pagine web dinamiche</p>	<p>Il linguaggio di markup HTML: i tag e gli attributi, struttura gerarchica e organizzazione di una pagina html</p> <p>Gestione di oggetti multimediali: testo, immagini, suoni, iper-link (collegamenti ipertestuali)</p> <p>Elementi avanzati del linguaggio: colori, tabelle, frame, mappe sensibili, form (moduli), style-sheets (fogli di stile)</p> <p>Cenni al codice html dinamico (generazione on-the-fly e risposta agli eventi): programmazione server-side e client-side e utilizzo di relativi linguaggi di script html-embedded, elementi di JavaScript lato client</p>	Esperienze di sviluppo di pagine web lato client	area ICT in generale come ORIENTAMENTO	<p>lezione frontale partecipata, problem-solving, didattica laboratoriale</p>

OBIETTIVI MINIMI PER IL PASSAGGIO ALLA CLASSE SUCCESSIVA: le competenze e gli obiettivi sono semplificati secondo il prospetto qui riportato.

All'alunno/a è richiesta una minore autonomia risolutiva e una minore capacità di collegamento.

Verifica di sosp. giudizio scritta e orale

ORIENTAMENTO: FORNIRE AGLI STUDENTI UNA VISIONE CHIARA E GENERALE DELLE MATERIE D'INDIRIZZO PRINCIPALI (Informatica, Sistemi e Reti, TPSIT), CHE VERRANNO INSEGNATE AL TRIENNIO, IN CASO DI SCELTA *DELL'ARTICOLAZIONE INFORMATICA* NELL'AMBITO DELL'INDIRIZZO *INFORMATICA E TELECOMUNICAZIONI*

STRUMENTI DIDATTICI: libro di testo, appunti del docente alla lavagna, materiale multimediale, lab. d'informatica, simulatore *Cisco Packet Tracer* e device di rete, Cisco, editor *HTML*

TIPOLOGIA VERIFICHE: Scritte/orali (prove orali come compensative/migliorative/sostitutive delle prove scritte)