

DISCIPLINA INFORMATICA

obiettivi specifici di apprendimento, declinati in termini di conoscenze, abilità e competenze e suddivisi in primo biennio, secondo biennio e quinto anno.

PRIMO BIENNIO		
CONOSCENZE	ABILITA'	COMPETENZE
<ul style="list-style-type: none"> • Differenza fra analogico e digitale. • Codifica dell'informazione in formato digitale, codifica delle immagini, codice ASCII e Unicode. • Sistema di numerazione binari. • La struttura di un sistema di elaborazione: hardware e software. • L'Architettura di Von Neumann e i suoi componenti. • La struttura generale di un sistema operativo, e in particolare del Sistema Operativo Windows. • Processi e programmi. • Le caratteristiche di un File System, delle operazioni su file e directories. • Il foglio di calcolo: concetti e funzioni • Gli strumenti di presentazione multimediali: concetti e funzioni. • Concetto di algoritmo e sue proprietà. • Concetto di variabile e sue proprietà. • Linguaggio di progetto, diagrammi di flusso e linguaggi di programmazione. • Costrutti sintattici fondamentali: sequenza, selezione, iterazione. • Documentazione di lavoro: tabella di traccia. • Differenza fra programma sorgente e programma eseguibile. • Linguaggio di programmazione Java. • Concetto di compilatore e Java Virtual Machine. • Approccio top-down e programmazione modulare. • Concetto di Record di attivazione 	<ul style="list-style-type: none"> • Riconoscere scheda madre, CPU, memoria centrale, dischi e periferiche principali di un calcolatore. • Saper convertire i numeri da formato decimale a binario e viceversa • Comprendere la codifica dell'informazione e la sua rappresentazione in formato numerico. • Saper individuare le proprietà di file e cartelle, e saper eseguire su di essi le operazioni fondamentali. • Saper applicare le varie funzioni del foglio di calcolo. • Saper applicare le varie funzioni del generatore di presentazioni multimediali. • Usare un browser. • Usare un motore di ricerca. • Saper analizzare un problema, individuare le diverse entità che lo compongono, e saperlo esprimere attraverso un modello formale. • Costruire algoritmi secondo gli schemi della programmazione strutturata. • Saper produrre la documentazione di lavoro. • Saper riconoscere file sorgente da file eseguibili e in Bytecode. • Saper distinguere i diversi tipi di variabile. • Utilizzare un ambiente di programmazione per la creazione di programmi Java. • Riconoscere la scomposizione di problemi in sotto-problemi. 	<ul style="list-style-type: none"> • Comprendere la struttura di un elaboratore. • Saper riconoscere le funzionalità di un Sistema Operativo in tutte le sue parti. • Riconoscere le diverse rappresentazioni interne dei dati. • Saper scegliere fra le varie funzioni del foglio di calcolo quella più idonea al raggiungimento di un dato obiettivo. • Saper creare una presentazione multimediale efficace. • Saper progettare un algoritmo per la soluzione di semplici problemi, ed esprimerlo tramite un diagramma a blocchi. • Saper ridurre un problema in sotto-problemi più piccoli ed elaborare una soluzione per i sotto-problemi. • Saper scegliere il tipo di variabile più idoneo per la soluzione di un problema. • Saper utilizzare il linguaggio di programmazione Java per la soluzione di problemi di media difficoltà. • Saper utilizzare la rete per comunicare, ricercare e condividere informazioni

SECONDO BIENNIO

CONOSCENZE	ABILITA'	COMPETENZE
<ul style="list-style-type: none"> • Concetto di classe e oggetto. • Concetto di metodo di una classe e di costruttore di una classe. • Strutture dati: i vettori. • Algoritmi fondamentali di ricerca • Algoritmi fondamentali di ordinamento • Strutture dati: le matrici • Concetto di ereditarietà • Concetto di interfaccia. • Funzioni di un Data Base Management System (DBMS) • Progettazione Concettuale di un Database • Progettazione Logica di un Database • Linguaggio SQL • Caratteristiche e funzionalità del linguaggio di markup. • Linguaggio HTML. • Scomposizione fra contenuto (HTML) e forma (CSS). • Le tabelle, le immagini in html e collegamenti ipertestuali in html. • Posizionamento dei fogli di stile (interno ed esterno). • Vantaggi dell'utilizzo dei fogli di stile. • Formattazione del testo con gli stili. • Box model e Layout con gli stili. • Differenza fra pagina statica e dinamica. • Posizionamento degli script (interno ed esterno). • Istruzione base di interazione con l'utente. 	<ul style="list-style-type: none"> • Capire la logica della programmazione ad oggetti. • Definire e riconoscere ambienti locali e globali. • Capire la logica della programmazione ad oggetti. • Definire e riconoscere il metodo costruttore. • Saper Utilizzare variabili strutturate, vettori e matrici. • Definire e riconoscere una classe ereditata. • Progettare una base di dati utilizzando specifici modelli • Realizzare tabelle e relazioni di un Data Base. • Saper riconoscere le caratteristiche dei linguaggi di markup al fine di distinguerli dai linguaggi di programmazione. • Saper scrivere codice in HTML. • Saper distinguere fra elementi di contenuto e di forma. • Saper gestire le immagini e tabelle. • Saper gestire i collegamenti fra pagine diverse. • Saper creare fogli di stile interni ed esterni. • Saper formattare il testo nella maniera desiderata (sfondo, font, colore, allineamento etc.) • Saper utilizzare in maniera appropriata il box model ed assegnare correttamente i suoi parametri. • Saper creare un semplice sito web statico. • Saper creare elementi di script interni ed esterni. • Saper creare script che interagiscono con l'utente utilizzando le istruzioni base. 	<ul style="list-style-type: none"> • Saper creare programmi che utilizzino classi e oggetti. • Saper creare semplici programmi che utilizzino strutture dati vettori e matrici. • Saper implementare gli algoritmi notevoli di ordinamento nel linguaggio di programmazione Java. • Saper creare programmi che sfruttino l'ereditarietà per ottimizzare la struttura dei programmi stessi. • Saper analizzare le informazioni necessarie alla progettazione di una base di dati. • Saper progettare lo schema concettuale di un Data Base • Saper progettare lo schema logico di una base di dati • Saper riconoscere attributi, entità e relazioni all'interno di uno schema concettuale e di uno schema logico • Saper utilizzare le funzioni di un DBMS per estrapolare informazioni. • Saper realizzare pagine web funzionali al sito che si vuole produrre. • Riuscire a realizzare pagine web accattivanti attraverso l'utilizzo dei fogli di stile. • Saper distinguere in maniera corretta fra pagina web statica e pagina web dinamica. • Saper creare semplici pagine web dinamiche attraverso l'utilizzo di Javascript. • Utilizzare le reti e gli strumenti informatici nelle attività di studio, ricerca e approfondimento disciplinare.

QUINTO ANNO

CONOSCENZE	ABILITA'	COMPETENZE
<ul style="list-style-type: none"> • Classificazione di reti di computer. • Commutazione di pacchetto e di circuito. • Concetto di protocollo. • Architettura a strati ISO/OSI e TCP/IP. • Incapsulamento dei dati. • Standard di livello 1, cavi in rame, fibra ottica e wi-fi. • Protocolli di livello 2, Token ring, Ethernet e relativa trama. • Indirizzi MAC. • Protocollo di livello 3. • Indirizzamento IP, reti e sottoreti e VLAN. • Protocolli di routing • DHCP, il DNS e altri servizi di rete • Concetto di errore di rappresentazione di un numero reale • Algoritmi di calcolo numerico: calcolo di radice quadrata, pigreco. • Colo degli zeri di funzioni: metodo di bisezione, tangenti, corde e secanti. • Calcolo massimi e minimi di una funzione. • Calcolo integrali. • Metodi Montecarlo. • Concetto di crittografia ed esempi di algoritmi. • Differenza fra crittografia simmetrica e asimmetrica. • Esempi di utilizzo di crittografia asimmetrica: la firma digitale. 	<ul style="list-style-type: none"> • Saper riconoscere i vari tipi di rete. • Saper identificare il livello di un protocollo relativamente allo stack ISO/OSI e TCP/IP. • Comprendere gli indirizzi di rete di livello 2 e 3. • Saper riconoscere i diversi servizi di rete. • Saper riconoscere i campi che compongono un pacchetto. • Saper riconoscere gli elementi dell'architettura Client-Server. • Capire la differenza fra numero reale e sua rappresentazione numerica. • Comprendere i Concetti di overflow e underflow e saperli riconoscere. • Comprendere gli algoritmi principali del calcolo numerico. • Comprendere il concetto di crittografia ed il suo utilizzo. • Saper riconoscere un algoritmo di crittografia simmetrico da uno asimmetrico. • Saper riconoscere i campi di utilizzo della crittografia. 	<ul style="list-style-type: none"> • Saper scegliere la tipologia di rete più adatta in base al contesto. • Saper scegliere i dispositivi più idonei alla realizzazione di una piccola rete. • Saper pianificare gli indirizzi di una piccola rete. • Capire la struttura di un pacchetto dati attraverso l'utilizzo dell'analizzatore di pacchetti. • Saper riconoscere le diverse rappresentazioni interne dei dati. • Saper implementare gli algoritmi principali del calcolo numerico applicati ad altre discipline e alla matematica in particolare. • Acquisire la consapevolezza dei vantaggi e dei limiti d'uso degli strumenti e dei metodi informatici. • Acquisire consapevolezza della necessità dell'utilizzo della crittografia nelle comunicazioni informatiche. • Saper applicare il concetto di crittografia a situazioni della vita reale, (es. la firma digitale)