

ISTITUZIONE SCOLASTICA DI ISTRUZIONE TECNICA

LYCÉE TECHNIQUE

**PROGRAMMAZIONE
EDUCATIVA E DIDATTICA ANNUALE
PER COMPETENZE**

A.S. 2019/2020

Materia Informatica

Classe 3 A IT

Professore Dell'Agosto Antonella – Nicod Elia

Metodologia (tiene conto delle competenze di cittadinanza)

Le ore di lezione sono intese come l'ambiente in cui si stimolano gli alunni ad organizzare il loro pensiero secondo ordinate strutture logiche e si offrono gli strumenti fondamentali per sviluppare capacità di ragionamento e di risoluzione di problemi di varia natura. Si pongono le basi affinché gli alunni sviluppino una crescente consapevolezza del ruolo cui vengono preparati e possano pervenire, anche autonomamente, a conoscenze più estese delle diverse applicazioni dell'informatica. Per favorire ciò, l'attività didattica sarà imperniata sulla verifica diretta da parte degli allievi delle connessioni esistenti tra aspetti teorici ed aspetti operativi. Si adotta, perciò, una metodologia adeguata che consente di pervenire alla conoscenza delle strutture informatiche con un processo che stimola al ragionamento attraverso modelli in cui si parte da situazioni concrete. Si alterneranno momenti di lezione frontale, con la presentazione da parte dell'insegnante dei contenuti essenziali dell'argomento, a momenti di discussione all'interno della classe per la soluzione di un problema presentato, in modo tale da sviluppare una coscienza critica negli studenti. I libri di testo saranno utilizzati come punto di riferimento per la consultazione soprattutto per quanto riguarda l'attività di apprendimento del linguaggio C++. Si farà riferimento al seguente testo in adozione: "PRO.TECH Informatica per Istituti Tecnici Tecnologici Vol. A" di *A. Lorenzi – R. Giupponi – V. Moriggia – E. Cavalli* – Ed. ATLAS.

Per quanto riguarda le competenze di cittadinanza, ci si atterrà a quanto deciso nei Collegi dei Docenti, nelle riunioni di Dipartimento e nel Consiglio di classe. Pertanto, gli studenti verranno aiutati a raggiungere gradualmente un certo livello di autonomia nello svolgimento delle attività proposte e di responsabilità nel rispetto delle attrezzature utilizzate, nell'uso consapevole del materiale scolastico, nell'interazione con gli insegnanti e con i compagni.

La ripartizione oraria relativa ai moduli potrà subire delle variazioni a seconda delle esigenze didattiche che emergeranno in fase operativa.

Qualora gli argomenti svolti a lezione prevedano una parte applicativa, quest'ultima sarà curata e valutata dall'Insegnante Tecnico Pratico.

Criteri e strumenti di valutazione

La valutazione verrà effettuata mediante verifiche periodiche, orali, scritte e pratiche, tenendo conto dei seguenti criteri, opportunamente pesati a seconda delle circostanze e del tipo di prova, in base alle indicazioni ministeriali:

- *Padronanza delle conoscenze disciplinari relative ai nuclei tematici oggetto della prova e caratterizzante/i l'indirizzo di studi;*

- *Padronanza delle competenze tecnico-professionali specifiche di indirizzo rispetto agli obiettivi della prova, con particolare riferimento all'analisi e comprensione dei casi e/o delle situazioni problematiche proposte e alle metodologie/scelte effettuate/procedimenti utilizzati nella loro risoluzione;*

- *Completezza nello svolgimento della traccia, coerenza/correttezza dei risultati e degli elaborati tecnici e/o tecnico grafici prodotti;*

- *Capacità di argomentare, di collegare e di sintetizzare le informazioni in modo chiaro ed esauriente, utilizzando con pertinenza i diversi linguaggi tecnici specifici.*

Il punteggio andrà da 1 a 10.

Il tipo di prova scritta sarà diversificato in base agli obiettivi da verificare mentre il colloquio orale, oltre a considerare le capacità di programmazione, tenderà a sviluppare le capacità espressive, di sintesi e l'abilità a usare vocaboli tipici del linguaggio informatico. La valutazione finale terrà conto sia dei risultati delle valutazioni periodiche (media degli scritti con peso 40%, degli orali con peso 40% e delle prove di laboratorio con peso 20%), sia dell'impegno e del progresso in itinere dell'alunno. In entrambi i due periodi scolastici, si prevede di effettuare almeno due prove scritte, due verifiche orali ed una di laboratorio, a cui si aggiungeranno le valutazioni sulle competenze qualora venga deciso dal Collegio dei Docenti, dal Dipartimento o dal Consiglio di Classe.

Attività di recupero

L'attività di recupero verrà svolta esclusivamente per gli studenti che hanno dimostrato impegno sia durante le lezioni sia nello svolgimento regolare delle attività assegnate a casa. Ci si atterrà a quanto deciso nei Collegi dei Docenti, nelle riunioni di Dipartimento e nel Consiglio di classe.

Competenze (Linee guida+competenze di cittadinanza):

- **CAI1:** utilizzare le strategie del pensiero razionale negli aspetti dialettici ed algoritmici per affrontare situazioni problematiche elaborando opportune soluzioni;
- **CAI2:** sviluppare applicazioni informatiche per reti locali o servizi a distanza;
- **CAI3:** scegliere dispositivi e strumenti in base alle loro caratteristiche funzionali;
- **CAI4:** gestire progetti secondo le procedure e gli standard previsti dai sistemi aziendali di gestione della qualità;
- **CAI5:** redigere relazioni tecniche e documentare le attività individuali e di gruppo relative a situazioni professionali;

Competenze Asse Scientifico Tecnologico

- **CAST6:** Utilizzare correttamente e descrivere il funzionamento di sistemi complessi, anche di uso corrente
- **CAST7:** Gestire Progetti

Programmazione (linee guida + competenze di cittadinanza):

CLASSE 3 A IT

MODULO	TIPOLOGIA PROVE
<i>Modulo 1</i> Fondamenti di informatica	verifica orale e/o scritta

Conoscenze	Abilità	Competenze
Terminologia di base. Linguaggi di programmazione per un elaboratore. Linguaggi ad alto livello, compilatori, interpreti. L'ambiente di sviluppo di un programma C++. La fase di debugging.	- Conoscere il lessico e terminologia tecnica di settore. - Stabilire le relazioni fondamentali tra macchine, problemi, informazioni e linguaggi. - Riconoscere i linguaggi e le macchine a vari livelli di astrazione. - Produrre un programma eseguibile attraverso le varie fasi.	CAST6 CAI3

MODULO	TIPOLOGIA PROVE
<i>Modulo 2</i> Introduzione agli algoritmi e alla programmazione di base	verifica scritta e/o orale e/o pratica

Conoscenze	Abilità	Competenze
------------	---------	------------

<p>Dal problema al programma:</p> <ul style="list-style-type: none"> - L'algoritmo - Diagrammi di flusso - Variabili e costanti - Tipi di dati - Struttura sequenziale, espressioni ben formate ed operatori aritmetici - Struttura condizionale: if, if annidati, switch case, espressioni condizionali ben formate ed operatori booleani. - Input e output 	<ul style="list-style-type: none"> - Saper rappresentare un algoritmo in modalità diverse. - Conoscere la rappresentazione delle informazioni alfanumeriche. - Conoscere la rappresentazione di altri tipi di dati. - Realizzare algoritmi che prevedono istruzioni di lettura, scrittura, assegnazione. - Saper descrivere algoritmi attraverso i diagrammi di flusso. - Saper utilizzare AlgoBuild. - Saper descrivere algoritmi attraverso il linguaggio C++. - Le strutture condizionale semplici. - Le basi della logica simbolica e del calcolo preposizionale. - Le strutture condizionali complesse. - Le strutture iterative con controllo in testa, in coda o con numero prefissato di cicli. - Utilizzare il debug per seguire l'evoluzione del programma. - Testare gli algoritmi creando insieme significativi di casi di prova. - Gli argomenti del main(). 	<p>CAST7 CAI1 CAI3 CAI5</p>
---	---	---

MODULO	TIPOLOGIA PROVE
<p style="text-align: center;"><i>Modulo 3</i></p> <p>Approfondimenti nella programmazione</p>	<p>verifica scritta e/o orale e/o pratica</p>

Conoscenze	Abilità	Competenze
<p>La progettazione top-down.</p> <p>Le variabili locali e globali.</p> <p>Le funzioni ed i parametri.</p> <p>Passaggio di parametri per valore o per indirizzo.</p> <p>Le aree di memoria coinvolte durante l'esecuzione di un programma.</p>	<ul style="list-style-type: none"> - Saper affrontare un problema scomponendolo in sottoproblemi. - Risolvere problemi usando le funzioni ed i parametri quando necessario. - Scegliere opportunamente tra un algoritmo iterativo ed uno ricorsivo. - Individuare i vantaggi e gli svantaggi di un algoritmo ricorsivo. 	<p>CAST6 CAST7 CAI1 CAI3 CAI5</p>

La ricorsione.	- Produrre la documentazione relativa al software.	
----------------	--	--

MODULO	TIPOLOGIA PROVE
<i>Modulo 4</i> I vettori	verifica scritta e/o orale e/o pratica

Conoscenze	Abilità	Competenze
Organizzare i dati nei vettori ad una dimensione. Algoritmi di ricerca. Algoritmi di ordinamento. Vettori in parallelo. Algoritmi di fusione e totalizzazione.	- Saper affrontare un problema con tipi di dato strutturato. - Saper applicare algoritmi di ricerca, di ordinamento, di fusione e totalizzazione. -Saper utilizzare vettori in parallelo.	CAST6 CAST7 CAI1 CAI3 CAI5

MODULO	TIPOLOGIA PROVE
<i>Modulo 5</i> Teoria della complessità algoritmica	verifica scritta e/o orale

Conoscenze	Abilità	Competenze
La complessità degli algoritmi. Algoritmi efficienti. La robustezza dei programmi.	-Analizzare e confrontare algoritmi diversi per la soluzione dello stesso problema.	CAI1 CAI3 CAI4 CAST7

MODULO	TIPOLOGIA PROVE
<i>Modulo 6</i> Strutture dati statiche	verifica scritta e/o orale e/o pratica

Conoscenze	Abilità	Competenze
Il record come insieme di elementi non omogenei. Algoritmi per la gestione delle matrici. La tabella come array di record. Algoritmi per la gestione delle tabelle.	- Risolvere problemi con il supporto di strutture dati statiche complesse. - Organizzare i dati nelle matrici. - Organizzare i dati nelle tabelle. - Applicare algoritmi di ricerca	CAST6 CAST7 CAI1 CAI3 CAI5

	all'interno di tabelle. - Applicare algoritmi di ordinamento sugli elementi di una tabella.	
--	--	--

MODULO	TIPOLOGIA PROVE
<i>Modulo 7</i> I files sequenziali - i files ad accesso diretto	verifica scritta e/o orale e/o pratica

Conoscenze	Abilità	Competenze
I files ad accesso sequenziale. I files testo. I files ad accesso diretto.	Risolvere problemi con il supporto di files sequenziali o ad accesso diretto	CAST7 CAI1 CAI3 CAI4

Obiettivi minimi

- Saper risolvere semplici problemi rappresentando la soluzione con una rappresentazione grafica. Saper elaborare/costruire diagrammi di flusso.
- Conoscere e saper utilizzare i costrutti fondamentali per la soluzione di semplici problemi: i costrutti sequenziali, di selezione, iterativi (con controllo in testa e in coda) e i cicli enumerativi.
- Saper scrivere una funzione individuando i parametri e restituendo correttamente l'output.
- Conoscere il concetto di riusabilità del codice. Saper individuare sottoprogrammi all'interno di un codice, dichiarare e invocare di funzioni, effettuare/realizzare il passaggio di parametri per valore e riferimento.
- Essere in grado di identificare le strutture dati più idonee alla rappresentazione dei dati in memoria e alla gestione da parte del programmatore. Conoscere e utilizzare in modo appropriato i tipi dato elementari e i tipi di dato derivati: gli array monodimensionali, le strutture e le tabelle.
- Saper gestire strutture dati vettoriali e tabelle (array di strutture) tramite algoritmi:
 - inserimento e visualizzazione di elementi (scansione di un vettore);
 - di ricerca (completa e dicotomica) in base ad un criterio;
 - di ordinamento (2 algoritmi a scelta).
 - fondamentali: ricerca di massimo e minimo, per effettuare somme e conteggi.
- Conoscere il concetto di visibilità di una variabile locale e globale.
- Conoscere il concetto di complessità di un algoritmo e confrontare le prestazioni degli algoritmi di ordinamento (2 algoritmi a scelta).
- Conoscere le tipologie di file e le operazioni elementari sui file (lettura, scrittura, controllo in fase di apertura del file in lettura, modifica, cancellazione) per la gestione di file sequenziali e binari.