

Istituto Tecnico Superiore Statale Majorana -

Giorgi VIA S.ALLENDE, 41, 16100 GENOVA (GE)

**SCUOLA SECONDARIA DI SECONDO GRADO**

# **PIANO DI LAVORO**

**Anno Scolastico 2022-2023**

Docente: NARBONA Alessandra

Disciplina d'insegnamento: INFORMATICA

Classe: 3BS

Modalità per l'analisi della situazione iniziale:

- test d'ingresso
- accertamento informale
- verifica orale
- altro:

Obiettivi formativi specifici e disciplinari:

**Obiettivi formativi relativi al modo di essere:** *correttezza*, cioè rispetto delle persone, della diversità, del sapere, del lavoro, delle regole e delle strutture materiali; *responsabilità*, cioè autonomia, capacità di organizzazione e di interazione; *attività*, cioè impegno nello studio e nella partecipazione al dialogo educativo, ampliamento degli interessi, capacità di valutazione e di autovalutazione.

**Obiettivi formativi relativi all'agire didattico:** *capacità di utilizzare autonomamente strumenti* (libri, manuali, indici, grafici, mappe e carte, letteratura specialistica, giornali e riviste); *sviluppo di competenze logiche* come quella di osservare, definire, gerarchizzare, individuare collegamenti, **descrivere**, generalizzare, sintetizzare, comparare; *sviluppo di capacità linguistiche*, in fase ricettiva e produttiva, come l'esposizione orale e scritta, l'acquisizione di un lessico specifico, la distinzione e la interpretazione dei linguaggi espressivi e/o informativi, la stesura di relazioni.

**Obiettivi formativi relativi all'agire nella realtà :** *competenze relative alla risoluzione di problemi semplici*, cioè capire la natura di un problema ed individuare le informazioni utili alla risoluzione; riflettere sulle relazioni alla base del problema e tenerne conto nella ricerca della soluzione; verificare la corrispondenza tra la soluzione e tutte le esigenze poste; trasformare le informazioni in una rappresentazione formale diversa; rielaborare anche in modo critico informazioni provenienti da fonti diverse; risolvere problematiche utilizzando dati e strumenti diversi; argomentare una problematica utilizzando le informazioni a disposizione

➤ Obiettivi in termini di abilità:

- Utilizzare la terminologia tecnica in modo appropriato
- Codificare e decodificare numeri e codici
- Riconoscere un errore sintattico o semantico
- Saper creare software per risolvere i diversi problemi
- Saper utilizzare le strutture fondamentali della programmazione
- Saper utilizzare i diagrammi di flusso per rappresentare gli algoritmi
- Saper utilizzare C++ per codificare gli algoritmi
- Scrivere algoritmi utilizzando le funzioni
- Utilizzare funzioni predefinite nei programmi
- Utilizzare funzioni ricorsive
- Trasformare funzioni iterative in ricorsive
- Scrivere, leggere e ricercare dati da un vettore
- Realizzare pagine HTML con contenuti multimediali
- Creare pagine web in HTML con link, tabelle, moduli e riquadri
- DIV Definire semplici moduli HTML

➤ Obiettivi in termini di conoscenze:

- Conoscere i metodi di rappresentazione dei dati e delle informazioni all'interno dell'elaboratore
- Comprendere la differenza tra compilatore ed interprete
- Conoscere i principi fondamentali della programmazione e le modalità di rappresentazione delle strutture fondamentali
- Individuare le diverse fasi di realizzazione di un programma
- Sapere progettare algoritmi riferiti a semplici problemi, scelti anche fra i temi proposti nelle altre discipline scolastiche
- Saper codificare in C++ gli algoritmi progettati
- Comprendere il meccanismo del passaggio dei parametri
- Comprendere le differenze tra il passaggio per valore per referenza
- Comprendere le regole di visibilità
- Individuare un problema ricorsivo
- Comprendere la differenza tra ricorsione e iterazione
- Come funziona il web
- I principali tag HTML

Il ruolo delle pagine web HTML  
La comunicazione nel web attraverso il browser

**Obiettivi MINIMI ABILITA' E CONOSCENZE "IRRINUNCIABILI" importanti e discriminanti ai fini del "successo formativo".**

- Obiettivi minimi in termini di abilità:

Utilizzare la terminologia tecnica in modo appropriato  
Codificare e decodificare semplici numeri e codici  
Riconoscere i più comuni errori sintattici  
Saper creare software per risolvere i diversi problemi  
Saper utilizzare le strutture fondamentali della programmazione nella risoluzione di problemi parzialmente corretti  
Saper utilizzare i diagrammi di flusso per rappresentare gli algoritmi nella risoluzione di problemi parzialmente corretti  
Saper utilizzare C++ per codificare gli algoritmi nella creazione di programmi parzialmente corretti  
Scrivere algoritmi utilizzando semplici funzioni  
Utilizzare le principali funzioni predefinite nei programmi  
Utilizzare semplici funzioni ricorsive  
Realizzare pagine HTML con contenuti multimediali  
Creare semplici pagine web in HTML con link, tabelle, moduli e riquadri DIV  
Definire semplici moduli HTML

- Obiettivi minimi in termini di conoscenze:

Conoscere i metodi di rappresentazione dei dati e delle informazioni all'interno dell'elaboratore e saperlo fare in modo parzialmente corretto  
Comprendere la relazione tra algoritmo e programma  
Acquisire il concetto di linguaggio di programmazione  
Comprendere le principali funzioni di un compilatore  
Conoscere la simbologia dei diagrammi di flusso  
Conoscere i principi fondamentali della programmazione e le modalità di rappresentazione delle strutture fondamentali  
Individuare le diverse fasi di realizzazione di un programma  
Sapere progettare algoritmi riferiti a semplici problemi, scelti anche fra i temi proposti nelle altre discipline scolastiche  
Saper codificare in C++ gli algoritmi progettati  
Comprendere il meccanismo del passaggio dei parametri  
Comprendere le differenze tra il passaggio per valore e per referenza  
Individuare, guidati, un problema ricorsivo  
Comprendere la differenza tra ricorsione e iterazione  
Come funziona il web  
I principali tag HTML  
Il ruolo delle pagine web HTML  
La comunicazione nel web attraverso il browser

CRITERI E MODALITA' di VALUTAZIONE:

- trattazione sintetica di argomenti
- quesiti a risposta singola
- quesiti a risposta multipla
- problemi a soluzione rapida
- esercitazioni pratiche in laboratorio/al PC a casa

UNITÀ DI APPRENDIMENTO Titolo	OBIETTIVI PERSEGUITI
<p><b>1. ALGORITMI</b>  Analisi del problema.  I tipi di dati: numerici (interi, reali), alfanumerici (caratteri, stringhe)  Dati di input, output e interni  Variabili e costanti  Regole per gli identificatori.  Le istruzioni di input, output e assegnazione  Il flow-chart per rappresentare la risoluzione dei problemi  Definizione di algoritmo  Le strutture fondamentali e il teorema di Bohm-Jacopini  La sequenza  La selezione  L'iterazione  Simulazione di un algoritmo  Esercitazioni con la realizzazione del flow-chart, della tabella di I/O, della tabella di descrizione dei dati e della simulazione  Esercitazioni in laboratorio</p>	<p>Conoscere i metodi di rappresentazione dei dati all'interno dell'elaboratore  Utilizzare la terminologia informatica  Descrivere la soluzione di semplici problemi mediante algoritmi  Affrontare in modo sistemico il problema dall'analisi del problema all'algoritmo</p>
<p><b>1. PROGRAMMARE in C++</b>  <b>Linguaggio C++</b>  Sezioni di un programma  Librerie del linguaggio  Dichiarazione delle variabili e costanti  I commenti  Indentare i programmi  Istruzioni di Input, Output e assegnazione  Gli operatori interi tra numeri interi: divisione e modulo  La selezione  Strutture iterative: pre-condizionale, post condizionale e ciclo enumerativo  La selezione multipla: l'istruzione switch  La variabile flag  Risoluzione esercizi  <b>Metodo top-down</b>  Lo sviluppo top-down: i sottoprogrammi  Le funzioni: dichiarazione e chiamata  Variabili locali e globali  I parametri attuali e formali  Il passaggio dei parametri per valore e referenza  Struttura del main di un progetto  Le variabili booleane  Simulazione di programmi  <b>La ricorsione</b></p>	<p>Utilizzare la tecnica top-down per descrivere gli algoritmi  Utilizzare uno strumento di editing visuale  Definire una funzione  Definire la modalità del passaggio dei parametri  Distinguere i parametri formali e attuali  Organizzare un programma con menù</p>

<p><b>IL LINGUAGGIO HTML</b>  Primi passi con l'HTML  Creazione di una pagina  La sintassi HTML  L'intestazione di un documento  Il corpo del documento  I paragrafi del testo e la formattazione del testo  Definizione del carattere  Le immagini  Elenchi numerati e puntati  I collegamenti ipertestuali</p>	<p>Distinguere i principali marcatori HTML  Definire pagine HTML con una struttura a link  Applicare i fogli di stile alle pagine web  Saper distinguere il significato dei diversi marcatori HTML  Utilizzare elementi multimediali in HTML</p>
--	--

**METODI, STRUMENTI E SUSSIDI:**

- Metodi
  - lezione frontale
  - lezione interattiva
  - problem solving
  - lavoro di gruppo
  - discussione guidata
  
- Strumenti
  - libro di testo
  - dispense e appunti
  - presentazioni multimediali
  - video
  - Aula virtuale - piattaforma Moodle
  - GSuite
  
- Spazi
  - aula
  - laboratorio
  
- Attività
  - Utilizzare GSuite
  - Utilizzare la piattaforma di e-learning Moodle
  - Utilizzare le risorse online del libro di testo adottato
  - Installare C++
  - Installare Notepad ++

ATTIVITA' DI RECUPERO, APPROFONDIMENTO E/O POTENZIAMENTO Saranno inizialmente previsti tentativi di recupero in itinere, mediante forme di didattica che preveda eventuale personalizzazione dei percorsi relativi ai programmi svolti, in cui si tenterà di commisurare gli argomenti alle capacità individuali, salva restando la valutazione oggettiva dell'impegno.

In tutte le esercitazioni svolte in classe e in laboratorio, e quindi anche prima di ogni verifica scritta, si effettuerà un ripasso e consolidamento degli argomenti della verifica coinvolgendo l'intera classe e lasciando spazio agli allievi che possano proporre domande su quanto non è risultato chiaro.

Genova, 28 novembre 2022

Il docente (firma)

(Prof.ssa Alessandra Narbona)