

Istituto: “Galilei – Artiglio”, Viareggio

Indirizzo di studi: Liceo Scientifico **Classe:** 2AS (a.s. 2024 / 2025)

Docente: Vincenzo Santoro

Disciplina: Informatica

Programma Svolto

1. Informatica e problem solving

- 1.1 L'informatica come scienza dell'informazione e dell'elaborazione
- 1.2 Concetto di problema e strategia di risoluzione
- 1.3 Fasi del *problem solving* (analisi – progettazione – implementazione – verifica)
- 1.4 Modellizzazione: astrazione, individuazione di input, vincoli, output attesi
- 1.5 Metodi euristici per la scelta della strategia risolutiva
- 1.6 Ruolo di **risolutore** ed **esecutore** nei sistemi informatici

2. Gli algoritmi

- 2.1 Azioni elementari e istruzioni
- 2.2 Definizione formale di algoritmo e proprietà (finitezza, non ambiguità, efficacia)
- 2.3 Rappresentazioni:

- Diagrammi di flusso (blocchi ISO)
- Pseudocodice strutturato
- 2.4 Variabili e costanti: tipologie, dichiarazione, ambito
- 2.5 Classificazione delle istruzioni:

- di inizio/fine
- operative (assegnazione, input, output)
- dichiarative
- 2.6 Strutture di controllo fondamentali:
- sequenza
- selezione
- iterazione (cicli)

3. Dall'algoritmo al programma

- 3.1 Evoluzione dei linguaggi di programmazione
- 3.2 Compilazione vs interpretazione
- 3.3 Introduzione al linguaggio C++: caratteristiche principali
- 3.4 Struttura generale di un sorgente (`#include`, `using`, `int main()`, `return 0`)
- 3.5 Identificatori in C++: variabili, costanti simboliche, funzioni
- 3.6 Dichiarazione ed inizializzazione delle variabili (tipi fondamentali)
- 3.7 Dichiarazione delle costanti (`const`)
- 3.8 Flusso di input/output con `cin` e `cout`
- 3.9 Operatori aritmetici, relazionali, logici
- 3.10 Costrutti di selezione (`if`, `if...else`, nidificazioni)
- 3.11 Costrutti di iterazione (`for`, `while`, `do...while`) con esempi pratici

4. Laboratorio e progetti svolti

- Traduzione di algoritmi in codice C++; esercizi su calcoli aritmetici, gestione di input/output formattato.
- Programmi con struttura sequenziale, selettiva e iterativa:
 - calcolo di statistiche (max/min)
 - verifica divisibilità e numeri primi
 - conversioni di basi numeriche
- Realizzazione di semplici giochi testuali (indovina-numero)
- Progetto finale: analisi, progettazione e codifica di un applicativo che elabora dati inseriti dall'utente usando array e funzioni di ordinamento (integrazione facoltativa per gli alunni interessati).

5. Competenze e abilità acquisite

- Analizzare un problema, individuare dati in ingresso/uscita e vincoli.
- Elaborare soluzioni algoritmiche corrette e strutturate.
- Rappresentare algoritmi con diagrammi e pseudocodice.
- Sviluppare, compilare e testare programmi C++ semplici, usando correttamente variabili, operatori e strutture di controllo.
- Documentare il codice e presentare la soluzione in modo chiaro.

6. Strumenti e metodologie didattiche

Lezioni frontali, esercitazioni guidate in aula informatica, lavori di gruppo, utilizzo del compilatore g++ e dell'IDE Code::Blocks, valutazioni formative con *problem solving* su carta e prove pratiche al computer.

7. Verifiche e criteri di valutazione

- Prove scritte: analisi di algoritmi, traduzione in pseudocodice/flow-chart.
- Prove pratiche: realizzazione e debugging di programmi in C++.
- Interrogazioni orali su teoria e riflessione metacognitiva sulla soluzione adottata.
- Griglia di valutazione basata su correttezza, efficienza, stile del codice e capacità di spiegazione.

8. Ed. Civica

- Cyberbullismo
- Generazioni e gap generazionale nell'era digitale
- Lavori del futuro nel settore digitale e impatto sociale
- Digitalizzazione elemento di disuguaglianza sociale
- Copyright e impatto della pirateria sulle imprese

Data: 06-06-2025

Firma del docente: [Firma]

Firma studenti: [Firma]
[Firma]