

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**

Docenti: NARBONA Alessandra

DI BRIGIDA Claudio

Disciplina d'insegnamento: TECNOLOGIE INFORMATICHE

Classe: 1B

Modalità per l'analisi della situazione iniziale:

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

Obiettivi formativi specifici e disciplinari:

1. **Obiettivi educativi:** correttezza, cioè rispetto delle persone, 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.
2. **Obiettivi didattici:** 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 presentazioni
3. **Obiettivi interdisciplinari:** competenze relative alla risoluzione di problemi semplici, cioè capire la natura di un problema ed individuare le informazioni utili alla risoluzione, trasformare le informazioni in una rappresentazione formale diversa, combinare e sintetizzare informazioni provenienti da fonti diverse, risolvere problematiche semplici utilizzando dati e strumenti diversi, argomentare una problematica utilizzando le informazioni a disposizione.

4. **Obiettivi in termini di abilità:**

- Utilizzare la terminologia tecnica in modo appropriato
- Codificare e decodificare numeri e codici
- Identificare componenti di un computer
- Confrontare le caratteristiche tecniche principali delle singole componenti
- Riconoscere le caratteristiche del Sistema Operativo
- 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 Scratch per codificare gli algoritmi

5. **Obiettivi in termini di conoscenze:**

- Conoscere i sistemi di numerazione decimale, binario, ottale, esadecimale
- Conoscere i metodi di rappresentazione dei dati e delle informazioni all'interno dell'elaboratore
- Conoscere l'architettura e i componenti fondamentali di un computer
- Conoscere le memorie: tipologia e capacità
- Conoscere le periferiche e le loro caratteristiche
- Conoscere le norme riguardanti la sicurezza dei dati, il diritto d'autore e gli aspetti legali
- Saper utilizzare le funzioni del Sistema Operativo
- Riconoscere il ruolo dei Sistemi operativi e quali sono i tipi più diffusi
- Conoscere le caratteristiche del desktop di Windows
- Saper distinguere i tipi di file in base all'estensione
- Riconoscere il significato e la struttura delle directory
- Identificare gli elementi dell'interfaccia grafica di Windows
- Identificare il significato del carattere jolly
- Creare e lavorare con i file (documenti, cartelle di lavoro, presentazioni, etc.)
- Acquisire la definizione e le caratteristiche di un algoritmo

Comprendere la relazione tra algoritmo e programma  
Acquisire il concetto di linguaggio di programmazione  
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 Scratch gli algoritmi progettati

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

### **1. Obiettivi in termini di abilità:**

Utilizzare la terminologia tecnica in modo sufficientemente appropriato  
Codificare e decodificare semplici numeri e codici  
Identificare le principali componenti di un computer  
Confrontare le caratteristiche tecniche principali delle singole componenti  
Riconoscere le principali caratteristiche del Sistema Operativo  
Saper creare semplici 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 Scratch per codificare gli algoritmi nella creazione di programmi parzialmente corretti

### **2. Obiettivi in termini di conoscenze:**

Conoscere i sistemi di numerazione decimale, binario, ottale, esadecimale  
Conoscere i metodi di rappresentazione dei dati e delle informazioni all'interno dell'elaboratore e saperlo fare in modo parzialmente corretto  
Conoscere l'architettura e i componenti fondamentali di un computer  
Conoscere le memorie: tipologia e capacità  
Conoscere le periferiche e le loro principali caratteristiche  
Conoscere le principali norme riguardanti la sicurezza dei dati, il diritto d'autore e gli aspetti legali  
Saper utilizzare le funzioni principali del Sistema Operativo  
Riconoscere il ruolo dei Sistemi operativi e quali sono i tipi più diffusi  
Conoscere le caratteristiche principali del desktop di Windows  
Saper distinguere i tipi di file in base all'estensione  
Riconoscere il significato e la struttura delle directory  
Identificare i principali elementi dell'interfaccia grafica di Windows  
Creare e lavorare con i file (documenti, cartelle di lavoro, presentazioni, etc.)  
Acquisire la definizione e le caratteristiche di un algoritmo  
Comprendere la relazione tra algoritmo e programma  
Acquisire il concetto di linguaggio di programmazione  
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 Scratch gli algoritmi progettati

## LABORATORIO

- Alfabetizzazione informatica:
  - Hardware e Software
  - Pc e sue componenti
  - Tipi di memorie
  - Cache
  - ROM
  - RAM
  - Hard Disk
  - SSD
- Il Sistema Operativo Windows
  - Un pò di terminologia (Desktop, barra delle applicazioni, finestre, barra del titolo, barra dei menù barre di scorrimento, etc etc)
- Cosa è una Suite (Office e LibreOffice)
- Tipologie di licenze sull'uso del software
- Introduzione a CALC (con alcuni riferimenti a Excel)
  - le celle
  - i riferimenti
  - le espressioni
  - le funzioni principali
  - i grafici
- MIT Scratch
  - Fondamenti di programmazione visuale
  - l'ambiente di sviluppo
  - primi programmi

## Parte TEORICA

### SISTEMI di NUMERAZIONE POSIZIONALI

Definizione di sistema di numerazione posizionale

I sistemi di numerazione decimale, binario, ottale ed esadecimale

Conversioni da decimale (parte intera e parte frazionaria) a binario, ottale, esadecimale e viceversa Addizione e sottrazione nei sistemi di numerazione decimale, binario, ottale ed esadecimale Conversione da binario a ottale e esadecimale e viceversa

### CODIFICA DELLE INFORMAZIONI NUMERICHE E ALFANUMERICHE

I dati e le informazioni

Dati numerici e alfanumerici

Rappresentazione delle informazioni alfanumeriche: ASCII, ASCII esteso e UNICODE

Rappresentazione dei numeri interi: valore assoluto e segno e complemento a due Rappresentazione dei numeri reali: notazione esponenziale normalizzata

### CONCETTI di BASE DELLA TECNOLOGIA DELL'INFORMAZIONE

#### Software

I tipi di software: software di base e software applicativi

I Sistemi Operativi e le loro principali funzioni

Graphical User Interface (GUI)

Software applicativi: le applicazioni, le suite, i Computer Based Training

Le fasi di sviluppo del software

I virus informatici, come si propagano

Millenium bug

Informatica e diritti d'autore, licenza d'uso del software, il copyright, software open source, freeware, shareware, i Creative Commons

## **Hardware**

I tipi di computer: Mini computer, Mainframe, Super computer, terminali stupidi ed intelligenti, desktop, laptop, tablet, netbook

I telefoni cellulari, PDA, smartphone

Il case (pannello anteriore e posteriore)

Le porte: seriali, parallele, USB, di rete

La scheda madre e i bus

La memoria interna (RAM e ROM), la memoria cache

Unità di misura della memoria e suoi multipli

Memorie di massa: hard disk interni/esterni, floppy, memory card, unità di rete , CD-ROM, DVD, pendrive, Blu-ray, nastro

L'Unità centrale di elaborazione, il ciclo macchina, la velocità del processore

Dispositivi di input: tastiera, mouse, trackball, touchpad, joystick , microfono, webcam e scanner (OCR) Dispositivi di output: monitor (dimensioni, frequenza di refresh, risoluzione

schermo, colori), stampanti ad aghi, getto di inchiostro, laser e 3D, altoparlanti, plotter, sintetizzatori vocali Dispositivi di input e output: modem, touch screen, stampante

multifunzione

Prestazioni di un PC

## **ALGORITMI**

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

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

## **Algebra di Boole**

Gli operatori logici: congiunzione, disgiunzione e negazione

Proposizioni logiche semplici e composte

Operatori relazionali

## **SCRATCH**

L'interfaccia di Scratch

Lo stage, gli sprite, gli script

Dal flow-chart alla codifica in Scratch

Istruzioni di Input, Output e assegnazione

La sequenza

La selezione, le selezioni in cascata e annidate

Esercitazioni su PC

## **ATTIVITÀ**

Utilizzare Gsuite

Utilizzare la piattaforma di e-learning Moodle

Scaricare Scratch e unirsi alla comunità online di Scratch

Genova, 28 novembre 2022

Il docente (firma)

(Prof.ssa Alessandra Narbona)

Il docente (firma)

(Prof. Claudio Di Brigida)