

ANNO SCOLASTICO: 2025-2026

PROGRAMMA EFFETTIVAMENTE SVOLTO
Docente Biagioli Mirko
Classe 3F
Materia Informatica

Architettura dei calcolatori

Approfondimenti sul funzionamento del calcolatore: ciclo di clock, transistor, registri, memoria cache, memoria ram, memoria di massa.

Ripasso programmazione C

Concetto di algoritmo, programma, processo. Stack. Ambiente di sviluppo DEV-C++ e compilatore gcc. Vettori e funzioni. Puntatori. Passaggio di parametri per valore e per riferimento. Struct C.

Introduzione alla programmazione Java

Conoscenze:

Brevi cenni storici. Vantaggi/svantaggi del linguaggio Java

La compilazione intermedia. Utilizzo del prompt dei comandi di Windows.

Concetti fondamentali: classe, metodo, oggetto e sue caratteristiche

L'ambiente di sviluppo Eclipse. Struttura di un programma

Principali elementi della sintassi

Tipi di dati fondamentali; identificatori

I tipi primitivi di dato. Operazioni sui dati. Stringhe

Gestione I/O su console. L'ambiente di sviluppo

Competenze:

Utilizzo elementi fondamentali programma Java

Acquisire i concetti di oggetto e di classe

Impiegare il metodo main

Saper utilizzare correttamente le regole sintattiche del linguaggio Java

Saper utilizzare i vari tipi di variabili, gli operatori e le librerie del linguaggio Java

Conoscere i tipi di dati complessi e le operazioni di input e output

Realizzare semplici programmi in linguaggio Java

Apportare modifiche in programmi già realizzati, per ottenere le funzioni desiderate

Utilizzare oggetti in classi definite

Java: algoritmi

Conoscenze:

Introduzione alle strutture di controllo

Strutture di controllo in Java

Strutture di controllo derivate



Competenze:

Saper utilizzare le strutture condizionali nella programmazione Java
Saper utilizzare le varie strutture di controllo nella programmazione Java
Ideare e implementare algoritmi più complessi in Java

Progettazione di classi

Conoscenze:

Variabili d'esemplare. Incapsulamento ed information hiding

Realizzare la classe. Gestione ingresso/uscita

Ereditarietà e polimorfismo. Gerarchie di ereditarietà

Realizzare sottoclassi. Sovrascrivere metodi.

Array di oggetti.

Classi astratte.

Interface. Gestione delle eccezioni.

Liste dinamiche (cenni)

Competenze:

Utilizzare i costruttori

Getters/Setters

Utilizzare le istruzioni di input/output

Definire una classe derivata

Ridefinire metodi.

Saper utilizzare gli array per organizzare le classi del progetto utilizzando aggregazione/composizione.

Linguaggio Python

Caratteristiche e costrutti di base del linguaggio

Definizione di classi ed ereditarietà

liste di oggetti.

Programmi ad oggetti in Python con interazione utente in modalità testuale.

Cuneo, lì 10/6/2026.

Firma del Docente: Mirko Biagioli