



A.S. 2020/2021

PERCORSO FORMATIVO DISCIPLINARE

Classe	3 DS A
Disciplina	Informatica
Docenti	Prof.ssa Maria Rita Di Berardini, Prof.ssa Maura Rogante.

OBIETTIVI DIDATTICI DISCIPLINARI

Gli alunni hanno acquisito conoscenze, abilità e competenze relative alle seguenti tematiche disciplinari:

- algoritmi e problemi;
- rappresentazione di algoritmi tramite diagrammi a blocchi e pseudocodifica;
- tipi di dato e dichiarazione di variabili e costanti;
- operatore di assegnamento;
- operatori di confronto e logici;
- strutture di controllo: if, cicli while, do-while e for;
- array in C;
- strutture in C.

METODOLOGIA DI LAVORO

La metodologia didattica ha previsto l'approccio ai diversi contenuti partendo dall'illustrazione di problemi connessi con la realtà, analizzati criticamente e alla luce delle conoscenze attuali. Gli alunni sono stato stimolati ad esprimere le proprie opinioni e a confrontarsi (tra di loro e con i docenti) in maniera serena e costruttiva. Si è fatto ricorso alla metodologia del "*problem solving*" e alla "*didattica laboratoriale*" e alla "*didattica per problemi*", abituando gli alunni a utilizzare e riorganizzare le conoscenze già acquisite per formulare ipotesi di risoluzione di un problema, in modo da fare acquisire loro una metodologia scientifica.

I docenti hanno inoltre dedicato molto tempo a far acquisire agli studenti una corretta metodologia di lavoro, metodologia che si compone di una serie di fasi da eseguire in sequenza per una corretta comprensione del problema e per la definizione di una corretta strategia risolutiva: analisi del problema e definizione di una strategia risolutiva, definizione di un algoritmo risolutivo (tramite diagrammi a blocchi e poi tramite pseudocodifica) e codifica in un linguaggio di programmazione.

CONTENUTI (ORGANIZZATI IN MODULI DIDATTICI)

MODULO 1: Problemi e algoritmi

- Il concetto di algoritmo e le proprietà di un algoritmo;
- Fasi di risoluzione di un problema: analisi dei dati del problema, identificazione di sottoproblemi, definizione di un algoritmo risolutivo;
- Rappresentazione di algoritmi tramite diagrammi a blocchi e pseudocodifica;

MODULO 2: Introduzione alla programmazione in C

- Struttura di un programma C: la libreria `stdio.h` e la funzione `main()`;
- La funzione `printf()`. Dichiarazione di variabili e costanti in C;
- Tipi di dato numerici: `int`, `float` e `double`;
- Operatore di assegnamento e operatori aritmetici;
- Acquisizione di dati in input con l'istruzione `scanf()`.

Fasi di sviluppo di un programma:

- Editing, compilazione e debug (o correzione degli errori);
- Il comando `gcc` in ambiente Linux.

Strutture di controllo in C:

- Istruzione `if`, `while`, `do-while` e `for`;
- Operatori di confronto (`>`, `<`, `>=`, `<=`, `==` e `!=`);
- Operatori logici: `&&` (and), `||` (or) e `!` (not). Espressioni condizionali.

MODULO 3: Array in C

- Il concetto di array;
- Dichiarazione di un array in C;
- Inizializzazione di array in C con valori costanti e con valori forniti in input dall'utente;
- Algoritmi su array: calcolo della somma, del valore medio, calcolo dell'elemento minimo e dell'elemento massimo;
- Generazione di numeri pseudo-casuali in C: funzioni `rand()` e `srand()`; la libreria `time.h` e la funzione `time()`;
- Inizializzazione di un array con valori pseudo-casuali;
- Algoritmo per il calcolo delle frequenze.

MODULO 4: Strutture e stringhe in C

Strutture in C

- Il concetto di struttura;
- Dichiarazione di strutture in C: la parola chiave `struct` e campi di una struttura;
- Dichiarazione e inizializzazione di variabili di tipo `struct`;
- Accesso ai campi di una variabile di tipo `struct` (dotted notation);

- Array di struct. Inizializzazione e manipolazione di array di struct.

Stringhe in C

- Il tipo di dato char e la codifica dei caratteri in C (codice ASCII).
- Le stringhe in C e la libreria *string.h*;
- Funzioni standard per la manipolazione di stringhe: *strcmp()*, *strlen()*, *strcpy()*.

Fermo, 05 Giugno 2021

Gli insegnanti:
Maria Rita Di Berardini,
Maura Rogante