

**PIANI DI STUDIO DI ISTITUTO – SECONDO CICLO
ISTITUTO COMPRENSIVO DI PRIMIERO**

Corso: Liceo Scienze Appl. Disciplina Informatica e Sistemi - 1° biennio del secondo ciclo

CONOSCENZE e ABILITÀ da promuovere nel 1° BIENNIO, attraverso le attività di insegnamento della programmazione annuale,

in vista dell'acquisizione delle COMPETENZE previste al termine del 1°BIENNIO SUPERIORE

PREMESSA:

L'insegnamento di informatica ha molteplici obiettivi: comprendere i fondamenti teorici delle scienze dell'informazione, acquisire la padronanza di strumenti dell'informatica da utilizzare per la soluzione di problemi anche connessi allo studio di altre discipline, scegliere i componenti hardware e software più adatti alle diverse situazioni e le loro configurazioni, valutarne le prestazioni e mantenerli in efficienza, acquisire la consapevolezza dei vantaggi e dei limiti dell'uso dell'informatica e delle tematiche sociali e culturali ad esso connesse, possedere i principi scientifici che stanno alla base delle strutture informatiche e delle loro applicazioni.

COMPETENZE al termine del 1° BIENNIO SUPERIORE

Al termine del percorso lo studente deve:

- comprendere della natura e degli scopi della disciplina e delle entità coinvolte in essa;
- essere in grado di valutare le caratteristiche dei diversi sistemi di elaborazione, con relative periferiche, in modo da scegliere i più adatti in base alle varie esigenze di uso;
- avere consapevolezza del modo in cui l'elaboratore gestisce i dati alfanumerici e numerici e dei limiti oggettivi nella rappresentazione dei valori da parte dell'elaboratore;
- essere consapevole del ruolo del software in un sistema di elaborazione e delle caratteristiche e funzionalità del sistema operativo; apprezzare le differenze e le similarità tra sistemi operativi diversi e saper scegliere il giusto software in base alle esigenze di elaborazione;
- individuare metodi di soluzione di semplici problemi e rappresentarli in modo rigoroso sotto forma di algoritmi;

- usare il programma Scratch per creare animazioni;
- saper tradurre procedimenti risolutivi (algoritmi) in programmi scritti con un linguaggio che segue il paradigma imperativo (linguaggio C);
- essere in grado di svolgere le operazioni di generazione di programmi eseguibili usando ambienti IDE;
- utilizzare le strutture di selezione e iterazione per realizzare programmi che risolvano problemi di tipo principalmente matematico.

Articolazione delle conoscenze e abilità

CONOSCENZE al termine del 1° BIENNIO SUPERIORE	ABILITA' al termine del 1° BIENNIO SUPERIORE
<p>Classe prima</p> <ul style="list-style-type: none"> • Definizioni generali e cenni alla nascita dell'informatica. • Trattamento automatico delle informazioni. • Accezioni dell'informatica: strumentale, tecnologica, scientifica. • Informazioni e dati. • Cenni alla teoria dei sistemi. • Hardware, software e sua classificazione. • Definizione e ruolo di sistema operativo. <ul style="list-style-type: none"> • La macchina di Von Neumann. • Cenni all'evoluzione e alla classificazione dei computer. • Scheda madre. • Componenti e clock della CPU. • Memoria centrale RAM e ROM, cache; tipi di accesso alle memorie. • Memorie di massa: magnetiche, ottiche, elettroniche. • Gerarchia delle memorie. • Periferiche di I/O. <ul style="list-style-type: none"> • Analogico e digitale • Sistemi di codifica. • Codici ASCII e ASCII esteso. • Codici Unicode e UTF-8. • Sistemi di numerazione posizionali. 	<p>Classe prima</p> <ul style="list-style-type: none"> • Essere consapevoli delle caratteristiche e delle funzioni dei componenti fondamentali del computer. • Sapere usare in modo appropriato le periferiche di memorizzazione dei dati e quelle per l'I/O. <ul style="list-style-type: none"> • Saper stimare l'occupazione di memoria necessaria per memorizzare in digitale vari tipi di informazioni. • Sapere convertire valori dalla base 10 ad altre basi e viceversa e saper fare semplici calcoli in binario, ottale, esadecimale. • Saper convertire i valori decimali, interi e frazionari usando gli

- Numeri binari, ottali, esadecimali; conversioni.
- Calcoli con valori in basi diverse dal 10.
- Rappresentazione degli interi in complemento a due.

- Classificazione del software.
- Linguaggi di programmazione.
- Struttura a livelli del sistema operativo.
- Funzioni del sistema operativo come gestore di risorse e interfaccia verso l'utente.
- Il BIOS.
- Tipologie di sistemi operativi; introduzione a Linux.

- Proprietà degli algoritmi.
- Rappresentazione degli algoritmi con i flow-chart.
- Strutture di controllo fondamentali.
- Ruolo delle variabili.
- Uso di Scratch per la costruzione di script di animazione con strutture iterative.
- Manipolazione delle variabili in Scratch.

Classe seconda

- Linguaggi a basso e ad alto livello; generazioni di linguaggi.
- Paradigmi di programmazione: il paradigma imperativo.
- Traduzione dei programmi: assembleri, compilatori e interpreti; linguaggi ibridi.

- Formalizzazione degli algoritmi con Algobuild.
- Sintassi del linguaggio C.
- Ambienti IDE in Windows (Dev-C++) e Linux (Geany).
- Struttura del programma, commenti e uso delle variabili.
- Istruzioni per l'I/O.
- Operatori matematici e casting.
- Compilazione e creazione del programma eseguibile.
- Selezione ad una via e a due vie, selezione multipla.
- Operatori logici.
- Iterazione precondizionale, postcondizionale e calcolata.

standard previsti dal computer.

- Saper installare e disinstallare software.
- Saper utilizzare un sistema di elaborazione in modalità multiboot con più sistemi operativi.

- Saper costruire algoritmi di soluzione di problemi rappresentandoli rigorosamente con i flow-chart.
- Sapere usare l'interfaccia di programmazione a blocchi di Scratch per acquisire i principi di base della programmazione.

Classe seconda

- Saper distinguere tra linguaggi compilativi e interpretativi valutando pro e contro delle due soluzioni

- Saper usare correttamente un ambiente IDE sfruttandone le facilitazioni per l'editing dei programmi e la generazione degli eseguibili.
- Rispettare la sintassi del linguaggio C.

- Variabili strutturate: i vettori.
- Ruolo della variabile indice.
- Inserimento, scorrimento e manipolazione di vettori.

- Saper usare correttamente le strutture di selezione e iterazione per risolvere semplici problemi con uso di vettori.