

CONTENUTI, CONOSCENZE, COMPETENZE DI INFORMATICA PER IL SECONDO BIENNIO DEL LICEO SCIENTIFICO OPZIONE SCIENZE APPLICATE

| CONOSCENZE | ABILITÀ | COMPETENZE DISCIPLINARI | COMPETENZE EUROPEE |
|--|---|---|--|
| <p>Modulo AL1 - Introduzione alla programmazione – Gli algoritmi</p> <ul style="list-style-type: none"> • Concetto di algoritmo • Caratteristiche fondamentali di un algoritmo • Le proprietà fondamentali di un algoritmo • La rappresentazione grafica degli algoritmi • Strutture fondamentali in un algoritmo: sequenza, selezione e iterazione • Concetto di loop infinito | <ul style="list-style-type: none"> • Tradurre un problema in algoritmo • Valutare se un algoritmo soddisfa i suoi requisiti fondamentali | <p>Fare l'analisi di un problema e formalizzare l'algoritmo risolutivo</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |
| <p>Modulo AL2 – Primi elementi di programmazione</p> <ul style="list-style-type: none"> • Linguaggi di programmazione e fasi della programmazione • Utilizzo di un IDE per la scrittura di un programma • Definizione dei tipi di dato • Istruzioni di input e output • Assegnazione | <ul style="list-style-type: none"> • Utilizzare un ambiente di sviluppo: scrittura, compilazione ed esecuzione di un programma • Distinguere e scegliere i vari tipi di dato • Utilizzare le istruzioni di acquisizione dei dati e gestire ed organizzare la visualizzazione dei risultati • Scrivere espressioni aritmetiche • Conoscere la sintassi delle istruzioni di base della programmazione (input, output e assegnazione) | <p>Acquisire familiarità con la struttura di un programma e apprendere i primi elementi di un linguaggio</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |
| <p>Modulo AL2 – Programmazione strutturata</p> <ul style="list-style-type: none"> • La struttura sequenziale • La struttura di selezione: tipi di selezione (semplice, annidata e multipla) • Operatori aritmetici e relazionali • Algebra booleana e operatori logici • Iterazione precondizionale e postcondizionale • Iterazione enumerativa | <ul style="list-style-type: none"> • Saper tradurre un semplice algoritmo in programma • Saper scrivere semplici programmi che utilizzino le fondamentali strutture di programmazione • Saper utilizzare e organizzare le istruzioni di sequenza, selezione e iterazione • Saper generare numeri casuali nei programmi che li richiedono • Essere in grado di individuare le strutture più idonee alla stesura di un algoritmo risolutivo • Scegliere con consapevolezza tra cicli con controllo in testa e con controllo in coda • Saper scrivere correttamente espressioni aritmetiche e logiche • Saper riconoscere e correggere i principali errori segnalati dal compilatore • Utilizzare e organizzare le strutture iterative • Individuare le strutture iterative più idonee alla stesura di un algoritmo risolutivo | <p>Apprendere gli elementi fondamentali di un linguaggio di programmazione</p> <p>Conoscere le fondamentali strutture di programmazione (sequenza, selezione, iterazione)</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |

| | | | |
|--|--|--|--|
| <p>AL3 – Procedure e funzioni</p> <ul style="list-style-type: none"> • Principi della programmazione top-down • Definizione e sintassi di utilizzo di funzioni e procedure • Ambito delle variabili: variabili globali e locali • Valori di ritorno delle funzioni • Passaggio di parametri: passaggio per valore e per riferimento • Funzioni ricorsive • Librerie utente | <ul style="list-style-type: none"> • Sapere organizzare un programma, suddividendolo in moduli • Utilizzare la sintassi per la dichiarazione e l'invocazione dei sottoprogrammi • Distinguere variabili locali e globali • Gestire il passaggio di parametri • Gestire l'utilizzo di una funzione ricorsiva • Creare e utilizzare una propria libreria di sottoprogrammi | <p>Sapere organizzare in sottoprogrammi un programma complesso</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |
| <p>Modulo AL4 – Dati strutturati: array (vettori e matrici) e stringhe</p> <ul style="list-style-type: none"> • Array monodimensionali • Array bidimensionali • Array paralleli • Algoritmi fondamentali <ul style="list-style-type: none"> ○ Shift ○ Merge ○ Rotazione • Algoritmi di ordinamento • Algoritmi di ricerca elementare <p>Stringhe e funzioni di manipolazione delle stringhe</p> | <ul style="list-style-type: none"> • Definire, caricare e utilizzare un array monodimensionale e bidimensionale • Costruire algoritmi con l'impiego di strutture di dati • Stimare la complessità computazionale di un algoritmo • Utilizzare le principali funzioni di manipolazione delle stringhe | <p>Conoscere la differenza tra variabili semplice e variabile strutturata</p> <p>Conoscere le modalità e il significato dell'organizzare i dati in un vettore e in una matrice</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |
| <p>Modulo AL5 – Approfondimenti sui dati strutturati: array</p> <ul style="list-style-type: none"> • Algoritmi di ricerca <ul style="list-style-type: none"> ○ sequenziale ○ a blocchi ○ binaria • Algoritmi di ordinamento <ul style="list-style-type: none"> ○ per inserimento ○ per selezione ○ bubblesort | <ul style="list-style-type: none"> • Definire, caricare e utilizzare un array monodimensionale e bidimensionale • Costruire algoritmi di ricerca ottimali sugli array • Stimare la complessità computazionale di un algoritmo | <p>Saper creare e scegliere gli algoritmi fondamentali ottimali sugli array</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |
| <p>Modulo AC3 – La rappresentazione delle informazioni all'interno del computer</p> <ul style="list-style-type: none"> • I circuiti logici. Le porte AND, OR e NOT • Il circuito half adder e full adder • Codifica dell'informazione • Rappresentazione delle informazioni alfanumeriche • Rappresentazione binaria dei numeri interi • Rappresentazione dei numeri reali (virgola fissa, virgola mobile, eccesso 127, eccesso 1023) | <ul style="list-style-type: none"> • Saper costruire e interpretare un semplice circuito logico • Saper rappresentare con software dedicati semplici circuiti logici (es. Logisim) • Saper comprendere le tecniche per la rappresentazione dei dati all'interno del computer • Saper codificare un'informazione nella rappresentazione interna al computer • Saper decodificare un'informazione partendo dalla sua rappresentazione interna al computer | <p>Sapere come il computer rappresenta nella memoria le informazioni numeriche</p> | <ul style="list-style-type: none"> • competenze di base in scienze e tecnologia • competenza digitale • imparare a imparare |

Indicativamente moduli AL1, AL2, AL3 vengono affrontati in terza; i restanti moduli AC3, AL4 E AL5 in quarta.

Obiettivi minimi classe 3°

Programmazione strutturata

- Utilizzare e organizzare tutte le strutture iterative studiate per la soluzione di problemi semplici
- Individuare le strutture iterative più idonee alla stesura di un algoritmo risolutivo

Procedure e funzioni

- Principi della programmazione top-down
- Definizione e sintassi di utilizzo di funzioni e procedure
- Ambito delle variabili: variabili globali e locali
- Valori di ritorno delle funzioni
- Passaggio di parametri: passaggio per valore e per riferimento

Obiettivi minimi classe 4°

Dati strutturati: array (vettori e matrici) e stringhe

- Scrittura di semplici programmi sugli array monodimensionali
- Scrittura di semplici programmi sugli array bidimensionali (operazioni sulle matrici: somma, differenza prodotto RICO)
- Scrittura di semplici programmi sulle stringhe con l'utilizzo di funzioni di manipolazione delle stesse
- Conoscenza delle logiche di funzionamento degli algoritmi fondamentali di ricerca e di ordinamento nei vettori

La rappresentazione delle informazioni all'interno del computer

- Rappresentazione di un numero intero e reale, così come viene codificato all'interno del computer
- Determinazione di un numero partendo dalla sua rappresentazione interna