

PROGRAMMAZIONE DIDATTICA

ASSE CULTURALE SCIENTIFICO TECNOLOGICO

competenze chiave	competenze base	conoscenze	abilità
<ul style="list-style-type: none"> • Imparare a Imparare • Competenza digitale • Comunicazione nella madrelingua • Comunicazione nelle lingue straniere 	<ul style="list-style-type: none"> • Scrivere un programma in linguaggio di progetto • Editare, testare e collaudare un programma • Effettuare l'input dei dati • Formattare l'output numerico sullo schermo • Scrivere programmi con istruzioni in sequenza e in blocchi 	<ul style="list-style-type: none"> • Conoscere i diversi formati di un programma • Descrivere le diverse fasi di sviluppo di un programma • Comprendere il concetto di variabile • Conoscere le istruzioni di comunicazione con l'utente • Comprendere l'importanza del commento del codice • Conoscere la tecnica top-down 	<ul style="list-style-type: none"> • Disporre l'output sullo schermo • Utilizzare le variabili nei programmi • Commentare il codice del programma • Utilizzare variabili intere e reali • Utilizzare l'operatore modulo (%) sui numeri interi
<ul style="list-style-type: none"> • Imparare a Imparare • Competenza digitale • Comunicazione nella madrelingua • Comunicazione nelle lingue straniere 	<ul style="list-style-type: none"> • Riconoscere quando utilizzare selezioni annidate • Codificare la selezione semplice e doppia • Effettuare l'annidamento delle istruzioni • Utilizzare le variabili di tipo bool • Utilizzare gli operatori logici 	<ul style="list-style-type: none"> • Comprendere l'importanza del commento del codice • Conoscere l'istruzione di selezione semplice e doppia • Conoscere le variabili di tipo bool e gli operatori logici • Comprendere il concetto di annidamento • Conoscere la notazione per i 	<ul style="list-style-type: none"> • Scrivere un codice con istruzioni condizionali • Scrivere un codice complesso con blocchi di istruzioni annidate • Combinare più selezioni con condizioni logiche composte • Scrivere un codice utilizzando l'istruzione di selezione multipla

		blocchi di istruzioni	
<ul style="list-style-type: none"> • Imparare a Imparare • Competenza digitale • Comunicazione nella madrelingua • Comunicazione nelle lingue straniere 	<ul style="list-style-type: none"> • Saper scegliere il tipo di iterazione adeguato alle diverse situazioni • Generare numeri casuali • Codificare l'iterazione indefinita • Codificare l'iterazione definita • Codificare programmi con cicli annidati 	<ul style="list-style-type: none"> • Acquisire il concetto di iterazione • Conoscere le diverse tipologie di iterazione • Conoscere la differenza tra controllo di testa e di coda • Conoscere la differenza tra iterazione definita e indefinita 	<ul style="list-style-type: none"> • Scrivere programmi con selezioni e iterazioni • Progettare programmi con cicli annidati • Utilizzare diverse tipologie di iterazione nello stesso programma • Utilizzare la trace table per individuare errori nel codice
<ul style="list-style-type: none"> • Imparare a Imparare • Competenza digitale • Comunicazione nella madrelingua • Comunicazione nelle lingue straniere 	<ul style="list-style-type: none"> • Definire una funzione • Definire la modalità del passaggio dei parametri • Distinguere i parametri formali e attuali • Organizzare un programma con menu 	<ul style="list-style-type: none"> • Comprendere il meccanismo del passaggio dei parametri • Comprendere le differenze tra il passaggio per indirizzo e per valore • Comprendere le regole di visibilità • Individuare un problema ricorsivo • Comprendere la differenza tra ricorsione e iterazione • 	<ul style="list-style-type: none"> • Scrivere algoritmi utilizzando le funzioni • Utilizzare funzioni predefinite nei programmi • Utilizzare funzioni personali
<ul style="list-style-type: none"> • Imparare a Imparare • Competenza digitale • Comunicazione nella madrelingua 	<ul style="list-style-type: none"> • Saper organizzare tipi di dati complessi • Definire array monodimensionali e bidimensionali 	<ul style="list-style-type: none"> • Saper gestire dati di tipo diverso in relazione tra loro • Avere il concetto di archivio • Conoscere le tipologie di accesso ai dati 	<ul style="list-style-type: none"> • Scrivere, leggere e ricercare dati da un array • Operare con stringhe e sottostringhe

<ul style="list-style-type: none">• Comunicazione nelle lingue straniere	<ul style="list-style-type: none">• Ricercare le simmetrie in una matrice quadrata• Definire una stringa• Definire record e strutture dati complessi	<ul style="list-style-type: none">• Riconoscere l'importanza dell'archiviazione dei dati• Conoscere la gestione dei dati sul supporto di massa	<ul style="list-style-type: none">• Utilizzare liste, tuple, set e dizionari
--	--	---	--

CONTENUTI DEL PROGRAMMA:

Programmare in Python:

- Istruzioni in sequenza
- La selezione
 - La selezione semplice
 - La selezione multipla
 - La selezione annidata
 - Operatori di confronto
 - Operatori logici
- Le iterazioni
 - Ciclo while
 - Ciclo for
- Le funzioni
- Strutture dati complesse
 - Creazione degli array in Python
 - Operazioni con gli array
 - Le stringhe
 - Liste e tuple
 - I set
 - I dizionari

Attività del docente e metodologia	Attività dello studente	Materiali e spazi utilizzati
<ul style="list-style-type: none"> • Esporre le conoscenze agli studenti nelle lezioni frontali utilizzando il supporto multimediale. • Assegnare agli alunni i riferimenti sul libro e i compiti settimanali da svolgere. • Assicurarsi che gli studenti comprendano le consegne • Programmare le unità di apprendimento. • Creare motivazioni, stimolare all'invenzione, orientare le attività pratiche. Valutare i risultati delle unità di apprendimento. • Insistere sull'uso di un linguaggio corretto • Insegnare a prendere appunti, a recepire, di un testo, ciò che è significativo • Favorire il dialogo in classe, incentivare la curiosità e sviluppare la creatività • Sviluppare la capacità di individuare, scegliere, utilizzare informazioni di varia natura • Fornire gli strumenti utili alla risoluzione di problemi • Riprogrammare le eventuali unità di apprendimento di recupero per gli alunni che non hanno conseguito gli obiettivi perseguiti 	<ul style="list-style-type: none"> • Comprendere un messaggio orale o scritto • Utilizzare il linguaggio proprio della disciplina • Esporre con chiarezza il proprio pensiero • Prendere appunti, annotare informazioni e saperli riorganizzarli in schemi logici, mappe concettuali, mappe mentali (usando anche tecnologie appropriate) • Scoprire analogie e differenze, sviluppare un modello utile alla risoluzione del problema in esame • Partecipare positivamente all'attività didattica, ponendo domande, proponendo soluzioni, riflettendo sugli interventi di docente e compagni • Lavorare in gruppo, condividendo informazioni e azioni e accogliendo suggerimenti o ipotesi alternative • Effettuare semplici esperimenti e trarne conclusioni pertinenti • Acquisire un metodo di studio proficuo che gli consenta di razionalizzare il tempo dedicato allo studio e di giudicare il livello della propria preparazione per condurlo a una corretta autovalutazione 	<ul style="list-style-type: none"> • Libri di testo, e-book, presentazioni PowerPoint • Laboratori e PC in ambiente Windows e Mac • Proiettore • Software di laboratorio

VALUTAZIONE:

CONOSCENZE	VAL.	ABILITA'	VAL.	COMPETENZE	VALUTAZIONE
Espone ed inquadra nel corretto contesto gli argomenti trattati. Usa con proprietà il linguaggio scientifico anche in lingua	9-10	Applica i procedimenti risolutivi in modo corretto e approfondito. Utilizza in modo completamente autonomo i software di laboratorio. Usa i formalismi dell'informatica in modo corretto.	9-10	Utilizza in modo completamente autonomo le strategie del pensiero razionale per risolvere problemi elaborando le opportune soluzioni dal punto di vista informatico.	Eccellente/ottimo
Espone in modo corretto gli argomenti trattati ed utilizza il lessico in modo appropriato anche in lingua	8	Utilizza in modo autonomo i software di laboratorio. Individua in modo autonomo le soluzioni ai problemi proposti e le realizza dal punto di vista informatico	8	Utilizza in modo autonomo le strategie del pensiero razionale per risolvere problemi elaborando le opportune soluzioni dal punto di vista informatico	buono
Espone in modo corretto gli argomenti trattati ed utilizza il lessico in modo accettabile anche in lingua	7	Utilizza in modo adeguato i software di laboratorio. Individua con discreta autonomia le soluzioni ai problemi proposti e le realizza dal punto di vista informatico.	7	Utilizza in modo corretto le strategie del pensiero razionale per risolvere problemi ed è in grado di elaborare in modo autonomo le opportune soluzioni dal punto di vista informatico.	discreto
Coglie gli elementi di base dell'argomento trattato. Qualche incertezza nell'uso del linguaggio tecnico.	6	Utilizza in modo parzialmente adeguato i software di laboratorio. Individua con parziale autonomia le soluzioni ai problemi proposti e le realizza dal punto di vista informatico.	6	Utilizza in modo parzialmente autonomo le strategie del pensiero razionale per risolvere problemi elaborando in modo quasi autonomo le opportune soluzioni dal punto	sufficiente

				di vista informatico.	
Conosce in modo frammentario l'argomento trattato. Usa il linguaggio tecnico in modo incerto	5	Utilizza in modo non completamente adeguato i software di laboratorio. Individua con difficoltà le soluzioni ai problemi proposti e le realizza dal punto di vista informatico.	5	Utilizza in modo improprio le strategie del pensiero razionale per risolvere problemi e elabora con errori le opportune soluzioni dal punto di vista informatico	mediocre
Conosce in modo lacunoso l'argomento trattato. Gravi improprietà nell'uso del linguaggio tecnico.	4	Utilizza in modo inadeguato i software di laboratorio. Non è in grado di individuare le soluzioni ai problemi proposti e li implementa con gravi errori.	4	Utilizza in modo improprio le strategie del pensiero razionale per risolvere problemi e elabora con gravi errori le opportune soluzioni dal punto di vista informatico	insufficiente
Conosce in modo gravemente lacunoso l'argomento trattato. Gravi improprietà nell'uso del linguaggio tecnico.	2-3	Ignora completamente il funzionamento dei software di laboratorio. Non è in grado di individuare le soluzioni ai problemi proposti e non è nemmeno in grado di implementarli.	2-3	Non è in grado di utilizzare le strategie del pensiero razionale per risolvere problemi e elabora con gravi errori le opportune soluzioni dal punto di vista informatico	Gravemente insufficiente