



MINISTERO DELLA ISTRUZIONE DELL'UNIVERSITA' E RICERCA
UFFICIO SCOLASTICO REGIONALE PER IL LAZIO
ISTITUTO DI ISTRUZIONE SUPERIORE "I.T.C. DI VITTORIO - I.T.I. LATTANZIO"

Via Teano, 223 - 00177 Roma ☎ 06121122405 / 06121122406- fax 062752492

Cod. Min. RMIS00900E ✉ rmis00900e@istruzione.it - rmis00900e@pec.istruzione.it

Cod. fiscale 97200390587

PROGRAMMAZIONE DIDATTICA

Materia	INFORMATICA
Classe	3H Lattanzio
Anno scolastico	2018/2019
Docente	<i>Pellecchia Carolina</i>

OBIETTIVI SPECIFICI DI APPRENDIMENTO

Competenze:

- Acquisizione di una metodologia di sviluppo del problema
- Formalizzazione del procedimento risolutivo
- Validazione del procedimento risolutivo
- Individuazione delle risorse informatiche
- Utilizzo degli strumenti informatici in relazione all'analisi dei dati e alla modellazione dei problemi

Conoscenze:

- Relazioni fondamentali tra macchine, problemi, informazioni e linguaggi
- Linguaggi e macchine a vari livelli di astrazione
- Paradigmi di programmazione
- Logica iterativa e ricorsiva
- Principali strutture dati e loro implementazione
- Strumenti per lo sviluppo del software e supporti per la robustezza dei programmi
- Lessico e terminologia tecnica di settore anche in lingua inglese

Abilità

- Saper implementare applicazioni in modalità console
- Applicare i principi di base della logica formale per studiare un algoritmo
- Codificare gli algoritmi e validare i programmi effettuando le necessarie correzioni
- Produrre un'efficace documentazione contestualmente allo sviluppo dei progetti software
- Identificare e ad applicare le metodologie e le tecniche per la progettazione e la realizzazione di pagine Web.

ARTICOLAZIONE ORARIA

Sono previste **1** ora di teoria e **1** ore di esercitazioni in laboratorio

ANALISI DELLA SITUAZIONE DI PARTENZA DELLA CLASSE

Profilo generale della classe

La classe è composta da 20 alunni di cui 2 provenienti da altre scuole, 3 ripetenti provenienti da questo istituto.

Buono appare il livello di partecipazione e la curiosità cognitiva degli alunni che affrontano con interesse le attività proposte. Dal punto di vista disciplinare la classe sa, in generale, seguire le norme che regolano la vita scolastica.

Gli alunni, fin dai primi giorni, non hanno evidenziato problemi relazionali e di socializzazione.

Il livello di partenza della classe rilevato, nella fase iniziale dell'anno scolastico, è che la maggior parte degli alunni si dimostra in possesso di adeguate competenze, mentre alcuni di essi manifestano difficoltà ed incertezze.

Alunni con disturbi specifici dell'apprendimento:

In questa classe non sono presenti alunni con bisogni educativi speciali.

UNITÀ DI APPRENDIMENTO DEL PERCORSO FORMATIVO

Le unità di apprendimento prevedono lezioni teoriche e lezioni pratiche di laboratorio

UNITÀ DI APPRENDIMENTO	PERIODO DI SVOLGIMENTO	COMPETENZE
A1 – LE BASI DEL LINGUAGGIO	Seconda metà di settembre ottobre	Avere una visione su: <ul style="list-style-type: none">• sistema di elaborazione e logica di funzionamento,• caratteristiche di un problema e modo di approcciare alla programmazione
Conoscenze		Abilità
<ul style="list-style-type: none">• Caratteristiche dei linguaggi C e C++• Passaggi fondamentali nello sviluppo del software: dal file sorgente al file eseguibile• Differenze tra compilatori e interpreti• La struttura di un programma in C/C++• Variabili, costanti e loro dichiarazione• La gestione dell'I/O in C e C++		<ul style="list-style-type: none">• Saper gestire un IDE per lo sviluppo di applicazioni C/C++• Saper impostare e comprendere la struttura di base di un programma C/C++• Saper tradurre un algoritmo in linguaggio C/C++• Riconoscere le direttive al preprocessore e la sezione dichiarativa all'interno di un programma C/C++• Riuscire a codificare semplici programmi che fanno uso di istruzioni di I/O• Comprendere le situazioni in cui si necessita di un casting
UNITÀ DI APPRENDIMENTO	PERIODO DI SVOLGIMENTO	COMPETENZE
A2 – LE ISTRUZIONI	Novembre	<ul style="list-style-type: none">• Pervenire alla traduzione di algoritmi utilizzando la logica di base dei linguaggi di programmazione.• Individuare la struttura generale di un programma in linguaggio C++ e le caratteristiche principali dei dati, delle istruzioni e degli operatori.• Scrivere i programmi in C++ utilizzando in modo corretto la sintassi delle istruzioni di input/output e delle strutture di controllo• Riconoscere le diverse fasi del lavoro di programmazione per codificare e validare gli algoritmi

Conoscenze		Abilità
<ul style="list-style-type: none"> • La struttura di selezione unaria e binaria • La struttura di selezione multipla • I vari tipi di costrutti iterativi • I costrutti selezione e iterazione in linguaggio C/C++ 		<ul style="list-style-type: none"> • Saper realizzare algoritmi che si servono del costrutto selezione • Saper scegliere tra i vari costrutti iterativi • Codificare in linguaggio C/C++ algoritmi che fanno uso di costrutti di selezione e iterativi
UNITÀ DI APPRENDIMENTO	PERIODO DI SVOLGIMENTO	COMPETENZE
A3 – LE FUNZIONI	Dicembre e prima metà di gennaio	<ul style="list-style-type: none"> • Scomporre il programma in funzioni e riutilizzare più volte le stesse funzioni assegnando diversi valori ai parametri • Organizzare dati dello stesso tipo o di tipo diverso, associando ad ogni situazione problematica la struttura di dati più idonea
Conoscenze		Abilità
<ul style="list-style-type: none"> • Metodologia top down • Scomposizione di problemi in sottoproblemi • Ambienti di visibilità delle variabili • Tipi di sottoprogrammi • Le procedure • Le funzioni void • Il passaggio dei parametri • Le function • La ricorsione 		<ul style="list-style-type: none"> • Saper scomporre un problema in sottoproblemi • Utilizzare consapevolmente le variabili locali e le variabili globali • Riconoscere procedure e funzioni • Implementare funzioni void e funzioni non void in C/C++ • Saper gestire il passaggio dei parametri • Riconoscere le differenze nel passaggio dei parametri in C e in C++ • Saper implementare funzioni ricorsive
UNITÀ DI APPRENDIMENTO	PERIODO DI SVOLGIMENTO	COMPETENZE
A4 – LE STRUTTURE DI DATI	Seconda metà di gennaio e prima metà di marzo	<ul style="list-style-type: none"> • Riconoscere l'esigenza di usare un vettore di variabili • Individuare l'esigenza di usare un algoritmo di ordinamento • Analizzare un problema e definire un nuovo tipo di dati
Conoscenze		Abilità
<ul style="list-style-type: none"> • Le strutture di dati: proprietà e caratteristiche • I vettori • Tecniche di gestione dei vettori • Tecniche di ordinamento di un vettore • Metodologie di ricerca delle informazioni all'interno di vettori ordinati e disordinati • Le stringhe • I vettori di caratteri • Le matrici • Operazioni sulle matrici 		<ul style="list-style-type: none"> • Realizzare algoritmi che fanno uso di strutture di dati • Implementare vettori in C/C++ • Realizzare algoritmi che usano tecniche di ricerca e ordinamento di strutture • Codificare algoritmi di ricerca e di ordinamento in C/C++ • Saper gestire le stringhe e vettori di caratteri anche in C/C++
UNITÀ DI APPRENDIMENTO	PERIODO DI SVOLGIMENTO	COMPETENZE
C1 – LA PROGETTAZIONE DI UN SITO WEB	Seconda metà di marzo e prima metà di aprile	<ul style="list-style-type: none"> • Operare con informazioni, documenti e oggetti multimediali in formato Web da pubblicare nei siti Internet

Conoscenze	Abilità
<ul style="list-style-type: none"> • L'ipertesto e le sue caratteristiche • La multimedialità • L'ipermedia e le sue caratteristiche • Tecniche di progettazione web • Architettura client server • Siti statici e siti dinamici • Servizi di housing e servizi di hosting • La pubblicazione del sito • Il dominio del sito web • I servizi offerti dalla rete 	<ul style="list-style-type: none"> • Saper distinguere ipertesti e ipermedia • Riconoscere le funzionalità e le innovazioni apportate dall'introduzione della multimedialità • Saper progettare un sito web • Saper scegliere i servizi web più utili per soddisfare differenti esigenze • Riconoscere il dominio di un sito web

UNITÀ DI APPRENDIMENTO	PERIODO DI SVOLGIMENTO	COMPETENZE
C2 – LINGUAGGI PER IL WEB: XHTML E CSS	Seconda metà di aprile e prima metà di giugno	<ul style="list-style-type: none"> • Utilizzare strumenti e linguaggi per personalizzare il layout e lo stile delle pagine Web • Creare contenuti web in grado di adattarsi graficamente in modo automatico al dispositivo con il quale vengono visualizzati
Conoscenze	Abilità	
<ul style="list-style-type: none"> • Caratteristiche e funzionalità dei linguaggi di markup • Il linguaggio XHTML(XML+HTML) • Regole di base del linguaggio XHTML • Elementi e tag • Le tabelle in XHTML • Le immagini in XHTML • La gestione dei collegamenti ipertestuali in XHTML 	<ul style="list-style-type: none"> • Riconoscere le caratteristiche dei linguaggi di markup al fine di distinguerli dai linguaggi di programmazione • Saper scrivere codice in XML + HTML • Identificare elementi e tag • Realizzare pagine web accattivanti • Saper inserire tabelle e immagini in una pagina XHTML • Saper gestire i collegamenti ipertestuali in XHTML 	

OBIETTIVI MINIMI

Conoscenze	Abilità
<ul style="list-style-type: none"> • Dati e informazioni • Concetti di problem solving • Algoritmi ed esecutori • La rappresentazione degli algoritmi • Schemi delle strutture di controllo dell'algoritmo • Codifica dell'algoritmo • Fasi di sviluppo di un programma • Ambiente di sviluppo di un programma in linguaggio C++ • Operazioni standard di input e output • Codifica delle strutture di controllo dell'algoritmo (sequenza, selezione, ripetizione) • Le funzioni in C++ : definizione e chiamata • Passaggio dei parametri per valore e per riferimento • Dati elementari e strutturati: gestione di stringhe di caratteri e di array monodimensionali • Algoritmi standard di ricerca e ordinamento • Il Web: browser, motori di ricerca, URL e DNS • Struttura di una pagina XHTML • Tag per inserimento di testo, elenchi, tabelle, link e immagini 	<ul style="list-style-type: none"> • Collaborare attivamente in un gruppo di lavoro • Riconoscere e usare correttamente la terminologia di base dell'Informatica • Formulare i passi per risolvere problemi, individuando i dati necessari e l'obiettivo da raggiungere • Impostare problemi con tecnica procedurale adeguata all'esecutore automatico PC • Rappresentare l'algoritmo risolutivo di un problema con diagrammi di flusso • Verificare la correttezza di una soluzione • Interagire con il Sistema Operativo installato sul PC in uso • Utilizzare l'ambiente di programmazione • Codificare algoritmi in C++ utilizzando le strutture di controllo dell'algoritmo, le funzioni, le variabili semplici, le stringhe e gli array unidimensionali • Documentare software a livello elementare • Navigazione consapevole in Internet per il recupero di informazioni • Distinguere lato client e server in un'applicazione web • Strutturare una pagina web statica con testo, immagini, elenchi, tabelle, link e sfondi

METODOLOGIE DIDATTICHE PREVISTE

- Lezione frontale e interattiva con l'uso della lim
- Apprendimento cooperativo attraverso la quale gli studenti apprendono in piccoli gruppi, aiutandosi reciprocamente e sentendosi corresponsabili del reciproco percorso
- Soluzione di problemi reali/ Problem solving che consente di analizzare, affrontare e cercare di risolvere positivamente situazioni problematiche. Obiettivi: trovare la soluzione e rendere disponibile una descrizione dettagliata del problema e del metodo per risolverlo.
- Brain-storming per consentire il confronto tra gli studenti e la valutazione di nuove idee.
- Gruppi di lavoro

Le lezioni si svolgeranno in laboratorio di informatica come previsto dall'orario scolastico.

STRUMENTI E ATTREZZATURE NECESSARIE ALLO SVOLGIMENTO DEL PERCORSO FORAMTIVO

- libro di testo
Informatica app 2. Vol. unico. Con CD-ROM. Con e-book. Con espansione online
Autori: Piero Gallo, Pasquale Sirsi
Casa editrice: Minerva Scuola
ISBN 9788829846979 - Prezzo: € 30,15
- contenuti digitali integrativi del libro di testo
- materiali tratti da Internet
- laboratorio di Informatica con
 - *PC connessi ad Internet*
 - *Browser web Chrome*
 - *Compilatore C++*
 - *Ambiente di sviluppo IDE per C++ (DEV C++)*
 - *Editor di pagine HTML(XHTML), file CSS e script JavaScript*
- LIM

TIPOLOGIA DELLE PROVE DI VERIFICA PREVISTE

Le prove di verifica che si intende utilizzare sono:

- Prove pratiche (laboratorio): lavori individuali e/o di gruppo, relazioni individuali.
- Prove orali: interrogazioni, interventi significativi e partecipazione al dialogo educativo.

Numero **minimo** di prove per quadrimestre: **2** (1 orale, 1 pratica).

CRITERI DI VALUTAZIONE, INDICATORI E GRIGLIE

Per i criteri di valutazione ci si atterrà a quelli illustrati nel P.O.F.