

ITCS Germano Sommeiller – Torino
A.S. 2023-2024

Classe: QUARTA SISTEMI INFORMATIVI AZIENDALI	Materia: INFORMATICA
Ore di lezione annue previste: 165	
Libro di testo: “EPROGRAM – SIA” – volume unico Jacobelli, Aime, Marrone – ed. Juvenilia – ISBN 9788874855575	
Bibliografia e sitografia: www.generazioniconnesse.it www.apprendereinrete.it www.test-ecdl.it http://sottositi.loescher.it/ecdl/ http://www.atuttascuola.it/ecdl/ http://www.auladigitale.rcs.it/special/tutorial/ www.raiscuola.rai.it www.html.it Videolezioni sul web segnalate dal docente durante l’anno scolastico	
Materiale didattico: libro di testo, manuali on line, dispense del docente, articoli di giornale, libri di saggistica e narrativa	
Profilo previsto dello studente a inizio anno: Lo studente ha conoscenza dei concetti teorici di base relativi ad hw e sw; sa applicare una metodologia per l’analisi di problemi elementari (sequenziali, condizionali e ciclici) al fine di ottenere un algoritmo risolutivo performante e documentarlo in modo chiaro ed esaustivo; conosce le problematiche connesse alla risoluzione di problemi mediante un computer e comprende la differenza tra fase di analisi, fase di implementazione e fase di esecuzione di un programma; comprende il ruolo del compilatore nello sviluppo dei programmi e ne sa utilizzare le principali funzionalità; sa implementare algoritmi sequenziali, condizionali e ciclici mediante un linguaggio di programmazione (preferibilmente visuale e object oriented) e mediante l’applicativo EXCEL; sa utilizzare le principali funzionalità di un ambiente di programmazione per il linguaggio scelto; sa progettare e realizzare interfacce user friendly per la comunicazione tra utente e programma; conosce la differenza tra dati elementari e dati strutturati; sa utilizzare le variabili statistiche contatore, accumulatore, minimo, massimo, media e percentuale.	
Profilo previsto dello studente al termine dell’anno scolastico: L’allievo conosce il ciclo di vita del sw e le problematiche ad esso connesse; sa applicare la metodologia TOP-DOWN per la progettazione di sistemi complessi e la loro documentazione; conosce e sa utilizzare le strutture logiche dei dati; sa implementare sistemi complessi mediante un linguaggio di programmazione (preferibilmente visuale e object oriented), conosce gli scopi di un sistema di gestione delle basi di dati e i suoi vantaggi rispetto ad un Foglio Elettronico nella gestione di archivi aziendali; sa operare con un DBMS relazionale (livello base) per creare la struttura di semplici Data Base ed estrarre report professionali dai dati dell’archivio; conosce i principali elementi del linguaggio HTML e sa realizzare semplici pagine WEB rispondenti ad una specifica.	
Modalità di recupero delle eventuali carenze formative degli studenti: recupero in itinere degli studenti anche con utilizzo di tutoraggio da parte delle eccellenze; supporto mediante esercitazioni e documentazione aggiuntive; sportello su richiesta degli studenti; indicazione di sitografia e bibliografia su specifici argomenti.	

Schema delle competenze da acquisire nel corso dell'anno¹

Modulo		Competenze Professional (CP)	Competenze relazionali (CR)	Competenze di Cittadinanza (CC)	Competenze Metacognitive (CM)	Comp.ze del benessere
1	RACCORDO con la classe precedente. Verifica obiettivi propedeutici	X	X	X	X	
2	Strutture Dati Logiche	X	X		X	
3	Ingegneria del Software: Metodologie e tecniche di progettazione	X	X		X	
4	Data Base e DBMS (concetti di base e strumenti)	X	X	X	X	
5	Introduzione alla Multimedialità	X	X	X	X	
6	Attività STEM	X	X	X	X	

¹ Si veda dettaglio nella descrizione dei singoli moduli

Modulo 1

Titolo: RACCORDO CON LA CLASSE PRECEDENTE **Verifica raggiungimento obiettivi propedeutici**

COMPETENZE

tutte quelle previste nei precedenti anni scolastici propedeutiche alle competenze dell'anno in corso (CP, CC, CM, CR, CB)

	Conoscenze (Sapere)	Abilità (Saper fare-saper essere)
	<ul style="list-style-type: none">• Conoscere le fasi della risoluzione di un problema mediante computer (figure, risultati e linguaggi coinvolti)• Comprendere la differenza tra linguaggi ad alto livello e linguaggi a basso livello e problematiche connesse• Comprendere la differenza tra linguaggi assembler e linguaggi macchina• Conoscere la differenza tra costanti e variabili• Comprendere la differenza tra dati di input, dati di output e variabili ausiliarie• Conoscere le fasi di una metodologia per l'analisi e la documentazione di un problema semplice• Comprendere la differenza tra errori sintattici ed errori semantici e le metodologie per individuarli• Conoscere le strutture fondamentali della programmazione strutturata• Comprendere la differenza tra struttura if ... then e struttura if ... then ... else• Comprendere la differenza tra ciclo FOR, ciclo per vero e ciclo per falso• Conoscere le variabili statistiche contatore, accumulatore, minimo, massimo, media e percentuale• Conoscere le istruzioni di base di un linguaggio di programmazione• Comprendere la differenza tra sorgente ed eseguibile	<ul style="list-style-type: none">• Associare ad ogni fase dello sviluppo di un programma figure, risultati e linguaggi coinvolti• Classificare in modo adeguato i dati di un problema• Applicare i passi della metodologia per l'analisi e la documentazione di un problema a casi via via più complessi• Saper progettare e implementare interfacce user friendly per la comunicazione utente/programma• Saper scegliere, in base al problema, le strutture fondamentali più adeguate alla sua soluzione• Valutare la correttezza sintattica e la correttezza semantica di un algoritmo risolutivo• Saper documentare i dati di una soluzione in presenza di cicli• Saper calcolare le variabili statistiche contatore, accumulatore, minimo, massimo, media e percentuale in presenza di dati elementari• Saper associare ad ogni dato elementare il tipo corretto• Saper utilizzare correttamente le istruzioni fondamentali del linguaggio di programmazione per implementare algoritmi via via più complessi• Saper utilizzare adeguatamente l'ambiente di programmazione per realizzare e testare programmi

Obiettivi Essenziali

	<ul style="list-style-type: none"> • Conoscere le funzioni di base dell'ambiente di programmazione del linguaggio scelto 	
Pre-requisiti	NESSUNO	NESSUNO
Tempo	h: 10 Periodo: I periodo	
Articolazione	Argomento U.D.	Tempi U.D.
	<ul style="list-style-type: none"> • Ripasso unità del precedente anno scolastico 	8h
	<ul style="list-style-type: none"> • Verifica sommativa 	3h
Materiali e strumenti	<ul style="list-style-type: none"> • Lezioni in aula e in laboratorio • LIM • Proiettore del laboratorio • Libro di testo, fotocopie e/o dispense in formato elettronico • Attività di laboratorio, sia individuali che in gruppo 	
Verifiche	<ul style="list-style-type: none"> • Scritto /Pratico 1 • Orale (nel corso dell'anno) 	
Valutazione	Si fa riferimento a quanto previsto nel PTOF	

Modulo 2

Titolo: STRUTTURE DATI LOGICHE

COMPETENZE

Comprendere le problematiche connesse alla modellazione di strutture logiche astratte mediante computer (**CP, CM**)

Risolvere mediante computer problemi che facciano uso di strutture logiche complesse (vettori, matrici, tabelle, ...) (**CP, CR, CM**)

Sviluppare sistemi complessi utilizzando un approccio TOP DOWN alla soluzione (**CP, CR, CM**)

	Conoscenze (Sapere)	Abilità (Saper fare-saper essere)
Obiettivi Essenziali	<ul style="list-style-type: none"> • Comprendere la differenza tra dati elementari e dati strutturati • Comprendere la differenza tra strutture dati omogenee (vettori) e non omogenee (record). • Comprendere la differenza tra strutture dati statiche e strutture dati dinamiche • Comprendere la funzione dell'indice in un dato strutturato di tipo vettore • Comprendere il concetto di tracciato record • Modalità di rappresentazione delle informazioni strutturate • Differenza tra strutture dati concrete e strutture dati astratte • Modalità di rappresentazione di strutture dati astratte mediante l'uso di strutture dati concrete. • Costrutti del linguaggio di programmazione per definire dati strutturati • Approccio TOP DOWN allo sviluppo di sistemi complessi • Comprendere i vantaggi di un approccio TOP DOWN 	<ul style="list-style-type: none"> • Utilizzare le strutture dati concrete per realizzare le principali strutture dati astratte: <ul style="list-style-type: none"> o Realizzazione di una tabella mediante vettori paralleli o Realizzazione di una tabella mediante vettori di record • Dichiarare la struttura di un record • Ricerca di un dato in un vettore ordinato e in tabella • Ricerca di un dato in un vettore non ordinato e in tabella • Ordinamento di un vettore e di tabella • Calcolo delle variabili statistiche in presenza di dati strutturati (Media, Max, Min,...)
Pre-requisiti	<ul style="list-style-type: none"> • Conoscere le strutture dati semplici • Conoscere i fondamenti della programmazione strutturata 	<ul style="list-style-type: none"> • Utilizzare correttamente le strutture dati semplici • Utilizzare i fondamenti della programmazione strutturata

		per lo sviluppo di programmi semplici
Tempo	h: 60 Periodo: I periodo	
Articolazione	Argomento U.D.	Tempi U.D.
	<ul style="list-style-type: none"> • Strutture dati concrete omogenee e non omogenee (vettori e record) • Strutture dati astratte • Realizzazione delle strutture dati astratte e loro gestione mediante un approccio TOP DOWN • Sviluppo di sistemi complessi mediante approccio TOP DOWN: problemi riassuntivi da risolvere con uso di dati semplici e strutture fondamentali della programmazione (es. gestione erboristeria con selezioni prodotti in base a categoria, prodotto con prezzo max, min, media, gestione biblioteca etc.) 	5h 5h 5h 45 h
Materiali e strumenti	<ul style="list-style-type: none"> • Lezioni in aula e in laboratorio • LIM • Proiettore del laboratorio • Libro di testo, fotocopie e/o dispense in formato elettronico • Attività di laboratorio, sia individuali che in gruppo 	
Verifiche	<ul style="list-style-type: none"> • Scritto /Pratico 1 • Orale 1 	
Valutazione	Si fa riferimento a quanto previsto nel PTOF	

Modulo 3

Titolo: INGEGNERIA DEL SOFTWARE: METODOLOGIE E TECNICHE DI PROGETTAZIONE

COMPETENZE

Acquisire il concetto di metodologia e le motivazioni che hanno portato alla nascita di un approccio ingegneristico allo sviluppo del sw (CP, CR, CC)

Comprendere l'importanza della realizzazione di sw comprensibile e facilmente modificabile (CP, CR, CM)

Saper progettare e realizzare sw di qualità (CP, CR, CC, CM)

Obiettivi Essenziali	Conoscenze (Sapere)	Abilità (Saper fare-saper essere)
	<ul style="list-style-type: none">• Comprendere l'importanza della realizzazione di software comprensibile e facilmente modificabile• Conoscere le varie fasi che attraversa un programma durante il suo ciclo di vita• Imparare le tecniche e i principali metodi per costruire velocemente programmi chiari, corretti e facilmente modificabili, cioè software di qualità.• Sapere dell'esistenza di strumenti automatici in grado di agevolare queste attività	<ul style="list-style-type: none">• Saper associare ad ogni attività la fase del ciclo di vita in cui viene svolta• Saper realizzare software comprensibile e facilmente modificabile.
Pre-requisiti	<ul style="list-style-type: none">• Conoscere i fondamenti della programmazione strutturata• Conoscere le istruzioni fondamentali di un linguaggio di programmazione	<ul style="list-style-type: none">• Saper realizzare programmi in linguaggio di programmazione
Tempo	h: 10 Periodo: I periodo	
Articolazione	Argomento U.D.	Tempi U.D.

	<ul style="list-style-type: none"> • La produzione e l'ingegneria del software e ciclo di vita del software • Metodologie per la programmazione strutturata (cenni alla tecnologia ad oggetti). • Manutenzione dei programmi e relativa documentazione 	<p>4 h</p> <p>4 h</p> <p>2h</p>
Materiali e strumenti	<ul style="list-style-type: none"> • Lezioni in aula e in laboratorio • LIM • Proiettore del laboratorio • Libro di testo, fotocopie e/o dispense in formato elettronico • Attività di laboratorio, sia individuali che in gruppo • Redazione della documentazione relativa alla risoluzione dei problemi proposti 	
Verifiche	<ul style="list-style-type: none"> • Orale 1 	
Valutazione	Si fa riferimento a quanto previsto nel PTOF	

Modulo 4

Titolo: DATA BASE E DBMS - CONCETTI DI BASE E STRUMENTI

COMPETENZE

Acquisire il concetto di DBMS e comprenderne i vantaggi rispetto ad un foglio elettronico nella gestione dei dati aziendali (CP, CC)

Comprendere le peculiarità di un programma DBMS rispetto agli altri programmi di produttività individuale (CP, CC)

Utilizzo consapevole di strumenti e linguaggi di un DBMS relazionale per creare la struttura dell'archivio aziendale, inserire in esso i dati, manipolarli, generare report professionali a supporto delle attività e delle decisioni aziendali e rendere più fruibili le informazioni da parte dell'utente. (CP, CC, CR, CM)

Utilizzo del linguaggio QBE per interrogare un Data Base e ottenere i dati di interesse (CP, CC, CR, CM)

Modellare i dati di un semplice sistema informativo aziendale mediante il modello Entity Relationship (CP, CC, CR, CM)

Dato un caso aziendale saper determinare la struttura di un archivio adeguata al contesto (senza preoccuparsi di problemi di performance e ottimizzazione) (CP, CC, CM)

Obiettivi Essenziali	Conoscenze (Sapere)	Abilità (Saper fare-saper essere)
	<ul style="list-style-type: none">• Comprendere la differenza tra Data Base e DBMS• Conoscere le principali funzioni di un DBMS• Comprendere il concetto di modello logico con particolare riferimento al Modello Logico Relazionale• Comprendere la struttura di un Data Base in ambiente Relazionale (tabelle, campi, record)• Comprendere il concetto di manipolazione dei dati• Conoscere le procedure per la creazione di un Data Base mediante il DBMS MS ACCESS• Conoscere le procedure per la creazione di una tabella e il significato delle clausole principali (tipo, dimensione, formato)• Comprendere la differenza tra visualizzazione struttura e visualizzazione foglio dati e saperle utilizzare nel giusto contesto	<ul style="list-style-type: none">• Saper assegnare il giusto significato ai concetti tabella, campo, record• Saper creare un Data Base mediante il DBMS MS ACCESS• Saper creare/modificare la struttura di una tabella utilizzando in modo adeguato i diversi tipi di dato• Saper utilizzare in modo opportuno visualizzazione struttura e visualizzazione foglio dati• Saper inserire, modificare, cancellare e aggiungere record in una tabella• Saper determinare il record corrente e il numero totale dei record di una tabella• Saper stampare struttura e record di una tabella• Saper utilizzare adeguatamente la barra dei record per navigare all'interno della tabella• Saper creare in modo automatico la maschera di una tabella

	<ul style="list-style-type: none"> • Comprendere il concetto di record corrente • Comprendere la differenza tra maschera e report • Comprendere la differenza tra etichetta e casella di testo e saperle utilizzare nel giusto contesto • Comprendere la differenza tra Livello di Gruppo e Ordinamento e saperli utilizzare nel giusto contesto • Conoscere i costrutti di base del modello Entity Relationship (ER) 	<ul style="list-style-type: none"> • Saper modificare il layout di una maschera utilizzando in modo opportuno etichette, caselle di testo e proprietà degli elementi • Saper creare un report in modalità guidata usando in modo opportuno livelli di gruppo e ordinamenti • Saper modificare il layout di un report • Saper individuare entità, attributi e relazioni a partire dalla specifica statica del sistema • Usare il QBE per richieste semplici
Pre-requisiti	Conoscere i tipi di dati	<ul style="list-style-type: none"> • Saper distinguere i tipi di dati • Saper utilizzare le principali funzionalità di un Sistema Operativo
Tempo	h: 40 Periodo: II periodo	
Articolazione	Argomento U.D.	Tempi U.D.
	<ul style="list-style-type: none"> • Data Base e DBMS: Definizione, Modelli Logici e Funzioni di un DBMS. • Perché utilizzare un DBMS. • Il DBMS MS ACCESS <ul style="list-style-type: none"> • Tipi di dati di MS ACCESS e confronto con i tipi generali <ul style="list-style-type: none"> • Creazione, salvataggio, Apertura, modifica di un Data Base • Utilizzo oggetti di base di MS Access (tabelle, maschere, report, etichette) • Query in QBE • Applicazioni gestionali mediante MS ACCESS (es. agenzia immobiliare, studio medico, stage studenti, casa editrice, biblioteca etc.) • Introduzione alla modellazione dei dati mediante ER 	<p>2h</p> <p>18h</p> <p>15 h</p> <p>5 h</p>

Materiali e strumenti	<ul style="list-style-type: none"> • Lezioni in aula e in laboratorio • LIM • Proiettore del laboratorio • Libro di testo, fotocopie e/o dispense in formato elettronico Attività di laboratorio, sia individuali che in gruppo
Verifiche	<ul style="list-style-type: none"> • Orale 1 • Scritto/pratico 1
Valutazione	Si fa riferimento a quanto previsto nel PTOF

Modulo 5

Titolo: INTRODUZIONE ALLA MULTIMEDIALITA'

Creare semplici siti WEB utilizzando il linguaggio HTML (CP, CR, CM)

Creare siti WEB professionali con il supporto di un CSM (a scelta degli studenti) (CP, CR, CC, CM)

	Conoscenze (Sapere)	Abilità (Saper fare-saper essere)
	<ul style="list-style-type: none">• Significato di ipertesto, ipermedia e applicazioni multimediali• Software e linguaggi per creare presentazioni per diverse situazioni.• Fasi di sviluppo di un ipermedia• Conoscere gli strumenti standard di un linguaggio per creare presentazioni per diverse situazioni. (HTML5)• Comprendere la differenza tra editor e wordprocessor• Acquisire le principali caratteristiche del CSM scelto	<ul style="list-style-type: none">• Saper usare le funzionalità di base del linguaggio HTML per comporre il testo, formattarlo, inserire grafici e immagini, gestire collegamenti multimediali, strutturare il layout di pagina mediante l'uso dei frame, aggiungere effetti speciali• Saper realizzare/modificare semplici pagine in linguaggio HTML• Saper visualizzare mediante un Browser pagine HTML• Creare un semplice sito mediante il CSM scelto
Obiettivi Essenziali		
Pre-requisiti	<ul style="list-style-type: none">• Conoscenza del Sistema Operativo• Conoscenza del concetto di pathname dei file• Comprensione della differenza tra pathname relativi e pathname assoluti	<ul style="list-style-type: none">• Saper utilizzare le principali funzionalità di un Browser• Saper utilizzare gli strumenti del Sistema Operativo per la gestione di file e cartelle
Tempo	h: 30 Periodo: II periodo	
	Argomento U.D.	Tempi U.D.

Articolazione	<ul style="list-style-type: none"> • Gli ipertesti e gli ipermedia • Strumenti Sw per creare presentazioni • Progettare una presentazione e/o un sito • Principali tag del linguaggio HTML • Realizzazioni di pagine HTML • Realizzazione di un sito Web mediante il programma CSM scelto • <u>PROGETTO CODING GIRLS</u> 	1h 1h 2h 4h 7h 15 h Il progetto verrà svolto solo parzialmente nelle ore curriculari
Materiali e strumenti	<ul style="list-style-type: none"> • Lezioni in aula e in laboratorio • LIM • Proiettore del laboratorio • Libro di testo, fotocopie e/o dispense in formato elettronico • Attività di laboratorio, sia individuali che in gruppo 	
Verifiche	<ul style="list-style-type: none"> • Orale 1 • Scritto/pratico 1 	
Valutazione	Si fa riferimento a quanto previsto nel PTOF	
NOTE	Eventuale realizzazione del sito mediante un CSM rientra nelle attività di PCTO. Pertanto il periodo di realizzazione potrebbe subire delle variazioni in base alle esigenze del referente del progetto.	

6 - ATTIVITA' STEM

Nel corso del secondo periodo didattico gli studenti saranno coinvolti nel progetto CODING GIRLS della Compagnia San Paolo in collaborazione con la Fondazione Mondo Digitale (spendibile anche come ore di PCTO)

Nel corso dell'anno scolastico (compatibilmente con il corretto avanzamento della programmazione didattica) verranno affrontate attività che richiedano l'uso del kit didattico di robotica LEGO Education SPIKE Prime