

a.s. 2019/2020

PROGRAMMAZIONE PER COMPETENZE 2° BIENNIO – 2a CLASSE

Dipartimento: INFORMATICA E TELECOMUNICAZIONI

Materia: **INFORMATICA**

Classi: **IV A, IV B, IV C**

Docenti: Bartolacci M., Mantini S., Alfieri V., Braccacini A., Pigni M.

OBIETTIVI rispetto alle COMPETENZE di CITTADINANZA	COMPETENZE IN AMBITO DISCIPLINARE	ABILITA'	CONOSCENZE	VERIFICHE rispetto alle competenze di ambito disciplinare
Imparare ad imparare Organizzare l'apprendimento scegliendo ed utilizzando varie fonti e modalità Comunicare Comprendere messaggi di genere diverso e con supporti diversi Rappresentare concetti utilizzando linguaggi diversi (verbale, matematico, simbolico- grafico) Risolvere problemi Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni	Utilizzare le tecniche e le strategie del pensiero razionale per affrontare problemi ed elaborare opportune soluzioni	Saper progettare algoritmi. Analizzare e confrontare algoritmi diversi per la soluzione dello stesso problema. Saper scegliere la giusta impostazione nello sviluppo software	MODULO 0: RIPASSO Ripresa ed approfondimento di elementi di programmazione C# introdotti nel precedente anno di corso. MODULO 1: APPROFONDIMENTI E COMPLETAMENTI Completamento della gestione degli array multidimensionali Conoscere i controlli avanzati: DataGridView, TabControl, finestre di dialogo. Creare controlli a Run- Time.	Soluzione di problemi reali di media complessità (Realizzare una applicazione semplice che preveda l'utilizzo degli elementi principali di una interfaccia grafica e l'accesso alla memoria di massa) Verifica sul ripasso Verifica di laboratorio Verifica scritta(ricorsione) <u>Mesi di settembre/ottobre</u>

			Gestione dei file binari La ricorsione: concetti generali; applicazione a problemi matematici.	
Risolvere problemi Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni Comunicare Comprendere messaggi di genere diverso e con supporti diversi Rappresentare concetti utilizzando linguaggi diversi (verbale, matematico, simbolico-grafico) Progettare Elaborare e realizzare progetti utilizzando le conoscenze apprese, verificando i risultati raggiunti	Sapere utilizzare la programmazione orientata agli oggetti(OOP): saper definire le classi di un programma saper organizzare un programma con metodi ed attributi Saper riconoscere le varie parti di un progetto UML	Scrivere programmi utilizzando classi ed oggetti Saper derivare sottoclassi da classi già esistenti Saper progettare il software usando la metodologia UML Definire le classi utilizzando l'UML	MODULO 2: PROGRAMMAZIONE OOP Definizione di Classe e Oggetto Aspetti principali della programmazione ad oggetti: Ereditarietà, Incapsulamento e Polimorfismo Definizione di classi di oggetti in C#, classi derivate, metodi statici e dinamici Classi astratte e interfacce Utilizzo delle metodologie di progettazione standard: diagrammi UML	Realizzare applicazioni object oriented in linguaggio C# di differente difficoltà associando i diagrammi UML Verifica scritta Test su Moodle Verifica laboratorio <u>Mesi di novembre / dicembre</u>
Risolvere problemi Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni	Riconoscere in che modo le strutture astratte di dati sono allocate fisicamente in memoria	Individuare la struttura dati più idonea per rappresentare i dati di un determinato problema	MODULO 3: LE LISTE-IMPLEMENTAZIONE STATICA E DINAMICA	Realizzare applicazioni in C# Verifica scritta Test Moodle

Progettare Elaborare e realizzare progetti utilizzando le conoscenze apprese, verificando i risultati raggiunti	Saper individuare la struttura dati più opportuna per risolvere un problema	Sapere come viene realizzato il puntatore nei diversi linguaggi di programmazione Saper gestire Lista, Pila, Coda sia con implementazione dinamica che statica	Teoria: Limiti della allocazione statica della memoria. Concetto di puntatore Concetto di lista concatenata e sua implementazione tramite strutture statiche e dinamiche Principali operazioni sulle liste concatenate Implementazione dinamica di strutture LIFO e FIFO Liste di oggetti Uso dei files per il salvataggio di tabelle o liste Laboratorio Collezioni predefinite in C# Laboratorio: Definizione delle strutture ad albero e la loro implementazione	ALMENO <u>Mese di gennaio/febbraio</u>
Imparare ad imparare Organizzare l'apprendimento scegliendo ed utilizzando varie fonti e modalità Risolvere problemi	Utilizzare le tecniche e le strategie del pensiero razionale per affrontare problemi ed elaborare opportune soluzioni	Saper progettare una semplice base di dati Saper operare con i principali operatori relazionali Saper normalizzare una relazione	MODULO 4: LE BASI DI DATI Conoscere il concetto di database e DBMS e le sue funzionalità. Conoscere i concetti di base relativi ai principali	Dall'analisi di una semplice situazione reale realizzare uno schema E/R, rappresentarlo secondo le regole del modello relazionale ed effettuare operazioni relazionali

Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni Progettare Elaborare e realizzare progetti utilizzando le conoscenze apprese, verificando i risultati raggiunti	Scegliere dispositivi e strumenti in base alle loro caratteristiche e funzionali Gestire progetti secondo le procedure e gli standard previsti dai sistemi aziendali Redigere relazioni tecniche e documentare le attività individuali e di gruppo relative a situazioni professionali	Saper utilizzare il linguaggio SQL (comandi base)	modelli, linguaggi e sistemi per le basi di dati Conoscere le principali fasi della progettazione di un database: modello concettuale, logico e fisico Conoscere le caratteristiche del modello relazionale, con i suoi principali operatori e le operazioni dell'algebra relazionale La normalizzazione Conoscere i comandi di base del linguaggio SQL Uso di Access	applicando anche le regole della normalizzazione. Effettuare le operazioni richieste utilizzando SQL Test Moodle Verifica scritta Verifica di laboratorio <u>Mese di marzo/aprile</u>
Risolvere problemi Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni Progettare Elaborare e realizzare progetti utilizzando le conoscenze apprese, verificando i risultati raggiunti	Saper applicare le tecniche per sviluppare pagine Web	Saper realizzare pagine Web statiche	MODULO 5: ELEMENTI PER LO SVILUPPO DI PAGINE WEB STATICHE. Elementi di HTML, CSS e JavaScript	Realizzare semplici pagine statiche Web Verifica di laboratorio <u>Mese di aprile/maggio</u>

Jesi 10/10/2019

Il coord. di dipartimento Prof. Luca Fabbracci

Docenti: Bartolacci M., Mantini S., Alfieri V., Braccacini A., Pigni M.