

A.S. 2020/2021
PROGRAMMAZIONE PER COMPETENZE
2°biennio
4A - 4B -4C

Dipartimento: INFORMATICA E TELECOMUNICAZIONI

Materia: **INFORMATICA**

Docenti: Bartolacci M., Mantini S., Piersigilli F.

Itp: Alfieri V., Braccacini A.

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	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 Correzione esercizi assegnati durante la pausa estiva Ripresa ed approfondimento di elementi di programmazione C# introdotti nel precedente anno di corso.	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 di laboratorio Verifica scritta <u>Mesi di settembre/ottobre</u>



Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni				
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 1: APPROFONDIMENTI E COMPLETAMENTI Completamento della gestione degli array multidimensionali. Conoscere i controlli avanzati: DataGridView, TabControl, finestre di dialogo. Creare controlli a Run-Time. La ricorsione: concetti generali; applicazione a problemi matematici. Le eccezioni: conoscerle e saperle gestire. Controlli sull'input dei dati: Try- Catch	Soluzione di problemi reali di media complessità (realizzare una applicazione semplice che preveda l'utilizzo degli elementi principali di una interfaccia grafica) Verifica di laboratorio Verifica scritta (ricorsione) <u>Mese di ottobre/novembre</u>



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 Eventuali recuperi ed approfondimenti	Progettare usando i diagrammi UML e realizzare in C#, applicazioni object oriented di differente difficoltà Verifiche scritte Test su Moodle Verifiche di laboratorio Verifiche orali <u>Mesi di novembre / dicembre/gennaio</u>
Risolvere problemi Affrontare situazioni problematiche facendo ipotesi, raccogliendo dati e proponendo soluzioni Progettare	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 Sapere come viene realizzato il puntatore nei	MODULO 3: LE LISTE-IMPLEMENTAZIONE DINAMICA Teoria: limiti della allocazione statica della memoria.	Realizzare applicazioni in C# in presenza Verifiche scritte Verifiche di laboratorio



Elaborare e realizzare progetti utilizzando le conoscenze apprese, verificando i risultati raggiunti	Saper individuare la struttura dati più opportuna per risolvere un problema	diversi linguaggi di programmazione Saper gestire Lista, Pila, Coda sia con implementazione dinamica che statica	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. Serializzazione. Uso dei files per il salvataggio di tabelle o liste. Collezioni predefinite in C#. Definizione delle strutture ad albero e la loro implementazione.	<u>Mesi di febbraio/marzo</u>
--	---	---	---	--------------------------------------



Imparare ad imparare Organizzare l'apprendimento scegliendo ed utilizzando varie fonti e modalità 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	Utilizzare le tecniche e le strategie del pensiero razionale per affrontare problemi ed elaborare opportune soluzioni 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 progettare una semplice base di dati Saper operare con i principali operatori relazionali Saper normalizzare una relazione Saper utilizzare il linguaggio SQL (comandi base)	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 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: 1FN, 2FN, 3FN. Conoscere i comandi di base del linguaggio SQL. Uso di Access	Dall'analisi di una semplice situazione reale realizzare uno schema E/R, rappresentarlo secondo le regole del modello relazionale ed effettuare operazioni relazionali applicando anche le regole della normalizzazione. Effettuare le operazioni richieste utilizzando SQL. <u>Mese di aprile/maggio</u>
--	--	---	---	--

Jesi 01/09/2020

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

Il coordinatore di dipartimento Prof. Luca Fabbracci



Via Raffaello Sanzio 8
60035 Jesi (AN) Italy
Tel. +39 0731 204550
Fax +39 0731 205706

Cod. MIUR **ANIS023002**
Cod. Fiscale 92044880422
Cod. Univoco Uff. UF0SE0

PEC anis023002@pec.istruzione.it
PEO anis023002@istruzione.it
www.iismarconipieralisi.edu.it