

PROGRAMMA CONSUNTIVO DOCENTE

Anno scolastico 2020/2021

Classe	4A SIA
Disciplina	Informatica
Docente	Francesco Gri
Docente	Loris Tissino
Data	10/06/2021

MODULI DIDATTICI - CONTENUTI/ABILITÀ/COMPETENZE

Contenuti

- Lista: rappresentazione in memoria (cenni), accesso ad una cella, inserimento, cancellazione, ordinamento, ricerca, la Pila, la Coda
- Archivi di dati: definizione di file, file sequenziali, file ad accesso diretto (cenni)
- Operazioni sui file: lettura e scrittura, operazioni fisiche e logiche, inserimento, modifica e cancellazione su file di testo
- Ciclo di vita del software: le fasi del ciclo di vita, tempi delle varie fasi, modelli di approccio al ciclo di vita del software (cenni)
- Il software di qualità: regole per realizzarlo, documentazione del software, testing, Information hiding, separazione tra interfacce e dati, riutilizzo
- La programmazione ad oggetti: definizioni, programmazione event-driven, vantaggio della programmazione ad oggetti, incapsulamento, ereditarietà (cenni), polimorfismo (cenni), gestione delle eccezioni
- Oggetti e classi: Costruttore, clausole public-private, proprietà, metodi, get/set
- Visual Studio C# come esempio di Linguaggio ad oggetti: il progetto, le unità, le form, la compilazione
- Visual Studio C#: gli oggetti predefiniti, le funzioni più comuni
- Visual Studio C#: sviluppare progetti complessi, creare un oggetto
- Introduzione alla programmazione su web (cenni)

Conoscenze

- Implementare applicazioni in Visual Studio C#
- Algoritmi fondamentali su Pila e Coda
- File: definizione di file, organizzazione fisica e logica (Cenni), istruzioni per la creazione, ricerca, inserimento, modifica, cancellazione in un file
- Oggetti: proprietà, metodi ed eventi, classi, istanze, ereditarietà, incapsulamento
- Programmazione visuale: forms, controlli, eventi

Abilità

- Utilizzare lessico e terminologia di settore
- Saper implementare algoritmi in linguaggio C#
- Individuare una struttura dati in base al problema da risolvere
- Realizzare procedure per l'utilizzo di una struttura dati
- Modellare un problema in modo orientato agli oggetti
- Gestire di un file di testo
- Sviluppare documentazione di un prodotto software secondo regole prestabilite
- Cogliere l'importanza della documentazione contestuale allo sviluppo di un progetto
- Validare i programmi
- Assemblare parti per ottenere la soluzione di problemi complessi
- Definire un processo per eventi
- Conoscere i paradigmi delle interfacce grafiche

Competenze

- Verificare la correttezza di una soluzione
- Analizzare situazioni complesse scomponendo il problema in parti
- Riutilizzare conoscenze note in ambiti differenti
- Cogliere analogie e differenze con conoscenze note
- Combinare in modo autonomo conoscenze note per risolvere nuovi problemi
- Inserire le proprie conoscenze in un ambito in evoluzione
- Produrre una documentazione adeguata
- Identificare e applicare le metodologie e le tecniche della gestione per progetti
- Saper identificare i tipi di informazione adeguati al dominio del problema
- Definire problemi attraverso la specificazione dei risultati attesi e dei dati disponibili
- Utilizzare le tecniche di sviluppo software apprese

Laboratorio (Programmazione in C#)

- le classi Veicolo, Contatore, Triangolo, Voto, Società, Prestito, Point
- Semplici applicazioni visuali: evidenziare con un segnale che un textBox è vuoto/pieno usando gli eventi, muovere un oggetto in una finestra con un timer, comandare il movimento di un oggetto in una finestra mediante tastiera, inserimento di punti in un piano cartesiano, selezione di un punto con visualizzazione delle coordinate, Gestione di un prestito
- Uso di textBox, Label, Button, Panel, Hint, Eventi, Tag
- Pila/Coda con visualizzazione grafica: uso di List, Timer, PictureBox
- Ereditarietà: la classe Cliente e le sottoclassi (Soggetto Privato, Ente Pubblico, Associazione, Impresa individuale, Impresa collettiva)
- lettura e scrittura di dati da/su file di testo
- Gestione dei Prodotti: comunicazione tra form, creazione dinamica di una form, DataGridView, classe Prodotto e classe Prodotti, separazione tra dati e interfaccia, sviluppo di un programma per parti, testing

- Il programma Tris come applicazione risassuntiva di quanto appreso: concetti della programmazione ad oggetti, separazione tra interfacce e dati, programmazione per parti, regole per lo sviluppo di interfacce grafiche