

Ministero dell'Istruzione e del Merito

Giacomo Leopardi – Battipaglia (SA)
Istituto Paritario di Istruzione Secondaria Superiore

LICEO SCIENTIFICO nuovo ordinamento – IST. TECN. SETT. TECNOLOGICO INFORMATICA E TELECOMUNICAZIONI art. INFORMATICA
Via Belluno, 19 –Codice Meccanografico SAPS09500G – STAFVU500C

Ente gestore: DEA12 SRL Via Belluno, 19 - 84091 Battipaglia (SA) tel/fax 0828371134 mail segreteria@dea12.it direzione@dea12.it
P.IVA 06196270653 www.dea12.it

UNITÀ DI APPRENDIMENTO DISCIPLINARI

ANNO SCOLASTICO 2024/2025

INDIRIZZO: ITI

DOCENTE: Salvatore Lanza / Fernando Cerrone

CLASSE: III A

N. ALUNNI: 10

DISCIPLINA: INFORMATICA e LABORATORIO

TITOLO UDA1	Introduzione All'informatica
LEZIONI UDA	<ol style="list-style-type: none">1. Informatica e Sistemi.2. L'hardware.3. Tipi di computer.4. Il software.5. Comunicare con il calcolatore.6. I linguaggi di programmazione.7. La traduzione di un programma.8. I linguaggi simbolici ad alto livello.
ATTIVITA'	<ol style="list-style-type: none">1. Lezione frontale.2. Attività di laboratorio.3. Mappe concettuali.4. Lim.5. Video e audio.6. Powerpoint.

<p>CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Relazioni fondamentali tra macchine, problemi, informazioni e linguaggi. 2. Classificazioni dei computer. 3. Caratteristiche dei linguaggi. 4. Principali componenti dell'elaboratore. 5. Padronanza della terminologia informatica. 6. Saper scegliere il linguaggio appropriato. 7. Saper riconoscere i principali hardware e i loro collegamenti.
<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Cogliere i motivi del crescente sviluppo dei calcolatori. 2. Capire la logica dei sistemi di elaborazione. 3. Capire la modalità di interazione tra i principali dispositivi hardware di un elaboratore.
<p>VERIFICA E VALUTAZIONE</p>	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA. 2. Valutazione delle attività di laboratorio.

<p>TITOLO UDA 2</p>	<p>Primi elementi di programmazione</p>
<p>LEZIONI UDA</p>	<ol style="list-style-type: none"> 1. Le definizioni di algoritmo. 2. Componenti di un algoritmo: i dati. 3. Componenti di un algoritmo: le istruzioni. 4. Rappresentazione e verifica degli algoritmi. 5. Dichiarazione dei dati. 6. Le istruzioni I/O. 7. La sequenza e l'assegnazione. 8. Incremento di variabili. 9. Uso di costanti e di variabili numeriche.

ATTIVITA'	<ol style="list-style-type: none"> 1. Lezione frontale. 2. Attività di laboratorio. 3. Mappe concettuali. 4. Lim. 5. Video e audio. 6. Powerpoint.
CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE	<ol style="list-style-type: none"> 1. Capire i problemi esistenti nella comunicazione tra l'uomo e l'elaboratore. 2. Conoscere il concetto di variabile come strumento per immagazzinare valori. 3. Conoscere la simbologia utilizzata nei diagrammi di flusso. 4. Saper realizzare algoritmi tramite i diagrammi di flusso. 5. Saper scrivere programmi che traducono l'algoritmo in linguaggio di programmazione
COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE	<ol style="list-style-type: none"> 1. Usare le istruzioni di lettura , scrittura e assegnazione per risolvere i problemi. 2. Individuare le fasi necessarie per passare da un problema alla sua soluzione. 3. Saper riconoscere i dati di input e di output di un problema.
VERIFICA E VALUTAZIONE	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA. 2. Valutazione delle attività di laboratorio.
TITOLO UDA 3	LA SELEZIONE
LEZIONI UDA	<ol style="list-style-type: none"> 1. La struttura di selezione. 2. Selezioni semplici. 3. Selezioni a una via. 4. Selezioni a cascata. 5. Selezioni annidate. 6. Selezione multipla. 7. Concetti di logica. 8. Utilizzo dei connettivi.

<p>ATTIVITA'</p>	<ol style="list-style-type: none"> 1. Lezione frontale. 2. Attività di laboratorio. 3. Mappe concettuali. 4. Lim. 5. Video e audio. 6. Powerpoint.
<p>CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Le strutture condizionali semplici. 2. Le basi della logica simbolica e del calcolo proposizionale. 3. Le strutture condizionali complesse. 4. Individuare le strutture di controllo più idonee a risolvere un determinato problema. 5. Codificare programmi che includono le strutture condizionali. 6. Testare algoritmi creando insiemi significativi di casi prova.
<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Formulare strutture condizionali sintatticamente corrette. 2. Applicare agli algoritmi i principi della logica proposizionale.
<p>VERIFICA E VALUTAZIONE</p>	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA. 2. Valutazione delle attività di laboratorio.

<p>TITOLO UDA 4</p>	<p>I Cicli</p>
---------------------	-----------------------

<p>LEZIONI UDA</p>	<ol style="list-style-type: none"> 1. Struttura iterativa 2. Le strutture iterative in C e C# 3. Uso delle diverse strutture iterative. 4. Ciclo con uscita per input particolare. 5. Ciclo per il calcolo della media 6. Ciclo con uscita per condizione avverata 7. Usare file di testo per I/O 8. Ciclo con condizioni e sommatorie 9. Ciclo per la ricerca del massimo 10. Sequenza ordinata 11. Cicli annidati
<p>ATTIVITA'</p>	<ol style="list-style-type: none"> 1. Lezione frontale. 2. Attività di laboratorio. 3. Mappe concettuali. 4. Lim. 5. Video e audio. 6. Powerpoint.
<p>CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Le strutture iterative con controllo in testa. 2. Le strutture iterative con controllo in coda. 3. Le strutture iterative con numero prefissato di cicli. 4. Individuare il tipo di ciclo più adatto a risolvere un determinato problema. 5. Saper trasformare un ciclo precondizionale in uno postcondizionale. 6. Codificare programmi che includono le strutture iterative. 7. Testare algoritmi creando insiemi significativi di casi prova.
<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Utilizzare le strategie del pensiero razionale negli aspetti dialettici e algoritmi per affrontare situazioni problematiche, elaborando opportune soluzioni. 2. Risolvere problemi che prevedono cicli.

VERIFICA E VALUTAZIONE	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA 2. Valutazione delle attività di laboratorio
---------------------------	--

TITOLO UDA 5	La programmazione complessa
LEZIONI UDA	<ol style="list-style-type: none"> 1. La complessità dei problemi. 2. Sottoprogrammi e funzioni. 3. Variabili locali e globali. 4. I parametri. 5. Il passaggio dei parametri. 6. La ricorsione.
ATTIVITA'	<ol style="list-style-type: none"> 1. Lezione frontale. 2. Attività di laboratorio. 3. Mappe concettuali. 4. Lim. 5. Video e audio. 6. Powerpoint.
CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE	<ol style="list-style-type: none"> 1. La progettazione top-down. 2. La scomposizione in sottoproblemi. 3. Le variabili locali e globali. 4. Le procedure e le funzioni. 5. Scrivere programmi divisi in sottoprogrammi. 6. Scrivere programmi con funzioni. 7. Scrivere programmi con variabili Globali. 8. Scrivere programmi con procedure in cui si usano variabili globali. 9. Saper usare le variabili locali e globali.

<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Saper affrontare un problema scomponendolo in sottoproblemi. 2. Risolvere problemi usando le procedure e le funzioni quando è necessario.
<p>VERIFICA E VALUTAZIONE</p>	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA. 2. Valutazione delle attività di laboratorio.

<p>TITOLO UDA 6</p>	<p>Le strutture dati semplici</p>
<p>LEZIONI UDA</p>	<ol style="list-style-type: none"> 1. Le caratteristiche di un vettore. 2. I vettori in C e C# 3. Il caricamento di un vettore. 4. La visualizzazione completa. 5. Estrarre valori in base a condizioni. 6. Calcolare la media. 7. Individuare un massimo o un minimo. 8. La ricerca di un elemento. 9. Creare un nuovo vettore. 10. Modificare gli elementi di un vettore. 11. Lo shift degli elementi di un vettore. 12. Gestire vettori paralleli. 13. L'ordinamento per selezione. 14. L'ordinamento per bubble sort. 15. Ordinare vettori paralleli. 16. Lavorare su totali parziali. 17. La ricerca dicotomica 18. La fusione di due vettori.

<p>ATTIVITA'</p>	<ol style="list-style-type: none"> 1. Lezione frontale 2. Attività di laboratorio 3. Mappe concettuali 4. Lim 5. Video e audio 6. Powerpoint
<p>CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Conoscere la differenza tra variabili semplici e variabili strutturate. 2. Conoscere gli algoritmi fondamentali applicati alle strutture dati. 3. Rappresentare e gestire dati con vettori paralleli. 4. Organizzare i dati nei vettori. 5. Applicare gli algoritmi di ricerca. 6. Applicare gli algoritmi ordinamento. 7. Applicare gli algoritmi di fusione e di totalizzazione.
<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Utilizzare le strategie del pensiero razionale negli aspetti didattici e algoritmici per affrontare situazioni problematiche, elaborando soluzioni. 2. Risolvere problemi con il supporto di strutture dati semplici (Vettori)
<p>VERIFICA E VALUTAZIONE</p>	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA 2. Valutazione delle attività di laboratorio
<p>TITOLO UDA 7</p>	<p>Le strutture dati complesse</p>

<p>LEZIONI UDA</p>	<ol style="list-style-type: none"> 1. Le caratteristiche di una matrice. 2. Caricare e stampare una matrice 3. Gestire una matrice 4. Operazioni su una matrice 5. Matrici quadrata e trasposta. 6. Le caratteristiche di una tabella 7. Caricare e visualizzare i dati di una tabella 8. Ordinare una tabella 9. La memorizzazione di strutture dati statiche.
<p>ATTIVITA'</p>	<ol style="list-style-type: none"> 1. Lezione frontale 2. Attività di laboratorio 3. Mappe concettuali 4. Lim 5. Video e audio 6. Powerpoint
<p>CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. La matrice come struttura dati bidimensionale. 2. Il record come insieme di elementi non omogenei. 3. La tabella come array di record. 4. Algoritmi per la ricerca e l'ordinamento. 5. Scrivere programmi in grado di trovare massimi, minimi, e medie delle righe e delle colonnelli una matrice. 6. Scrivere programmi in grado di implementare i principali algoritmi per la ricerca e l'ordinamento di elementi in una tabella.
<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Utilizzare le strategie del pensiero razionale negli aspetti dialettici e algoritmici per affrontare situazioni problematiche, attraverso l'elaborazione di soluzioni. 2. Risolvere problemi con il supporto di strutture dati complesse.
<p>VERIFICA E VALUTAZIONE</p>	<ol style="list-style-type: none"> 1. Prove di verifica di fine UDA. 2. Valutazione delle attività di laboratorio.

<p>TITOLO UDA 8</p>	<p>Programmare per il web</p>
<p>LEZIONI UDA</p>	<ol style="list-style-type: none"> 1. Le pagine Web e HTML. 2. Formattare documenti. 3. Pagine complesse. 4. HTML5. 5. I fogli di stile. 6. Classi e settori. 7. Gli attributi.
<p>ATTIVITA'</p>	<ol style="list-style-type: none"> 1. Lezione frontale. 2. Attività di laboratorio. 3. Mappe concettuali. 4. Lim. 5. Video e audio. 6. Powerpoint.
<p>CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Internet. 2. HTML. 3. CSS. 4. Creare pagine collegate ipertestualmente. 5. Scrivere pagine HTML. 6. Usare i CSS incorporati ed esterni.
<p>COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE</p>	<ol style="list-style-type: none"> 1. Realizzare pagine HTML. 2. Descrivere problemi con un approccio multimediale.

VERIFICA E
VALUTAZIONE

- 1. Prove di verifica di fine UDA.**
- 2. Valutazione delle attività di laboratorio.**

LUOGO E DATA

FIRMA DEL DOCENTE

BATTIPAGLIA, 28/10/2024

