

**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](http://www.dea12.it)

# UNITÀ DI APPRENDIMENTO DISCIPLINARI

ANNO SCOLASTICO 2024/2025

**INDIRIZZO: ITI****DOCENTE: Alfonso Fortunato****CLASSE: 4 A****N. ALUNNI: 11**

DISCIPLINA: INFORMATICA

TITOLO UDA1	Progettazione di Sistemi e l'Object-Oriented – <b>settembre / ottobre</b>
LEZIONI UDA	<ol style="list-style-type: none"><li>1. L'ingegneria del software</li><li>2. Il ciclo di vita del software</li><li>3. La programmazione orientata agli oggetti</li><li>4. Approccio metodologico agli oggetti</li><li>5. Le classi degli oggetti</li><li>6. I costruttori</li><li>7. Gestione di classi semplici</li><li>8. Gestione Array con le classi</li></ol>
ATTIVITA' DIDATTICA	<ul style="list-style-type: none"><li>• Lezioni di tipo interattivo</li><li>• Lezioni di tipo frontale</li><li>• Lezioni pratiche in laboratorio</li></ul>
CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE	<ul style="list-style-type: none"><li>• Comprendere i principi fondamentali della programmazione orientata agli oggetti.</li><li>• Conoscere il ciclo di vita del software e le diverse metodologie di sviluppo.</li><li>• Conoscere le strutture dati fondamentali (classi, oggetti, attributi, metodi).</li><li>• Progettare e implementare software utilizzando un linguaggio di programmazione orientato agli oggetti.</li><li>• Utilizzare strumenti di sviluppo integrati (IDE).</li><li>• Scrivere codice efficiente e ben strutturato.</li></ul>

TITOLO UDA3	Basi di dati – <b>gennaio / febbraio</b>
LEZIONI UDA	<ol style="list-style-type: none"> <li>1. I file</li> <li>2. Le basi di dati e dbms</li> <li>3. Le basi di dati relazionali</li> <li>4. Le basi di dati con Access</li> <li>5. Le tabelle in Access</li> <li>6. L’inserimento e la visualizzazione dei dati</li> <li>7. Le query</li> <li>8. Visualizzazione tramite i report</li> </ol>
ATTIVITA’ DIDATTICA	<ul style="list-style-type: none"> <li>• Lezioni di tipo interattivo</li> <li>• Lezioni di tipo frontale</li> <li>• Lezioni pratiche in laboratorio</li> </ul>
CONOSCENZE E ABILITA’ CHE L’UDA SI PREFIGGE DI SVILUPPARE	<ul style="list-style-type: none"> <li>• Conoscere i database, i DBMS e le loro funzioni principali.</li> <li>• Capire il modello relazionale dei dati e le sue caratteristiche.</li> <li>• Saper utilizzare Microsoft Access come DBMS.</li> <li>• Conoscere la struttura delle tabelle in Access e le relazioni tra di esse.</li> <li>• Comprendere il funzionamento delle query per estrarre e manipolare i dati.</li> <li>• Saper creare report per visualizzare i dati in modo chiaro e conciso.</li> <li>• Creare e gestire database utilizzando Microsoft Access.</li> <li>• Definire la struttura delle tabelle e le relazioni tra di esse.</li> <li>• Inserire, modificare ed eliminare dati nelle tabelle.</li> <li>• Creare query semplici e complesse per estrarre informazioni dai dati.</li> <li>• Progettare e creare report personalizzati.</li> <li>• Importare ed esportare dati da e verso altri formati.</li> </ul>
COMPETENZE CHE L’UDA SI PREFIGGE DI SVILUPPARE	<ul style="list-style-type: none"> <li>• Organizzare e gestire grandi quantità di dati.</li> <li>• Analizzare i dati per estrarre informazioni utili.</li> <li>• Risolvere problemi legati alla gestione dei dati.</li> <li>• Documentare la struttura e il contenuto di un database.</li> </ul>
VERIFICA E VALUTAZIONE	<ul style="list-style-type: none"> <li>• Prove di verifica di fine modulo</li> <li>• Valutazione delle attività di laboratorio</li> </ul>

TITOLO UDA5	Complessità e Robustezza dei Programmi – <b>maggio</b>
LEZIONI UDA	<ol style="list-style-type: none"> <li>6. Complessità degli algoritmi</li> <li>7. La valutazione degli algoritmi</li> <li>8. Robustezza dei programmi</li> <li>9. I Sistemi CASE</li> </ol>
ATTIVITA' DIDATTICA	<ul style="list-style-type: none"> <li>• Lezioni di tipo interattivo</li> <li>• Lezioni di tipo frontale</li> <li>• Lezioni pratiche in laboratorio</li> </ul>
CONOSCENZE E ABILITA' CHE L'UDA SI PREFIGGE DI SVILUPPARE	<ul style="list-style-type: none"> <li>• Comprendere il concetto di complessità algoritmica e i diversi modelli di analisi (caso migliore, caso medio, caso peggiore).</li> <li>• Saper valutare le prestazioni di diversi algoritmi in termini di tempo e spazio.</li> <li>• Comprendere i concetti di robustezza, affidabilità e manutenibilità del software.</li> <li>• Conoscere i sistemi CASE e le loro funzionalità.</li> <li>• Analizzare la complessità di algoritmi semplici e più complessi.</li> <li>• Scegliere l'algoritmo più adatto a risolvere un problema specifico.</li> <li>• Scrivere codice efficiente e ben strutturato.</li> <li>• Identificare e correggere errori nei programmi.</li> <li>• Utilizzare strumenti di profilazione per misurare le prestazioni del software.</li> <li>• Utilizzare un sistema CASE per progettare e sviluppare applicazioni.</li> </ul>
COMPETENZE CHE L'UDA SI PREFIGGE DI SVILUPPARE	<ul style="list-style-type: none"> <li>• Pensare in modo algoritmico e risolvere problemi in modo efficiente.</li> <li>• Valutare criticamente le soluzioni proposte.</li> <li>• Lavorare in modo autonomo e collaborativo.</li> <li>• Documentare il proprio lavoro in modo chiaro e conciso.</li> </ul>
VERIFICA E VALUTAZIONE	<ul style="list-style-type: none"> <li>• Prove di verifica di fine modulo</li> <li>• Valutazione delle attività di laboratorio</li> </ul>