

<b>Insegnamento:</b> Elementi di Informatica	
<b>Modulo</b> (ove presente la suddivisione in moduli):	
<b>Anno di corso:</b> I	<b>Semestre:</b> I
<b>Codice:</b>	<b>SSD:</b> ING---INF/05
<b>CFU:</b> 6	<b>Ore:</b> 48
<b>Ore di lezione:</b> 26	<b>Ore di esercitazione:</b> 22
<p><b>Obiettivi formativi:</b>          Conoscenza delle nozioni di base relative alla struttura ed al modello funzionale di un elaboratore. Conoscenza delle fondamentali strutture di dati e degli strumenti e metodi per lo sviluppo di programmi, su piccola o media scala, per applicazioni di tipo tecnico---scientifico. Capacità di progettare e codificare algoritmi in linguaggio C++, secondo le tecniche di programmazione strutturata e modulare, per la risoluzione di problemi di calcolo numerico di limitata complessità e di gestione di insiemi di dati, anche pluridimensionali.</p>	
<p><b>Contenuti:</b>          Nozioni di carattere introduttivo sui sistemi di calcolo: Cenni storici. Il modello di von Neumann. Caratteristiche della Memoria Centrale e della Unità Centrale di Elaborazione. L'hardware e il software. Software di base e software applicativo. Evoluzione del modello di Von Neumann. Cenni sui Sistemi Operativi.          Tipi e strutture di dati. I tipi di dati fondamentali del C++: int, float, double, bool, char, void. Cenni di algebra booleana. Conversioni dei sistemi di numerazione (binario, ottale, esadecimale). Rappresentazione dei dati nei registri di memoria: virgola fissa, virgola mobile, complementi alla base. Codice ASCII per la rappresentazione dei caratteri. Cenni sulla codifica di immagini e suoni. Array e stringhe di caratteri. I Record.          Strumenti e metodi per la progettazione dei programmi: Algoritmo e programma. Le fasi di analisi, progettazione e codifica. Sequenza statica e dinamica delle istruzioni. Stato di un insieme di informazioni nel corso dell'esecuzione di un programma. Metodi di progetto dei programmi. La programmazione strutturata.          L'approccio top-down per raffinamenti successivi.          Componenti di un programma: documentazione, dichiarazioni, istruzioni eseguibili. Le istruzioni di controllo del linguaggio C++.          Costrutti seriali, selettivi e ciclici: sintassi, semantica, esempi d'uso. Nesting di strutture. Modularità dei programmi.          Sottoprogrammi: le funzioni. Modalità di scambio fra parametri formali ed effettivi; effetti collaterali. Visibilità delle variabili.          Algoritmi fondamentali di elaborazione: Metodi iterativi per il calcolo numerico.          Gestione di array: ricerca, eliminazione, inserimento, ordinamento. Cenni sulla complessità computazionale di un algoritmo.          Esercitazioni: impiego di un ambiente di sviluppo dei programmi (Dev C++) con esempi di algoritmi fondamentali e di tipo numerico.</p>	
<b>Prerequisiti / Propedeuticità:</b> Nessuna	
<p><b>Metodo didattico:</b>          L'insegnamento comprende lezioni frontali ed esercitazioni sullo sviluppo di programmi in linguaggio C++. Le esercitazioni vengono svolte in aula ed in laboratorio in ambiente di sviluppo integrato Dev-C++.</p>	
<p><b>Materiale didattico:</b></p> <ul style="list-style-type: none"> <li>• Libri di testo             <ul style="list-style-type: none"> <li>○ A. Chianese, V. Moscato, A. Picariello: Alla scoperta dei fondamenti dell'informatica – Liguori Editore, 2008.</li> <li>○ E. Burattini, A. Chianese, V. Moscato, A. Picariello, C. Sansone: Che C serve. Per iniziare a programmare. – CUES Editore, 2011</li> </ul> </li> <li>• Materiale integrativo (slide ed esercitazioni) disponibile sul sito <a href="http://www.docenti.unina.it">www.docenti.unina.it</a></li> </ul>	
<p><b>Modalità di esame:</b>          L'esame è costituito da due prove: una prova pratica al calcolatore, che accerta la capacità di progettare e codificare un programma in C++, ed una prova (orale/scritta) tendente ad accertare la conoscenza degli argomenti teorici.</p>	