



SCHEDA DI TRASPARENZA DELL'INSEGNAMENTO

Insegnamento	Sistemi di Elaborazione
Livello e corso di studio	Laurea Triennale in Ingegneria Elettronica e Informatica (L8)
Settore scientifico disciplinare (SSD)	ING-INF/05 – Sistemi di Elaborazione delle Informazioni
Anno di corso	II
Numero totale di crediti	9 CFU
Propedeuticità	–
Docente	Salvatore Monteleone https://ricerca.unicusano.it/author/salvatore-monteleone/ Nickname: salvatore.monteleone E-mail: salvatore.monteleone@unicusano.it (preferire l'invio di messaggi in piattaforma alle e-mail) Orario di ricevimento: Consultare il calendario videoconferenze sul sito d'Ateneo, disponibile al link https://www.unicusano.it/calendario-lezioni-in-presenza/calendario-area-ingegneristica .
Presentazione	Il Corso di Sistemi di Elaborazione ha lo scopo di far acquisire allo studente una buona conoscenza dei principi che governano la programmazione strutturata. Il Corso propone i concetti basilari della rappresentazione ed elaborazione binaria delle informazioni. Inoltre, obiettivo formativo del Corso è fornire allo studente una conoscenza nel dettaglio sulla costruzione di algoritmi mediante la gestione del controllo di flusso, e sulle strutture dati e loro gestione. Le Etivity associate al Corso sviluppano le competenze necessarie ad implementare algoritmi e strutture dati in programmi completi attraverso l'uso del linguaggio C.
Obiettivi formativi	Il Corso di Sistemi di Elaborazione ha i seguenti obiettivi formativi: <ol style="list-style-type: none">1. Illustrare i concetti di base della rappresentazione e l'elaborazione delle informazioni dal livello del bit.2. Illustrare i concetti di base, e relativi modelli, necessari alla comprensione della struttura dei calcolatori elettronici digitali programmabili3. Illustrare i concetti di base nell'ambito della programmazione dei calcolatori elettronici4. Illustrare le tecniche di sviluppo di programmi secondo il paradigma della programmazione strutturata attraverso l'uso del linguaggio C5. Orientare l'approccio, allo sviluppo, alla costruzione di algoritmi e alla strutturazione e gestione dei dati.
Prerequisiti	Conoscenze matematiche di base.



<p>Risultati di apprendimento attesi</p>	<p>Conoscenza e capacità di comprensione (knowledge and understanding) Lo studente al termine del Corso avrà conoscenza sia teorica che di tipo progettuale riguardante la programmazione strutturata in linguaggio C e le architetture di base dei moderni calcolatori elettronici. Inoltre, lo studente acquisirà la conoscenza di elementi di rappresentazione delle informazioni nei calcolatori. Inoltre, tramite le Etivity gli studenti acquisiranno la capacità di impiegare i principali strumenti di programmazione.</p> <p>Applicazione delle conoscenze (applying knowledge and understanding) Lo studente sarà in grado di valutare le possibili soluzioni software a problemi di complessità moderata, ed affrontarne l'implementazione; sarà inoltre in grado di applicare i principi appresi per affrontare semplici problemi di rappresentazione binaria delle informazioni. Le Etivity prevedono l'applicazione delle conoscenze teoriche all'implementazione di semplici programmi in linguaggio C, utilizzando gli appositi strumenti e ambienti di sviluppo.</p> <p>Autonomia di giudizio/Capacità di trarre conclusioni (ability to draw conclusions) Lo studente sarà in grado di affrontare in autonomia l'analisi, la progettazione e l'implementazione di software basato sulla programmazione strutturata; sarà inoltre in grado di valutare la qualità delle sue soluzioni, in termini di semplicità, leggibilità, modularità, efficienza e idoneità al riuso.</p> <p>Abilità comunicative (communication skills) Lo studente sarà in grado di esporre, efficacemente e con proprietà di linguaggio, analisi e soluzioni di problemi affrontabili con la programmazione strutturata, nonché di problemi di rappresentazione delle informazioni, adoperando una terminologia adeguata.</p> <p>Capacità di apprendere (learning skills) Lo studente al termine del Corso avrà conoscenza delle nozioni fondamentali necessarie per affrontare in maniera autonoma problemi di programmazione strutturata individuando e integrando soluzioni parziali già disponibili, sia formalizzate sia implementate. Questo gli consentirà di approfondire in autonomia la conoscenza di moduli software e di interfacce di programmazione, fornendo basi utili per poter apprendere quanto verrà proposto nei corsi specialistici di ingegneria informatica, con particolare riferimento agli argomenti legati alla conoscenza dei linguaggi e dei paradigmi di programmazione, oltre che delle architetture dei calcolatori.</p>
<p>Organizzazione dell'insegnamento</p>	<p>Il corso è sviluppato attraverso le lezioni preregistrate audio-video che compongono, insieme a slide e dispense, i materiali di studio disponibili in piattaforma. Sono poi proposti dei test di autovalutazione, di tipo asincrono, che corredano le lezioni preregistrate e consentono agli studenti di accertare sia la comprensione, sia il grado di conoscenza acquisita dei contenuti di ognuna delle lezioni. La didattica interattiva è svolta nel forum della "classe virtuale" e comprende delle Etivity, elencate nel seguito, in cui lo studente applica le conoscenze acquisite nelle lezioni di teoria alla soluzione di problemi. In tale corso le Etivity richiedono l'approfondimento di un argomento complesso corredato da esempi pratici.</p> <p>Il carico totale di studio per questo modulo di insegnamento è di circa 225 ore così suddivise:</p> <ul style="list-style-type: none">• Didattica Erogetiva: circa 171 ore per la visualizzazione e lo studio del materiale videoregistrato• Didattica Interattiva: circa 54 ore per l'elaborazione e la consegna delle Etivity e per l'esecuzione dei test di autovalutazione.



	<p>Si consiglia di distribuire lo studio della materia uniformemente in un periodo di 8 settimane dedicando circa 30 ore di studio a settimana.</p>
Contenuti del corso	<p>Modulo 1 – Sistemi di numerazione (impegno di 19 ore) dove sono affrontati i seguenti argomenti: Numerazione binaria; Bit, byte e multipli; Conversioni binario-decimale; Operazioni aritmetiche nel sistema binario; Sistema di numerazione ottale; Sistema di numerazione esadecimale; Codici; Codice BCD; Rappresentazione di interi con segno. Esercitazione su sistemi di numerazione.</p> <p>Modulo 2 – Rappresentazione delle informazioni ed architettura del calcolatore (impegno di 19 ore) dove sono affrontati i seguenti argomenti: Rappresentazione delle informazioni; Rappresentazione in complemento alla base; Rappresentazione di numeri reali: virgola fissa, virgola mobile; Rappresentazione di caratteri alfanumerici; Codice ASCII; Rappresentazione di immagini; Struttura di un calcolatore elettronico digitale programmabile; Modello Von Neumann: CPU, memoria, input/output, bus; Microprocessori. Esercitazione su rappresentazione delle informazioni.</p> <p>Modulo 3 – Introduzione alla programmazione C (impegno di 21 ore) dove sono affrontati i seguenti argomenti: Introduzione al modulo; Introduzione alla programmazione; Introduzione alla programmazione nel linguaggio C - Aritmetica e decisione; Sviluppo di un programma strutturato in linguaggio C.</p> <p>Etivity 1 (3 ore di carico di studio) – Introduzione all'uso della linea di comando e agli ambienti di sviluppo.</p> <p>Modulo 4 – Controllo di flusso e funzioni (impegno di 28,5 ore) dove sono affrontati i seguenti argomenti: Controllo nei programmi in C; Funzioni in C e ricorsione. Esercitazione sullo sviluppo di programmi strutturati.</p> <p>Etivity 2 (9 ore di carico di studio) – Sviluppo di programmi strutturati in C.</p> <p>Modulo 5 – Array (impegno di 21,5 ore) dove sono affrontati i seguenti argomenti: Array in C, ricerca e ordinamento. Esercitazione sulla manipolazione di vettori e matrici.</p> <p>Etivity 3 (9 ore di carico di studio) – Sviluppo di programmi in C per la manipolazione di vettori e matrici.</p> <p>Modulo 6 – Puntatori (impegno di 24 ore) dove sono affrontati i seguenti argomenti: Puntatori in C. Esercitazione sulla manipolazione di puntatori.</p> <p>Etivity 4 (9 ore di carico di studio) – Sviluppo di programmi in C che fanno uso di puntatori.</p> <p>Modulo 7 – Tipi dati strutturati e file (impegno di 14 ore) dove sono affrontati i seguenti argomenti: Strutture, unioni, manipolazione di bit ed enumerazioni in C; Elaborazione di file in C.</p> <p>Modulo 8 – Strutture dati dinamiche (impegno di 24 ore) dove sono affrontati i seguenti argomenti: Strutture di dati dinamiche in C. Esercitazione sulla gestione di strutture di dati dinamiche.</p>



	<p>Etivity 5 (12 ore di carico di studio) – Sviluppo di programmi in C per la gestione di strutture di dati dinamiche.</p> <p>Etivity 6 (12 ore di carico di studio) – Simulazione esame.</p>
Materiali di studio	<p>Materiale didattico a cura del docente</p> <p>Il materiale didattico presente in piattaforma è suddiviso in 8 moduli. Essi ricoprono interamente il programma e ciascuno di essi contiene dispense, slide e videolezioni in cui il docente commenta le slide. Tale materiale contiene tutti gli elementi necessari per affrontare lo studio della materia.</p> <p>Testi consigliati</p> <ul style="list-style-type: none">• P. J. Deitel, H. M. Deitel, “Il linguaggio C. Fondamenti e tecniche di programmazione”, Pearson, IX edizione.• B. W. Kernighan; D. M. Ritchie, “Il linguaggio C”, Pearson Prentice Hall, II edizione.
Modalità di verifica dell'apprendimento	<p>L'esame consiste nello svolgimento di una prova scritta tendente ad accertare le capacità di analisi e rielaborazione dei concetti acquisiti. In accordo con il modello formativo del Corso di Studi, la valutazione finale dell'insegnamento, espressa in trentesimi, prende in considerazione anche l'attività svolta in itinere dallo studente e la valutazione delle Etivity proposte (descritta in dettaglio nella scheda di ciascuna Etivity).</p> <p>La prova scritta è divisa in due parti, di seguito indicate come Parte 1 e Parte 2, ciascuna delle quali prevede l'assegnazione di un massimo di 15 punti, per un totale di 30 punti. Nel dettaglio:</p> <p>Parte 1 (moduli 1 e 2): Consiste di 3 esercizi, da svolgere carta e penna, relativi ai sistemi di numerazione e alla rappresentazione delle informazioni nelle architetture dei calcolatori. Ciascun esercizio prevede l'assegnazione di un massimo di 5 punti.</p> <p>Parte 2 (moduli da 3 a 8): Riguarda la programmazione in linguaggio C e richiede la stesura, al calcolatore, di un programma che implementi 4 passi secondo delle specifiche date. Ciascun passo prevede l'assegnazione di un massimo di 3 punti, l'aderenza alle specifiche date prevede l'assegnazione di un massimo di 3 punti.</p> <p>I risultati di apprendimento attesi circa le conoscenze della materia e la capacità di applicarle sono valutate dalla prova scritta, mentre le abilità comunicative, l'autonomia di giudizio (intesa anche come capacità di trarre conclusioni) e la capacità di autoapprendimento sono valutate in itinere attraverso le Etivity.</p> <p>È possibile effettuare l'esame attraverso due prove parziali scritte consistenti nella Parte 1 e Parte 2 sopra descritte.</p>
Criteri per l'assegnazione dell'elaborato finale	<p>L'assegnazione dell'elaborato finale avverrà sulla base di un colloquio con il docente in cui lo studente manifesterà i propri specifici interessi in relazione a qualche argomento che intende approfondire.</p> <p>Non esistono preclusioni alla richiesta di assegnazione della tesi e non è prevista una media particolare per poterla richiedere.</p>