



FONDAMENTI DI INFORMATICA A - Co

ING-INF/05 - 9 CFU - Insegnamento annuale

Docente titolare dell'insegnamento

VINCENZA CARCHIOLO

Email: vincenza.carchiolo@unict.it

Edificio / Indirizzo: Edificio 3 - piano V - stanza 11

Telefono: 095 7382359

Orario ricevimento: martedì e venerdì dalle ore 11 alle ore 12

OBIETTIVI FORMATIVI

Conoscenza e capacità di comprensione (Knowledge and understanding abilities)

- conoscenza elementare dell'architettura del calcolatore
- conoscenza dei principi dell'informatica e della programmazione di tipo procedurale.
- conoscenza delle principali strutture dati e comprensione dei meccanismi di funzionamento
- conoscenza delle tecniche di ricerca e ordinamento
- conoscenza dei rudimenti di complessità computazionale

Capacità di applicare conoscenza e comprensione (Applying knowledge and understanding abilities)

- capacità di scrivere programmi in ANSI C per risolvere problemi utilizzando le strutture dati più importanti (pile, liste, code)
- capacità di analizzare il codice e correggere gli errori durante le fasi di sviluppo
- capacità di risolvere i problemi e di definire gli algoritmi risolutivi: "problem solving"
- capacità di utilizzare un ambiente di sviluppo (IDE)

Autonomia di giudizio (Ability of making judgements)

- lo studente è in grado di valutare l'algoritmo più adatto a risolvere un determinato problema

Abilità comunicative (Communication skills)

- lo studente acquisisce la conoscenza del linguaggio informatico e della terminologia tecnica

MODALITÀ DI SVOLGIMENTO DELL'INSEGNAMENTO

Il corso è organizzato in unità didattiche elementari (UDE) in base ai contenuti e alle capacità che si intende sviluppare.

il corso prevede come metodo di insegnamento principale le lezioni frontali per acquisire le conoscenze teoriche di base e tutti gli elementi sintattici e lo svolgimento di esercitazioni, da svolgere anche in modo autonomo, proposte dal docente per acquisire la capacità di risolvere i problemi, applicare la conoscenza e utilizzare gli ambienti e le metodologie di sviluppo.

Il docente propone, inoltre, delle esercitazioni individuali che consistono nella soluzione di un problema che lo studente deve affrontare in autonomia che vengono successivamente corrette o discusse in classe.

Qualora l'insegnamento venisse impartito in modalità mista o a distanza potranno essere introdotte le necessarie variazioni rispetto a quanto dichiarato in precedenza, al fine di rispettare il programma previsto e riportato nel syllabus.

PREREQUISITI RICHIESTI

Notazione esponenziale o scientifica dei numeri. Arrotondamento dei valori numerici

Numeri reali. Operazioni con numeri reali. Potenza ad esponente razionale e reale. Logaritmo di un numero reale positivo

Concetto di funzione

Vettori e Matrici: Operazioni con vettori e matrici

FREQUENZA LEZIONI

Lo studente è tenuto a frequentare almeno il 70% delle lezioni del corso per poter sostenere le prove in itinere

La frequenza non è richiesta, seppure fortemente consigliata, per sostenere la prova di esame

CONTENUTI DEL CORSO

Modulo 1

- Elaborazione automatica dell'informazione e algoritmi - Algoritmi e programmi - Una notazione grafica per esprimere algoritmi - Linguaggi di programmazione - Il progetto di programmi
- Rappresentazione dell'informazione: *Sistemi numerici - Conversione fra sistemi numerici - *Sistema di numerazione binaria - *Operazioni tra numeri binari - *overflow e underflow - *Rappresentazione dei numeri interi - *Rappresentazione dei numeri con segno -

*Rappresentazione in virgola fissa e virgola mobile - Codici e Rappresentazione dei Caratteri -
*Algebra di Boole, *Funzioni logiche, *Espressioni logiche, Applicazioni dell'algebra booleana

- Cenni della struttura di un elaboratore e sistema di elaborazione: Struttura di un elaboratore: memoria centrale, unità centrale, funzionamento dell'elaboratore.
- Cenni sui sistemi di elaborazione: software di base: Traduzione ed esecuzione dei programmi - ambiente di programmazione - Linguaggi di programmazione: linguaggi imperativi - Compilatori, Linker, Cenni sulla realizzazione di applicazioni: Preprocessore - Commenti - Librerie e file di intestazione

Modulo 2

- Elementi fondamentali del linguaggio C: *Sintassi del C - *Struttura di un programma C - *Compilazione di un programma - *Tipi di dato e rappresentazioni - *Tipi di dato principali - *Identificatori - *Variabili - *Modificatori di accesso - *Costanti - *Operatori - *Strutture di controllo - *Istruzione di selezione - *Istruzioni di iterazione - *Istruzioni di salto - *Istruzioni di espressione - *Istruzione blocco
- Array, stringhe e puntatori: *Array monodimensionali - Puntatori ad array - *Array come argomento di una funzione - *Stringhe - Array di stringhe - *Array multidimensionali - *Variabili puntatore - *Operatori ed espressioni con puntatori - *Puntatori ad array - *Puntatori a funzioni
- Funzioni: *Regole di visibilità delle funzioni - *Argomenti delle funzioni - Argomenti di main - *Istruzione return - *Valori restituiti da una funzione - *Ricorsione - *Dichiarazioni e campo di azione degli identificatori - Tecniche di legame dei parametri - *effetti collaterali ed implementazione delle funzioni
- Strutture, unioni e tipi definiti dall'utente: *Strutture- Array di strutture - *Strutture come argomenti di funzioni - *Puntatori a strutture - *Array e strutture all'interno di altre strutture - Unione - Enumerazioni - *Sizeof - *Typedef
- I/O da console da file: *Lettura e scrittura di caratteri e stringhe da console - *I/O formattato da console - *Canali - *File

Modulo 3

- *Allocazione dinamica della memoria.
- Complessità Computazionale: Efficienza dei Programmi, le Notazioni O e W, Valutazione della Complessità di un Programma.
- Algoritmi di Ordinamento: *Classi di algoritmi di ordinamento - ordinamento per selezione (selection sort) - algoritmi per inserzioni (insertion sort) - algoritmi per scambio bubble sort, quick sort, merge sort.
- Tipi di dato astratto: *Liste, *Code, *Pile, Alberi binari, Alberi generali, Dizionari.

Gli argomenti segnati con (*) rappresentano le conoscenze minime.

Il corso è organizzato in 7 Unità Didattiche Elementari (UDE) il cui contenuti è descritto negli argomenti del corso.

U
N
I
T
A
,
D
I
D
D
U
A
B
I
L
I
T
À
C
H
O
R
E
F
E
R
E
M
E
N
T
A
R
I
U
D
E
1
U
D
E
2
U
D
E
3

TESTI DI RIFERIMENTO

[Pel] Pellegrino Principe. C guida alla programmazione. Apogeo

[BeGu] Bellini, Guidi. Linguaggio C. Guida alla programmazione. McGraw-Hill

Materiale distribuito dal docente attraverso la piattaforma Studium.UniCT e/o Microsoft Team

ALTRO MATERIALE DIDATTICO

Il materiale didattico è pubblicato sul portale Studium di Ateneo e/o sulla piattaforma Microsoft Team

PROGRAMMAZIONE DEL CORSO

	Argomenti	Riferimenti testi
1	UDE 1: Rappresentazione dei numeri interi e dei numeri reali. Algebra di boole, funzione ed espressioni logiche	Appunti del docente - [Pel] Capitolo 1, [Pel] Appendice C - [BeGu] Capitolo 1, 2 [BeGu] App. D, E
2	UDE 1: Cenni sui sistemi di elaborazione; Traduzione ed esecuzione dei programmi; Ambiente di Programmazione; Linguaggi di Programmazione; Operazioni tra numeri binari; Codifica dei caratteri	Appunti del docente - [Pel] Capitolo 1, [Pel] Appendice C - [BeGu] Capitolo 1, 2 [BeGu] App. D, E

3	UDE 2: Tipo di dato principale; Identificatori; Variabili; Modificatori di Accesso; Specificatori di classe di memorizzazione; Costanti; Operatori; Strutture di Controllo; Istruzioni di selezione, di iterazione, di salto, di espressione, di blocco	Pel] Capitolo 2,4,5 - [BeGu] Capitolo 3, 4, 5, 6, 7, 8, 9
4	UDE 2: Algoritmi e programmi; Uso di notazione grafica per esprimere algoritmi; Espressioni Complesse	Pel] Capitolo 2,4,5 - [BeGu] Capitolo 3, 4, 5, 6, 7, 8, 9
5	UDE 3: • Array Monodimensionali; Puntatori; Puntatori ad array; Stringhe; Strutture e strutture nidificate; Array di strutture	[Pel] Capitolo 3,7,8,11 - [BeGu] Capitolo 10, 13, 14, 16
6	UDE 3: Puntatori a strutture; Unione; Enumerazione	[Pel] Capitolo 3,7,8,11 - [BeGu] Capitolo 10, 13, 14, 16
7	UDE 4: Lettura e scrittura di caratteri e stringhe; I/O formattato; Canali; File di testo; Esercizi sugli argomenti svolti nelle UDE 1,2,3,4	[Pel] Capitolo 11 - [BeGu] Capitolo 17, 19 [Pel] Capitolo 6,9,10 - [BeGu] Capitolo 11, 1
8	UDE 4: File Binari	[Pel] Capitolo 11 - [BeGu] Capitolo 17, 19 [Pel] Capitolo 6,9,10 - [BeGu] Capitolo 11, 15
9	UDE 5: Funzioni; Istruzione return; Passaggio di parametri; Allocazione dinamica della memoria	[Pel] Capitolo 6,9,10 [BeGu] Capitolo 11, 15
10	UDE 5: Ricorsione e Record di Attivazione; Variabili locali, regole di visibilità e tempo di vita;	[Pel] Capitolo 6,9,10 [BeGu] Capitolo 11, 15
11	UDE 6: • Algoritmi di ricerca e ordinamento in memoria interna: classi di algoritmi • Esempi di Algoritmi di ordinamento	DISPENSE DEL DOCENTE
12	UDE 6: Complessità computazionale, la notazione O e omega, Cenni sulla valutazione della complessità di un programma	DISPENSE DEL DOCENTE
13	UDE 7: Strutture dati lineari: Liste, Pile, Code; Strutture dati annidate; Esercizi sugli argomenti svolti nelle UDE 5,6,7	DISPENSE DEL DOCENTE
14	UDE 7: Strutture dati non lineari: Alberi, Hash	DISPENSE DEL DOCENTE

VERIFICA DELL'APPRENDIMENTO

MODALITÀ DI VERIFICA DELL'APPRENDIMENTO

sono previste due prove in itinere o un esame finale che permettono il superamento dell'esame.

Le competenze minime richieste per il superamento dell'esame sono le seguenti:

Conoscenze:

- Conoscenze elementari di paradigmi di programmazione
- Conoscenza della sequenza completa di produzione di programmi eseguibili
- Capacità di realizzazione di semplici conversioni decimali/binarie e viceversa
- Capacità di utilizzo di funzioni ed espressione booleane

- Comprensione delle differenti tipologie di algoritmi di ordinamento
- Capacità di utilizzare le strutture dati elementari (pile, code e liste)
- Capacità di risoluzione di problemi informatici tramite la decomposizione in funzioni elementari
- Sintassi e semantica del Linguaggio ANSI-C:

Capacità

- Capacità di risoluzione di problemi informatici elementari tramite l'utilizzo delle istruzioni di selezione ed iterazione
- Capacità di risoluzione di problemi informatici che richiedono l'utilizzo di vettori monodimensionali
- Capacità di utilizzo delle funzioni e dei parametri formali e dell'istruzione return
- Capacità di gestione della memoria Heap, tramite semplici operazioni di allocazione dinamica di elementi e vettori
- Capacità di utilizzo dei file di testo per rendere persistenti le informazioni.

- Capacità di implementare semplici algoritmi di ordinamento utilizzando la memoria interna
- Capacità di implementare ed utilizzare le strutture dati elementari (pile, code e liste)

I criteri adottati nella valutazione finale sono basati sulla pertinenza delle risposte rispetto alle domande formulate, sulla qualità dei contenuti, sulla capacità di riportare esempi, e sulla proprietà di linguaggio tecnico e la capacità espressiva complessiva dello studente.

PROVE IN ITINERE

Sono previste due prove in itinere durante il corso.

Prima prova in itinere

La prima prova in itinere è costituita da un elaborato al calcolatore della durata da **30 a 60 minuti**, che potrà svolgersi a distanza e/o in laboratorio, che prevede la capacità di sviluppo di un semplice programma in ANSI C (in accordo alle competenze erogate nelle Unità Didattiche 1,2,3,4).

Il superamento della prima prova in itinere permette l'accesso alla seconda prova in itinere.

Seconda prova in itinere

La seconda prova in itinere è costituita da una prova al calcolatore della durata da **60 a 90 minuti** che potrà svolgersi a distanza e/o in laboratorio **e da un colloquio orale**. Non ci sono limiti al massimo voto ottenibile con le prove in itinere. La prova al calcolatore funge da ammissione alla prova orale.

Il colloquio orale, in presenza e/o a distanza, prevede una discussione sulle prove preliminari svolte e approfondimenti sull'intero programma.

PROVE DI FINE CORSO

La prova d'esame è composta da due prove preliminari al calcolatore e da una prova orale, che si potranno svolgere in presenza e/o a distanza. L'accesso alla prova orale è subordinato al superamento di ambedue le prove al calcolatore.

Prima prova preliminare

La prima prova preliminare (in analogia con la prima prova in itinere) è costituita da un elaborato al calcolatore, che potrà svolgersi a distanza e/o in laboratorio, della durata di **30 minuti** che prevede la capacità di sviluppo di un semplice programma in ANSI C (in accordo alle competenze erogate nelle Unità Didattiche 1,2,3,4,5).

Seconda prova preliminare

La seconda prova preliminare è costituita da un elaborato al calcolatore della durata da **60 a 90 minuti (non è prevista valutazione numerica)** che potrà svolgersi a distanza e/o in laboratorio.

Colloquio orale

Il colloquio orale prevede una discussione sulle prove preliminari svolte e approfondimenti sull'intero programma.

ESEMPI DI DOMANDE E/O ESERCIZI FREQUENTI

Gli esercizi, esempi di prove in itinere o compiti d'esame sono disponibili su studium (studium.unict.it)
