



UNIVERSITÀ DEGLI STUDI DI PALERMO

DIPARTIMENTO	Matematica e Informatica		
ANNO ACCADEMICO OFFERTA	2021/2022		
ANNO ACCADEMICO EROGAZIONE	2021/2022		
CORSO DILAUREA	MATEMATICA		
INSEGNAMENTO	PROGRAMMAZIONE CON LABORATORIO		
CODICE INSEGNAMENTO	10664		
MODULI	Si		
NUMERO DI MODULI	2		
SETTORI SCIENTIFICO-DISCIPLINARI	INF/01		
DOCENTE RESPONSABILE	TEGOLO DOMENICO	Professore Associato	Univ. di PALERMO
ALTRI DOCENTI	TEGOLO DOMENICO	Professore Associato	Univ. di PALERMO
CFU	9		
PROPEDEUTICITA'			
MUTUAZIONI			
ANNO DI CORSO	1		
PERIODO DELLE LEZIONI	1° semestre		
MODALITA' DI FREQUENZA	Facoltativa		
TIPO DI VALUTAZIONE	Voto in trentesimi		
ORARIO DI RICEVIMENTO DEGLI STUDENTI	TEGOLO DOMENICO Mercoledì 15:00 17:00 Dipartimento Matematica e Informatica Via Archirafi 3490123Palermo		

DOCENTE: Prof. DOMENICO TEGOLO

PREREQUISITI	Nessun prerequisito è richiesto
RISULTATI DI APPRENDIMENTO ATTESI	Conoscenza e capacita' di comprensione: Acquisizione dei fondamenti sui sistemi di elaborazione e dei principi della programmazione strutturata; Apprendere i concetti di base sulle strutture dati statiche e dinamiche; Comprendere semplici algoritmi fondamentali sull'ordinamento o sulla ricerca alfanumerica; Assimilare i concetti sulla soluzione di semplici problemi attraverso la programmazione ricorsiva; possesso delle competenze sui costrutti fondamentali del linguaggio di programmazione C. Capacita' di applicare conoscenza e comprensione: Comprendere i processi cognitivi utili per l'individuazione di una soluzione ad un problema numerico semplice o complesso. Capacita' di programmazione nel linguaggio di programmazione C. Capacita' di individuare e risolvere errori sintattici e semantici emersi in fase di compilazione i primi ed esecuzione i secondi. Essere in grado di suddividere problemi complessi in problemi elementari. Autonomia di giudizio: Essere in grado di valutare la bonta' di metodi e i contenuti nella redazione di un programma. Ampia comprensione dei concetti avanzati sull'uso dei costrutti e delle strutture dati. Essere in grado di valutare le implicazioni sull'uso dei costrutti e sul passaggio dei parametri alle funzioni. Il raggiungimento di un'autonomia di giudizio sugli applicativi sviluppati basati su di una soluzione algoritmica efficiente. Abilita' comunicative: Proprieta' di espressione nella presentazione delle nozioni di base dell'arte della programmazione e del linguaggio di programmazione C. Capacita' d'apprendimento: Sapere approcciarsi alla programmazione, al problem solving e a sapere contestualizzare le abilita' acquisite in problemi concreti.
VALUTAZIONE DELL'APPRENDIMENTO	Il voto finale terra' conto del voto della prova integrata finale costituita da una prova scritta e/o di laboratorio sugli argomenti dei due moduli e da un colloquio orale. Le prove scritte saranno costituite da 10 tra domande a risposte multiple o aperte, e da 2 esercizi di programmazione. Le domande saranno valutate con un punteggio da 0 a 2, mentre ogni esercizio avra' un punteggio massimo di 5, la prova scritta sara' considerata superata con un punteggio maggiore o uguale a 15/30. La prova orale, valutata in trentesimi, sara' costituita dalla discussione delle prove scritte e/o di laboratorio, e da domande al fine di accertare la conoscenza degli argomenti del corso. La valutazione finale, sara' formulata sulla base delle seguenti condizioni: a) Conoscenza molto buona, ampia ed elevata degli argomenti proposti; capacita' di applicarli con rigore e in piena autonomia; possesso di ottime capacita' comunicative (voto 29-30L). b) Buona conoscenza degli argomenti proposti e capacita' di applicarli con rigore matematico e metodologico, ma non in piena autonomia; possesso di buona proprieta' di linguaggio (voto 26-28); c) Conoscenza discreta degli argomenti proposti e sufficiente capacita' di applicarli autonomamente; discreta capacita' di portare a termine un ragionamento rigoroso e buona proprieta' di linguaggio (voto 22-25); d) Conoscenza di base degli argomenti proposti e capacita' limitata di applicarli autonomamente; sufficiente capacita' di portare a termine un ragionamento rigoroso e sufficiente proprieta' di linguaggio (voto 18-21); e) Conoscenza insufficiente: lo studente non possiede una conoscenza accettabile dei contenuti e degli argomenti trattati nell'insegnamento e non ha alcuna capacita' di applicare autonomamente le conoscenze acquisite.
ORGANIZZAZIONE DELLA DIDATTICA	Il raggiungimento degli obiettivi didattici sara' ottenuto mediante lezioni frontali ed esercitazioni in laboratorio.

MODULO PROGRAMMAZIONE STRUTTURATA

Prof. DOMENICO TEGOLO

TESTI CONSIGLIATI

Paul J. Deitel - Harvey M. Deitel, Il linguaggio C - Fondamenti e tecniche di programmazione • 8/Ed.
A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill.

TIPO DI ATTIVITA'	A
AMBITO	50194-Formazione informatica
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	94
NUMERO DI ORE RISERVATE ALLE ATTIVITA' DIDATTICHE ASSISTITE	56

OBIETTIVI FORMATIVI DEL MODULO

Il modulo si propone di fornire allo studente metodologie teoriche e di laboratorio finalizzate all'acquisizione dei concetti base per la programmazione di un sistema di calcolo. Saranno altresì analizzati sia la rappresentazione dei dati sia i costrutti elementari di un linguaggio di programmazione imperativo, inoltre saranno somministrati semplici algoritmi che fanno uso delle strutture di controllo, di sequenza, selezione e iterazione.
Dovuta alla propedeuticità e alla notevole diffusione nel mercato sarà preso in considerazione il linguaggio di programmazione C, e ciò al fine di garantire una più ampia versatilità nel mondo del lavoro.

PROGRAMMA

ORE	Lezioni
3	Introduzione al modulo, organizzazione di un Computer, evoluzione dell'hardware, evoluzione dei sistemi operativi, evoluzione dei linguaggi di programmazione.
4	Computer: Hardware e software. Il sistema binario: definizione, operazioni. Passaggio dal sistema decimale a quello binario e viceversa. L'informazione e le unità di misura: bit e byte. Introduzione agli ambienti di sviluppo in C, introduzione alla programmazione. Definizione di Algoritmo. Semplici programmi in meta-linguaggio. Cenni sulla complessità di un algoritmo.
4	Introduzione ai diversi paradigmi di programmazione: paradigma imperativo (programmazione strutturata e programmazione ad oggetti), paradigma dichiarativo (programmazione funzionale e programmazione logica). Programmazione strutturata. Teorema di Böhm - Jacopini. Il linguaggio C e la struttura di un programma. Il costrutto di sequenza, Il costrutto di selezione If...else. Il costrutto di selezione switch...case. Identificatori. Funzioni di input/output.
3	Dichiarazione di costanti e variabili, istruzione di assegnazione. Tipi di dato standard: intero, carattere e loro rappresentazione, i tipi float e double. Il codice ASCII e altri codici di caratteri. Rappresentazione dei numeri reali in memoria. Gli operatori in C. e loro ordine di priorità. Operatori di incremento e decremento di una variabile intera.
4	I costrutti di iterazione: Il costrutto di iterazione for, il costrutto di iterazione while, il costrutto while...do. Equivalenza dei costrutti di iterazione.
3	Gli Array in C. Array a una dimensione. Applicazioni. Definizione e visualizzazione degli array. Array n-dimensionali: le matrici. Definizione assegnazione e visualizzazione di una matrice.
3	Ricerca Lineare, ricerca binaria iterativa. Algoritmi elementari di ordinamento e loro definizione. Algoritmi di ordinamento iterativi: SelectionSort, InsertionSort, BubbleSort. Le stringhe e la libreria delle funzioni sulle stringhe <string.h>.
4	Le funzioni in C: dichiarazione, definizione e chiamata di funzioni. Il passaggio dei parametri e visibilità delle variabili.
4	I puntatori. Array e puntatori. Aritmetica dei puntatori. Funzioni iterative su array.

ORE	Esercitazioni
2	Esercitazione in laboratorio sugli operatori di assegnamento, incremento e decremento. Istruzioni elementari di Input/output
2	Esercitazioni in laboratorio sull'aritmetica sugli interi e sui reali, gli operatori di uguaglianza e relazionali del C.
2	Esercitazioni in laboratorio sui costrutti di selezione con e senza nidificazione.
2	Esercitazioni in laboratorio sul costrutto iterativo FOR
2	Esercitazioni in laboratorio sul costrutto iterativo WHILE
2	Esercitazione in laboratorio sul costrutto iterativo DO..WHILE
2	Esercitazioni in laboratorio sulla selezione multipla SWITCH, sul BREAK-CONTINUE.
2	Esercitazioni in laboratorio sugli operatori logici e esercitazione sull'equivalenza dei cicli iterativi.
2	Esercitazioni in laboratorio su funzioni elementari e definite dall'utente: prototipi e passaggio di parametri. Regole di visibilità delle variabili.
2	Esercitazioni in laboratorio sugli array: ricerca e ordinamento. Applicazioni sugli ordinamenti: SelectionSort, InsertionSort, BubbleSort, MergeSort, QuickSort.

4	Esercitazioni in laboratorio sui puntatori, espressioni e aritmetica. Relazione tra puntatore e vettore.
MODULO PROGRAMMAZIONE AVANZATA <i>Prof. DOMENICO TEGOLO</i>	
TESTI CONSIGLIATI	
Paul J. Deitel - Harvey M. Deitel, Il linguaggio C - Fondamenti e tecniche di programmazione • 8/Ed. A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill.	
TIPO DI ATTIVITA'	C
AMBITO	10709-Attività formative affini o integrative
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	47
NUMERO DI ORE RISERVATE ALLE ATTIVITA' DIDATTICHE ASSISTITE	28
OBIETTIVI FORMATIVI DEL MODULO	
Questo modulo approfondisce alcune tematiche avanzate riguardanti la programmazione. In particolare vengono analizzate alcune strutture dati dinamiche definite tramite i puntatori.	

PROGRAMMA

ORE	Lezioni
2	La ricorsione. Funzioni ed algoritmi ricorsivi. Iterazione e ricorsione a confronto.
2	Algoritmi di ordinamento ricorsivi (es. MergeSort, QuickSort).
4	Strutture di dati. Liste unidirezionali di puntatori: creazione di una lista, inserimento e cancellazione di una cella.
4	Liste bidirezionali di puntatori, inserimento e cancellazione di una cella in una lista bidirezionale, pile e code.
4	Introduzione all'elaborazione di File. (File Management.)
ORE	Esercitazioni
4	Esercitazioni sulla creazione e l'accesso ai File.
2	Esercitazioni in laboratorio sulla ricorsione; un esempio di algoritmo: le Torri di Hanoi.
2	Esercitazioni in laboratorio sulla creazione di una lista unidirezionale
2	Esercitazioni in laboratorio sulla creazione di una lista bidirezionale
2	Cancellazione e inserimento di una cella sia nelle liste unidirezionali che nelle bidirezionali.