

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2012/2013
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Programmazione e Laboratorio C.I.
TIPO DI ATTIVITÀ	Attività di base
AMBITO DISCIPLINARE	Formazione informatica di base
CODICE INSEGNAMENTO	05880
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	Giuseppa Castiglione Ricercatrice Università di Palermo
DOCENTE COINVOLTO (MODULO 2)	Marinella Sciortino Professore Associato Università di Palermo
CFU	15
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	255
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	120
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Primo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Dipartimento di Matematica e Informatica di Palermo Via Archirafi, Aula 4
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Lezioni di laboratorio ed esercitazioni
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Pratica di laboratorio, Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre (modulo 1), Secondo semestre (modulo 2)
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito www.cs.unipa.it
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Giovedì ore 15:00-16:00

<p>RISULTATI DI APPRENDIMENTO ATTESI</p> <p>Conoscenza e capacità di comprensione Acquisizione dei concetti fondamentali della programmazione strutturata, strutture dati elementari statiche e dinamiche, semplici algoritmi fondamentali di ordinamento o di ricerca, definizione ricorsiva di soluzioni. Padronanza dei costrutti fondamentali del linguaggio di programmazione C.</p> <p>Capacità di applicare conoscenza e comprensione Capacità di problem solving per semplici problemi numerici, di ricerca e ordinamento. Capacità di programmazione in linguaggio di programmazione C. Capacità di validare, mediante la scrittura di semplici programmi, i concetti appresi. Capacità di comprensione degli errori rilevati in fase di compilazione ed esecuzione di semplici programmi scritti in C. Capacità di decomporre problemi complessi in problemi più semplici da un punto di vista</p>

computazionale. Essere in grado di formulare strategie risolutive per semplici problemi con l'eventuale utilizzo di opportune strutture dati traendo spunto da quanto studiato durante il corso.

Autonomia di giudizio

Saper individuare le strutture dati più idonee per efficienza nella soluzione algoritmica di problemi. Saper individuare le modalità più appropriate nel passaggio dei parametri. Saper confrontare due semplici programmi in termini di efficienza di calcolo e invarianza rispetto ai cambiamenti.

Abilità comunicative

Proprietà di espressione nella presentazione delle nozioni di base dei linguaggi di programmazione e della programmazione imperativa

Capacità d'apprendimento

Sapere approcciarsi ai vari linguaggi di programmazione e sapere contestualizzare le abilità acquisite in problemi concreti nell'ambito lavorativo.

OBIETTIVI FORMATIVI DEL MODULO 1

Il modulo si propone di fornire allo studente gli strumenti teorici e pratici per la progettazione di un programma per calcolatore elettronico nei suoi aspetti fondamentali: la rappresentazione dei dati e la formulazione di semplici algoritmi che fanno uso delle fondamentali strutture di controllo, di sequenza, selezione e iterazione. La ricorsione. Il linguaggio di programmazione utilizzato è il C, per la sua diffusione e per essere di fatto paradigmatico rispetto alla maggior parte dei moderni linguaggi di programmazione. L

MODULO 1	Programmazione Strutturata in C
ORE FRONTALI	LEZIONI FRONTALI
2	Introduzione al corso di Programmazione. Cenni sull'Architettura del calcolatore. Risoluzione dei problemi tramite un calcolatore. La nozione di Algoritmo. Esempi di algoritmi. Cenni sulla complessità di un algoritmo. I linguaggi di Programmazione. Il paradigma dichiarativo e il paradigma imperativo. I principi della Programmazione strutturata.
2	Il linguaggio C. Struttura di un programma in C. Identificatori. Programmi di input/output. Programmi che utilizzano il costrutto di sequenza. Le costanti e le variabili. Dichiarazione e assegnazione. Il tipo Int. La rappresentazione degli interi e degli interi relativi in binario. Il tipo char. Rappresentazione dei caratteri. Il codice ASCII e altri codici di caratteri. I tipi float e double. Rappresentazione dei numeri reali in memoria.
6	I costrutti di sequenza, selezione e iterazione in C. I costrutti di selezione. Il costrutto di selezione If...else. Il costrutto di selezione switch...case. Equivalenza dei cicli di iterazione. Gli operatori in C. Ordine di priorità degli operatori. I costrutti di iterazione: Il costrutto di iterazione for. Operatori di incremento e decremento di una variabile intera. Il costrutto di iterazione while, il costrutto while...do. Equivalenza dei costrutti di iterazione.
4	Il tipo strutturato array. Array a una dimensione. Applicazioni. Codici per l'inserimento e la visualizzazione degli array. Array a più dimensioni. Matrici. Inserimento e visualizzazione di una matrice. Applicazioni degli array. Ricerca Lineare, Ricerca binaria. Algoritmo di ordinamento Bubblesort. Mergesort. Le stringhe. Varie applicazioni e utilizzo delle librerie
10	Le funzioni in C. La dichiarazione, la definizione e la chiamata di funzioni. La visibilità. Il passaggio dei parametri. I puntatori. Array e puntatori. Aritmetica dei puntatori. La ricorsione. Esempi di funzioni ricorsive: il fattoriale, la somma di una successione di interi, i numeri di Fibonacci. Confronto tra iterazione e ricorsione. Algoritmo di ordinamento Quicksort.
	LEZIONI DI LABORATORIO ed ESERCITAZIONI
24	Esempi ed esercizi di applicazione degli argomenti trattati nel corso.
TESTI CONSIGLIATI	A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill. Kim N. King. Programmazione in C. Apogeo

	B. W. Kernighan, D. M. Ritchie. Il linguaggio C - Principi di Programmazione e Manuale di riferimento. Pearson Education Italia.
--	--

OBIETTIVI FORMATIVI DEL MODULO 2
 Il modulo si propone di utilizzare e trattare dati memorizzati su file esterni. Si studieranno i puntatori e il loro uso nel passaggio dei parametri. Si approfondiranno inoltre semplici strutture dati dinamiche definite mediante l'ausilio dei puntatori.

MODULO 2	Strutture Dati Astratte (modulo 2)
ORE FRONTALI	LEZIONI FRONTALI
8	Puntatori e oggetti dinamici. Allocazione e de-allocazione di memoria. Le strutture. Strutture e puntatori. Tipi derivati composti tramite struttura.
6	La gestione dei file di testo e dei file binari.
8	Strutture Dati Astratte (ADS). Una semplice ADS: la lista. Implementazione tramite array. Implementazione tramite lista concatenata. Operazioni (iterative e ricorsive) di inserimento, ricerca e cancellazione di elementi in liste concatenate.
6	La struttura dati astratta PILA. Implementazione tramite array e lista concatenata. La struttura dati astratta CODA. Implementazione tramite array circolare e lista concatenata.
8	La struttura dati astratta ALBERO. Definizione generale. Albero radicato, albero ordinato, albero k-ario. Definizione ricorsiva di albero binario. Implementazione di alberi binari. Livello di un nodo. Altezza di un albero. Albero binario completo. Relazione tra numero di nodi e altezza in un albero completo. Albero binario bilanciato. Esplorazione dei nodi di un albero binario: visita in preordine, postordine e ordine simmetrico; visita per livelli. Rappresentazione parentetica di un albero binario. Creazione di un albero binario a partire dalle visite in preordine e in ordine simmetrico.
	ATTIVITA' in LABORATORIO
36	Esempi ed esercizi sugli argomenti trattati nel corso.
TESTI CONSIGLIATI	K. N. King. Programmazione in C. Apogeo. A. Bellini, A. Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill. B. W. Kernighan, D. M. Ritchie. Il linguaggio C - Principi di Programmazione e Manuale di riferimento. Pearson Education Italia.