

| | |
|---|--|
| FACOLTÀ | Scienze MM.FF.NN. |
| ANNO ACCADEMICO | 2013/2014 |
| CORSO DI LAUREA | Matematica |
| INSEGNAMENTO | Programmazione con Laboratorio |
| TIPO DI ATTIVITÀ | Base |
| AMBITO DISCIPLINARE | Formazione informatica |
| CODICE INSEGNAMENTO | 16161 |
| ARTICOLAZIONE IN MODULI | NO |
| NUMERO MODULI | 1 |
| SETTORI SCIENTIFICO DISCIPLINARI | INF/01 |
| DOCENTE RESPONSABILE | Chiara Epifanio Ricercatore confermato Università degli Studi di Palermo |
| CFU | 6 |
| NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE | 102 |
| NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE | 48 |
| PROPEDEUTICITÀ | Nessuna |
| ANNO DI CORSO | Primo |
| SEDE DI SVOLGIMENTO DELLE LEZIONI | Consultabile al sito: http://www.scienze.unipa.it/matematica/mate/ |
| ORGANIZZAZIONE DELLA DIDATTICA | Lezioni frontali Prove pratiche |
| MODALITÀ DI FREQUENZA | Facoltativa |
| METODI DI VALUTAZIONE | Prova Scritta Prova Orale |
| TIPO DI VALUTAZIONE | Voto in trentesimi |
| PERIODO DELLE LEZIONI | Primo semestre |
| CALENDARIO DELLE ATTIVITÀ DIDATTICHE | Consultabile al sito: http://www.scienze.unipa.it/matematica/mate/ |
| ORARIO DI RICEVIMENTO DEGLI STUDENTI | Consultabile al sito: http://www.math.unipa.it/~epifanio/programmazione/ |

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione.

Conoscenza della struttura di un computer. Acquisizione degli strumenti per l'analisi ed il progetto di algoritmi. Padronanza dei costrutti .

Capacità di applicare conoscenza e comprensione.

Capacità di progettazione di algoritmi efficienti, mediante l'utilizzo delle strutture dati più adatte. Capacità di traduzione degli algoritmi scelti nel linguaggio C. Capacità di comprensione degli errori rilevati in fase di compilazione ed esecuzione di semplici programmi scritti in C.

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 di parametri. Saper discernere tra algoritmi più o meno efficienti.

Abilità comunicative

Capacità di esposizione degli argomenti studiati. Capacità di utilizzare una terminologia corretta e un linguaggio tecnico appropriato alla materia.

Capacità d'apprendimento

Capacità di decomporre problemi complessi in problemi più semplici da un punto di vista computazionale. Essere in grado di formulare strategie risolutive per semplici problemi.

OBIETTIVI FORMATIVI DEL CORSO

L'insegnamento ha lo scopo di illustrare i principi e le tecniche della programmazione, con l'obiettivo di presentare i principali concetti e costrutti, e di descrivere la tecnica di programmazione procedurale. Ciò sarà affiancato da esercitazioni utili ad un miglior apprendimento.

| CORSO | PROGRAMMAZIONE CON LABORATORIO |
|--------------------------|--|
| ORE | ARGOMENTI |
| 4 | Computer: Hardware e software. I diversi sistemi di numerazione. Il sistema binario: definizione, operazioni. Passaggio dal sistema decimale a quello binario e viceversa. L'informazione e le sue unità di misura: bit e byte. |
| 2 | Algoritmi e programmi. |
| 5 | Linguaggi di programmazione: linguaggi di basso ed alto livello. Dal linguaggio ad alto livello al linguaggio macchina: interpreti e compilatori. 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. Strutture di controllo fondamentali: sequenza, selezione, iterazione |
| 4 | La ricorsione. Funzioni ed algoritmi ricorsivi. Iterazione e ricorsione a confronto. Un esempio di algoritmo ricorsivo: le Torri di Hanoi. |
| 2 | La complessità di un algoritmo. |
| 31 | 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. I costrutti di selezione. Il costrutto di selezione If...else. Il costrutto di selezione switch...case. 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. Il tipo strutturato array. Array a una dimensione. Applicazioni. Codici per l'inserimento e la visualizzazione degli array. Array a più dimensioni. Alcune applicazioni degli array. Le stringhe. Varie applicazioni e utilizzo delle librerie. Le funzioni in C. La dichiarazione, la definizione e la chiamata di funzioni. La visibilità. Il passaggio dei parametri. Esempi di funzioni ricorsive. I puntatori. Array e puntatori. Aritmetica dei puntatori. |
| | |
| TESTI CONSIGLIATI | Alfred V. Aho, Jeffrey D. Ullman, <i>Fondamenti di Informatica</i> , Zanichelli. C. Demetrescu, I. Finocchi, G.F. Italiano, <i>Algoritmi e strutture dati</i> , McGraw-Hill. A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill. |