

CORSO DI LAUREA IN INFORMATICA

Anno di corso	Corsi di insegnamento o Attività formative ai sensi del DM 270	
I	Analisi Matematica	X
I	Matematica Discreta	X
I	Programmazione e Laboratorio	X
I	Geometria	X
I	Sistemi operativi I	X
I	Architetture	X
I	Inglese	

II	Fisica	X
II	Algoritmi e Strutture dati	X
II	Informatica Teorica	X
II	Linguaggi di Programmazione	X
II	Basi di dati I	X
II	Calcolo delle Probabilità e Statistica	X

Anno di corso	Corsi di insegnamento o Attività formative ai sensi del DM 509	
III	Basi di dati	Mutuato dal II Anno
III	Teoria e Tecniche di Compilazione	X
III	Analisi di Immagini	X
III	Reti di Calcolatori	X
III	Logica II	Mutuato da Matematica

Anno di corso	Corsi specialistici	
III	Architetture II	
III	Cibernetica	
III	Linguaggi per il Web	Mutuato dal II Anno
III	Laboratorio di Basi di Dati	
III	Linguaggi di Programmazione	Mutuato dal II Anno
III	Tecniche Innovative Comunicazione	Mutuato da Ingegneria

FACOLTÀ	Scienze MM. FF. NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Sistemi Operativi I
TIPO DI ATTIVITÀ	Di Base
AMBITO DISCIPLINARE	Formazione informatica di base
CODICE INSEGNAMENTO	11083
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Giosuè Lo Bosco Ricercatore non confermato Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	98
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	52
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Primo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Aula 4 dip. Matematica, Laboratorio Via Ingrassia
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali ed Esercitazioni in laboratorio.
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Scritta (di laboratorio), Prova Orale.
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare il docente: lobosco@unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione della conoscenza dell'architettura generale di un elaboratore moderno con particolare riferimento al microprocessore (CPU). Programmazione a basso livello.

Capacità di applicare conoscenza e comprensione

Capacità di sviluppare programmi in linguaggio assembly, in particolare durante il corso verrà presa in esame l'architettura e la programmazione della CPU Intel 80x86.

Autonomia di giudizio

Essere in grado di valutare esattamente l'efficienza di un algoritmo in termini di cicli di clock impiegati dalla CPU dell'elaboratore.

Abilità comunicative

Capacità di esporre le operazioni fondamentali utilizzate da una CPU nell'esecuzione di un qualsiasi algoritmo.

Capacità d'apprendimento

Maggiore conoscenza di un elaboratore dal punto di vista hardware, maggiore conoscenza dei linguaggi di programmazione a basso livello che implica una migliore conoscenza dei linguaggi di programmazione ad alto livello.

OBIETTIVI FORMATIVI

Obiettivo del modulo è approfondire la conoscenza dell'architettura di un elaboratore moderno, con particolare riferimento alla programmazione a basso livello del microprocessore. Esso sarà il modulo hardware principalmente studiato, prima dal punto di vista generale, e poi considerando il caso particolare dell'architettura e programmazione di CPU Intel 80x86. Durante le lezioni in laboratorio, verrà utilizzato l'assemblatore Netwide Assembler che consentirà lo sviluppo di codice in linguaggio PC assembly. Verrà introdotta la rappresentazione degli interi e dei numeri decimali in singola e doppia precisione, le istruzioni per le relative operazioni aritmetiche e logiche, le comparazioni, i salti, i cicli, i sottoprogrammi normali e rientranti, le strutture dati e l'interfacciamento del codice assembly con il linguaggio ad alto livello C. Durante le lezioni in laboratorio verrà richiesto allo studente di sviluppare il codice assembly relativo ad alcuni esempi pratici.

ORE FRONTALI	LEZIONI FRONTALI
2	Il calcolatore, storia ed evoluzione alla tecnologia attuale.
2	I chip in silicio. Legge di Moore. Gerarchia moderna dei calcolatori. Hardware e software. Componenti fondamentali di un calcolatore: CPU e memorie. Dispositivi di ingresso-uscita. Linguaggi a basso e ad alto livello.
3	Sistemi di numerazione decimale, binario, esadecimale. CPU 80x86 e suoi registri. Modalità reale e protetta. Il NASM. Istruzioni di base linguaggio assembly, direttive.
3	Rappresentazione degli interi senza segno e con segno. Rappresentazione Signed magnitude, complemento a uno, complemento a due. Operazioni aritmetiche con gli interi, precisione estesa.
3	Comparazioni, istruzioni di salto, istruzioni iterative. Traduzione delle strutture di controllo e dei cicli in assembly. Operazioni sui bit: Shift logici, aritmetici e di rotazione, and, or, xor, not.
3	Lo stack, convenzioni di chiamata delle procedure e funzioni. Passaggio di parametri. Programmi multi modulo. Interfacciamento del C e dell'assembly. Sottoprogrammi assembly ricorsivi.
3	Array unidimensionali, multidimensionali e loro rappresentazione in memoria. Istruzioni e registri per il trattamento degli array. Strutture generiche. Stringhe.
3	Rappresentazione in virgola mobile a singola e doppia precisione. Operazioni. Il coprocessore numerico delle CPU 80x86 e le sue istruzioni di calcolo e comparazione.
2	Chiamata delle funzioni di libreria C. Utilizzo ed interfacciamento con le strutture.

	ESERCITAZIONI
3	Somma dei primi n interi, potenza di un intero.
3	Conteggio di bit, divisione e moltiplicazione per potenza di due, calcolo del logaritmo, funzione di crossover.
3	Calcolo dei numeri di Fibonacci
3	Calcolo della lunghezza di una stringa, conversione da stringa ad intero.
3	Calcolo di un espressione in notazione polacca inversa.
3	Ordinamento tra interi, Ordine lessicografico, Ordinamento tra stringhe.
3	Calcolo del perimetro, Ordinamento tra elementi a doppia precisione.
3	Calcolo del minimo rettangolo di ricoprimento e del diametro di un insieme di punti.
TESTI CONSIGLIATI	<i>Patterson, Hennessy</i> : Struttura e progetto dei calcolatori. L'interfaccia Hardware-Software, Zanichelli

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Analisi Matematica
TIPO DI ATTIVITÀ	Attività Affini e Integrative
AMBITO DISCIPLINARE	Attività formative affini o integrative
CODICE INSEGNAMENTO	01238
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	MAT/05
DOCENTE RESPONSABILE (MODULO 1)	Cristina Di Bari Ricercatore Università degli Studi di Palermo
DOCENTE COINVOLTO (MODULO 2)	Pasquale Vetro Professore Ordinario Università di appartenenza
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	200
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	100
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Primo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il sito: http://www.scienze.unipa.it/informatica/
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in aula
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Scritta, Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre, Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito: http://www.scienze.unipa.it/informatica/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Modulo 1: Mercoledì dalle 9:00 alle 10:30 e/o studio 16, I piano, Dipartimento di Matematica ed Applicazioni, via Archirafi 34. Modulo 2: Mercoledì dalle 15:00 alle 16:30 e/o studio 18, I piano, Dipartimento di Matematica ed Applicazioni, via Archirafi 34

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Gli studenti devono essere in grado di studiare qualitativamente le funzioni di una e due variabili reali, risolvere problemi di integrazione semplice e doppia, determinare la soluzione generale di equazioni differenziali lineari del primo e secondo ordine. Gli studenti devono avere, inoltre, conoscenze di base sulle successioni e sulle serie di funzioni.

Capacità di applicare conoscenza e comprensione

Capacità di riconoscere, ed organizzare autonomamente, utilizzando gli strumenti di calcolo a loro disposizione, lo studio delle funzioni di una o più variabili reali, problemi semplici di ottimizzazione.

Autonomia di giudizio

Essere in grado di valutare le implicazioni degli studi e dei risultati ottenuti.

Abilità comunicative

Capacità di enunciare correttamente e dimostrare i principali risultati presentati nel corso.

Capacità d'apprendimento

Capacità di seguire, utilizzando le conoscenze acquisite nel corso, corsi d'approfondimento nel settore dell'Analisi Matematica.

OBIETTIVI FORMATIVI DEL MODULO “ANALISI MATEMATICA I”

Presentare i fondamenti dell'Analisi Matematica fornendo allo studente metodologie di calcolo applicabili ad altre discipline scientifiche. Conoscere strumenti quali il calcolo differenziale e integrale per le funzioni di una variabile reale e le successioni.

MODULO 1	ANALISI MATEMATICA I
ORE FRONTALI	LEZIONI FRONTALI
6	Nozioni di base: Teoria elementare degli insiemi – Relazioni di ordine e di equivalenza – Cenni sui numeri naturali, interi e razionali – Assiomi che caratterizzano l'insieme dei numeri reali – Insieme esteso dei numeri reali e intervalli – Insiemi limitati – Estremo superiore, inferiore, massimo e minimo di un sottoinsieme dei numeri reali.
14	Funzioni reali di una variabile reale: Funzioni iniettive, surgettive e biiettive -Funzioni elementari e loro grafico - Successioni di numeri reali - Nozione di limite per le funzioni reali di una variabile reale – Nozione di limite per le successioni – Teoremi sulle funzioni dotate di limite e regole per il calcolo di limiti – Limiti notevoli – Applicazioni.
6	Funzioni continue: definizioni e teoremi – Teorema di esistenza degli zeri e dei valori intermedi – Teorema di Weierstrass – Continuità delle funzioni inverse delle funzioni trigonometriche – Applicazioni.
24	Derivata di una funzione reale di una variabile reale: Definizione di derivata e proprietà delle funzioni derivabili – Regole per il calcolo delle derivate – Derivate delle funzioni elementari – Significato geometrico e cinematico della derivata – Derivate successive – Punti di massimo e di minimo relativo – Teorema di Rolle, di Lagrange e di Cauchy – Applicazioni – Regola di de L'Hôpital e formula di Taylor – Applicazioni al calcolo di limiti – Studio di funzioni.
14	Primitive e integrali indefiniti: Definizioni e regole di calcolo – Integrali indefiniti immediati. Integrale di Riemann – Proprietà dell'integrale di Riemann e regole di calcolo – Funzione integrale - Applicazioni al calcolo di aree e di volumi.
	ESERCITAZIONI
12	Esempi ed esercizi sugli argomenti trattati nel corso.
TESTI CONSIGLIATI	C. Di Bari – P. Vetro, Matematica Teoria ed esercizi, Libreria Dante Editrice C. Di Bari – P. Vetro, Solutions 1 e 2, Libreria Dante

OBIETTIVI FORMATIVI DEL MODULO “ANALISI MATEMATICA II”

Approfondire lo studio dell'Analisi Matematica, sviluppando nozioni di base e strumenti propri del calcolo differenziale ed integrale con particolare riferimento alle funzioni di due variabili reali. Presentare tecniche per determinare la soluzione generale di equazioni differenziali lineari, la convergenza di successioni e serie di funzioni.

MODULO 2	ANALISI MATEMATICA II
ORE FRONTALI	LEZIONI FRONTALI
12	Funzioni reali di due o più variabili: Spazi vettoriali e spazi Euclidei – Elementi di topologia - Limiti di successioni – Limiti e continuità per le funzioni reali di due o più variabili reali. Uso delle coordinate polari – Teoremi sulle funzioni dotate di limite e sulle funzioni continue – Derivate parziali e differenziabilità – Massimi e minimi relativi e metodi per la ricerca dei punti di massimo e minimo relativo – Massimi e minimi relativi vincolati – Moltiplicatori di Lagrange – Integrali multipli e metodi di calcolo.
4	Serie numeriche e successioni e serie di funzioni: Definizioni Criteri di convergenza e criteri di convergenza assoluta - Successioni e serie di funzioni – Serie di potenze.
8	Equazioni differenziali: Definizioni Metodi risolutivi per le equazioni differenziali lineari del primo ordine e a variabili separabili - Problema di Cauchy – Equazioni differenziali di Bernoulli – Equazioni differenziali lineari del secondo ordine a coefficienti costanti - Soluzioni linearmente indipendenti – Problema di Cauchy - Soluzione generale – Metodo della variazione delle costanti per le equazioni non omogenee.
	ESERCITAZIONI
TESTI CONSIGLIATI	C. Di Bari – P. Vetro, Analisi Matematica, Volume secondo, Libreria Dante Editrice

FACOLTÀ	SCIENZE MM. FF. NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Geometria
TIPO DI ATTIVITÀ	Attività affinità e integrative
AMBITO DISCIPLINARE	Formazione affine
CODICE INSEGNAMENTO	03675
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	MAT/03
DOCENTE RESPONSABILE	Angela Speciale Assistente Ordinario R.E. Università 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	Anfiteatro 4 – Dip. Mat. E Appl. Facoltà Scienze. Via Archirafi 34 PA
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in aula
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale, Prova Scritta
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito;: http://www.scienze.unipa.it/informatica/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Lunedì e Martedì ore 8.30-12.30

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Gli studenti devono essere in grado di risolvere problemi di diagonalizzazione di matrici ed endomorfismi; geometria analitica nel piano e nello spazio tridimensionale

Capacità di applicare conoscenza e comprensione Capacità di individuare autonomamente il metodo più idoneo, scegliendo tra quelli a loro disposizione, per risolvere problemi attinenti all'oggetto dell'insegnamento. **Autonomia di giudizio** Essere in grado di valutare le implicazioni degli studi e dei risultati ottenuti. **Abilità comunicative** Capacità di enunciare e dimostrare correttamente i principali risultati presentati nel corso. **Capacità d'apprendimento** Capacità di utilizzare le conoscenze acquisite per ulteriori approfondimenti e per un futuro utilizzo nell'ambito di altri corsi.

OBIETTIVI FORMATIVI DEL CORSO

Presentare i fondamenti dell'algebra lineare e della geometria analitica e fornire allo studente strumenti e metodologie applicabili ad altre discipline.

MODULO	GEOMETRIA
ORE FRONTALI	LEZIONI FRONTALI
3	Nozioni di base: Strutture algebriche. Campo dei numeri complessi.
6	Spazi vettoriali: Spazio vettoriale su un campo K. Sottospazi. Sistemi di vettori linearmente dipendenti e linearmente indipendenti. Basi. Dimensione. Spazio somma. Spazio intersezione. Relazione di Grassmann.. Applicazioni lineari. Nucleo e immagine di un' applicazione lineare. Teorema della dimensione. Composizione di applicazioni lineari.
3	Matrici: Matrici: matrice rettangolare, quadrata, trasposta, simmetrica, antisimmetrica, diagonale, triangolare. Moltiplicazione tra matrici. Matrici permutabili, invertibili, ortogonali. Spazio delle matrici.
8	Applicazioni lineari: Applicazioni lineari e matrici. Matrice di un' applicazione lineare composta. Matrici del cambiamento di base. Sistemi di equazioni lineari. Sistemi di Cramer. Matrice inversa di una matrice quadrata. Rango di una matrice. Sistemi lineari omogenei. Sistemi lineari non omogenei. Teorema di Rouchè-Capelli. Endomorfismi.
8	Diagonalizzazione Autovettori. Autovalori. Autospazi. Polinomio caratteristico. Diagonalizzazione. Forma canonica di Jordan.
8	Geometria Cartesiana: Riferimento sulla retta e segmenti orientati. Coordinate cartesiane. Vettori geometrici. Vettori paralleli e complanari. Coordinate dei vettori. Spazio affine. Allineamento e complanarità tra punti. Equazioni parametriche di rette e piani. Equazione cartesiana di un piano. Fasci di piani e di rette. Stella di piani. Equazioni cartesiane di una retta. Stella di rette. Condizione di complanarità di due rette. Rette sghembe. Spazio euclideo. Nozioni angolari e modulo di un vettore. Prodotto scalare. Misura di distanze e angoli. Distanza di due punti. Sfera. Circonferenza (nel piano e nello spazio) Coniche come luoghi geometrici. Coseni direttori di una retta. Significato geometrico dei parametri di giacitura di un piano. Angolo di due rette. Distanza di un punto da un piano. Minima distanza di due rette sghembe. Retta di minima distanza di due rette sghembe. Coni. Cilindri. Superficie di rotazione.
	ESERCITAZIONI
12	Esempi ed esercizi sugli argomenti trattati nel corso.
TESTI CONSIGLIATI	A. Basile: Algebra lineare e geometria cartesiana – Margiacchi-Galeno editrice L. Stramaccia: Esercitazione di algebra lineare e geometria cartesiana- Margiacchi-Galeno editrice.

FACOLTÀ	SCIENZE MM.FF.NN
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	INFORMATICA
INSEGNAMENTO	MATEMATICA DISCRETA
TIPO DI ATTIVITÀ	Di base
AMBITO DISCIPLINARE	Formazione Matematico-Fisica
CODICE INSEGNAMENTO	10371
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	MAT/02
DOCENTE RESPONSABILE	FABIO DI FRANCO PROFESSORE ASSOCIATO UNIVERSITA' DI PALERMO
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	204
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	96
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	I
SEDE DI SVOLGIMENTO DELLE LEZIONI	Aula 4, Dipartimento di Matematica ed Applicazioni, Via Archirafi 34, Palermo
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale e Prova Scritta
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre e Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare il docente: difranco@math.unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Si riferiscono all'insegnamento e non ai singoli moduli che lo compongono. Vanno espressi utilizzando i descrittori di Dublino **Conoscenza e capacità di comprensione:** Conoscere alcuni principi di base della matematica, con particolare riferimento all'aritmetica dei numeri interi, alle strutture combinatorie e ai grafi **Capacità di applicare conoscenza e comprensione:** Applicare le conoscenze matematiche acquisite ad argomenti legati all'informatica, come la crittografia **Autonomia di giudizio:** Essere in grado di valutare quali delle conoscenze matematiche acquisite possono essere applicate alla risoluzione di problemi algebrici e combinatori **Abilità comunicative:** Capacità di esporre in modo coerente le conoscenze matematiche acquisite **Capacità d'apprendimento:** Essere in grado di apprendere conoscenze matematiche supplementari con la lettura di testi di medio livello

OBIETTIVI FORMATIVI DEL MODULO: Acquisire una preparazione matematica di base

MODULO	MATEMATICA DISCRETA
ORE FRONTALI	LEZIONI FRONTALI
15	Logica ed insiemistica
20	Calcolo combinatorio
10	Aritmetica degli interi
20	Grafi
10	Quadrati latini, disegni e piani proiettivi
9	Strutture algebriche
12	Crittografia
TESTI CONSIGLIATI	Alberto Facchini “ Algebra e Matematica Discreta ” Ed. Decibel-Zanichelli

--

FACOLTÀ	Scienze MM. FF. NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	ARCHITETTURE
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline Informatiche
CODICE INSEGNAMENTO	14034
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Gaetano Gerardi Professore Associato Università di Palermo
CFU	9
NUMERO DI ORE RISERVATE ALLO PERSONALE STUDIO P	153
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	72
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Primo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Aula 4, Dipartimento di Matematica e Applicazioni
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in aula, Esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale + Prova pratica di progetto e verifica funzionale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre + Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Martedì ore 15-17

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione dei concetti fondamentali di codifica binaria per la descrizione dei fenomeni che si evolvono in un numero finito o indefinito di modalità statiche e/o dinamiche e/o stati.

Capacità di applicare conoscenza e comprensione

Progettazione di architetture hardware per la elaborazione dei fenomeni, codificati in binario, al fine di realizzare sistemi di processamento e controllo dei fenomeni in modo adeguato alle esigenze dei vari problemi.

Autonomia di giudizio

Valutazione della adeguatezza delle diverse architetture di elaborazione in relazione alla loro complessità e al problema da risolvere.

Abilità comunicative

Sapere presentare in modo semplice e lineare sistemi complessi, come elaboratori elettronici,

anche con l'ausilio di tabelle di verità, tabelle di stato, diagrammi di flusso temporale delle informazioni e di stato.

Capacità d'apprendimento

Avere solide basi dei concetti fondamentali perché, in modo completamente autonomo, si possa capire, apprezzare in modo critico, ed usare i nuovi sistemi, sempre più complessi (che il mercato mette a disposizione) ed illustrati con i vari sistemi di comunicazione come articoli scientifici e/o divulgativi.

OBIETTIVI FORMATIVI DELL'INSEGNAMENTO

L'insegnamento si propone di fornire allo studente i concetti teorici generali alla base delle definizioni e realizzazioni delle moderne CPU e calcolatori e la conoscenza dei metodi fondamentali pratici di progetto e realizzazione hardware, sia con logica random che con logica programmata.

Modulo	ARCHITETTURE I
ORE FRONTALI	LEZIONI FRONTALI
2	Introduzione al corso e illustrazione del materiale didattico PSPICE e XILINX
4	Introduzione ai Sistemi Numerici e codifica delle informazioni con e senza ridondanze
5	Operatori binari tabelle di verità e mappe. Realizzazione hardware delle funzioni logiche elementari con tecnologia MOSFET (CMOS).
4	Algebra booleana e teoremi relativi alle funzioni con il metodo delle ivi. Minimizzazione delle mappe e tabulare
4	Uso della logica combinatoria: and, or, xor, inverter, buffer, buffer tri-state, etc. Corse ed hazard dinamici e statici; Circuiti discriminatori, trigger di Schmitt, circuiti di ritardo e di rivelazione di fronti.
4	Funzioni combinatorie fondamentali: Encoder, Decoder, Priority encoder, Multiplexer, Demultiplexer, Generatori/Rivelatori di parità, convertitori di codici, Shifter, Barrel shifter, Conversione di codici, Comparatori.
4	Circuiti aritmetici fondamentali: incremento e decremento binario; Half adder, Full adder, Parallel adder, Look-Ahead Adder. Sottrazione Binaria. Rappresentazioni numeriche in complemento a 10 e a 9, complemento a 2 e ad 1. Sottrazione con numeri scritti in complemento ad 2. Somma/Sottrazione BCD
7	Logica sequenziale: Gated Oscillator. Circuiti CR ed RC per la realizzazione di: Clocks, Impulsi, Ritardi, Duplicatori di Frequenza. Implementazione dei principali Latch e Flip Flop (RS, D, T, JK) e Flip-Flop Master/Slave e Edge Triggered e loro equazione di stato. on clocked e clocked. Triggering con il primo o secondo fronte del clock. Circuiti Monostabili retriggerabili e non retriggerabili.
3	Procedure di progetto di un Circuito Sequenziale ed uso delle Tabelle di eccitazione, Equazioni di stato, Diagrammi temporali.
5	Circuiti sequenziali fondamentali. Rivelatori di sequenze, Registri di tipo latched e non latched con e senza Enable, Registri a scorrimento. Circuito di Calcolo/Check del CRC Contatore ad anello, Contatore Johnson. Contatore Gray con uscite decodificate, Contatori Binari e BCD di tipo ripple, paralleli, Up, Down. Contatori modulo N. Contatori programmabili asincroni e sincroni.
3	Memorie Statiche RAM e ROM, Register File dual port; Static Random Access Memory. Uso di ROM, PLA, PAL, CPLD, FPGA per la realizzazione di funzioni logiche.
5	Microprogrammazione: Controllore micro programmato per sequenze singole e multiple; Mapping-Prom; Realizzazione hardware di Salti condizionati e non. Sovrapposizione temporale delle operazioni di Fetch ed Execution: Registro Pipeline. Program counter realizzato con contatori e Incrementer+storage register. Stack LIFO, Chiamate a Subroutine; Overlapping dei campi di microistruzioni e gestione di loop. Gestione delle periferiche: Status polling e Polled interrupt.

5	Linguaggio di programmazione Hardware: VHDL. Livelli di astrazione per la definizione di un progetto: Livello Strutturale, Dataflow, Behavioral; Linguaggio VHDL (ealy e di Moore). Definizione VHDL di: , per la definizione di circuiti digitali combinatori e Funzione Booleane quali: AND, OR, NOR, Multiplexer, Decoders, Convertitori di codice, Demultiplexer, Parallel Adder e Parallel Subtractor, ALU, Shifter. Definizione VHDL di Latches e Flip-Flop con e senza abilitazione, sincroni ed asincroni. Definizione VHDL di un circuito sequenziale a partire da un diagramma di stati (macchine di M Shift Register di tipo PIPO, SIPO, PISO, Universale, Contatori Binari UP, DOWN e UP/DOWN con clear e preset sincroni ed asincroni. Registri di Memoria statica. Sintesi implementazione hardware, map e Place&Route, simulazione behavioural e Post-Route
	ESERCITAZIONI
5	Esercitazioni in aula di analisi e progetto di circuiti logici
5	Simulazione PSPICE (student edition) Simulazione di sistemi combinatori e sequenziali
8	Sintetizzatore Xilinx (ISE WebPACK) Sintesi e Simulazione di sistemi combinatori e sequenziali definite in VHDL.
TESTI CONSIGLIATI	G. Gerardi “Introduzione alle reti logiche Combinatorie e Sequenziali” (2008-09) G. Gerardi “Architettura Microprogrammata di Computer” (2008-09) M. Morris Mano & Charles R. Kime "Reti Logiche Digitali" Addison WESLEY Donnamaie E. White “Bit-Slice Design: Controllers and ALUs” 10.edacafe.com) file da web) Garland STPM press, 1981(scaricabile da: http://www.HARDY Electronics AB “VHDL Handbook” (scaricabile) PSPICE: Manuale d'uso Simulatore (a corredo del simulatore) Xilinx: Manuale d'uso ISE Web PACK (a corredo del sintetizzatore)

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
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)	Cesare Valenti Ricercatore Università degli Studi di Palermo
DOCENTE COINVOLTO (MODULO 2)	Marinella Sciortino Professore Associato Università di Palermo
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	204
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	96
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Primo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Dipartimento di Matematica ed Applicazioni di Palermo
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Attività in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Pratica, 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 web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare o docenti: cvalenti@math.unipa.it ; mari@math.unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione dei concetti fondamentali relativi alla programmazione strutturata, alle strutture dati elementari statiche e dinamiche, a semplici algoritmi fondamentali di ordinamento o di ricerca, alla definizione ricorsiva di soluzioni. Padronanza dei costrutti fondamentali del linguaggio di programmazione C.

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.

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

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 con l'eventuale utilizzo di opportune strutture dati traendo spunto da quanto studiato durante il corso.

OBIETTIVI FORMATIVI DEL MODULO 1

Il modulo si propone di illustrare le modalità di 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. 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.

MODULO 1	PROGRAMMAZIONE E LABORATORIO C.I. (modulo 1)
ORE FRONTALI	LEZIONI FRONTALI
4	Introduzione al corso di Programmazione. Cenni sull'Architettura del calcolatore. Risoluzione dei problemi tramite un calcolatore. La nozione di Algoritmo. Esempi di algoritmi. Complessità di un algoritmo. I linguaggi di Programmazione. Il paradigma dichiarativo e il paradigma imperativo. I principi della Programmazione strutturata. Il teorema di Boem Jacopini. I costrutti di sequenza, selezione e iterazione. Equivalenza di cicli.
4	Il linguaggio C. Struttura di un programma in C. Identificatori. Programmi di input/output. Programmi che utilizzano il costrutto di sequenza. Le costanti, 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 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. Equivalenza dei costrutti di iterazione.
4	Il tipo strutturato array. Array a una dimensione. Applicazioni. Codici per l'inserimento e la visualizzazione degli array. Somma e Prodotto scalare degli array. Array a più dimensioni. Matrici. Inserimento e visualizzazione di una matrice. Prodotto di Matrici. Alcune applicazioni degli array. Ricerca Lineare, Ricerca binaria. Algoritmi di Ordinamento. Selectionsort. Bubblesort. 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.
	ATTIVITA' in LABORATORIO
20	Esempi ed esercizi sugli argomenti trattati nel corso.

TESTI CONSIGLIATI	<p>Libro di testo: A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill.</p> <p>Libro consigliato:B. W. Kernighan, D. M. Ritchie. Il linguaggio C - Principi di Programmazione e Manuale di riferimento. Pearson Education Italia.</p>
--------------------------	---

<p>OBIETTIVI FORMATIVI DEL MODULO 2</p> <p>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.</p>

MODULO 2	PROGRAMMAZIONE E LABORATORIO C.I. (modulo 2)
ORE FRONTALI	LEZIONI FRONTALI
4	Puntatori e oggetti dinamici. Allocazione e de allocazione di memoria. Le strutture. Strutture e puntatori. Tipi derivati composti tramite struttura.
4	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.
2	La struttura dati astratta PILA. Implementazione tramite array e lista concatenata. Uso della pila per la valutazione delle espressioni aritmetiche in forma postfissa.
2	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. Alberi binari di ricerca (ABR). Operazioni di inserimento, ricerca e cancellazione e relativo costo computazionale: caso pessimo, caso ottimo e caso medio.
ATTIVITA' in LABORATORIO	
20	Esempi ed esercizi sugli argomenti trattati nel corso.
TESTI CONSIGLIATI	<p>Libro di testo: A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill.</p> <p>Libro consigliato:B. W. Kernighan, D. M. Ritchie. Il linguaggio C - Principi di Programmazione e Manuale di riferimento. Pearson Education Italia.</p>

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Linguaggi di Programmazione
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline informatiche
CODICE INSEGNAMENTO	04758
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Lenzitti Biagio R.U. Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	102
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	48
PROPEDEUTICITÀ	Analisi Mat., Mat Discreta, Progr. e Labor. C.I.
ANNO DI CORSO	2°
SEDE DI SVOLGIMENTO DELLE LEZIONI	Dipartimento di Matematica
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare il docente: lenzitti@math.unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Conoscenza delle macchine astratte. Conoscenza panoramica storica dei vari paradigmi di programmazione. Conoscenza del paradigma di programmazione funzionale, logico e dichiarativo e strutturale. Conoscenza del paradigma orientato agli oggetti. Conoscenza del linguaggio C++. Conoscenza dei linguaggi orientati alla programmazione in rete. Conoscenza dei linguaggi XML. Conoscenza del linguaggio PHP.

Capacità di applicare conoscenza e comprensione

Capacità di valutare le funzionalità dei diversi paradigmi di programmazione. Capacità di scrivere un codice nei linguaggi di programmazione C++ e PHP. Capacità di utilizzare un linguaggio XML

Autonomia di giudizio

Capacità di valutare e comparare autonomamente le soluzioni di un problema di limitata complessità.

Abilità comunicative

Capacità di organizzarsi in gruppi di lavoro.

Capacità di comunicare efficacemente in forma orale anche utilizzando termini in inglese.

Capacità di apprendere

Capacità di catalogare, schematizzare e rielaborare le nozioni acquisite.

OBIETTIVI FORMATIVI DEL MODULO

Fornire le competenze di base, sia metodologiche che tecniche, su Linguaggi di Programmazione, in particolar modo sui linguaggi orientati agli oggetti e sui linguaggi orientati alla programmazione nel WEB.

MODULO	Linguaggi di Programmazione
ORE FRONTALI	LEZIONI FRONTALI
4	Concetti base sulle macchine astratte
2	panoramica storica dei vari paradigmi di programmazione
2	paradigma di programmazione funzionale e logico
2	paradigma di programmazione dichiarativo e strutturale
4	paradigma orientato agli oggetti
10	linguaggio C++ Esempi ed esercizi
1	linguaggi orientati alla programmazione in rete.
	ESERCITAZIONI
4	Esempi ed esercizi in linguaggio C++ ,
TESTI CONSIGLIATI	M. Gabrielli, S. Martini Linguaggi di programmazione, principi e paradigmi Mc GrawHill Harvey M. Deitel e Paul J. Deitel C++ Tecniche avanzate di programmazione 2a ed. Apogeo

1	linguaggi orientati alla programmazione nel web
2	linguaggi XML
3	Strumenti XML ((Xpath , Xlink Xpointer , Xquery)
3	Linguaggi XML (XHTML XSL SVG)
1	Web Scripting CGI
10	linguaggio php Esempi ed esercizi
2	Le Classi in PHP
2	Librerie XML in PHP5
	ESERCITAZIONI
8	Esempi ed esercizi in linguaggio, XML e php
TESTI CONSIGLIATI	Anders Moller, Michael Schwartzbach, Introduzione a XML Pearson Education Italia Converse, Park, Morgan PHP5 & MySQL LA Guida McGraw-Hill

FACOLTÀ	Scienze MM. FF. NN.
ANNO ACCADEMICO	2009-2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Calcolo delle Probabilità e Statistica
TIPO DI ATTIVITÀ	Affine o Integrativa
AMBITO DISCIPLINARE	Attività formative Affini o Integrative
CODICE INSEGNAMENTO	01737
ARTICOLAZIONE IN MODULI	NO
SETTORE SCIENTIFICO DISCIPLINARE	MAT/06
DOCENTE RESPONSABILE	Gaetana Gambino Ricercatore non confermato Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	98
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	52
PROPEDEUTICITÀ	Analisi Mat, Mat. Discreta, Progr e Lab C.I.
ANNO DI CORSO	Secondo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il sito web: http://www.cs.unipa.it/
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova scritta e prova orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Giovedì, dalle 11 alle 13

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Conoscenza degli strumenti basilari del calcolo delle probabilità, della statistica inferenziale classica e bayesiana.

Capacità di applicare conoscenza e comprensione

Capacità di utilizzare i predetti strumenti, contribuendo ad arricchire l'insieme dei propri già noti strumenti di analisi attraverso moderni metodi non deterministici

Autonomia di giudizio

Lo studente dovrà essere in grado di interpretare i principali risultati degli esercizi svolti, così come di organizzare e leggere i dati statistici.

Abilità comunicative

Acquisire la capacità di comunicare ed esprimere problematiche inerenti fenomeni non deterministici

Capacità d'apprendimento

Apprendere i principi della metodologia statistica e probabilistica. Acquisire nuove informazioni e leggere i risultati.

OBIETTIVI FORMATIVI

Obiettivo del corso e' introdurre lo studente allo studio del Calcolo delle Probabilita' senza creare antitesi con altre discipline a carattere prevalentemente deterministico. La caratteristica fondamentale del corso sara' il ricorso ad una visione "concreta" del calcolo delle probabilita' (non ristretta ad introdurre il concetto astratto di probabilita' in termini di teoria della misura e dell'integrazione). Verra' introdotta la statistica come strumento utile per capire i fenomeni non deterministici, naturali e non. Verranno forniti gli strumenti metodologici fondamentali per il calcolo delle probabilita'. Il corso, essendo dedicato a studenti che utilizzano la probabilita' e la statistica a fini strettamente applicativi, sara' ricco di esercitazioni, esempi e si avvarra' dell'utilizzo di software statistici. D'altra parte non verranno trascurati gli aspetti concettuali della probabilita' e della statistica.

MODULO	DENOMINAZIONE DEL MODULO
ORE FRONTALI	LEZIONI FRONTALI
3	Statistica descrittiva, campionamento
5	Misure statistiche di sintesi
5	Rappresentazioni grafiche (stem and leaf, box plot, istogrammi)
3	Calcolo combinatorio
3	Definizione classica, frequentista e soggettiva di probabilita'
7	Impostazione assiomatica del calcolo delle probabilita', legami stocastici tra eventi
10	Variabili aleatorie discrete (Bernoulli, Poisson), continue (uniforme, normale) ed inferenziali (Chi quadrato, Student, Fisher)
5	Test delle ipotesi (ipotesi nulla, livello di significativita' e potenza di un test)
3	Metodo Monte Carlo
4	Analisi di regressione
TESTI CONSIGLIATI	W. Navidi, Probabilita' e Statistica per l'ingegneria e le scienze , McGraw-Hill, 2006 • R. Scozzafava, Incertezza e Probabilita' , Zanichelli, 2001 • P. Erto, Probabilita' e Statistica per le scienze e l'ingegneria , McGraw-Hill, 2008

FACOLTÀ	Scienze Matematiche Fisiche Naturali
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Fisica
TIPO DI ATTIVITÀ	Attività di base
AMBITO DISCIPLINARE	Formazione Matematico-Fisica
CODICE INSEGNAMENTO	03245
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	FIS/01
DOCENTE RESPONSABILE (MODULO 2)	Giuseppina Andaloro Professore Associato Università di Palermo
DOCENTE COINVOLTO (MODULO 1)	Giovanni Peres Professore Ordinario Università di Palermo
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	204
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	96
PROPEDEUTICITÀ	Matematica Discreta, Analisi Matematica, Programmazione e Laboratorio C.I.
ANNO DI CORSO	Secondo
SEDE	Aula ex Consorzio Agrario
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali Esercitazioni in aula
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Scritta ed Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Annuale
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Da concordare con il docente: giuseppina.andaloro@libero.it; peres@astropa.unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione dei concetti e delle leggi della fisica classica. Capacità di applicare le leggi alla soluzioni di semplici problemi..

Capacità di applicare conoscenza e comprensione

Autonomia di giudizio

Abilità comunicative

Capacità d'apprendimento**OBIETTIVI FORMATIVI DEL MODULO I “MECCANICA E IDRODINAMICA”**

Obiettivo del modulo è introdurre lo studente alla conoscenza delle grandezze, dei concetti e delle leggi della meccanica classica e della idrodinamica. Le applicazioni riguarderanno i moti più facilmente osservabili e governati da forze descritte da semplici leggi matematiche.

MODULO I	MECCANICA E IDRODINAMICA
ORE FRONTALI	LEZIONI FRONTALI
1	Obiettivi della disciplina e cenni alle conoscenze matematiche utilizzate.
2	Moto in una e due dimensioni.
2	Dinamica del punto materiale.
2	Lavoro ed energia.
2	Quantità di moto.
2	Cinematica e dinamica del moto rotatorio.
1	Oscillazioni.
2	Gravitazione.
2	Idrodinamica.
ESERCITAZIONI	
13	Applicazioni numeriche sulle leggi di Newton e sulle leggi di conservazioni
TESTI CONSIGLIATI	R. A. SERWAY FISICA Edises

OBIETTIVI FORMATIVI DEL MODULO II “ELETTRROMAGNETISMO E OTTICA”

Obiettivo del modulo è quello di introdurre allo studio dell'Elettricità e del Magnetismo con cenni alla struttura della

materia. I fenomeni dell'Optica sono affrontati con le leggi dell'Optica Geometrica e dell'Optica Fisica.

MODULO 2	ELETTRROMAGNETISMO E OTTICA
ORE FRONTALI	LEZIONI FRONTALI
1	Obiettivi del corso e richiami alle conoscenze fisiche e matematiche che saranno utilizzate.
2	Campo elettrico
2	Legge di Gauss
3	Potenziale elettrico
2	Capacità e dielettrici.
4	Correnti e resistenze, circuiti in corrente continua.
8	Forze magnetiche, campi magnetici, sorgenti magnetiche
6	Legge di Faraday-Lenz

6	Onde, equazione d'onda, onde meccaniche ed elettromagnetiche.
14	Ottica geometrica ed ottica ondulatoria..
ESERCITAZIONI	
20	Applicazioni numeriche su elettricit�, magnetismo e induzione elettromagnetica
19	Applicazioni numeriche su ottica geometrica e ottica fisica
TESTI CONSIGLIATI	R. A. SERWAY FISICA EdiSES

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Basi di Dati I
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline Informatiche
CODICE INSEGNAMENTO	11084
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Sabrina Mantaci Professore Associato Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	102
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	48
PROPEDEUTICITÀ	Analisi Mat. , Mat. Discreta, Prog. e Labor. C.I.
ANNO DI CORSO	II
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il sito web: http://www.cs.unipa.it/
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale, Prova Scritta
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Martedì 15.00-17.00 Giovedì 15.00-17.00

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Il corso punta a far acquisire allo studente gli strumenti di base per il progetto e l'utilizzo delle basi di dati. Capacità di utilizzare il linguaggio SQL.

Capacità di applicare conoscenza e comprensione

Capacità di progettare semplici database, e sviluppo di programmi in PL/SQL per il loro funzionamento

Autonomia di giudizio

Nel progetto di database si tende a sviluppare la capacità di scegliere le strategie che rendono efficiente e di semplice uso il database.

Abilità comunicative

Si vuole sviluppare la capacità di documentare il database progettato al fine di comunicare al committente il funzionamento del prodotto ottenuto. Essere in grado di evidenziare le ricadute

tecnologiche delle teorie studiate.

Capacità d'apprendimento

Si testeranno le capacità di apprendimento dello studente mediante esercitazioni che saranno svolte in classe per un certo numero di ore.

OBIETTIVI FORMATIVI DEL MODULO

Si vogliono impartire agli studenti delle nozioni di base per lo sviluppo e l'utilizzo dei database. Essendo un corso di base, parte dal corso sarà dedicato ai principi teorici alla base della creazione dei database, e una parte all'acquisizione del linguaggio SQL per l'interrogazione dei database

MODULO	DENOMINAZIONE DEL MODULO
ORE FRONTALI	LEZIONI FRONTALI
10	Sistemi informativi e informatici. DB e DBMS. Funzionalità di un DBMS: Definizione, uso e controllo della base di dati. Progettazione di un DB. Entità, collezioni, tipi, proprietà, associazioni
10	Il modello relazionale. Algebra relazionale: unione, intersezione, differenza, ridenominazione, selezione, proiezione, join naturale. proprietà e cardinalità del join naturale. Prodotto cartesiano, thajoin, equi-join.interrogazioni
10	Calcolo relazionale su domini. Calcolo relazionale su tuple con dichiarazioni di range.Terza forma normale, BCNF e relativi algoritmi di normalizzazione.
8	Il linguaggio SQL. Data Definition Language e Data Manipulation Language. DDL: Definizione e modifica dello schema del database inserimento, cancellazione e aggiornamento di dati. Interrogazione del database.
10	Ordinamento, operatori aggregati, raggruppamenti, join. Select nidificate. Viste. Elementi di PI/SQL. Trigger, Procedure e Funzioni.
TESTI CONSIGLIATI	<ul style="list-style-type: none"> • Albano, Ghelli, Orsini, <i>Fondamenti di Basi di dati</i>, Zanichelli. • Atzeni, Ceri, Paraboschi, Torlone, <i>Basi di dati - Modelli e linguaggi di interrogazione</i>, McGraw-Hill.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009-2010
CORSO DI LAUREA	INFORMATICA
INSEGNAMENTO	INFORMATICA TEORICA
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline Informatiche
CODICE INSEGNAMENTO	03946
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	DUE
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	Antonio Restivo Professore Ordinario Università di Palermo
DOCENTE COINVOLTO (MODULO 2)	Settimo Termini Professore Ordinario Università di Palermo
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	96
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	104
PROPEDEUTICITÀ	Analisi Matematica, Matematica Discreta, Progr e Labor C.I.
ANNO DI CORSO	II
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il sito web: http://www.cs.unipa.it/
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in aula
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Scritta, Prova Orale, Presentazione di alcuni argomenti integrativi e complementari del programma sotto forma di seminari degli studenti
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo e secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Martedì e Giovedì dalle 15.00 alle 17.00

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione degli strumenti avanzati per leggere gli aspetti basilari della letteratura specialistica della disciplina.
Capacità di utilizzare il linguaggio tecnico proprio della disciplina.

Capacità di applicare conoscenza e comprensione

Capacità di riconoscere, ed organizzare in autonomia argomenti base dell'informatica teorica.

Autonomia di giudizio

Essere in grado di valutare la rilevanza generale di argomenti della disciplina.

Abilità comunicative

Capacità di esporre le tematiche generali dell'informatica teorica anche a un pubblico non esperto..

Capacità d'apprendimento

Capacità di aggiornamento con la consultazione delle pubblicazioni scientifiche proprie del settore. Capacità di seguire, utilizzando le conoscenze acquisite nel corso, sia master di secondo livello, sia corsi d'approfondimento sia seminari specialistici nei settori trattati.

OBIETTIVI FORMATIVI DEL MODULO I

Conoscere le capacità computazionali degli automi a stati finiti e la capacità generativa delle grammatiche non contestuali. Rapporti tra modelli deterministici e non deterministici. Capacità di convertire un formalismo in un altro equivalente: ad esempio, grammatiche e automi, automi e espressioni regolari, automi deterministici e non deterministici. Saper progettare automi che riconoscono linguaggi fissati. Saper progettare grammatiche che generano linguaggi fissati. Saper usare automi e grammatiche nella progettazione di algoritmi. Conoscere l'utilizzo degli automi e delle grammatiche come modello in alcune importanti di applicazioni: ad esempio, progetto di compilatori, software per progettare circuiti digitali, software per esaminare vaste collezioni di testi.

Riportati nel Regolamento Didattico del Corso di Studio

MODULO	TEORIA DEGLI AUTOMI E DEI LINGUAGGI FORMALI
ORE FRONTALI	LEZIONI FRONTALI
40 (5 CFU)	
6 ore	Automi a Stati Finiti Motivazioni e descrizione informale. Definizione di automa a stati finiti deterministico (DFA). Linguaggio riconosciuto da un DFA. Rappresentazione di un DFA con grafo degli stati. Automi a stati finiti non deterministici (NFA). Teorema di equivalenza tra DFA e NFA. La "subset construction". Discussione sulla "state complexity" di DFA e NFA. Applicazioni alle ricerche testuali. Automi con ϵ -transizioni. Eliminazione delle ϵ -transizioni.
6 ore	Espressioni regolari. Linguaggi regolari. Equivalenza tra linguaggi regolari e linguaggi riconosciuti da DFA (Teorema di Kleene). Algoritmo di eliminazione degli stati per convertire un automa in un'espressione. Algoritmo di Berry e Sethi per convertire un'espressione in un automa.
2 ore	Il "pumping lemma" per i linguaggi regolari. Applicazioni del pumping lemma.
6 ore	Equivalenza e minimizzazione di automi. La relazione di indistinguibilità degli stati. Automa ridotto. Equivalenza tra automa ridotto e automa minimale. Teorema di Myhill-Nerode. Unicità dell'automa minimale. Algoritmo di minimizzazione di un DFA. Algoritmo per decidere l'equivalenza di due DFA
2 ore	Automi bidirezionali (2-DFA). Equivalenza tra 2-DFA e 1-DFA (Teorema di Rabin-Shepherdson).
2 ore	Problemi di decisione per i linguaggi regolari
6 ore	Grammatiche e Linguaggi Liberi dal Contesto (CF) Motivazioni e descrizione informale. Definizione di grammatica. Derivazioni delle grammatiche. Linguaggio generato da una grammatica. La gerarchia di Chomsky. Le grammatiche e i linguaggi CF. Alberi sintattici. Ambiguità nelle grammatiche e nei linguaggi CF: grammatiche ambigue, eliminazione delle ambiguità, ambiguità inerente.
6 ore	Forme normali. Forma normale di Chomsky. Pumping lemma per i linguaggi CF. Applicazioni del pumping lemma. Proprietà di chiusura dei linguaggi CF. Proprietà di decisione per i linguaggi CF

4 ore	Automi a Pila (PDA). Linguaggi riconosciuti da PDA. Equivalenza di PDA e grammatiche CF.
	ESERCITAZIONI
12 ore (1 CFU)	Seminari su argomenti integrativi e complementari svolti dagli studenti stessi dopo una messa a punto e una preparazione degli argomenti discussa assieme
TESTI CONSIGLIATI	J. E. Hopcroft, R. Motwani, J. D. Ullman, Automi, Linguaggi e Calcolabilità, Addison-wesley (PearsonEducation Italia) 2003. R. McNaughton, Elementary Computability, Formal Languages and Automata, Prentice-Hall, 1082 D. Perrin, Finite Automata, Capitolo 1 del Vol.2 del Handbook of Theoretical Computer Science, Elsevir, 1990.

OBIETTIVI FORMATIVI DEL MODULO II

L'obiettivo del modulo è quello di fornire agli studenti gli elementi di base, concettuali e formali, della teoria della calcolabilità, mettendo in evidenza i rapporti esistenti tra alcuni risultati teorici di carattere generale e alcuni problemi e domande che sorgono a partire da aspetti apparentemente solo "tecnici" della programmazione.
Riportati nel Regolamento Didattico del Corso di Studio

MODULO	TEORIA DELLA CALCOLABILITA'
ORE FRONTALI	LEZIONI FRONTALI
40 (5 CFU)	
4 ore	Aspetti generali della nozione di calcolabilità Centralità della nozione di calcolabilità. Analisi di Turing del processo di calcolo. Enunciato e discussione della tesi di Church-Turing. Primi esempi di funzioni Turing-calcolabili. Definizione di produttività di una Macchina di Turing (MdT). Definizione della funzione p (produttività massima delle MdT a n stati). Dimostrazione della non Turing calcolabilità della funzione p. Presentazione intuitiva di vari "explicata" formali dell' "explicandum" informale di funzione calcolabile. Le funzioni ricorsive primitive. Costruzione della funzione di Ackermann. Definizione di funzione ϵ -ricorsiva e μ ricorsiva.
6 ore	Le funzioni ricorsive primitive. Definizione e proprietà principali. Dimostrazione della ricorsività primitiva di varie funzioni elementari. Metodi di codifica ricorsivi primitivi. Numeri di Godel. Il linguaggio di programmazione S di Davis/Weyuker Dimostrazione della definibilità di varie funzioni in S. Introduzione del concetto di macro. Codifiche dei programmi di S. Calcolabilità in S delle funzioni ricorsive primitive. Il teorema della "fermata". Esistenza di programmi "universali".
5 ore	Definizione di insieme ricorsivamente enumerabile (r.e.) e di insieme ricorsivo. Teoremi sulle relazioni intercorrenti tra insiemi ricorsivamente enumerabili e insiemi ricorsivi. Il teorema di Post. Dimostrazione dell' esistenza di insiemi ricorsivamente enumerabili ma non ricorsivi. Il teorema s-m-n (o del parametro) di Kleene. Alcune sue conseguenze.
5 ore	Il linguaggio di programmazione LOOP di Meyer e Ritchie . Dimostrazione della LOOP-calcolabilità delle funzioni ricorsive primitive. Dimostrazione dell'equivalenza tra funzioni ricorsive primitive e funzioni calcolabili da programmi LOOP. Teoremi di limitazione alla crescita delle funzioni ricorsive primitive. Profondità di nidificazione dei cicli LOOP. La gerarchia Ln. Dimostrazione della non ricorsività primitiva della funzione di Ackermann. Inverso del teorema di limitazione alla crescita. Dimostrazione della calcolabilità in S della funzione di Ackermann. Introduzione del linguaggio WHILE come estensione del linguaggio LOOP. Dimostrazione della equivalenza tra il linguaggio S e il linguaggio WHILE.
5 ore	Linguaggi di programmazione S_n per il calcolo di stringhe su un alfabeto di n simboli. Simulazione in S_n delle funzioni calcolabili in S. Introduzione del linguaggio di T di Post - Turing e dimostrazione della calcolabilità in T delle funzioni parzialmente calcolabili in S_n . Dimostrazione della calcolabilità in S delle funzioni calcolabili da programmi di Post-Turing. Dimostrazione dell'equivalenza tra MdT a quadruple, MdT a quintuple e programmi di Post Turing. Dimostrazione dell'equivalenza tra MdT con nastro infinito bidirezionale e MdT con nastro infinito in una sola direzione. Macchine di Turing non deterministiche.

5 ore	Processi di Thue e simulazione di MdT non deterministiche mediante processi di Thue. Definizione di grammatica. Dimostrazione dell'equivalenza tra i linguaggi accettati da MdT non deterministiche e i linguaggi generati da grammatiche. Ricorsività primitiva degli operatori di derivabilità in una grammatica. Dimostrazione dell'equivalenza tra linguaggi ricorsivamente enumerabili e linguaggi generati da una grammatica. Varie caratterizzazioni degli insiemi ricorsivamente enumerabili. Il teorema della forma normale di Kleene. Il problema della corrispondenza di Post e dimostrazione della sua insolubilità algoritmica. Dimostrazione dell'equivalenza tra funzioni calcolabili in S e funzioni μ -ricorsive. Non ricorsiva enumerabilità dell'insieme di indici delle funzioni ricorsive totali.
2 ore	Cenni al decimo problema di Hilbert e agli insiemi diofantei. Il teorema di Matjasievic (senza dimostrazione)
4 ore	Il problema della complessità. Difficoltà di fornire modelli formali generali ed onnicomprensivi della nozione di complessità. La complessità astratta. Gli assiomi di Manuel Blum. Teorema del collegamento ricorsivo tra misure di complessità. Il teorema della lacuna. il teorema dell'accelerazione di Blum (senza dimostrazione).
4 ore	La complessità concreta. Calcolabilità in tempo polinomiale. Le classi di problemi P ed NP. Definizione di problema NP completo. . Il problema della soddisfacibilità. Il teorema di Cook e la tesi di Cook-Karp. Cenni al problema $P=?NP$. I sette Problemi del Millennio come riproposizione dei problemi di Hilbert al Convegno del 1900.
ESERCITAZIONI	
12 ore (1 CFU)	Seminari su argomenti integrativi e complementari svolti dagli studenti stessi dopo una messa a punto e una preparazione degli argomenti discussa assieme a tutta la classe.
TESTI CONSIGLIATI	M. Davis, E. Weyuker, Computability, Complexity and Languages, Academic Press (1983) G. S. Boolos, R. C. Jeffrey, Computability and Logic, Cambridge University Press (1989) Si suggerisce, inoltre, la lettura di E. Casari, Computabilità e ricorsività, Quaderni della Scuola Superiore di Idrocarburi dell'ENI (1959)

FACOLTÀ	Scienze
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Algoritmi e Strutture Dati
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline informatiche
CODICE INSEGNAMENTO	01175
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	Raffaele Giancarlo Professore Ordinario Università di Palermo
DOCENTE COINVOLTO (MODULO 2)	Sabrina Mantaci Professore Associato Università di Palermo
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	104
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	96
PROPEDEUTICITÀ	Analisi Mat., Mat Discreta, Progr. e Labor C.I.
ANNO DI CORSO	II
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il sito: http://www.cs.unipa.it
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Obbligatoria
METODI DI VALUTAZIONE	Prova Orale, Prova Scritta
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito: http://www.cs.unipa.it
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Martedì e Giovedì Ore 15-17 Per entrambi i moduli

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione degli strumenti di base per l'analisi ed il progetto di algoritmi. Capacità di utilizzare il linguaggio specifico proprio della disciplina.

Capacità di applicare conoscenza e comprensione

Capacità di sviluppare software basati su algoritmi efficienti per problemi elementari

Autonomia di giudizio

Essere in grado di valutare le implicazioni e i risultati degli studi algoritmici che segue e della complessità computazionale dei problemi ad essi associati.

Abilità comunicative

Capacità di esporre i risultati salienti degli studi algoritmici, anche ad un pubblico non esperto. Essere in grado di evidenziare le ricadute tecnologiche delle teorie studiate.

Capacità d'apprendimento

Capacità di aggiornamento con la consultazione di testi avanzati propri del settore dell'algoritmica. Capacità di seguire, utilizzando le conoscenze acquisite nel corso, sia master di primo livello, che corsi di laurea magistrali

OBIETTIVI FORMATIVI DEL MODULO

Esporre lo studente a tecniche fondamentali di progetto ed analisi di algoritmi. In particolare, si copre tutto lo spettro delle strutture dati e dei principali paradigmi algoritmi, incluso lo studio di complessità computazionale di problemi intrattabili.

MODULO	Algoritmi e Strutture Dati- Fondamenti Teorici
ORE FRONTALI	LEZIONI FRONTALI
8	MODELLI di CALCOLO, COMPLESSITA' COMPUTAZIONALE E ALGORITMI Random Access Machines, Complessità Computazionale RAM, Macchine di Turing e loro Complessità di Tempo. Relazione tra Macchina di Turing e RAM. Complessità Computazionale e Linguaggi di Programmazione ad Alto Livello. Alberi di Decisione e Lower Bounds per l'ordinamento.
10	PARADIGMI PER IL PROGETTO DI ALGORITMI EFFICIENTI Divide et Conquer, Programmazioni Dinamica, Tecniche Greedy. Esempi: Ricerca Minimo e Massimo, Moltiplicazione d'interi, Moltiplicazione di Matrici; Mergesort; Il Quicksort. Analisi worst case e analisi caso medio. Prodotto di n matrici. Longest Common Subsequence, Riconoscimento Grammatiche Context Free. Algoritmi Greedy: Optimal Storage on Tapes. Il Problema dello Zaino (versione “greedy”)
10	STRUTTURE DATI ED OPERAZIONI SU INSIEMI Operazioni Fondamentali su Insiemi. Tabelle Hash. Union-find. Alberi di Ricerca Ottimi, Schemi di Alberi Bilanciati, Dizionari e Code a Priorità, Mergeable Heaps, Code Concatenabili.
10	ALGORITMI SU GRAFI Rappresentazione di Grafi, Visite su Grafi, Biconnettività e Connettività Forte, Algoritmi di Spanning Tree Minimo, Algoritmi per Cammini Ottimi.
10	TEORIA DELL' NP- COMPLETEZZA Macchine di Turing Non Deterministiche, Le classi P ed NP, NP Completezza del Problema della Soddisfattibilità. Ulteriori Problemi Np Completi.
	ESERCITAZIONI
TESTI CONSIGLIATI	<i>Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano, Algoritmi e Strutture dati, McGraw Hill, 2005 H. Cormen. C. Leiserson, R. Rivest, C. Stein</i> Introduzione agli algoritmi e strutture dati, McGraw Hill, 2001 <i>A.V. Aho, J.E. Hopcroft, J.D. Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley, 1974 R. Sedgewick, Algoritmi in C, Addison Wesley, 1980</i>

OBIETTIVI FORMATIVI DEL MODULO

L'obbiettivo formativo del modulo è sia di complemento che duale al primo, in quanto si presentano sia tecniche matematiche che aspetti ingegneristici di natura fondamentale per l' analisi e l'implementazione di algoritmi efficienti

MODULO	Algoritmi e Strutture Dati-Fondamenti Matematici ed Ingegneristici
ORE FRONTALI	LEZIONI FRONTALI
6	NOZIONI INTRODUTTIVE Algoritmi e Strutture Dati. Il Problema della connettività, algoritmi di Union-Find: quick-find, quick-union, quick-union pesata. Implementazione in C.
10	TECNICHE EMPIRICHE E MATEMATICHE PER L'ANALISI DI ALGORITMI Analisi Empiriche. Valutazione empirica degli algoritmi di Union-Find. Analisi degli algoritmi. Velocità di crescita delle funzioni. Ricorrenze Fondamentali. Master Theorem e sua generalizzazione. Studio della crescita di alcune serie fondamentali. Metodo del confronto tra serie e integrali.
8	PARADIGMI DI PROGETTO DI ALGORITMI Divide et Impera: ricerca del minimo e del massimo, ricerca binaria e loro implementazione in C. Programmazione Dinamica: algoritmo per il calcolo dei numeri di Fibonacci, calcolo ad un passo dei numeri di Fibonacci, complessità della soluzione ricorsiva e iterativa e loro implementazione in C. Knapsack problem e sua implementazione in C. Algoritmo per le Torri di Hanoi e sua complessità.
8	ALGORITMI DI ORDINAMENTO Lower bound per gli algoritmi di ordinamento: caso pessimo e caso medio. Mergesort, Heapsort. Quicksort e loro implementazione in C.
2	STRUTTURE DATI ELEMENTARI Array, liste concatenate, stringhe e loro implementazione in C.
4	STRUTTURE DATI ASTRATTE Pile, Code e loro implementazione in C mediante array e liste concatenate. Valutazione di un'espressione in forma postfixa mediante una pila e sua implementazione in C.
10	GRAFI ED ALBERI Strutture dati per la rappresentazione di grafi ed alberi in C. Algoritmi di visita su alberi e applicazione alla valutazione delle espressioni in forma postfixa e prefixa. Albero di ricoprimento di costo minimo di un grafo non orientato: algoritmo di Kruskal e sua implementazione in C, algoritmo di Prim. Visita DFS in un grafo orientato e non orientato. Connettività forte in un grafo orientato, componenti fortemente connesse e loro proprietà, algoritmo per il calcolo delle componenti fortemente connesse e sua implementazione in C.
	ESERCITAZIONI
TESTI CONSIGLIATI	<i>R. Sedgwick</i> – Algoritmi in C, Addison-Wesley. <i>T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein</i> - Introduzione agli Algoritmi e strutture dati, McGraw Hill. <i>A.V. Aho, J.E. Hopcroft, J.D. Ullman</i> , The Design and Analysis of Computer Algorithms, Addison Wesley. <i>C. Demetrescu, I. Finocchi, G.F. Italiano</i> , Algoritmi e Strutture Dati, McGraw-Hill.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Teoria e Tecniche di Compilazione
TIPO DI ATTIVITÀ	Attività caratterizzanti
AMBITO DISCIPLINARE	Formazione informatica
CODICE INSEGNAMENTO	07488
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Marinella Sciortino Professore Associato Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	102
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	48
PROPEDEUTICITÀ	Analisi Matematica, Matematica Discreta, Programmazione e Laboratorio, Informatica Teorica, Algoritmi e Strutture Dati.
ANNO DI CORSO	Terzo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Dipartimento di Matematica ed Applicazioni di Palermo
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Attività in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale con discussione di un elaborato
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Consultare il docente: mari@math.unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Il corso intende fornire agli studenti le nozioni necessarie per comprendere ed affrontare le diverse problematiche relative alle diverse fasi della compilazione con particolare attenzione all'analisi lessicale, sintattica e semantica ma che trovano applicazione anche in altri contesti (traduzioni di linguaggi, parser, scanner). Il corso si prefigge anche di trasmettere la conoscenza di importanti strumenti di generazione automatica di parser e scanner (BISON, FLEX)

Capacità di applicare conoscenza e comprensione

Il corso ha come obiettivo rendere lo studente capace di comprendere il funzionamento degli analizzatori lessicali e sintattici, gli strumenti pratici per la realizzazione di tali analizzatori, il procedimento richiesto per trasformare gli analizzatori in traduttori, alcuni aspetti avanzati della compilazione di linguaggi moderni ed alcune tecniche di analisi automatica di correttezza di programmi.

Autonomia di giudizio

Gli studenti sono guidati ad apprendere in maniera critica e responsabile tutto ciò che viene spiegato loro in classe e ad arricchire le proprie capacità di giudizio attraverso lo studio del materiale didattico indicato dal docente.

Abilità comunicative

Attraverso le attività di laboratorio previste, il corso tenderà a favorire lo sviluppo della capacità di saper comunicare in modo chiaro le conclusioni, nonché le conoscenze e le ragioni sottostanti. Gli studenti dovranno anche sviluppare la capacità di saper lavorare in gruppo, di confrontarsi sulle problematiche al fine di individuare le soluzioni in base alle conoscenze acquisite durante il corso.

Capacità d'apprendimento

Attraverso approfondimenti e consultazione dei testi di riferimento, gli studenti saranno stimolati ad una conoscenza più approfondita e critica dei linguaggi di programmazione a loro già noti, tramite lo studio di come tali linguaggi possono essere compilati.

OBIETTIVI FORMATIVI DEL CORSO

L'insegnamento ha due obiettivi. Il primo è di tipo culturale. Si vuole cioè introdurre una teoria, quella alla base della realizzazione dei compilatori per i linguaggi di programmazione, che rappresenta probabilmente il più importante contributo scientifico allo sviluppo e alla diffusione delle tecnologie informatiche. Il secondo obiettivo è più applicativo. Infatti, anche qualora lo studente non venga mai coinvolto (nella propria carriera lavorativa) nel progetto di un nuovo compilatore, è comunque altamente probabile che egli/ella si trovi spesso ad utilizzare metodologie e tecniche che di tale teoria fanno parte: dagli automi e le grammatiche come formalismi per definire il comportamento di un sistema, alle tecniche per realizzare traduttori molto più semplici di un compilatore vero e proprio (ad esempio, l'interprete di un file di configurazione). Verrà studiata la struttura generale di un compilatore, ponendo poi particolare enfasi agli aspetti di analisi lessicale e di parsing. Verranno studiati parser di tipo top-down (a discesa ricorsiva, predittivi e LL(1)) e di tipo bottom-up (LR e LALR). Infine, saranno esaminati esempi di traduttori per semplici linguaggi.

CORSO	TEORIA E TECNICHE DI COMPILAZIONE
ORE FRONTALI	LEZIONI FRONTALI
4	Introduzione: Linguaggi di Programmazione e Processori di Linguaggi di Programmazione. Compilatori e Linguaggi. Linguaggi Macchina, Linguaggi Assembly ed evoluzione dei linguaggi di programmazione. Compilatori e Interpreti. Macchina Virtuale. Struttura di un compilatore: Preprocessore. Linker e Loader. Fasi della compilazione. Front End e Back End. Passate di un compilatore.
6	Analisi lessicale: Operazioni preliminari. Token e Lessemi. Errori lessicali. Token e espressioni regolari. Definizioni regolari. Eliminazioni di ambiguità. Automi a stati finiti. Implementazioni di DFA. Simulazioni di NFA. Generazione automatica di scanner. Flex.
14	Analisi sintattica: Grammatiche context-free. Alberi di derivazione. Grammatiche ambigue. Automi a pila deterministici e non deterministici. Complessità di calcolo di un automa a pila. Algoritmo di Earley. Errori sintattici e metodi di gestione degli errori. Parser discendenti o top-down: Parser a discesa ricorsiva. Parser LL(1). Eliminazione della ricorsione sinistra. Fattorizzazione sinistra. Insieme First e Follow. Parser ascendenti o bottom-up: Parser shift-reduce. Parser LR(0). Parser SLR. Parser LR(1). Parser LALR(1). Proprietà dei linguaggi e delle grammatiche LR(k). Confronto tra le grammatiche LL(k) e LR(k). Generatori automatici di parser. Uso Bison

6	<p>Analisi semantica: Semantica statica e dinamica. Grammatiche con attributi. Semantica guidata dalla sintassi. Albero sintattico decorato. Calcolo degli attributi. Grafo delle dipendenze. Grammatiche con S-attributi. Grammatiche con L-attributi. Ordinamento topologico del grafo delle dipendenze. Tabella dei simboli. Vari tipi di implementazioni tramite array, liste concatenate e ABR. Implementazione tramite hash table with chaining. Attributi di visibilità e metodi di realizzazione. Type checking. Equivalenza di tipi. Type coercion.</p>
2	<p>Generazione del codice: Codice intermedio. Codice a tre indirizzi. Strutture dati per l'implementazione del 3AC. Codice per macchina virtuale. P-code. Ottimizzazione del codice. Esempi di ottimizzazioni indipendenti dalla macchina. Generatori di codice oggetto. Esempi di ottimizzazioni dipendenti dalla macchina.</p>
ATTIVITA' in LABORATORIO	
4	Esercizi risolti con l'ausilio di Flex
10	Esercizi risolti con l'ausilio di Bison. Uso della tabella dei simboli
TESTI CONSIGLIATI	<p>A. Aho, M. Lam, R. Sethi, J. Ullman Compilers, Principles, Techniques & Tools Addison Wesley Giorgio Bruno Linguaggi Formali e Compilatori Utet Libreria Stefano Crespi Reghizzi Linguaggi Formali e Compilazione Pitagora Editrice</p>

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Reti di Calcolatori
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Formazione informatica
CODICE INSEGNAMENTO	06232
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Biagio Lenzitti Ricercatore Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	94
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	56
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	3°
SEDE DI SVOLGIMENTO DELLE LEZIONI	Laboratorio, via Ingrassia
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Mercoledì delle ore:15:00 alle17:00

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Conoscenza delle Reti e dei protocolli circuit switching e packet switching Conoscenza dello stack ISO/OSI e dello stack TCP/IP. Conoscenza del livello host-to-network e delle tecnologie e componenti di Ethernet Conoscenza del livello IP. Conoscenza degli autonomous systems, degli algoritmi di Routing in Internet, del protocollo ICMP. Conoscenza del livello di trasporto e dei protocolli UDP e TCP. Conoscenza dei Domain Name System. Conoscenza del livello applicativo e di un esempio di protocollo: WWW, client e server Web.

Capacità di applicare conoscenza e comprensione

Capacità di valutare le funzionalità dei diversi protocolli di rete. Capacità di valutare le prestazioni di una rete di calcolatori.

Autonomia di giudizio

Capacità di valutare e comparare autonomamente le soluzioni di un problema di limitata complessità.

Abilità comunicative

Capacità di organizzarsi in gruppi di lavoro. Capacità di comunicare efficacemente in forma orale anche utilizzando

termini in inglese.

Capacità di apprendere

Capacità di catalogare, schematizzare e rielaborare le nozioni acquisite.

OBIETTIVI FORMATIVI DEL MODULO

Fornire le competenze di base, sia metodologiche che tecniche, sulle reti di calcolatori

MODULO	Reti di Calcolatori
ORE FRONTALI	LEZIONI FRONTALI
2	Concetti base sulla comunicazione
1	Reti di Calcolatori ed Internet
1	La rete esterna ed interna
2	La rete di accesso e Mezzi fisici
1	Ritardi e perdite nella commutazione di pacchetto
1	I modelli di riferimento OSI e TCP/IP
6	Livello Applicazione Caratteristiche principali Programmare con le socket
6	Word Wide Web e HTTP FTP Posta Elettronica DNS
8	Il Livello di Trasporto UDP TCP Gestione della connessione Trasferimento dati affidabile Controllo della congestione
4	Il livello di Rete Protocollo IP NAT Protocollo ICMP
4	I protocolli di routing in Internet
6	Il livello di collegamento tipi di collegamenti e protocolli relativi. Indirizzamento LAN e ARP
2	Le Reti LAN Interconnessione
4	Reti Wireless
	ESERCITAZIONI
8	Traceroute Client/server TCP Client/server UDP httpdclient.c httpdserver.c Ftp con telnet SMTP con telnet comando host esercitazione con Wireshark
TESTI CONSIGLIATI	Reti di Calcolatori e Internet, 3a ed. , James F. Kurose e Keith W. Ross, Pearson, 2005 Reti di Calcolatori , 4a ed. , Andrew S. Tanenbaum, Pearson, 2003 Internetworking con TCP/IP , 5a ed. , Douglas E. Comer, Pearson, 2006

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2009/2010
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Analisi di Immagini
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Formazione informatica
CODICE INSEGNAMENTO	09474
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	Cesare Fabio Valenti Ricercatore Università di Palermo
CFU	3
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	51
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	24
PROPEDEUTICITÀ	
ANNO DI CORSO	Terzo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Laboratorio C
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale, Presentazione di una Tesina
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Da definire
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Da programmare
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Su appuntamento

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Acquisizione degli strumenti avanzati per l'analisi di immagini digitali e per la progettazione di sistemi di elaborazione e di visione artificiale. Capacità di utilizzare il linguaggio specifico del settore.

Capacità di applicare conoscenza e comprensione

Capacità di riconoscere le principali caratteristiche informative delle immagini e di progettare un sistema ad-hoc per la loro elaborazione e interpretazione.

Autonomia di giudizio

Essere in grado di valutare le implicazioni e i risultati del sistema di elaborazione di immagini, in considerazione della loro natura e dell'uso delle informazioni prodotte (ad esempio, per indagini biomediche o dati satellitari).

Abilità comunicative

Capacità di esporre lo specifico problema affrontato e i risultati previsti dal sistema sviluppato. Essere in grado di sostenere ed evidenziare l'importanza e l'attendibilità dell'elaborazione prodotta (ad esempio, validazione della discriminazione non

supervisionata).

Capacità d'apprendimento

Capacità di aggiornamento con la consultazione delle pubblicazioni scientifiche proprie del settore dell'analisi di immagini, della visione artificiale e, più in generale, della teoria degli algoritmi. Capacità di seguire, utilizzando le conoscenze acquisite nel corso, sia master di secondo livello, sia corsi d'approfondimento sia seminari specialistici nel settore dell'analisi di immagini e della visione artificiale.

OBIETTIVI FORMATIVI DEL MODULO 1

Obiettivo del modulo è lo studio degli strumenti di base per l'analisi di immagini digitali. In particolare, sono presentati il teorema della convoluzione, esempi di filtri non lineari, operatori spaziali, morfologia matematica a scala di grigio, tecniche di miglioramento della qualità, algoritmi di segmentazione e compressione. Sono descritti i principali metodi di acquisizione delle immagini e i formati grafici più diffusi per la loro corretta memorizzazione. Case study sono presentati durante il corso per evidenziare l'applicazione delle tecniche discusse su dati reali. È altresì realizzato un ambiente minimale d'elaborazione delle informazioni per correggere eventuali artefatti, individuare/intrepretare le caratteristiche distintive e classificare gli oggetti presenti nella scena, con ridotto intervento da parte dell'utente.

MODULO 1	ANALISI DI IMMAGINI
ORE FRONTALI	LEZIONI FRONTALI
1	Introduzione alla visione artificiale e al sistema percettivo umano
2	Sistemi di acquisizione digitale; confronto apparecchiature con 1 ccd e 3 ccd.
2	Percezione del colore e luminanza. Spazi colore rgb/yuv/hsv.
2	Retina digitale; intorni digitali; metriche discrete; teorema di Shannon; paradosso di Jordan.
2	Immagini truecolor e indicizzate; quantizzazione.
2	Operatori aritmetici e logici. Bitplane e codici di Gray.
3	Principali filtri (media, mediano, sharpen, Gauss, Laplace, Sobel, Prewitt).
1	Rotazione e ridimensionamento di immagini (interpolazioni nearest, bilineare e bicubica).
2	Morfologia matematica a scala di grigio (erosione, dilatazione, apertura, chiusura, individuazione contorni, top-hat, bottom-hat, kappa and sharpen); formula di Eulero; minimo rettangolo di ricoprimento; granulometria.
1	Istogrammi; stretching; equalizzazione; fixed/optimal/adaptive/iterative/dynamic threshold.
1	Segmentazione; quadtree (compressione e split+merge).
2	Compressione di immagini digitali (lossy/lossless); misure di errore; codifica interlacciata; cenni ai principali formati grafici (bmp/gif/jpg).
3	Trasformata discreta coseno; short-time-fourier-transform; wavelets e filter bank; trasformata di Haar; decomposizione standard e non-standard; algoritmo "a trous".
	ESERCITAZIONI
	Ciascuna lezione è integrata da esempi ed esercitazioni al calcolatore

TESTI CONSIGLIATI	<ul style="list-style-type: none">• R.C.Gonzales, R.E.Woods. Elaborazione delle Immagini Digitali. Pearson – Prentice Hall, 2008.• R.C.Gonzales, R.E.Woods, S.L.Eddins. Digital Image Processing using Matlab. Prentice Hall, 2004.• A.S.Glassner. Principles of Digital Image Synthesis. Morgan Kaufmann Publishers, 1995.• L.G.Shapiro, G.C.Stockman. Computer Vision. Prentice Hall, 2001.• P.Soille. Morphological Image Analysis. Springer-Verlag, 2003.
------------------------------	---