

CORSO DI LAUREA IN INFORMATICA

Sito web del CdL: <http://www.cs.unipa.it/>

Insegnamenti		
I	Geometria	X
I	Analisi Matematica	X
I	Matematica Discreta	X
I	Programmazione e Laboratorio	X
I	Fisica	X
I	Architetture	X
I	Inglese	Non disponibile

Insegnamenti		
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	Non disponibile

Insegnamenti		
III	Sistemi di Elaborazione	X
III	Sistemi Operativi II	X
III	Compilatori	X
III	Reti di Calcolatori	X
III	Grafica al Calcolatore	X
A Scelta	Linguaggi XML	X
A Scelta	Basi di Dati II	Non disponibile

FACOLTÀ	Scienze MM. FF. NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	ARCHITETTURE
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline Informatiche
CODICE INSEGNAMENTO	01478
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	--
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Gaetano Gerardi Prof. Associato Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	120
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	60
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
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il calendario sul sito web del CdL
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 l'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 dell'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 solidi 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 con logica random.

DELL'INSEGNAMENTO	ARCHITETTURE I
ORE FRONTALI	LEZIONI FRONTALI
2	Introduzione al corso e illustrazione del materiale didattico PSPICE e XILINX
5	Introduzione ai Sistemi Numerici e codifica delle informazioni con e senza ridondante
4	Operatori binari tabelle di verità e mappe. Realizzazione hardware delle funzioni logiche elementari con tecnologia MOSFET (CMOS).
4	Algebra booleana e teoremi relativi. Minimizzazione delle funzioni con il metodo 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) non clocked e clocked. Triggering con il primo o secondo fronte del clock. Flip-Flop Master/Slave e Edge Triggered e loro equazione di stato. 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.
7	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.
Totale ore 47	
	ESERCITAZIONI
8	Esercitazioni in aula di analisi e progetto di circuiti logici
5	Simulazione PSPICE (student edition) Simulazione di sistemi combinatori e sequenziali

FACOLTÀ	Scienze
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA (o LAUREA MAGISTRALE)	Informatica
INSEGNAMENTO	Algoritmi e Strutture Dati
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	INF/01
CODICE INSEGNAMENTO	01175
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE COINVOLTO (MODULO 1)	Sabrina Mantaci Professore Associato Università di Palermo
DOCENTE RESPONSABILE (MODULO 2)	Raffaele Giancarlo 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	Analisi Matematica Matematica Discreta Programmazione
ANNO DI CORSO	II
SEDE DI SVOLGIMENTO DELLE LEZIONI	Vedere Calendario Lezioni sul sito web del CdL
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Obbligatoria
METODI DI VALUTAZIONE	Prova Orale, Prova Scritta, Prova Pratica
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo e secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Vedere Calendario Lezioni sul sito web del CdL
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.

Modulo 1 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 algoritmici, incluso lo studio di complessità computazionale di problemi intrattabili.

MODULO 1	Algoritmi e Strutture Dati-Fondamenti Matematici ed Ingegneristici
ORE FRONTALI	LEZIONI FRONTALI
5	NOZIONI INTRODUTTIVE Algoritmi e Strutture Dati. Nozioni introduttive per la soluzione algoritmica di un problema, diverse soluzioni per uno stesso problema. Il problema dell'efficienza di un algoritmo.
10	TECNICHE EMPIRICHE E MATEMATICHE PER L'ANALISI DI ALGORITMI Analisi Empiriche. Analisi degli algoritmi. Velocità di crescita delle funzioni. Ricorrenze Fondamentali. Soluzioni delle equazioni di ricorrenza. Metodo dell'iterazione, Metodo di Sostituzione e Master Theorem. Studio della crescita di alcune serie fondamentali. Metodo del confronto tra serie e integrali.
1	STRUTTURE DATI ELEMENTARI Array, liste concatenate, stringhe e loro implementazione in C.
3	STRUTTURE DATI ASTRATTE Pile, Code e loro implementazione in C mediante array e liste concatenate. Valutazione di un'espressione in forma postfissa mediante una pila e sua implementazione in C.
15	ALGORITMI DI ORDINAMENTO Lower bound per gli algoritmi di ordinamento: caso pessimo e caso medio. Selectionsort, Bubblesort, Mergesort, Heapsort. Quicksort, Countingsort e Radixsort e loro implementazione in C.
3	Il Problema della connettività, algoritmi di Union-Find: quick-find, quick-union, quick-union pesata. Implementazione in C.

6	<p>PARADIGMI DI PROGETTO DI ALGORITMI</p> <p>Ricorsione. Divide et Impera: ricerca del minimo e del massimo, ricerca binaria e loro implementazione in C. Programmazione Dinamica: Distanza di Hamming fra due stringhe. La massima sottosequenza comune. Paradigma Greedy. Il distributore automatico di resto. Knapsack problem e sua implementazione in C. Algoritmo per le Torri di Hanoi e sua complessità.</p>
5	<p>GRAFI ED ALBERI</p> <p>Strutture dati per la rappresentazione di grafi ed alberi in C. Algoritmi di visita su alberi. Visita DFS e BFS. Applicazione alla valutazione delle espressioni in forma postfissa e prefissa.</p>
ESERCITAZIONI	
TESTI CONSIGLIATI	<p><i>R. Sedgewick</i> – Algoritmi in C, Addison-Wesley.</p> <p><i>T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein</i> - Introduzione agli Algoritmi e strutture dati, McGraw Hill.</p> <p><i>A.V. Aho, J.E. Hopcroft, J.D. Ullman</i>, The Design and Analysis of Computer Algorithms, Addison Wesley.</p> <p><i>C. Demetrescu, I. Finocchi, G.F. Italiano</i>, Algoritmi e Strutture Dati, McGraw-Hill.</p>

Modulo 2 : OBIETTIVI FORMATIVI DEL MODULO

L'obiettivo formativo del modulo è quello di presentare sia tecniche matematiche che aspetti ingegneristici di natura fondamentale per l'analisi e l'implementazione di algoritmi efficienti

MODULO	Algoritmi e Strutture Dati- Fondamenti Teorici
ORE FRONTALI	LEZIONI FRONTALI
8	<p>MODELLI di CALCOLO, COMPLESSITA' COMPUTAZIONALE E ALGORITMI</p> <p>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.</p>
10	<p>PARADIGMI PER IL PROGETTO DI ALGORITMI EFFICIENTI</p> <p>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")</p>
10	<p>STRUTTURE DATI ED OPERAZIONI SU INSIEMI</p> <p>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.</p>

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	<p><i>Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano, Algoritmi e Strutture dati, McGraw Hill, 2005</i></p> <p><i>H. Cormen. C. Leiserson, R. Rivest, C. Stein</i> Introduzione agli algoritmi e strutture dati, McGraw Hill, 2001</p> <p><i>A.V. Aho, J.E. Hopcroft, J.D. Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley, 1974</i></p> <p><i>R. Sedjewick, Algoritmi in C, Addison Wesley, 1980</i></p>

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Analisi di Immagini
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline informatiche
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	48
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	24
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Terzo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il calendario sul sito web del CdL
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale, presentazione di un elaborato
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Consultare il calendario sul sito web del CdL
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il calendario sul sito web del CdL
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare il docente: cvalenti@math.unipa.it

<p>RISULTATI DI APPRENDIMENTO ATTESI</p> <p>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.</p> <p>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.</p> <p>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).</p> <p>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).</p> <p>Capacità d'apprendimento Capacità di aggiornamento con la consultazione delle pubblicazioni scientifiche proprie del settore</p>

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; hard / 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
	Le lezioni saranno integrate da esempi e/o esercitazioni al calcolatore. Apposite dispense saranno distribuite durante il corso.
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.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Analisi Matematica
TIPO DI ATTIVITÀ	Attività formative di base
AMBITO DISCIPLINARE	Formazione matematico-fisica
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 degli Studi di Palermo
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	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	http://www.scienze.unipa.it/informatica/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Modulo 1: Lunedì dalle 15:30 alle 17:00 e/o studio 16, I piano Modulo 2: Lunedì dalle 15:30 alle 17:00 e/o studio 18, I piano, Dipartimento di Matematica ed Informatica, 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 1"

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 1
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.
12	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.
3	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.
15	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.
4	Primitive e integrali indefiniti: Definizioni e regole di calcolo – Integrali indefiniti immediati.
	ESERCITAZIONI
12	Esempi ed esercizi sugli argomenti trattati nel corso.
TESTI CONSIGLIATI	C. Di Bari – P. Vetro, Matematica Teoria ed esercizi, Libreria Dante Editrice

OBIETTIVI FORMATIVI DEL MODULO "ANALISI MATEMATICA 2"

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 2
ORE FRONTALI	LEZIONI FRONTALI
6	Integrale di Riemann – Proprietà dell'integrale di Riemann e regole di calcolo – Funzione integrale - Applicazioni al calcolo di aree e di volumi.
18	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.
10	Serie numeriche, successioni e serie di funzioni: Definizioni - Criteri di convergenza e criteri di convergenza assoluta - Criterio di Leibniz - Successioni e serie di funzioni – Serie di potenze.

14	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	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Basi di Dati 1
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	INF/01
CODICE INSEGNAMENTO	11084
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	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 Matematica Matematica Discreta Programmazione
ANNO DI CORSO	II
SEDE DI SVOLGIMENTO DELLE LEZIONI	Vedere Calendario Lezioni sul sito del CdL
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Scritta, Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Vedere Calendario Lezioni sul sito del CdL
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 i linguaggi di interrogazione di Database.

Capacità di applicare conoscenza e comprensione

Capacità di progettare reali Basi di dati per la gestione di dati di aziende e organizzazioni.

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
2	Il concetto di Base di Dati all'interno di un sistema informativo. I problemi di ridondanza e inconsistenza dei dati. Condivisione. Il DBMS. Basi di dati versus file systems.
8	I modelli di dati. Schema fisico Schema logico e Schema esterno. L'indipendenza dei dati. I linguaggi per basi di dati. DDL e DML. Il modello relazionale. Il concetto di relazione. Informazioni incomplete. La gestione dei valori nulli. I vincoli di integrità. Chiavi e superchiavi. Vincoli di integrità referenziali.
8	Linguaggi di interrogazione di basi di dati. L'Algebra Relazionale. Gli operatori dell'algebra Relazionale. Gli operatori insiemistici. Ridenominazione. Proiezione e Selezione. Il Join. Join Naturale, Theta Join, Equi Join, Self Join, Join esterno. Esercizi. Equivalenza fra espressioni algebriche. Viste in algebra relazionale.
10	Il linguaggio di interrogazione SQL. Il Data Definition Language. Creazione di tabelle. attributi, domini e valori di default. Vincoli di integrità. Primary Key, Unique, not null. Vincoli di Integrità referenziale. Foreign key. Reazioni alla violazione. Il vincolo Check. Cancellazione di tabelle e modifica di tabelle. Indici. Data Manipulation Language. Interrogazioni semplici in SQL. Proiezione, selezione e ridenominazione in SQL. Espressione delle condizioni mediante la clausola where. Ordinamenti, Funzioni di gruppo e raggruppamenti in SQL. Il Join in SQL. Le due sintassi. Self Join. Gli operatori Booleani: Unione, Intersezione e Differenza. Subquery. Funzioni di gruppo e raggruppamenti.
10	La Progettazione di una Base di Dati. Metodologia di Progetto. Progetto Concettuale, logico e fisico. La Progettazione Concettuale. Il modello Entity-Relationship. I Costrutti del modello Entity-Relationship. Entità. Associazione. Attributo. Cardinalità: associazioni uno a uno, uno a molti, molti a molti. Identificatore interno ed esterno. Generalizzazione, ereditarietà. Generalizzazione totale o parziale. Generalizzazione esclusiva o sovrapposta. Esercizi. Documentazione associata agli schemi concettuali La progettazione Concettuale. Analisi dei requisiti. Costruzione del modello concettuale. Scelta tra Entità e Associazione. Riconoscere le generalizzazioni. Pattern di progetto. Strategie di Progetto. Bottom-Up, Top-Down e Inside-Out. Metodologie per la creazione del modello concettuale. Esempi
5	La Progettazione Logica. Ristrutturazione del modello concettuale. Valutazione delle prestazioni. Analisi delle ridondanze, eliminazione delle generalizzazioni, Partizionamento/accorpamento di concetti, Scelta degli identificatori principali. Traduzione nel modello logico. Associazioni molti a molti, associazioni uno a molti, associazioni uno a uno. Traduzione di schemi complessi.

5	<p>La Normalizzazione. Anomalie. Il concetto di Dipendenza Funzionale. Anomalie derivanti dalla presenza di dipendenza funzionali. La Forma Normale di Boyce e Codd. Decomposizione senza perdita. Conservazione delle dipendenze. Terza forma Normale. Decomposizione in terza forma normale. La teoria delle dipendenze. Chiusura funzionale di un insieme di attributi. Algoritmo per il calcolo della chiusura funzionale. Insiemi di dipendenze funzionali equivalenti. Insiemi di dipendenze funzionali non ridondanti e ridotti. Trasformazione di un sistema di dipendenze funzionali in uno non ridondante e ridotto. Algoritmo per la decomposizione in terza forma normale.</p>
<p>TESTI CONSIGLIATI</p>	<ul style="list-style-type: none"> • Atzeni, Ceri, Paraboschi, Torlone, <i>Basi di dati - Modelli e linguaggi di interrogazione</i>, McGraw-Hill. • Albano, Ghelli, Orsini, <i>Fondamenti di Basi di dati</i>, Zanichelli.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Compilatori
TIPO DI ATTIVITÀ	Attività caratterizzanti
AMBITO DISCIPLINARE	Formazione informatica
CODICE INSEGNAMENTO	14049
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	Vedere calendario sul sito del CdL
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Da programmare (contattare la 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. Insiemi 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	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.
2	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.
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	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

FACOLTÀ	Scienze Matematiche Fisiche Naturali
ANNO ACCADEMICO	2010/2011
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	Giovanni Peres - Professore Ordinario - Università di Palermo
DOCENTE COINVOLTO	Giuseppina Andalaro - Professore Associato - Università di Palermo
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	196
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE	104
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	PRIMO
SEDE DI SVOLGIMENTO DELLE LEZIONI	AULA IV – Dip. Matematica
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 e secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Lunedì 11,30-13,30, Mercoledì 9,30-10,30 Venerdì 8,30-10,30
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Da concordare con gli studenti (contattare peres@astropa.unipa.it; giuseppina.andalaro@libero.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

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

Capacità di applicare conoscenza e comprensione

Capacità di estendere l'analisi scientifica a contesti più ampi di quelli della Fisica e di applicare il metodo scientifico nella soluzioni di diversi problemi.

Autonomia di giudizio

Nel corso delle esercitazioni viene stimolato un approccio critico nell'apprendimento dei vari concetti e nella soluzione di problemi di Fisica, confrontando, ove possibile, diversi approcci o metodologie ad una trattazione, eventualmente scartando quelli meno adeguati o, ove applicabile, quelli inappropriati.

Abilità comunicative

Gli studenti sono invitati ad interagire nel corso della lezione e delle esercitazioni, esponendo la propria valutazione e la propria soluzione nel contesto affrontato al momento.

Capacità d'apprendimento

Si stimola l'approccio autonomo al testo scritto, alla sua analisi ed utilizzo.

Tutte le capacità vengono vagliate attentamente nel corso dell'esame scritto ed orale.

OBIETTIVI FORMATIVI DEL MODULO I - "MECCANICA E TERMODINAMICA"

Obiettivo del modulo è introdurre lo studente alla conoscenza delle grandezze, dei concetti e delle leggi della meccanica classica e della termodinamica.

MODULO I	MECCANICA E TERMODINAMICA
ORE FRONTALI	LEZIONI FRONTALI
1	Obiettivi della disciplina e cenni alle conoscenze matematiche utilizzate.
3	Moto in una e due dimensioni.
3	Dinamica del punto materiale.
4	Lavoro ed energia.
4	Impulso e quantità di moto.
3	Cinematica e dinamica del moto rotatorio.
4	Oscillazioni.
4	Onde
6	Termodinamica
	ESERCITAZIONI
2	Moto in una e due dimensioni
2	Dinamica del punto materiale
2	Lavoro ed energia.
2	Impulso e quantità di moto.
2	Cinematica e dinamica del moto rotatorio.
2	Oscillazioni.
2	Onde
2	Termodinamica
TESTI CONSIGLIATI	Halliday, Resnick, Walker - Fisica - Ambrosiana R. A. Serway - Fisica - EdiSES

OBIETTIVI FORMATIVI DEL MODULO II "ELETTROMAGNETISMO 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'Ottica sono affrontati con le leggi dell'Ottica Geometrica e dell'Ottica Fisica.

MODULO II	ELETTROMAGNETISMO 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.
6	Forze magnetiche, campi magnetici, sorgenti magnetiche
3	Legge di Faraday-Lenz
3	Onde, equazione d'onda, onde elettromagnetiche.
10	Ottica geometrica ed ottica ondulatoria.
	ESERCITAZIONI
12	Applicazioni numeriche su elettricità, magnetismo e induzione elettromagnetica
8	Applicazioni numeriche su ottica geometrica e ottica fisica
TESTI CONSIGLIATI	Halliday, Resnick, Walker - Fisica - Ambrosiana R. A. Serway - Fisica - EdiSES

FACOLTÀ	SCIENZE MM. FF. NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Geometria
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Formazione Matematico-fisica
CODICE INSEGNAMENTO	03675
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	MAT03
DOCENTE RESPONSABILE	Angela Speciale Prof. Stabilizzato 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 Informatica. 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	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	http://www.scienze.unipa.it/informatica/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Lunedì 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 artesiana: 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	M. Abate- C. de Fabritiis: Geometria analitica con elementi di algebra lineare – 2/ed – McGraw-Hill. M. Abate- C. de Fabritiis: Esercizi di geometria – McGraw-Hill.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Grafica al Calcolatore
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline informatiche
CODICE INSEGNAMENTO	14050
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	--
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	Cesare Fabio Valenti Ricercatore Università di Palermo
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	64
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	48
PROPEDEUTICITÀ	
ANNO DI CORSO	Terzo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare il sito: http://www.cs.unipa.it/
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale, presentazione di un elaborato
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Consultare il sito: http://www.cs.unipa.it/
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito: http://www.cs.unipa.it/
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare il docente: cvalenti@math.unipa.it

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. Acquisizione degli strumenti per la grafica al calcolatore. 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, per la realizzazione di sistemi virtuali).

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 grafica al calcolatore.

OBIETTIVI FORMATIVI DEL MODULO 1

Obiettivo del modulo è lo studio degli strumenti di base per l'analisi di immagini digitali e per la grafica al calcolatore. 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. Dopo una breve introduzione agli spazi vettoriali e alla trigonometria, saranno anche affrontati i problemi legati alla resa grafica di ambienti virtuali, nonché, ad esempio, i principali metodi per mosaicing, rendering, stereo vision e riduzione delle aberrazioni dovute all'ottica di acquisizione. 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/intepretare 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; hard / 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".
1	Introduzione agli algoritmi di grafica tridimensionale. Panoramica sui sistemi di acquisizione per dati tridimensionali.
4	Sistemi di ricostruzione a partire dalle proiezioni (tomografia). Algoritmi di retro-proiezione, ART e Kaczmarz. Tomografia discreta.
4	Rendering e raytracing.
2	Aberrazioni ottiche e loro riduzioni.
2	SIFT e Harris detector.
2	RANSAC e mosaicing.
4	Flusso ottico e registrazione di immagini. Super-risoluzione.
3	Stereoscopia e ricostruzione tridimensionale da viste multiple.
2	Ridimensionamento "intelligente" di immagini digitali.

	ESERCITAZIONI
	Le lezioni saranno integrate da esempi e/o esercitazioni al computer. Apposite dispense saranno distribuite durante il corso.
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. • R.Szeliski. Computer Vision: Algorithms and Applications. Springer 2010. • A.Fusiello. Visione Computazionale. Creative Commons, 2009.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010-2011
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	196
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	104
PROPEDEUTICITÀ	Analisi Matematica, Matematica Discreta, Programmazione
ANNO DI CORSO	II
SEDE DI SVOLGIMENTO DELLE LEZIONI	Da definire nel Calendario Didattico
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 Calendario Didattico pubblicato nel sito del CdL
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Martedì e Giovedì dalle 15.00 alle 17.00

<p>RISULTATI DI APPRENDIMENTO ATTESI</p> <p>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.</p> <p>Capacità di applicare conoscenza e comprensione Capacità di riconoscere, ed organizzare in autonomia argomenti base dell'informatica teorica.</p> <p>Autonomia di giudizio Essere in grado di valutare la rilevanza generale di argomenti della disciplina.</p> <p>Abilità comunicative Capacità di esporre le tematiche generali dell'informatica teorica anche a un pubblico non esperto..</p>
--

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	<i>Automi a Stati Finiti</i> 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. <u>Eliminazione delle ϵ-transizioni.</u>
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	<i>Grammatiche e Linguaggi Liberi dal Contesto (CF)</i> 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	Automati 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	<p><i>J. E. Hopcroft, R. Motwani, J. D. Ullman, Automati, Linguaggi e Calcolabilità, Addison-wesley (PearsonEducation Italia) 2003.</i></p> <p><i>R. McNaughton, Elementary Computability, Formal Languages and Automata, Prentice-Hall, 1982</i></p> <p><i>D. Perrin, Finite Automata, Capitolo 1 del Vol.2 del Handbook of Theoretical Computer Science, Elsevir, 1990.</i></p>

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	<p><i>Aspetti generali della nozione di calcolabilità</i></p> <p>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 del! "explicandum" <i>informale</i> di funzione calcolabile. Le funzioni ricorsive primitive. Costruzione della funzione di Ackermann. Definizione di funzione ϵ-ricorsiva e μ-ricorsiva.</p>
6 ore	<p>Le funzioni ricorsive primitive. Definizione e proprietà principali. Dimostrazione della ricorsività primitiva di varie funzioni elementari. Metodi di codifica ricorsivi primitivi. Numeri di Godel.</p> <p>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".</p>
5 ore	<p>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 del l' esistenza di insiemi ricorsivamente enumerabili</p>

	<p>ma non ricorsivi.</p> <p>Il teorema s-m-n (o del parametro) di Kleene. Alcune sue conseguenze.</p>
5 ore	<p>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 L_n. 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.</p>
5 ore	<p>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.</p> <p>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.</p>
5 ore	<p>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.</p> <p>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.</p> <p>Dimostrazione dell'equivalenza tra funzioni calcolabili in S e funzioni μ-ricorsive. Non ricorsiva enumerabilità dell'insieme di indici delle funzioni ricorsive totali.</p>
2 ore	<p>Cenni al decimo problema di Hilbert e agli insiemi diofantei. Il teorema di Matjasievic (senza dimostrazione)</p>
4 ore	<p>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).</p>
4 ore	<p>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>

	P=?NP. I sette <i>Problemi del Millennio</i> 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	<p><i>M. Davis, E. Weyuker, Computability, Complexity and Languages</i>, Academic Press (1983)</p> <p><i>G. S. Boolos, R. C. Jeffrey, Computability and Logic</i>, Cambridge University Press (1989)</p> <p>Si suggerisce, inoltre, la lettura di <i>E. Casari, Computabilità e ricorsività</i>, Quaderni della Scuola Superiore di Idrocarburi dell'ENI (1959)</p>

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Linguaggi di Programmazione
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline informatiche
CODICE INSEGNAMENTO	04758
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF01
DOCENTE RESPONSABILE	Laura Giambruno Assegnista 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	2°
SEDE DI SVOLGIMENTO DELLE LEZIONI	Laboratorio via Ingrassia Aula Ex-Consortio
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Martedì 8:30-9:30 Giovedì 11:30-13:30 Venerdì 10:30-12:30
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Martedì 15:00-17:00

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 Java.

Capacità di applicare conoscenza e comprensione

Capacità di valutare le funzionalità dei diversi paradigmi di programmazione. Capacità di scrivere un codice nel linguaggio di programmazione Java.

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.

MODULO	Linguaggi di Programmazione
ORE FRONTALI	LEZIONI FRONTALI
4	Concetti base sulle macchine astratte
6	Descrizione basilare dei linguaggi di programmazione
4	I nomi e l'ambiente
4	Gestione della memoria
6	Strutturare e astrarre sul controllo e sui dati
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
14	Linguaggio Java Esempi ed esercizi
TESTI CONSIGLIATI	M. Gabrielli, S. Martini <i>Linguaggi di programmazione, principi e paradigmi</i> Mc GrawHill Cay Horstmann <i>Concetti di informatica e fondamenti di Java</i> , Apogeo

FACOLTÀ	SCIENZE MM.FF.NN:
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	INFORMATICA
INSEGNAMENTO	MATEMATICA DISCRETA
TIPO DI ATTIVITÀ	Di base
AMBITO DISCIPLINARE	Formazione Matematico-Fisica
CODICE INSEGNAMENTO	10371
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	MAT/02
DOCENTE RESPONSABILE (MODULO 1)	FABIO DI FRANCO Professore Associato Università di Palermo
DOCENTE COINVOLTO (MODULO 2)	SILVANA MAUCERI Ricercatore Confermato 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	I
SEDE DI SVOLGIMENTO DELLE LEZIONI	Aula 4, Dipartimento di Matematica e Informatica, 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 e Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Martedì ore 10.30-12.30 Mercoledì ore 12.30-13.30 Venerdì ore 10.30-12.30
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Da concordare con i docenti (di franco@math.unipa.it; mauceri@math.unipa.it)

RISULTATI DI APPRENDIMENTO ATTESI

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 1 "FONDAMENTI DI MATEMATICA DISCRETA":

Acquisire una preparazione matematica di base fornendo agli studenti alcune nozioni relative agli insiemi discreti,

evidenziando anche le tecniche risolutive e dimostrative connesse con il loro studio.

MODULO 1	“FONDAMENTI DI MATEMATICA DISCRETA”
ORE FRONTALI	LEZIONI FRONTALI
20	Logica e insiemistica
10	Aritmetica degli interi
18	Teoria dei grafi
TESTI CONSIGLIATI	Facchini “ Algebra e Matematica Discreta ” Ed. Decibel-Zanichelli Piacentini Cattaneo “ Matematica Discreta ed Applicazioni ” Ed. Zanichelli

OBIETTIVI FORMATIVI DEL MODULO 2 “STRUTTURE COMBINATORIE”:
Utilizzare le strutture combinatorie per la risoluzione di problemi matematici. Acquisire le nozioni base dell'algebra astratta per la loro applicazione alla crittografia.

MODULO 2	STRUTTURE COMBINATORIE
ORE FRONTALI	LEZIONI FRONTALI
14	Calcolo combinatorio
12	Quadrati latini e Teoria dei disegni
10	Strutture algebriche
12	Crittografia
TESTI CONSIGLIATI	Facchini “ Algebra e Matematica Discreta ” Ed. Decibel-Zanichelli Piacentini Cattaneo “ Matematica Discreta ed Applicazioni ” Ed. Zanichelli

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	20010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Reti di Calcolatori
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline informatiche
CODICE INSEGNAMENTO	06232
ARTICOLAZIONE IN MODULI	NO
SETTORI SCIENTIFICO DISCIPLINARI	INF01
DOCENTE RESPONSABILE (MODULO 1)	Biagio Lenzitti Ricercatore 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	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	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare il sito web del CdL
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. Conoscenza del linguaggio php.

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. Capacità di realizzare applicazioni di rete in C e applicazioni web in php

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, Traceroute
1	I modelli di riferimento OSI e TCP/IP
4	Livello Applicazione Caratteristiche principali Programmare con le socket, Client/server TCP, Client/server UDP
2	Word Wide Web e HTTP
6	Programmare lato server con php
2	FTP , Ftp con telnet, Posta Elettronica, SMTP con telnet ,DNS, comando host
6	Il Livello di Trasporto UDP TCP Gestione della connessione Trasferimento dati affidabile Controllo della congestione , vista dei pacchetti con Wireshark
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
TESTI CONSIGLIATI	<i>Reti di Calcolatori e Internet</i> , 5a ed. , James F. Kurose e Keith W. Ross, Pearson, 2005 <i>Reti di Calcolatori</i> , 4a ed. , Andrew S. Tanenbaum, Pearson, 2003 <i>Internetworking con TCP/IP</i> , 5a ed. , Douglas E. Comer, Pearson, 2006 <i>PHP5 & MySQL LA Guida</i> Converse, Park, Morgan McGraw-Hill

FACOLTÀ	Scienze MM. FF. NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Laurea in Informatica
INSEGNAMENTO	Sistemi Operativi II
TIPO DI ATTIVITÀ	Caratterizzante
AMBITO DISCIPLINARE	Discipline Informatiche
CODICE INSEGNAMENTO	06510
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1)	Domenico Tegolo Professore Associato, Università di Palermo
DOCENTE COINVOLTO (MODULO 2)	Franco Bellavia Docente a contratto
CFU	9
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	153
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	72
PROPEDEUTICITÀ	Nessuna
ANNO DI CORSO	Terzo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Consultare sito web del CdL
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali ed Esercitazioni in laboratorio.
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Modulo 1-2: Prova in laboratorio e Orale.
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Primo e secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	5 ore settimanali primo semestre 5 ore settimanali secondo semestre
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare i docenti: Prof. D. Tegolo: domenico.tegolo@unipa.it Dott. Bellavia: fbellavia@unipa.it

RISULTATI DI APPRENDIMENTO ATTESI: Conoscenza e capacità di comprensione

- Conoscenza della gestione dei processi e dei thread (dalla creazione alla loro schedulazione);
- Acquisizione le problematiche relative al deadlock;
- Acquisizione delle metodologie di base per la gestione della memoria centrale;
- Individuazione delle caratteristiche per la gestione dell'I/O
- Conoscenza delle soluzioni adottate da un generico S.O. per la gestione del file system.

Capacità di applicare conoscenza e comprensione

- Capacità di applicare tali conoscenze a specifici problemi di laboratorio (Processi e Gestione della memoria).
- Capacità ad individuare la giusta metodologia per la soluzione di problematiche relativi alla gestione di più processi.

Autonomia di giudizio

- Essere in grado di valutare le qualità di base di un Sistema Operativo.

Abilità comunicative

- Capacità di applicare le metodologie apprese alle problematiche dei S.O., non necessariamente inerenti al settore scientifico della materia in oggetto.
- Abilità nell'integrare le capacità apprese nei due moduli.

<p>OBIETTIVI FORMATIVI DEL MODULO 1 “Principi Generali dei Sistemi Operativi”</p> <p>L’insegnamento si propone di fornire allo studente i concetti teorici che sono alla base degli attuali Sistemi Operativi. Inoltre, saranno date le caratteristiche di base dei sistemi di elaborazione con riferimento agli attuali sistemi operativi. Processi e thread: Introduzione ai processi, Thread, comunicazione tra processi, problemi di comunicazione tra processi, schedulazione tra processi. Deadlock: introduzione ai Deadlock, identificare e risolvere dei deadlock, Evitare i deadlock, prevenzione da Deadlock. Gestione della Memoria: sistemi di base per gestire della memoria, swapping, memoria Virtuale, algoritmi di riposizionamento delle pagine, segmentazione. Input/output: principi dell’hardware e del software, i livelli software dell’I/O, i Dischi, i Clock. I File System: i file, le directory, implementazione del File System.</p> <p>OBIETTIVI FORMATIVI DEL MODULO 2 “Modelli e Tecniche dei Sistemi Operativi”</p> <p>Obiettivo del modulo è fornire metodologie avanzate per affrontare attraverso metodi e tecniche dei S.O i problema presenti nella gestione dei processi e della memoria. Tali metodologie saranno applicate in esempi di problemi reali non necessariamente inerenti al settore scientifico della materia in oggetto ma utili per la formazione dello studente. Le esercitazioni comprendono l’implementazione di applicativi in linguaggio C.</p>
--

MODULO 1	Principi Generali dei Sistemi Operativi
ORE FRONTALI	LEZIONI FRONTALI
8	Introduzione ai sistemi di elaborazione. Introduzione ai sistemi operativi: che cosa è un S.O., cenni storici sui sistemi operativi, classificazione dei sistemi operativi, concetti base sui sistemi operativi, chiamate di sistema.
8	Processi e thread: Introduzione ai processi, Thread, comunicazione tra processi, problemi di comunicazione tra processi, schedulazione tra processi.
8	Deadlock: introduzione ai Deadlock, identificare e risolvere dei deadlock, Evitare i deadlock, prevenzione da Deadlock.
8	Gestione della Memoria: sistemi di base per gestire della memoria, swapping, memoria Virtuale, algoritmi di riposizionamento delle pagine, segmentazione.
8	Input/output: principi dell’hardware e del software, i livelli software dell’I/O, i Dischi, i Clock.
8	I File System: i file, le directory, implementazione del File System.

TESTI CONSIGLIATI	Andrew S. Tanenbaum, “I moderni Sistemi Operativi 3th Ed.”, Pearson-Prentice Hall.
--------------------------	--

MODULO 2	Modelli e Tecniche dei Sistemi Operativi
ORE FRONTALI	LEZIONI FRONTALI
2	Il linguaggio C
2	File System in C
2	Gestione Memoria
2	Processi in Windows
2	Thread in Windows
2	Sincronizzazione threads in Window

	ESERCITAZIONI
2	Sintassi, Strutture e Funzioni in C
2	Gestione della Memoria
2	Funzioni per il richiamo di programmi esterni in Windows

2	Sincronizzazione threads in windows, Mutex, Semafori
2	Gestione e Sincronizzazione dei processi in Unix/Linux
2	Creazione e Sincronizzazione dei Threads in Linux

TESTI CONSIGLIATI	H. M. Deitel, P. J. Deitel "C Corso completo di programmazione 3ed" Apogeo P. Ancilotti, M. Boari, A. Ciampolini, G. Lipari "Sistemi Operativi 2ed" McGraw-Hill
------------------------------	--

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Programmazione e Laboratorio
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)	Rosone Giovanna Docente a contratto
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 sito web del CdL
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare la docente: mari@math.unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

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.

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 (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 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 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	Libro di testo: A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill. Libro consigliato: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	PROGRAMMAZIONE E LABORATORIO (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.
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	Libro di testo: A. Bellini, A.Guidi. Linguaggio C - guida alla programmazione. Mc Graw Hill. Libro consigliato: B. W. Kernighan, D. M. Ritchie. Il linguaggio C - Principi di Programmazione e Manuale di riferimento. Pearson Education Italia.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Sistemi di Elaborazione
TIPO DI ATTIVITÀ	Attività caratterizzanti
AMBITO DISCIPLINARE	Discipline informatiche
CODICE INSEGNAMENTO	06457
ARTICOLAZIONE IN MODULI	SI
NUMERO MODULI	2
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE (MODULO 1) - Algoritmi Genetici	Millonzi Filippo Professore a Contratto
DOCENTE RESPONSABILE (MODULO 2) - Simulated annealing e Tabu search	Vaglica Roberto Professore a Contratto
CFU	12
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	204
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	96
PROPEDEUTICITÀ	Analisi Matematica ; Matematica Discreta ; Programmazione e Laboratorio; Algoritmi e Strutture dati ; Informatica Teorica
ANNO DI CORSO	Terzo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Laboratorio informatico - via Ingrassia, Palermo
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in laboratorio
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova di Laboratorio e Prova Orale
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Consultare sito web del CdL
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare sito web del CdL
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Su prenotazione via e-mail (vaglica@math.unipa.it; millonzi@unipa.it)

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Il corso fornisce le conoscenze teoriche e pratiche per ideare ed implementare algoritmi euristici di tipo evolutivo e algoritmi di ricerca locale (Tabu Search, Simulated Annealing, etc.)

per la risoluzione di problemi di ottimizzazione ad elevata complessità computazionale.

Inoltre fornisce le conoscenze necessarie per l'uso di base e avanzato di Matlab che sarà il linguaggio utilizzato per sviluppare tali algoritmi. Gli studenti acquisiranno la capacità di codificare opportunamente i dati di un problema in modo da poterlo risolvere all'interno di del frame evolutivo e affineranno le proprie capacità di problem solving.

Capacità di applicare conoscenza e comprensione

Il corso ha come obiettivo rendere lo studente capace di comprendere il funzionamento dei principali algoritmi euristici e metaeuristici, fornendogli non solo le basi teoriche che giustificano la

convergenza, ma anche gli strumenti pratici per la realizzazione di tali algoritmi mediante un linguaggio ormai divenuto standard in ambito scientifico.

Autonomia di giudizio

Gli studenti sono guidati ad apprendere in maniera critica e responsabile tutto ciò che viene spiegato in aula e ad arricchire le proprie capacità di giudizio sia attraverso lo studio del materiale didattico indicato o fornito dal docente, sia attraverso la realizzazione di elaborati pratici consistenti nell'ideazione, implementazione e validazione di euristiche.

Abilità comunicative

Attraverso l'interazione durante le lezioni e le attività di laboratorio previste, il corso tenderà a favorire lo sviluppo della capacità di comunicare in modo chiaro ed esaustivo le proprie ragioni ed argomentare le proprie conclusioni. Gli studenti dovranno altresì sviluppare la capacità di lavorare in gruppo, di confrontarsi sulle problematiche proposte rispettando i punti di vista diversi dal proprio e arricchendo le conoscenze acquisite durante il corso con la dialettica e il confronto tra pari.

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 MODULO I – Algoritmi Genetici

Il corso ha il duplice obiettivo di fornire agli studenti la conoscenza del linguaggio Matlab e di alcuni strumenti di calcolo non convenzionale per la risoluzione di problemi ad elevata complessità. Il linguaggio Matlab è, *de facto*, lo strumento di programmazione standard per applicazioni scientifiche in moltissimi ambiti di ricerca. A tal fine, il corso si propone di fornire gli strumenti e le metodologie necessarie per scrivere efficientemente un codice Matlab e sfruttare alcune delle capacità grafiche di visualizzazione e analisi dei dati. Il corso vuole dare agli studenti la possibilità di apprendere l'uso, di base e avanzato, del linguaggio, senza pretendere di coprire tutti gli aspetti di Matlab. Pertanto durante il corso non verranno trattati argomenti quali GUI, Object Oriented, Calcolo Parallelo o distribuito, Toolbox, etc.

L'obiettivo primario del corso resta comunque quello di fornire gli strumenti sia teorici che pratici per affrontare la risoluzione di problemi di minimizzazione *multi-obiettivo* ad elevata complessità di calcolo, mediante paradigmi computazionali basati sull'evoluzione di sistemi complessi. In particolare verranno affrontate le tecniche di *problem solving* basate sugli *algoritmi genetici*, seriali e paralleli, e sugli *algoritmi memetici*, con cenni alla *Ant Colony Optimization*.

L'ottimizzazione di un sistema spesso si riduce alla risoluzione efficiente di un problema di minimizzazione di una o più funzioni, dette funzioni obiettivo. Tuttavia per molti problemi di ottimizzazione questa funzione target si basa su un elevato numero di parametri, spesso in competizione tra loro. Pertanto la ricerca della soluzione ottimale, che potrebbe anche non esistere, risulta comunque, estremamente dispendiosa in termini di potenza e tempi di calcolo. Gli algoritmi evolutivi sono euristiche di ottimizzazione stocastiche, basati sulla ricerca parallela nello spazio delle soluzioni di un dato problema, capaci di determinare soluzioni *sub-ottimali* a problemi di ottimizzazione complessi (anche multi-obiettivo) in tempi ragionevoli e con basso errore rispetto alle soluzioni ottime degli stessi.

Al termine del corso lo studente dovrebbe essere in grado di scrivere codice Matlab efficiente e, nello stesso tempo, possedere gli strumenti necessari per generare un codice prototipale per il Calcolo Scientifico basato sui paradigmi evolutivi.

CORSO	SISTEMI DI ELABORAZIONE MOD. I – ALGORITMI GENETICI
ORE FRONTALI	LEZIONI FRONTALI
1	Introduzione Presentazione del corso e del docente.
16	Matlab Comandi base; variabili e matrici; tipi di dato; funzioni built-in; script e funzioni definite dall'utente; funzioni con input e output variabile; strutture di selezione; strutture iterative; funzioni grafiche; funzioni di esportazione di dati ed importazione di dati esterni; funzioni di analisi dei dati.
2	Ottimizzazione funzioni obiettivo; funzioni multi-obiettivo; minimizzazione delle funzioni obiettivo; spazio delle soluzioni; Pareto front; convergenza in probabilità.
15	Fondamenti teorici degli algoritmi evolutivi e genetici Algoritmi stocastici; algoritmi evolutivi; algoritmi genetici; ricerca nello spazio delle soluzioni; codifica dei dati; cromosomi e popolazione; operatore di crossover su: dati binari, dati reali, stringhe, permutazioni ed alberi; operatore di mutazione su: dati binari, dati reali, stringhe, permutazioni ed alberi; funzione di fitness; funzioni di selezione; modelli generazionali e modelli steady-state; struttura degli algoritmi genetici; complessità computazionale; probabilità di crossover; probabilità di mutazione; fitness landscape; teorema dello schema; algoritmi memetici; individui e popolazioni; operatori competitivi; operatori cooperativi; algoritmi evolutivi costruttivi; voting; Ant Colony Optimization; toy problem: TSP, SPP; algoritmi genetici paralleli; operatore di migrazione.
14	Implementazione di algoritmi evolutivi Design e implementazione con Matlab di algoritmi genetici per risolvere problemi di ottimizzazione anche multi-obiettivo; design e implementazione con Matlab di algoritmi memetici per risolvere problemi di ottimizzazione anche multi-obiettivo; risoluzione di esercizi.
TESTI CONSIGLIATI	Holly Moore, Matlab per l'ingegneria , Pearson Education Italia (2008) - ISBN: 978-8-871-92447-2 Daniel Ashlock - Evolutionary Computation for Modeling and Optimization , Springer (2006) - ISBN: 978-0-387-22196-0 Frank Neumann, Carsten Witt, Bioinspired Computation in Combinatorial Optimization , Springer (2010) - ISBN: 978-3-642-16543-6 Zbigniew Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs , Springer (1996) - ISBN: 978-3-540-60676-5

OBIETTIVI FORMATIVI DEL MODULO II -Simulated annealing e Tabu search

Il modulo ha come obiettivo principale quello di fornire agli studenti la conoscenza di alcune

metodologie per la risoluzione di problemi complessi, con particolare riguardo ad alcune tecniche non tradizionalmente riconducibili al settore del Soft Computing. In particolare, per quanto riguarda le tecniche di ottimizzazione, si intende approfondire lo studio di alcune euristiche di ricerca locale quali il Simulated Annealing ed il Tabu Search.

Al termine del corso lo studente dovrebbe essere in grado di ricorrere a moderne ed avanzate tecniche in grado di calcolare soluzioni approssimate o euristiche (sia da un punto di vista teorico che pratico mediante l'uso di Matlab) per problemi di ottimizzazione intrinsecamente difficili dal punto di vista computazionale, individuandone i metodi algoritmici più adeguati per la loro risoluzione.

CORSO	SISTEMI DI ELABORAZIONE MOD. I I – SIMULATED ANNEALING E TABU SEARCH
ORE FRONTALI	LEZIONI FRONTALI
2	Introduzione Presentazione del modulo: obiettivi e modalità
4	Algoritmi di ricerca locale e metaeuristici Classificazione e breve panoramica sugli algoritmi di ricerca locale e sui metaeuristici: Algoritmi di ricerca locale, Simulated Annealing , Algoritmi Genetici , Tabu Search, Ant Colony Optimization .
10	Spazi di Ricerca e Tecniche di Ottimizzazione Ricerca Locale (Local Search o Hill Climbing). Tabu search: definizioni generali e funzionamento. Simulated Annealing: definizioni generali e funzionamento, teoria del Simulated Annealing e suo comportamento asintotico. Esempi.
10	Tabu Search in Matlab Implementazione con Matlab di algoritmi di tabu search per il problema del commesso viaggiatore e per il problema delle n regine; processi di intensificazione e diversificazione per intorni di permutazioni; gestione automatica della dimensione della tabu list.
10	Simulated Annealing in Matlab Soluzione mediante algoritmo simulated annealing del problema del commesso viaggiatore, gestione automatica della velocità di raffreddamento del processo.
12	Implementazione di euristiche in Matlab Implementazione in Matlab di euristiche per la risoluzione dei seguenti problemi: il problema delle gemme, il problema degli idranti, il problema degli anelli, il problema del ricoprimento mediante croci, il problema del tautopangramma, il problema del Sudoku, il problema del convex hull, il problema dello schedatore ottimale, il problema dei dischi sovrapposti.
TESTI CONSIGLIATI	D.T. Pham and D. Karaboga, “Intelligent Optimisation Techniques”, Springer, 2000. Dispense a cura del docente.

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2010/2011
CORSO DI LAUREA	Informatica
INSEGNAMENTO	Linguaggi XML
TIPO DI ATTIVITÀ	Altre attività formative
AMBITO DISCIPLINARE	A scelta dello studente (consigliata)
CODICE INSEGNAMENTO	14056
ARTICOLAZIONE IN MODULI	NO
NUMERO MODULI	1
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Marco Elio Tabacchi Docente a Contratto
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	Secondo/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 Pratica
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultare sito web del CdL
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Contattare il docente: tabacchi@unipa.it

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

Il corso intende fornire agli studenti le nozioni necessarie per districarsi nel complicato labirinto dei linguaggi a marcatore. Gli studenti saranno in grado di scrivere, leggere ed interpretare codice XML, riconoscere i vari dialetti, preparare delle informazioni da fonti eterogenee in maniera che possano essere servite in maniera strutturata.

Capacità di applicare conoscenza e comprensione

Gli studenti dovranno essere in grado di progettare un sistema comunicativo basato su informazioni da fonti eterogenee, e di realizzarlo attraverso i più comuni strumenti di elaborazione dei linguaggi a marcatore.

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, anche se queste frasi hanno contenuto informativo prossimo allo

zero.

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 XML, tramite lo studio di come tali linguaggi possono essere utilizzati nella comunicazione di informazioni.

OBIETTIVI FORMATIVI DEL CORSO

Questo corso si pone obiettivi su due direttivi fondamentali: la prima è quella di suggerire agli studenti dei mezzi critici per la comprensione del problema della comunicazione delle informazioni da fonti eterogenee usando sistemi automatizzati e semi automatizzati. Gli studenti dovranno sviluppare l'autonomia di giudizio ed il grado di expertise necessari per il proficuo completamento del corso, e contemporaneamente essere in grado di procedere in autonomia. La seconda vuole fornire invece degli strumenti professionali completi e coerenti con le sfide che il mondo del lavoro nel settore ICT propone a coloro che vogliono entrarvi. Per questo motivo il corso sarà integralmente basato su case history reali, e le applicazioni prenderanno sempre in considerazione problemi, mezzi e strumenti disponibili ordinariamente al tipico profilo ICT.

CORSO	LINGUAGGI XML
ORE FRONTALI	LEZIONI FRONTALI ED ATTIVITÀ AL CALCOLATORE
48	XHTML, CSS e ECMAScript HTML XML, XHTML integrazione tra componenti di un sito dinamico; il protocollo HTTP. Layout - Tabelle Layout - CSS-P Client Side scripting PHP AJAX - integrare l'oggetto XMLHttpRequest configurazione e personalizzazione di un CMS Applicazioni Avanzate SOAP
TESTI CONSIGLIATI	Il corso farà uso dei manuali ufficiali delle singole applicazioni utilizzate, e dei testi disponibili gratuitamente in rete. Si farà particolare attenzione all'utilizzo di applicativi e tecnologie Open Source ed Open Access. Gli studenti rimarranno liberi di scegliere la suite applicativa, i pacchetti ed i manuali a loro più congeniali, sempre seguendo seguendo il programma didattico.