

FACOLTÀ	Scienze MM.FF.NN.
ANNO ACCADEMICO	2014/2015
CORSO DI LAUREA	INFORMATICA
INSEGNAMENTO	INFORMATICA TEORICA II modulo (teoria della calcolabilità)
TIPO DI ATTIVITÀ	
AMBITO DISCIPLINARE	
CODICE INSEGNAMENTO	03946
SETTORI SCIENTIFICO DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	Settimo Termini
CFU	6
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	102
NUMERO DI ORE RISERVATE ALLE ATTIVITÀ DIDATTICHE ASSISTITE	48
PROPEDEUTICITÀ	nessuna
ANNO DI CORSO	Secondo
SEDE DI SVOLGIMENTO DELLE LEZIONI	Comunicato successivamente
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali, Esercitazioni in aula
MODALITÀ DI FREQUENZA	Facoltativa
METODI DI VALUTAZIONE	Prova orale. Prova scritta. Invito agli studenti a presentare alcuni argomenti integrativi e complementari del programma sotto forma di seminari.
TIPO DI VALUTAZIONE	Voto in trentesimi
PERIODO DELLE LEZIONI	Secondo semestre
CALENDARIO DELLE ATTIVITÀ DIDATTICHE	Consultabile sul sito del corso di laurea:
ORARIO DI RICEVIMENTO DEGLI STUDENTI	Sarà comunicato all'inizio del corso

RISULTATI DI APPRENDIMENTO ATTESI

Conoscenza e capacità di comprensione

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

Capacità di applicare conoscenza e comprensione

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

Autonomia di giudizio

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

Abilità comunicative

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

Capacità d'apprendimento

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

OBIETTIVI FORMATIVI

L'obiettivo del corso è quello di fornire agli studenti gli elementi di base, concettuali e formali, della teoria della calcolabilità, mettendo in evidenza - da un lato - i rapporti esistenti tra alcuni risultati di carattere generale e alcuni problemi e domande che sorgono a partire da aspetti apparentemente solo "tecnici" della programmazione e - da un altro lato - come questa teoria, alla base dell'informatica sia nata a partire da questioni e problemi di stretto interesse teorico.

	TEORIA DELLA CALCOLABILITA'
ORE FRONTALI	LEZIONI FRONTALI
48	
6 ore	<p><i>Aspetti generali della nozione di calcolabilità.</i> Caratteristiche dell'informatica teorica. 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 e-ricorsiva e μ- ricorsiva.</p>
12 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. 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>
6 ore	<p>Il teorema "contapassi" e sua relazione col teorema della fermata. Definizione di insieme ricorsivamente enumerabile (r.e.) e di insieme ricorsivo. Teoremi sulle relazioni intercorrenti tra insiemi ricorsivamente enumerabili e insiemi ricorsivi. Il teorema di Post. Dimostrazione dell'esistenza di insiemi ricorsivamente enumerabili ma non ricorsivi. Il teorema s-m-n (o del parametro) di Kleene. Alcune sue conseguenze.</p>
4 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. 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>
6 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> <p>Cenni agli automi finiti e alla gerarchia di Chomsky</p>
6 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 nidificazioni dei cicli LOOP. La gerarchia L. 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>
8 ore	<p>Alcuni degli argomenti che seguono, a seconda della situazione del momento e degli interessi degli studenti:</p> <p>Una breve introduzione al Lisp. Cenni al X problema di Hilbert. I sette Problemi del Millennio come riproposizione dei problemi di Hilbert al Convegno del 1900. Aspetti della Complessità. La complessità astratta di Blum. La complessità visuale. Il lambda calcolo e suoi rapporti col linguaggio Lisp. Altri modelli di computo. Il gioco della vita di Conway. Cenni alla logica fuzzy.</p>

TESTI CONSIGLIATI	<p>- <i>Martin Davis, E. Weyuker, Computability, Complexity and Languages, Academic Press (1983).</i></p> <p>- <i>George. S. Boolos, Richard C. Jeffrey, Computability and Logic, Cambridge University Press (1989).</i></p> <p>- <i>Ettore Casari, Computabilità e ricorsività, Quaderni della Scuola Superiore di Idrocarburi dell'ENI (1959).</i></p> <p>- <i>Martin Davis, Hilbert's Tenth Problem is Unsolvable The American Mathematical Monthly, Vol. 80, No. 3 (Mar., 1973), pp. 233-269</i></p>
--------------------------	--