



# UNIVERSITÀ DEGLI STUDI DI PALERMO

<b>DEPARTMENT</b>	Matematica e Informatica		
<b>ACADEMIC YEAR</b>	2015/2016		
<b>FIRST CYCLE COURSE</b>	COMPUTER SCIENCE		
<b>SUBJECT</b>	THEORETICAL COMPUTER SCIENCE		
<b>TYPE OF EDUCATIONAL ACTIVITY</b>	B		
<b>AMBIT</b>	50166-Discipline Informatiche		
<b>CODE</b>	16671		
<b>SCIENTIFIC SECTOR(S)</b>	INF/01		
<b>HEAD PROFESSOR(S)</b>	CASTIGLIONE GIUSEPPA	Ricercatore	Univ. di PALERMO
<b>OTHER PROFESSOR(S)</b>			
<b>CREDITS</b>	9		
<b>INDIVIDUAL STUDY (Hrs)</b>	153		
<b>COURSE ACTIVITY (Hrs)</b>	72		
<b>PROPAEDEUTICAL SUBJECTS</b>	05880 - PROGRAMMING AND LABORATORY - INTEGRATED COURSE 16448 - MATHEMATICAL METHODS FOR COMPUTER SCIENCE		
<b>MUTUALIZATION</b>			
<b>YEAR</b>	2		
<b>TERM (SEMESTER)</b>	1° semester		
<b>ATTENDANCE</b>	Not mandatory		
<b>EVALUATION</b>	Out of 30		
<b>TEACHER OFFICE HOURS</b>	<b>CASTIGLIONE GIUSEPPA</b> Tuesday 14:00 15:00 Dipartimento di Matematica e Informatica. Stanza 209 secondo piano. Thursday 14:00 15:00 Dipartimento di Matematica e Informatica. Stanza 209 secondo piano.		

DOCENTE: Prof.ssa GIUSEPPA CASTIGLIONE

<b>PREREQUISITES</b>	
<b>LEARNING OUTCOMES</b>	<p>Conoscenza e capacità di comprensione</p> <p>Acquisizione dei concetti fondamentali della Teoria degli Automi, dei Linguaggi Formali e della Teoria della calcolabilità con particolare riferimento ai principali modelli matematici utilizzati: automi a stati finiti, espressioni regolari, grammatiche. Sviluppo delle capacità di formalizzazione, astrazione, modellazione di sistemi e analisi di problemi complessi. Conoscere l'esistenza di problemi non risolubili o di problemi "difficilmente" risolubili, in termini di risorse di calcolo, e quindi la necessità di una loro classificazione in classi di complessità. Acquisizione degli strumenti per leggere e comprendere 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</p> <p>Capacità di utilizzare le conoscenze acquisite alla costruzione di automi e grammatiche in campi applicativi specifici, con particolare riferimento all'elaborazione di testi e alla descrizione astratta di un algoritmo. Comprendere la chiara distinzione tra aspetti sintattici e semantici</p> <p>Autonomia di giudizio</p> <p>Essere in grado di valutare la rilevanza degli argomenti della disciplina, e di contestualizzare gli aspetti teorici della teoria degli automi, dei linguaggi formali e della teoria della calcolabilità nei diversi ambiti applicativi.</p> <p>Abilità comunicative</p> <p>Capacità di esporre in modo matematicamente chiaro e rigoroso le tematiche trattate della teoria degli automi, dei linguaggi formali e della teoria della calcolabilità anche a un pubblico non esperto, mostrando come metodi e risultati matematici si rapportano a ambiti applicativi specifici.</p> <p>Capacità d'apprendimento</p> <p>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.</p>
<b>ASSESSMENT METHODS</b>	Prova Pratica/Scritta, Prova Orale.
<b>EDUCATIONAL OBJECTIVES</b>	<p>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.</p>
<b>TEACHING METHODS</b>	Lezioni frontali, Esercitazioni in aula e in laboratorio
<b>SUGGESTED BIBLIOGRAPHY</b>	<p>J. E. Hopcroft, R. Motwani, J. D. Ullman, Automi, Linguaggi e Calcolabilità, Addison-wesley (PearsonEducation Italia) III edizione 2009.</p> <p>R. McNaughton, Elementary Computability, Formal Languages and Automata, Prentice-Hall, 1082</p> <p>D. Perrin, Finite Automata, Capitolo 1 del Vol.2 del Handbook of Theoretical Computer Science, Elsevir, 1990.</p>

## SYLLABUS

Hrs	Frontal teaching
8	<p>Automi a Stati Finiti</p> <p>Motivazioni, applicazioni e descrizione informale. I concetti centrali della teoria degli automi. Definizione di automa a stati finiti deterministico (DFA). Automi riconoscitori. Rappresentazione di un DFA con grafo degli stati e tabella delle transizioni. 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 <math>\epsilon</math>-transizioni. Eliminazione delle <math>\epsilon</math>-transizioni.</p>

## SYLLABUS

Hrs	Frontal teaching
6	Espressioni regolari. Linguaggi regolari. Applicazioni di espressioni 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 regolare. Algoritmo di Berry e Sethi per convertire un'espressione in un automa.
6	Chiusura dei linguaggi regolari rispetto alle operazioni booleane e reverse. Il "pumping lemma" per i linguaggi regolari. Applicazioni del pumping lemma. Problema di decisione se un linguaggio regolare è vuoto. Problema di inclusione dei linguaggi regolari.
12	Equivalenza tra automi. Problema di decisione dell'equivalenza di due DFA. Minimizzazione di automi deterministici tramite gli algoritmi classici di minimizzazione. La relazione di indistinguibilità degli stati. Automa ridotto. Equivalenza tra automa ridotto e automa minimale. Teorema di Myhill-Nerode. Unicità dell'automa deterministico minimale.
10	Grammatiche e Linguaggi Liberi dal Contesto (CF) Motivazioni e descrizione informale. Definizione di grammatica. Derivazioni delle grammatiche. Linguaggio generato da una grammatica. La gerarchia di Chomsky. Le grammatiche e i linguaggi CF. Alberi sintattici. Ambiguità nelle grammatiche e nei linguaggi CF: grammatiche ambigue, eliminazione delle ambiguità, ambiguità inerente. Alcune applicazioni delle grammatiche libere dal contesto.
6	Forme normali. Forma normale di Chomsky. Pumping lemma per i linguaggi CF. Applicazioni del pumping lemma. Proprietà di chiusura dei linguaggi CF. Problemi di decisione per i linguaggi CF
8	Breve introduzione alla teoria della calcolabilità. La macchina di Turing. Funzioni calcolate da una macchina di Turing. Linguaggi riconosciuti da una macchina di Turing. La tesi di Turing-Church. La macchina universale di Turing. Esistenza di funzioni non calcolabili. Il problema della "fermata" di una macchina di Turing. Problemi decidibili e indecidibili. Problemi intrattabili. Modelli particolari di macchine di Turing. Gerarchia di Chomsky e decidibilità.
16	Implementazione di automi riconoscitori in C tramite matrice delle transizioni. Svolgimento di esercizi sugli algoritmi presentati durante il corso. Uso di strumenti software per la manipolazione di automi e linguaggi formali.