UNIVERSITÀ DEGLI STUDI DI PALERMO

DEPARTMENT	Matematica e Informatica
ACADEMIC YEAR	2021/2022
BACHELOR'S DEGREE (BSC)	COMPUTER SCIENCE
SUBJECT	THEORETICAL COMPUTER SCIENCE
TYPE OF EDUCATIONAL ACTIVITY	B
AMBIT	50166-Discipline Informatiche
CODE	16671
SCIENTIFIC SECTOR(S)	INF/01
HEAD PROFESSOR(S)	CASTIGLIONE GIUSEPPA Ricercatore Univ. di PALERMO
OTHER PROFESSOR(S)	
CREDITS	9
INDIVIDUAL STUDY (Hrs)	153
COURSE ACTIVITY (Hrs)	72
PROPAEDEUTICAL SUBJECTS	05880 - PROGRAMMING AND LABORATORY - INTEGRATED COURSE
MUTUALIZATION	
YEAR	2
TERM (SEMESTER)	1° semester
ATTENDANCE	Not mandatory
EVALUATION	Out of 30
TEACHER OFFICE HOURS	**CASTIGLIONE GIUSEPPA** 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

PREREQUISITES	Basic notions of mathematical logic, discrete mathematics, set theory.
LEARNING OUTCOMES	Knowledge and understanding To learn the main concepts of the Theory of Automata, Formal Languages and Theory of computability with particular attention to their mathematical models: finite state automata, regular expressions and grammars. Ability of formalization, abstraction, systems modeling and analysis of complex problems. To understand the clear distinction between syntactic and semantic aspects. To know the existence of problems not solvable or "difficult" solvable problems, in terms of computational resources, and therefore their classification into classes of complexity. Applying knowledge and understanding Ability to apply the acquired knowledge to the construction of automata and grammars in specific application fields. Design a deterministic finite state machine that accepts a specified language for an engineering firm and generates a regular expression to represent the language. To be able to read and understand the basic aspects of the specialist literature. To use the technical language of the discipline. Making judgments To be able to assess the relevance of the topics of the discipline, and to contextualize the theoretical aspects of the theory of automata, formal languages and theory of computability in various application areas. Communication To use clear and mathematically rigorous exposition of the issues of the automata theory, formal languages and theory of computability even to a non-expert audience, showing how mathematical methods and results relate to specific application environments. Apply rules of inference to construct proofs and present results to a group of professionals, appropriate proofs, or logical reasoning to solve a strategic problem. Lifelong learning skills Ability to upgrade with the consultation of its scientific literature. To be able to follow, using the knowledge acquired in the course, master's degrees, second level masters and specialized seminars.
ASSESSMENT METHODS	An intermediate written test (optional for the students) a final written test and an oral exam. The intermediate test lasts 1 and half hours and consists of three exercises regarding automata and regular expressions. Each exercise will have a rating from 0/30 to 5/30 and is passed with a total score of at least 9/30.The final written test lasts 2 hours and consists of six exercises (three in the case of passed intermediate test). Each exercise will have a rating from 0/30 to 5/30 that will be added to the score of the intermediate test. The test is passed with a total score of at least 18/30. The oral exam consists of three or more questions on topics of the course with reference to the recommended book and materials provided during the lectures (slides, handouts, exercises with solutions). The test has the aim to evaluate the acquired knowledge, understanding of the topics, and the acquisition of specific language. The minimum rating will be achieved if the written test reachs the rate 18/30 and the student will know the basic concepts and will be able to expose them. Below this threshold, the examination will be insufficient. The valuation rates are proportional to language skills, correct and detailed exposition of the topics and the ability to apply and connect the concepts. In particular, there are four evaluation levels of the oral test: 18-21/30 the student outlines the essential aspects of arguments with simple language, solves simple problems only with support, contextualises the basic concepts 22-24/30 the student exposes all aspects of the topics with proper language, solves simple problems independently, contextualises only with support. 25-27/30 the student exposes all aspects of the topics with adequate language, is able to apply the concepts to all the proposed problems, contextualises only with support. 28-30/30 the student exposes all aspects of the topics with proper language, is able to apply the concepts to all the proposed problems, contextually autonomously. The laud will be given to students of the leve L 28-30 that are able to prove the theorems proposed. The assessment is carried out of thirty and will be the average of the evaluations of the two tests (intermidiate + final and oral exam).
EDUCATIONAL OBJECTIVES	To know the computational power of finite state automata and the generative power of context-free grammars. Relations between deterministic and non-deterministic models. Ability to convert a formalism to another equivalent: for example, grammars and automata, automata and regular expressions, deterministic And non-deterministic automata. To be able to design automata that recognize fixed languages. To know how to design grammars that generate

	fixed languages. To know how to use automata and grammars in designing algorithms. Learn the use of automata and grammars as a model in several important applications for example, compilers, software design, digital circuits, software for large collections of texts and for industrial applications.
TEACHING METHODS	Lectures
SUGGESTED BIBLIOGRAPHY	J. E. Hopcroft, R. Motwani, J. D. Ullman, Automi, Linguaggi e Calcolabilita, Addison-wesley (PearsonEducation Italia) III edizione 2009.

SYLLABUS

Hrs	Frontal teaching
6	Finite State Automata Motivations, applications and informal description. The central concepts of theory of automata. Definition of deterministic finite state automaton (DFA). Automata recognizers. Representation of a DFA graph of states and transitions table. Non-deterministic finite state automata (NFA). Equivalence of DFA and NFA. The "subset construction". Discussion on "state complexity" of DFA and NFA. Applications to text searches. Automata with epsilon-transitions. Elimination of epsilon-transitions.
6	Regular expressions of regular languages. Equivalence between regular languages and languages recognized by DFA (Kleene's Theorem). Algorithm of elimination of states to convert an automaton in a regular expression. Berry and Sethi algorithm to convert an expression to an automaton.
6	Closure of regular languages with respect to reverse, and Boolean operations. The "pumping lemma" for regular languages. Applications of the pumping lemma. Decision problems for regular languages: equivalence, emptyness and inclusion.
6	Equivalence of automata. Decision problem of the equivalence of two DFA. Minimization of deterministic automata using classical minimization algorithms. The relation of indistinguishability of states. Reduced automaton. Equivalence between reduced automaton and minimal automaton. Myhill-Nerode Theorem. Uniqueness of the minimal deterministic automaton.
6	Grammars and context-free languages (CF) Motivations and informal description. Definition of grammar. Derivations of grammars. Language generated by a grammar. The Chomsky hierarchy. Grammars and CF-languages. Parse trees. Ambiguity in grammars and languages in CF: ambiguous grammars, elimination of ambiguity, inherent ambiguity. Some applications of context-free grammars.
6	Push down automata. Accepted language. Equivalence between push down automata and context-free grammars.
6	Normal forms. Chomsky normal form. Pumping lemma for CF. Applications of the pumping lemma. Closure properties of the CF -languages. Decision problems for the CF languages.
6	Brief introduction to the theory of computability. The Turing machine. Functions computed by a Turing machine. Languages recognized by a Turing machine. The Church-Turing thesis.
8	The universal Turing machine. Existence of non-computable functions. The halting problem for a Turing machine. Decidable and undecidable problems. Intractable problems. P vs. NP. NP-complete problems. Special models of Turing machines. Chomsky hierarchy and decidability.
4	Software tools for the manipulation of automata. Determination, minimization of automata with Jflap.
4	Software tools for the manipulation of regular expressions.
4	Software tools for the manipulation of grammars. Derivation trees with Jflap.
4	Pumping lemma for context-free languages with Jflap.