



### Programme of Course "Teoria dei Linguaggi"

- Code: DT0269
- Type of course unit: Elective (Bachelor Degree in Computer Science curriculum General), Elective (Master Degree in Computer Science curriculum NEDAS), Elective (Master Degree in Computer Science curriculum SEAS), Elective (Master Degree in Computer Science curriculum UBIDIS)
- Level of course unit: Undergraduate Degrees, Postgraduate Degrees
- Semester: 2

Number of ects credits: (Master Degree in Computer Science) 6 (workload 150 hours), (Bachelor Degree in Computer Science) 6 (workload 150 hours)

Teachers: Sergio Orefice (Sergio.Orefice@univaq.it)

1	<b>Course objectives</b>	The course offers students an in-depth overview of formal language theory
2	<b>Course content and learning outcomes (dublin descriptors)</b>	<p>Topics of the module include:</p> <ul style="list-style-type: none"> <li>• - Introduction to language theory: Formal language basic notions. Chomsky hierarchy.</li> <li>• - Regular languages: Formalisms (DFA, NFA, epsilon-NFA, regular expressions). Equivalence results (subset construction, Kleene theorem, Thompson algorithm). Pumping Lemma. Closure properties. Decision properties (testing emptiness, testing membership, testing equivalence).</li> <li>• - Context-free languages: Formalisms (context-free grammars, PDA, DPDA). Chomsky Normal Form. Pumping Lemma. Closure properties. Decision properties (testing membership: CYK algorithm).</li> </ul> <p>On successful completion of this module, the student should :</p> <ul style="list-style-type: none"> <li>• have knowledge of the main formalisms of automata and formal language theory</li> <li>• understand and apply properties of regular and context free languages</li> <li>• demonstrate skill in formal reasoning and ability to conceive a proof</li> <li>• understand and explain language specifications by using formal notations (regular expressions, automata, grammars)</li> <li>• demonstrate capacity for reading and understanding other texts on related topics</li> </ul>
3	<b>Course prerequisites</b>	Knowledge of basic courses of programming, familiarity with common data structures, recursion and basic concepts of compilers
4	<b>Teaching methods and language</b>	<p>Lectures and exercises</p> <p><b>Language:</b> Italian</p> <p><b>Reference textbooks</b></p> <ul style="list-style-type: none"> <li>• Hopcroft, Motwani, Ullman, <i>Automi, Linguaggi e Calcolabilità</i>. Addison-Wesley.</li> </ul>
5	<b>Assessment methods</b>	<p>Formative assessment: the formative assessment is performed via interaction between teacher and students during lectures. The students are encouraged to actively participate to the lectures by making questions and discussing the theoretical concepts presented and the solutions adopted in the developed examples. Summative assessment: written test on the subjects treated in the course. An optional mid-term written test will also be provided, which is meant to cover the first part of the course, in order to help the students to split the workload. The written test (lasting 2 hours) consists of a set of questions for the verification of theoretical/formal competences and for the verification of skills in understanding and solving significant exercises. Criteria of evaluation will be: the level of knowledge of the notions and formalisms presented in the course, as well as the ability to apply them; the clarity and completeness of explanations.</p>