COURSE DESCRIPTION – ACADEMIC YEAR 2017/2018

Course title	**Theory of Computing**
Course code	72001
Scientific sector	ING-INF/05
Degree	Master in Computer Science (LM-18)
Semester	1
Year	1
Credits	8
Modular	No

Total lecturing hours	48
Total lab hours	--
Total exercise hours	24
Attendance	Not compulsory
Prerequisites	There are no prerequisites in terms of courses to attend. Students should be familiar with notions of mathematics and set theory, and with basic proof techniques, as taught in the mathematics courses of a bachelor in computer science.
Course page	http://www.inf.unibz.it/~calvanese/teaching/tc/

Specific educational objectives	The course belongs to the type "caratterizzanti – discipline informatiche" in the curriculum "Data and Knowledge Engineering". The objective of the Theory of Computing course is to introduce and study abstract, mathematical models of computation (such as Turing machines, formal grammars, recursive functions), and to use the abstract computation models to study the ability to solve computational problems, by identifying both the intrinsic limitations of computing devices, and the practical limitations due to limited availability of resources (time and space). A second objective is to show how to reason and prove properties about computations in a precise, formal, abstract way.

Lecturer	Diego Calvanese
Contact	Piazza Domenicani 3, Room 2.07, calvanese@inf.unibz.it, 0471-016160
Scientific sector of lecturer	ING-INF/05
Teaching language	English
Office hours	By prior arrangement via e-mail.
Lecturing Assistant (if any)	--
Contact LA	--
Office hours LA	--
List of topics	• Formal languages • Formal grammars • Turing Machines • Recursive functions • Undecidability • Computational complexity • NP-completeness • Time and space complexity classes

Teaching format	Frontal lectures, exercises.

Learning outcomes	Knowledge and understanding: • Thoroughly understand the scientific method of investigation. • Understand the methods of mathematics and statistics which are of support to information technology and its applications. Applying knowledge and understanding: • Be able to extend or modify a formal calculation model in an original way, taking into account altered conditions or requirements. • Be able to define an algorithmic solution to a computational problem and to estimate its complexity. Making judgments • Be able to identify reasonable work goals and estimate the resources required to achieve the objectives. Communication skills • Be fluent, in written and oral form, in at least one European language other than English, with reference also to the specific specialized vocabulary. • Be able to present in a fixed time the content of a scientific / technical report in front of an audience also composed of non-specialists. Learning skills • Be able to autonomously extend the knowledge acquired during the study course by reading and understanding scientific and technical documentation mostly in English. • Be able, in the context of a problem-solving activity, to extend even incomplete knowledge taking into account the objective of the project.

Assessment	The assessment of the course consists of two parts: • midterm or final exam on the first half of the syllabus (50%); • final exam on the second half of the syllabus (50%). Each part of the examination may be either written or oral.
Assessment language	English
Evaluation criteria and criteria for awarding marks	At the exam, the student has to solve exercises and answer questions on the course topics in written or oral form. The solution of the exercises requires on the one hand knowledge and understanding of the notions and techniques studied in the course, and on the other hand the ability to apply these to novel contexts. The two parts of the examination can be taken independently of each other within the three exam sessions of an academic year. In case of a positive mark for one of the two parts (obtained at the midterm or at one of the first two regular exam sessions), that part will count for all 3 regular exam sessions.

Required readings	• Introduction to Automata Theory, Languages, and Computation (3rd edition). J.E. Hopcroft, R. Motwani, J.D. Ullman. Addison Wesley, 2007. Unibz Library location: ST 130 H791(3.07)

	• Languages and Machines (3rd edition). Thomas A. Sudkamp. Addison Wesley, 2005. Only Chapter 13 (available on the course webpage).
Supplementary readings	• Elements of the Theory of Computation (2nd edition). H.R Lewis, C.H. Papadimitriou. Prentice Hall. 1998. • Introduction to the Theory of Computation. M. Sipser. PWS Publishing Company. 1997. • Complexity Theory. Ingo Wegener. Springer, 2005. • Computational Complexity. C.H. Papadimitriou. Addison Wesley. 1995.
Software used	--