**COURSE PRESENTATION FORM**

| | |
|---|---|
| **COURSE NAME** | Theory of Computing |
| **COURSE CODE** | 72001 (MSc New – DM 270) / 70101 (MSc Old – DM 509) |
| **LECTURER** | Diego Calvanese |
| **TEACHING ASSISTANT** | To be determined |
| **TEACHING LANGUAGE** | English |
| **CREDIT POINTS** | 8 |
| **LECTURE HOURS** | 48 |
| **EXERCISE HOURS** | 24 |
| **TIMESPAN** | 28/9/2009 – 23/1/2010 |
| **TIMETABLE** | See http://www.unibz.it/en/inf/progs/mcs/timetable/ |
| **OFFICE HOURS LECTURER** | See http://www.inf.unibz.it/~calvanese/teaching/ |
| **OFFICE HOURS TEACHING ASSISTANT** | Time to be determined<br>Via Sernesi 1, office C 5.01 |
| **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. |
| **OBJECTIVES** | 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. |
| **SYLLABUS** | Formal languages, formal grammars, Turing Machines, recursive functions, undecidability, computational complexity, NP-completeness, time and space complexity classes, non-uniform computing models. |
| **TEACHING FORMAT** | Frontal lectures; exercises in class. |
| **ASSESSMENT** | Midterm or final examination on the first half of the syllabus (50%) + final examination on the second half of the syllabus (50%). The two parts of the |

examination can be taken independently of each other within the three exam sessions of an academic year. Each part of the examination may be either written or oral.

**READING LIST**

Textbooks:

- *Introduction to Automata Theory, Languages, and Computation (3rd edition)*. J.E. Hopcroft, R. Motwani, J.D. Ullman. Addison Wesley, 2007.
- *Languages and Machines (3rd edition)*. Thomas A. Sudkamp. Addison Wesley, 2005. Only Chapter 13.
- *Complexity Theory.* Ingo Wegener. Springer, 2005. Only Chapter 14.

Further reading material:

- *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.
- *Computational Complexity*. C.H. Papadimitriou. Addison Wesley. 1995.

**SOFTWARE USED**

None

**LEARNING OUTCOME**

After the course, students will know the fundamental models of computation, and the intrinsic and practical limitations of computing devices. They will also be familiar with formal techniques of computer science, and will be able to formally prove properties about computations.

**COURSE PAGE**

http://www.inf.unibz.it/~calvanese/teaching/tc/