



Programme of Module "Formal Methods"

- Code: DT0202
- Type of course unit:
- Level of course unit:
- Semester: 2

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

Teachers: Monica Nesi (monica.nesi@univaq.it)

1	Course objectives	The goal of this course is to introduce symbolic techniques for the specification and verification of systems properties based on equational reasoning and theorem proving. On successful completion of this course, the student should understand the basic notions of first-order rewriting and logic, and be able to reason on properties of terms by means of symbolic manipulation modulo an equational theory or the deduction rules of a given logic.
2	Course content and learning outcomes (dublin descriptors)	<p>Topics of the module include:</p> <ul style="list-style-type: none"> • Abstract reduction systems, normal form, convertibility. Confluence, Church-Rosser property and their equivalence. Local confluence, termination, canonicity. Principle of Noetherian Induction and Newman's lemma (proof included). • First-order terms, substitutions, match and mgu. Algorithm of syntactic unification. Equational theories and equational deduction. Term rewriting systems. Termination: reduction ordering, simplification ordering, recursive path ordering. • Confluence: overlapping of rewrite rules, critical pairs, Huet's lemma (proof included). The word problem, completion procedures, divergence of completion. E-unification of terms, narrowing relation, E-unification procedure based on normalized and basic narrowing. • Boolean formulae, satisfiability, tautology. Formulae in CNF and Davis-Putnam's algorithm. Natural deduction. Predicate logic: predicates, functions, variables, quantifiers, rules of natural deduction. Prenex DNF. • Introduction to higher-order logics and lambda-calculus. Untyped lambda-calculus, beta-reduction, simple type theory, type assignment calculus, polymorphism. <p>On successful completion of this module, the student should :</p> <ul style="list-style-type: none"> • have profound knowledge of the basic concepts of first-order rewriting and first-order logic, relate the termination and confluence properties, have knowledge and understanding of pattern matching, syntactic and semantic unification, have knowledge and understanding of the natural deduction rules for propositional logic and predicate calculus, understand lambda-calculus as the base of the syntax for higher-order logic and functional programming; • understand and apply definitions, inference rules and theorems; • analyse and discuss different variants of a concept, discuss different proof techniques for deriving properties of terms and formulae; • explain and illustrate the fundamental notions of unification of terms, reduction ordering and critical pairs, explain the word problem and the completion of equational theories; • demonstrate skill in equational reasoning, formal derivation and symbolic manipulation, demonstrate ability to derive types for higher-order terms and properties of terms and formulae, demonstrate capacity for building proofs.
3	Course prerequisites	Basic notions of mathematical logic and functional programming are helpful.
4	Teaching methods and language	<p>Lectures and exercises Language: English Reference textbooks</p> <ul style="list-style-type: none"> • L. Thery, Lectures notes. http://www-sop.inria.fr/marelle/Laurent.Thery/formal/ • J.-G. Smaus, Pearls of Computer-Supported Modeling and Reasoning - Lecture

		<p>in l'Aquila. http://www.informatik.uni-freiburg.de/~ki/teaching/ws0910/csmr/aquila.html</p> <ul style="list-style-type: none">• M. Nesi e M. Venturini Zilli, Sistemi di riduzione astratti. Research Report SI-98/06. Facoltà di Scienze MM.FF.NN., Università degli Studi di Roma La Sapienza. 1998. http://www.di.univaq.it/monica/MFI/NoteARS.pdf• P. Inverardi, M. Nesi e M. Venturini Zilli, Sistemi di Riscrittura per Termini del Prim'Ordine. Dipartimento di Matematica Pura e Applicata, Università degli Studi di L'Aquila. 1999. http://www.di.univaq.it/monica/MFI/NoteSRT.pdf
5	Assessment methods	Written and oral examination. The written exam can be split into a midterm exam + a final exam at the end of the course.