MODELLO D (inglese)	
General Information	Formal Methods in Computer Science
Academic subject	Theoretical Computer Science
Degree course	Computer Science
Curriculum	6
ECTS credits	
Compulsory attendance	No
Language	English

Subject teacher	Name Surname	Mail address	SSD
	Giovanni Pani	**giovanni.pani@uniba.it**	Inf01

ECTS credits details			
Basic teaching activities	Lectures	videos	laboratory

Class schedule	
Period	First term
Year	First
Type of class	Lecture, laboratory, videos.

Time management	
Hours	150
Hours of lectures	32
Tutorials and lab	30

Academic calendar	
Class begins	25th September 2019
Class ends	12th January 2020

Syllabus	
Prerequisites/requirements	Programming languages Computability and Complexity
Expected learning outcomes (according to Dublin Descriptors) (it is recommended that they are congruent with the learning outcomes contained in A4a, A4b, A4c tables of the SUA-CdS)	*Syntax and semantics of programming languages.* *Applying knowledge to define programming languages and to define interpreter using functional programming.* *Choice programming languages.* *Design and develop software, also suggesting to evaluating alternative solutions and define most appropriate programming languages.* *Capacities to continue learning.*
Contents	Formal Semantics of Programming languages. Functional Programming.
Course program	Logical notations. Sets. Relations and functions. Introduction to operational semantics: the evaluation of arithmetic and Boolean expressions; the execution of commands. Mathematical induction, Structural induction, well founded induction. Inductive definitions. The denotational semantics. Introduction to domains theory. Recursion equations. Operations and denotational semantics for call by name and

	call by value. Languages with higher types. Eager operational and denotational semantics. Lazy operational and denotational semantics. Full abstraction. Recursive types. Laboratory. Haskell, The hugs system. Types and classes. Functions. List comprehensions. Recursive functions. Higher order functions. Functional parser. Interpreter.
Bibliography	G. Winskel , The formal semantics of programming languages, Mit press. G. Hutton, Programming in Haskell, Cambridge University Press.
Notes	
Teaching methods	Lectures, videos, laboratory
Assessment methods (indicate at least the type written, oral, other)	Laboratory. Theoretical part exam either written or oral
Evaluation criteria (Explain for each expected learning outcome what a student has to know, or is able to do, and how many levels of achievement there are.	Marks. Laboratory minimum 6 max 10 plus theoretical part minimum 12 max 20.
Further information	