SUBJECT TEACHING GUIDE

G646 - Logic

Degree in Computer Systems Engineering

Academic year 2023-2024

1. IDENTIFYING DATA					
Degree	Degree in Computer Systems Engineering			Type and Year	Core. Year 2
Faculty	Faculty of Sciences				
Discipline	Subject Area: Mathematical Foundations of Computer Science Basic Training Module				
Course unit title and code	G646 - Logic				
Number of ECTS credits allocated	6	Term		Semester based (2)	
Web	https://aulavirtual.unican.es				
Language of instruction	Spanish	English Friendly	Yes	Mode of delivery	Face-to-face

Department	DPTO. MATEMATICAS, ESTADISTICA Y COMPUTACION
Name of lecturer	INES GONZALEZ RODRIGUEZ
E-mail	ines.gonzalez@unican.es
Office	Facultad de Ciencias. Planta: + 3. DESPACHO PROFESORES (3003A)
Other lecturers	

3.1 LEARNING OUTCOMES
- Understand the interaction between Logic and Computer Science.
- Learn and understand the logic concepts of syntax, semantics, logical consequence, correct reasoning, correctness and completeness, decidability in propositional and predicate logic.
- Learn and use the resolution algorithm in propositional and predicate logic.
- Be acquaintanced with axiomatic methods and natural deduction in logic. Be acquaintanced with the problems of undecidability and incompleteness.
- Learn, use and implement basic concepts and algorithms of Logic Programming.

4. OBJECTIVES
Be fluent in the propositional and predicate logic languages and prove the validity of a simple formula in both logics using different procedures.
Model simple situations and sentences in natural language with the most appropriate logical language.
Check the correctness of a simple reasoning.
Understand the process to transform a logical formula into equivalent or equisatisfiable formulae (normal and clausal form).
Know and use different proof procedures: semantic trees, general resolution...
Understand the computational mechanisms associated to automated theorem proving and logic programming.
Write a simple program in Prolog language.
Introduce the ideas of decidability and completeness as well as natural deduction as an axiomatic system.

6. SUBJECT PROGRAM	
CONTENTS	
1	INTRODUCTION Logic in Computer Science, history. Fundamental concepts.
2	PROPOSITIONAL LOGIC 1. Syntax and semantics - Introduction: propositions and connectives. - Syntax: alphabet and grammar. - Semantics: interpretation and evaluation, satisfiability, logical consequence, equivalence. 2. Normal forms: - Conjunctive and disjunctive normal form. - Clausal form, equisatisfability, Horn clauses, clause elimination strategies. 3. Proof methods: - Proof by refutation - Truth tables - Semantic trees - Resolution: resolution rule (consistency), proof by resolution (consistency, completeness), resolution algorithm. - Natural deduction (*)
3	PREDICATE LOGIC: 1. Syntax and semantics: - Introduction: motivation, extension of L0 - Syntax: alphabet, formulae, quantifiers - Semantics: interpretation and evaluation, satisfiability, logical consequence, equivalence. 2. Normal forms: - Prenex normal form: renaming, definition and existence, transformation. - Skolem normal form: definition, skolemisation algorithm, equisatisfiability - Clausal form: definition, transformation. 3. Herbrand's Theorem (*) - Semidecidability, Herbrand's universe - Herbrand's base and interpretation, Herbrand's Theorem - The methods of Gilmore and Davis and Putnam 4. Resolution: - Substitutions - Unification, most general unifier - Resolution: resolution rule, proof by resolution. - Strategies and refinements of resolution (*)
4	LOGIC PROGRAMMING - Introduction, SLD resolution - Prolog basic syntax, resolution process - Arithmetic, recursion, tail recursion - Static and dynamic structures - Control: the cut (*)

7. ASSESSMENT METHODS AND CRITERIA

Description	Type	Final Eval.	Reassessn	%
Written exam including theoretical and problem-solving questions.	Written exam	Yes	Yes	60,00
Programming assignments.	Others	No	Yes	20,00
Other coursework activities.	Others	No	Yes	20,00
TOTAL				100,00

Observations

The exact nature of coursework activities will depend on the course progress and the student needs and interests. The goal is to provide feedback to the students as well as coordinating coursework activities with the remaining courses.
Programming assignment grading is subject to a regular attendance to the lab sessions.
Programming assignment grades can be regained with a programming exam and coursework grades can be regained with additional test questions in the written exam.
Exam dates will be established by the Faculty (ordinary and resit period).

Observations for part-time students

Part-time students must sit for the written exam with the rest of the students. For the remaining activities, alternative formulae will be agreed between the student and the lecturer, taking into account the student's circumstances. The student will always have the opportunity to regain the grades as the rest of students.

8. BIBLIOGRAPHY AND TEACHING MATERIALS

BASIC

I. Bratko, “Prolog Programming for Artificial Intelligence”, Addison-Wesley, (1986).

J. Kelly, “The Essence of Logic”. Prentice Hall (1997)

U. Schöning, “Logic for Computer Scientists", Birkhäuser, 1st ed. 2nd printing (2008)