COURSE DESCRIPTION

1. Program Information

1.1 University	**Alexandru Ioan Cuza University of Iași**
1.2 Faculty	**Computer Science**
1.3 Department	**Computer Science**
1.4 Study Domain	**Computer Science**
1.5 Study Cycle	**Bachelor**
1.6 Study Program / Qualification	**Computer Science / Licentiate in Computer Science**

2. Course Information

2.1 Course Name			**Logic for computer Science**				
2.2 Course Instructor			**Ștefan Ciobâcă**				
2.3 Tutorial Class Instructor			**Andrei Arusoaie**				
2.4 Study Year	**I**	**2.5** Semester	**2**	**2.6** Evaluation	**E**	**2.7** Course Status*	**OB**

** OB – Compulsory / OP – Optional*

3. Total estimated hours (hours per semester and didactic activities)

3.1 Hours per week	**4**	of which: **3.2** lecture	**2**	**3.3** tutorial/laboratory class	**2**
3.4 Hours in curriculum	**56**	of which: **3.5** lecture	**28**	**3.6**tutorial/laboratory class	**28**
Time Distribution					hours
Study of textbook, lecture notes, bibliography, and others					**14**
Supplementary documentation in the library, in electronic forums, and on the field					**28**
Preparation of tutorial/laboratories classes, homework, reports, portfolios and essays					**82**
Tutoring					-
Evaluation					**4**
Other activities					-

3.7 Total hours of individual study	**124**
3.8 Total hours per semester	**180**
3.9 Credits	**6**

4. Preconditions (if any)

4.1 Curriculum	**-**
4.2 Skills	**Ability to correctly understand a text, ability to express oneself, basic knowledge of mathematics**

5. Conditions (if any)

5.1 Course Operation	-
5.2 Tutorial/Laboratory Class Operation	**-**

6. Specific Skills Acquired

Professional Skills	**C1.**The ability to understand the difference between the syntax and the semantics of a logic. **C2.**The ability to understand and use inductive definitions and proofs. **C3.**The ability to model a real-world problem as a logical formula. **C4.** The ability to model computational problems as instances of SAT. **C5.**The ability to understand the differences between Propositional Logic and First-Order Logic. **C6.** The ability to compute normal forms and study the satisfiability, equivalence and validity of logical formulae. **C7.** The ability to understand a proof system and perform proofs using the Resolution Calculus and the Natural Deduction Calculus.

Transversal Skills	CT1. The ability to understand mathematical statements as first-order formulae. CT2. The ability to tell when an argument is deductively valid. CT3. The ability to write proper mathematical proofs.

7. Course Objectives (from the grid of specific skills acquired)

7.1 General Objectives	The students should have a basic understanding of formal logic and of its applications in computer science.
7.2 Specific Objectives	O1. To understand and reason about inductive definitions. O2. To model reality as propositional or first-order formulae. O3. To identify satisfiable, equivalent or valid formulae and respectively valid entailments. O4. To understand the difference between syntax and semantics. O5. To model computational problems as instances of SAT. O6. To prove propositional and/or first-order formulae using syntactic means such as Natural Deduction or Resolution. O7. To compute normal forms for formulae in Propositional Logic and First-Order Logic. O8. To compute most general unifiers and use first-order resolution to prove formulae. O9. To appreciate the importance of Logic in Computer Science.

8. General Description

8.1	**Course**	**Teaching Methods**	**Observations** (hours and bibliographic references)
1.	Introduction to Informal Logic	Lecture	2
2.	The Syntax of Propositional Logic	Lecture	2
3.	The Semantics of Propositional Logic	Lecture	2
4.	The Satisfiability Problem	Lecture	2
5.	Natural Deduction	Lecture	2
6.	Resolution	Lecture	2
7.	Functions, Predicates, Structures	Lecture	2
8.	Midterm Examination	Written test	6
9.	Syntax of First-Order Logic	Lecture	2
10.	Semantics of First-Order Logic	Lecture	2
11.	Normal Forms 1	Lecture	2
12.	Normal Forms 2, Ground Resolution	Lecture	2
13.	Unification and Resolution	Lecture	2
14.	Historical Considerations. Advanced applications of Logic to Computer Science: descriptive complexity, relational databases, program verification.	Lecture	2

Bibliography

Main references:

M. Huth, M. Ryan – *Logic in Computer Science: Modelling and Reasoning about Systems*, Cambridge University Press, 2000, ISBN 0-521-65200-6.
*** Ștefan Ciobâcă - *Logic in Computer Science (Lecture Notes)*

Supplementary references:

P. D. Magnus - *forall x - An Introduction to Formal Logic*

8.2	**Tutorial / Laboratory Class**	**Teaching methods**	**Observations** (hours and bibliographic references)
1.	Exercises corresponding to Week 1 Lecture	Review of the topics presented at the lecture, proposing a set of exercises, individual work, interactive methods on the board	2
2.	Exercises corresponding to Week 2 Lecture	Idem	2
3.	Exercises corresponding to Week 3 Lecture	Idem	2
4.	Exercises corresponding to Week 4 Lecture	Idem	2
5.	Exercises corresponding to Week 5 Lecture	Idem	2
6.	Exercises corresponding to Week 6 Lecture	Idem	2
7.	Exercises corresponding to Week 7 Lecture	Idem	2
8.	Midterm Examination	Discussion on the solutions of the problems from the written test	2
9.	Exercises corresponding to Week 9 Lecture	Review of the topics presented at the lecture, proposing a set of exercises, individual work, interactive methods on the board	2

10.	Exercises corresponding to Week 10 Lecture	Idem	2
11.	Exercises corresponding to Week 11 Lecture	Idem	2
12.	Exercises corresponding to Week 12 Lecture	Idem	2
13.	Exercises corresponding to Week 13 Lecture	Idem	2
14.	Review	Idem	2

Bibliography

*** Ștefan Ciobâcă - *Logic in Computer Science (Exercises and Support Applications for the Tutorial Class)*

9. Course content synchronization with the expectations of the community representatives, professional associations and employers from the program domain

The subject introduces basic notions like inductive definitions, syntax, semantics, propositional logic, first-order logic, satisfiability, which are required knowledge of any competent computer scientist.
The subject also promotes the development of critical thinking and exact reasoning.

10. Evaluation

Activity Type	**10.1 Evaluation criteria**	**10.2 Evaluation methods**	**10.3** The **weight of each evaluation form (%)**

10.4 Lecture	Quality of the answers to the questions in the written tests.	Two written tests	45% + 45%
10.5 Tutorial/ Laboratory Class	Quality of solving the proposed exercises. Very good answers.	Attendance Assessment of classroom activity Bonus (max 10%)	10% (attendance) 10% (bonus)
10.6 Minimal standards to pass			
At least 45% of the total number of points.			

Date	Lecturer	Tutorial/Laboratory Instructor
15 February 2018	Conf. Dr. Ștefan Ciobâcă	Lect. Dr. Andrei Arusoaie

Date of Approval in the Department Head of Department