

Wright State University

CORE Scholar

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

Spring 2010

CS 499/699: Logic for Computer Scientists

Pascal Hitzler

Wright State University - Main Campus, pascal.hitzler@wright.edu

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

Repository Citation

Hitzler, P. (2010). CS 499/699: Logic for Computer Scientists. .
https://corescholar.libraries.wright.edu/cecs_syllabi/433

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.

Logic for Computer Scientists

CS 499/699
Wright State University
Spring 2010

Brief Description:

Logic is often called the Calculus of Computer Science. Indeed, logic permeates most areas of Computer Science in one way or the other, sometimes more prominently, and sometimes as motivational or formal underpinning. Examples are database schema, program verification, semantics of programming languages, computer security, artificial intelligence, cognitive robotics, Web information systems, computer hardware circuitry, or modeling in software engineering.

In this course, we convey the foundations of logic for Computer Science, covering the central topics about which every Computer Scientist should have a basic knowledge. The covered material is fundamental and will help to prepare you to obtain a better understanding of topics which have a logical underpinning and which you are bound to encounter in your future studies or work life.

Student Learning Outcomes:

- Knowledge about the basics of propositional and first-order predicate logic, including algorithms.
- Ability to model knowledge in propositional and first-order predicate logic.
- Ability to apply basic mathematical proof techniques including (structural) induction, contraposition, proof by contradiction.
- Understanding of non-deterministic algorithms.
- Knowledge about the importance of logic in Computer Science.
- Understanding of the value of rigorous formal approaches.

Instructor:

Dr. Pascal Hitzler, 389 Joshi.
pascal@pascal-hitzler.de, <http://www.knoesis.org/pascal>
Office hours: Wednesdays 3pm to 4pm and by appointment.
Please use email as main means of communication with me outside class.

Class Hours:

Mondays and Wednesdays 8pm to 9:15pm, Joshi 193.

Prerequisites:

None.

Course Materials:

Required: Uwe Schöning, Logic for Computer Scientists, Birkhäuser, 2008
Recommended: Mordechai Ben-Ari, Mathematical Logic for Computer Science, Springer, 1993

Method of Instruction:

Lecture

Evaluation:

Homework (20%), mid-term exam (30%), final exam (50%)
Grading will follow a standard scale (A: 100-90, B: 89-80, C: 79-70, D: 69-60, F: 59-0). These may be adjusted in favor of the students.

Course Outline:

- Week 1 Introduction. Syntax of propositional logic.
- Week 2 Semantics of propositional logic. Notion of logical consequence.
- Week 3 Normal forms for propositional logic.
- Week 4 Tableaux algorithm for propositional logic.
- Week 5 Syntax of first-order predicate logic.
- Week 6 Semantics of first-order predicate logic.
- Week 7 Modeling in first-order predicate logic.

Week 8 Normal forms for first-order predicate logic.
Week 9 Theoretical aspects: soundness, completeness, (semi-)decidability, compactness.
Week 10 Tableaux algorithm for first-order predicate logic.