T-79.5102

Special Course in Computational Logic (4 cr)

Autumn 2007

Agenda for Autumn 2007

➤ Contents according to the TKK Study Programme 2007–2008:

Knowledge representation, reasoning, and decision-making.

Automated reasoning.

➤ The course concentrates on *declarative programming* which is much about specifying *what* is to be computed rather than *how* the computation actually takes place.

➤ From the methodological point of view, the course provides an in-depth introduction to *answer set programming* (ASP) which is a new rule-based approach to constraint programming.

Practical Arrangements

Lectures: Mondays, 12–14, room TB353

Lecturer: Docent, D.Sc.(Tech.) Tomi Janhunen, office TB335, tel. 09 451 3255, email `@tkk.fi`

Tutorials: Tuesdays, 15–16, room TB353

Course assistant: M.Sc.(Tech.) Antti Hyvärinen, office TB358, tel. 09 451 4774, email `@tkk.fi`

Home page: `http://www.tcs.hut.fi/Studies/T-79.5102/`

News: `opinnot.tik.logiikka`

Email: `t795102 @tcs.hut.fi`

Formal Course Requirements

Course credits (4cr) are granted on the following basis:

1. Three **home assignments** are made.
   - Modelling the given problem domain using rules
   - Using ASP tools to find a solution
   - Grading on scale 0–5 (a nonzero grade is required to pass)
2. An **examination** is passed with a grade 1–5.
   - The first exam is arranged on

     the 17th of December, 2007, 9–12, in hall T1.
   - Additionally, two other exams are arranged in 2008.

The course grade is based on the exam (70%) and the home assignments (30%, i.e., 10% weight is given for each assignment).

Course Material

- Lecture notes
- Questions from tutorials
- Articles and links
- Tools and manuals
- References to other similar courses
- Supplementary reading (not required):

  Chitta Baral: *Knowledge Representation, Reasoning, and Declarative Problem Solving*, Cambridge University Press, 2003.

Course Contents in Detail

Period I: Practical Introduction to ASP (lectures 1–6)

1. Introduction
2. Basic concepts
3. Negation and non-monotonicity
4. Further primitives
5. Extensions
6. Applications

Course Contents in Detail

Period II: Theoretical Background of ASP (lectures 7–12)

1. Complexity and approximations
2. Implementation techniques
3. Equivalence checking
4. Modular program development
5. Relationship with propositional logic
6. Further topics

General Objectives

- In-depth understanding of the ASP methodology
- Practical modelling/programming skills
- Knowledge of current ASP tools and systems
- Awareness of some applications and their main characteristics
- Basic understanding of the theoretical background of ASP and its relationship to other similar disciplines