T-79.5101

Advanced Course in Computational Logic (4 cr) L

Spring 2008

Practicalities

Prerequisites: T-79.3001/144 Logic in computer science: foundations (or corresponding knowledge)

Lectures: Tuesdays 10–12, TB353
Lecturer: Prof. Ilkka Niemelä, TB337,
puh. 451 3290, e-mail: Ilkka.Niemela@tkk.fi.

Tutorials: Mondays 15–16, TB353
Assistant : Lic.Sc.(Tech) Matti Järvisalo, TB354,
puh. 451 2896, e-mail: mjj@tcs.tkk.fi.

Homepage: `http://www.tcs.tkk.fi/Studies/T-79.5101/`

New group: `opinnot.tik.logiikka`

email: `t795101@tcs.tkk.fi`

Course Content

- Modal logics
  - — syntax
  - — semantics
  - — proof methods
- Applications of temporal logic in concurrent and distributed systems.
- Recurring concepts in computer science:
  - — formal model
  - — consistency and completeness
  - — efficiency; computational complexity of problems

Material

- Lectures (see slides on the course home page)
- M. Fitting: *Basic Modal Logic*, Handbook of Logic in Artificial Intelligence and Logic Programming, Volume 1, Logical Foundations, 1993.
- E. Clarke, O. Grumberg and D. Peled: *Model Checking*, The MIT Press, 1999. (Chapters 1-4)
- E.A. Emerson: *Automated Temporal Reasoning about Reactive Systems*, Logics for Concurrency, F. Moller and G. Birtwistle (Eds.), Springer-Verlag, 1996, LNCS 1043, pp. 39–99.

Passing the Course

- In order to pass the course one has to
  - pass compulsory home assignments (3 sets) and
  - pass the final exam (with a grade greater than 0).
- Course grade: the grade of the final exam
- You can earn at most 4 bonus points for the exam (maximum poins in exam 50):
  - Attendance in tutorials (at most 3p.):
    attending at least 4 times / 1 p.
    attending at least 8 times / 2 p.
    attending at least 11 times / 3 p.
  - Filling out the feed back form (1 p.)

Logic in Computer Science

- Formal methods are gaining popularity
- Logic-based tools widely used

  The use of formal verification tools is well established and becoming more so. Simulation- and emulation- based methodologies aren't sufficient to guarantee correctness with today's complex chips.

  (Carl Pixley, Motorola Inc. in IEEE Spectrum, Jan 1997, p. 61)
- This is due to a number of factors
  - The performance and memory capacity of computers are rapidly increasing
  - Algorithms and implementation techniques for logic based methods and tools have advanced dramatically

Modal Logic in Computer Science

- Modal logic is the logic of concepts such as *necessary, obligatory, known, believed, true in the future, provable,* ...
- Specification, analysis, verification of systems

  distributed and concurrent systems, reactive systems, protocols, security, ...
- Knowledge representation, natural language processing, software agents, semantic web, ontologies, ...

Challenges in System Design

- System specification and design error prone activities
- Early mistakes very expensive to repair.
- Need for more formal, mathematical methods.
  - "One should take a small portion of every dish"
  - "Define small"

Distributed and Concurrent Systems

- Several distributed and concurrent process
- Shared resources, process coordination, communication
- Unending operation cycle
- Examples: operating systems, communication protocols, device drivers, instrumentation and control systems, ...
- Designing such systems very challenging and new methods and tools are needed.

Example. (Fairness)

How to express precisely and to verify properties such as:

(i) Every process is excuted infinitely often

(ii) Every process that is constantly ready to be scheduled is excuted infinitely often

(iii) Every process that is infinitely often ready to be scheduled is excuted infinitely often

☞ Needed: a model describing the behavior of the system and a language for expressing properties of the system.

Example—cont'd.

Temporal logic provides a suitable framework.

$\Box P$: always (in the future) P.
$\Diamond P$: sometimes (in the future) P.

(i) $\Box\Diamond ex$

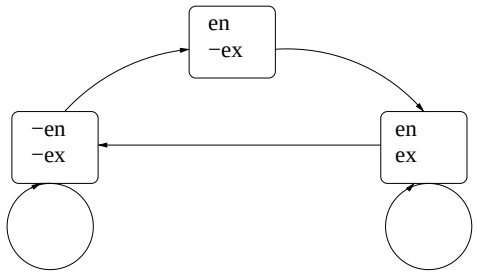
(ii) $\Box en \rightarrow \Box\Diamond ex$

(iii) $\Box\Diamond en \rightarrow \Box\Diamond ex$

How can temporal logic used in system design

1. Model checking: Is a given property true in a model?
2. Satisfiability: Are there models satisfying given properties?
3. Validity: Do all models satisfy given conditions?

Example—cont'd.



(i) $\Box\Diamond ex$?

(ii) $\Box en \rightarrow \Box\Diamond ex$?

(iii) $\Box\Diamond en \rightarrow \Box\Diamond ex$?