

Outline of contents of 2if25

- A simple formal model for system behavior: Transition system.

I. The finite case: propositional logics

1. Specification languages to specify properties of systems.
 - (a) Propositional logic.
 - (b) Propositional modal logic.
 - (c) Propositional linear time logic: *LTL*.
 - (d) Propositional branching time logic, limited version: computation tree logic, *CTL*.
 - (e) Propositional branching time logic, full version: *CTL**.
 - (f) Propositional modal μ -calculus.
2. Model checking formulas in the above propositional logics against finite transition systems, i.e., models of system behavior.
 - (a) The labeling approach for modal logic and *CTL*.
 - (b) The intersection of automata approach for *LTL*.
 - (c) The combination of the above two for *CTL**.
 - (d) The recursive algorithm for modal μ -calculus.

II. The potentially infinite case: first order logics

1. Specification languages to specify properties of systems.
 - (a) Hoare logic for programming languages.
 - (b) First order *LTL* for parallel components.
2. Theorem proving.
 - (a) Automated theorem proving for first order logic.
 - i. Sequent calculus for automated proving of first order formulas (PVS).
 - ii. Sequent calculus for automated proving first order formulas over of first order axiomatizations of programs (PVS).
 - iii. Sequent calculus for automated proving of Hoare logic formulas (Cocktail, PVS, ESC/Java?, Spec#?).
 - (b) Automated theorem proving for first order *LTL*.
 - i. Proof system for (automating) proving of first order *LTL* formulas (Manna/Pnueli).
 - ii. Proof system for (automating) proving of first order *LTL* formulas over first order *LTL* descriptions of programs (Manna/Pnueli).
 - iii. Proof system for (automating) proving of first order *LTL* formulas over programs (STEP, Manna).

2if25 is an introduction to formalization and the then possible (partial) automation of verification. It both provides an overview of what is available for those that take this as the only course in that direction, as well as provide a basis for more advanced or specialized lectures.

During the course we shall discuss what formalization and automation is about, what can and cannot be done, completeness, decidability, etc.

Two 2-hour lectures plus two 2-hour exercise sessions on each topic are the basis for the course. Material is handouts that will also appear on study-web and, additionally, copies of papers. Successful participation in the exercise sessions provides one bonus-point towards the exam result.