

The Calculus of Computation

Decision Procedures with Applications to Verification

(ECEN 4053/5053)
MWF 12:00-12:50, ECEE 1B28
Prof. Aaron Bradley

*It is reasonable to hope that the relationship between **computation** and **mathematical logic** will be as fruitful in the next century as that between **analysis** and **physics** in the last.*

— John McCarthy

A Basis for a Mathematical Theory of Computation, 1963

Computational logic is a fast-growing field with applications in the design and verification of software and hardware systems, artificial intelligence, and constraint solving. We introduce computational logic from the foundations of first-order logic to state-of-the-art decision procedures for arithmetic, data structures, and combination theories.

We also present a logical approach to engineering correct software. The increasing ubiquity of computers makes implementing correct systems more important than ever. Verification exercises using the π VC verifying compiler (see <http://theory.stanford.edu/~arbrad/pivc>) will develop your skills in designing, specifying, and verifying systems using logic.

Finally, we discuss a methodology for algorithmically deducing facts about programs, an important part of modern design and verification systems.

Prerequisites: Undergraduate background in programming at the level of CSCI 2270 and in discrete mathematics at the level of ECEN 4703 or CSCI 2824.

Text: A.R. Bradley and Z. Manna, *The Calculus of Computation: Decision Procedures with Applications to Verification*, Springer 2007.

Topics:

- Foundations:
 - Propositional and first-order logic, first-order theories, induction
 - Verification: inductive assertion method, ranking function method
- Algorithmic Reasoning:
 - Satisfiability decision procedures for linear arithmetic, equality, data structures, and combinations of theories
 - Static analysis of software

For more information, please email arbrad@cs.stanford.edu.