Computer Science 792 (Spring Term, 2000)

Computational Logic

Course instructor: Harry Mairson (`mairson@cs.bu.edu`), MCS 283, phone 353-8926. Office hours 11am–12pm Tuesday and Friday, and by arrangement. I especially encourage you to communicate with me via electronic mail, for fastest and most reliable responses to your questions. I try to read e-mail every 5 minutes, 24 hours a day.

Time and place: Tuesday and Thursday, 9.30–11am, MCS B46.

What is this course about? This course is a (non-comprehensive) introduction to topics in logic that are relevant to computer science. Unlike typical logic courses, it stresses computational metaphors: algorithmics and constructive mathematics, computational complexity, and decidability. The basic topics of the course are

1. Propositional logic *(6 lectures)*.
2. First-order logic *(5 lectures)*.
3. Intuitionistic logic *(7 lectures)*.
4. Higher-order logic and realizability *(4 lectures)*.
5. Linear logic *(7 lectures)*.

The discussion of propositional logic includes a study in miniature of resolution-based theorem proving procedures, and of logical characterizations of complexity classes. The survey of first-order logic continues this relation of language formalism to expressive power, and includes discussion of logic as a database query language, as well as proof of the completeness theorem that can be used as a proof search technique (i.e., resolution). Intuitionistic logic introduces λ-calculus (the theory on which Lisp, Scheme, and other functional languges are based) as a mechanism for representing and manipulating proofs. Our discussion of higher-order logic stresses complexity issues, and how proofs can be analyzed to produce computational information, proving a version of the famous "Dialectica" theorem of Gödel. The course concludes with a discussion of *linear logic*, a so-called *resource-conscious* logic; we will discuss the relationship between cut-elimination (proof simplification) in linear logic, and evaluation of programs written in λ-calculus.

Required work: Work for the course will include several problem sets, and some class presentations. Even though I love grading problem sets, I'm going to try to have it done by you on a round-robin basis. Items marked $\star$ in the syllabus are lectures where you may *volunteer* to do the presentation if you think you would find the subject matter interesting.

There is no textbook for the course—there will, however, be book chapters and papers to read, as well as class notes.

Grading for the course will be liberal. I am mainly concerned in presenting a good selection of lectures, in the hope that it might be of future research interest to the participants. Undergraduate courses are meals that are meant to be eaten in full. A graduate seminar like this one is a smorgasbord—*bon appetit.*

Tentative syllabus (30 lectures overall)

Recall that lectures marked $\star$ are available if you would like to do the presentation. Let me know in advance, and we can talk about what's required.

Administrivia and Introduction *[1 lecture]*

January 11: Course overview.

Propositional logic *[6 lectures]*

January 13: Syntax. Semantics: truth tables. Natural deduction. Sequent calculus. *[van Dalen: 1.1, 1.2, 1.5; Girard, Lafont, and Taylor: 2, 5]*

January 18: Conjunctive normal form. Resolution theorem proving and completeness. *[Chang and Lee: 2.4, 5.1, 5.2, 5.7; class notes]*

January 20: Cut elimination and proof normalization. *[Girard, Lafont, and Taylor: 5, 13; Girard]*

January 25: Circuit value problem and PTIME-completeness. Satisfiability and NP-completeness. *[Hopcroft and Ullman: 13; Papdimitriou: 8]*

January 27*: Quantified Boolean formulas and PSPACE-completeness. Savitch's theorem (PSPACE=NPSPACE). *[Hopcroft and Ullman: 13; Papdimitriou: 7]*

February 1*: Immerman's theorem (NSPACE[$f(n)$]=co-NSPACE[$f(n)$]). *[Immerman; Papdimitriou: 7]*

First-order logic *[5 lectures]*

February 3: Syntax. Semantics (model theory). Natural deduction and sequent calculus. Undecidability of provability. *[Chang and Lee: 3.1, 3.2; van Dalen: 2]*

February 8: Datalog and data complexity. Linear queries and LOGSPACE-completeness. Nonlinear queries and PTIME-completeness. *[van Dalen: 2.8; class notes]*

February 10: Herbrand's theorem. Unification. *[Chang and Lee: 4,5; class notes]*

February 15: Resolution theorem proving. *[Chang and Lee: 4,5]*

February 17*: Boundedness and deductive databases. *[Hillebrand et al.]*

Intuitionistic logic *[7 lectures]*

February 24: Foundations and philosophy of mathematics. Simply-typed λ-calculus and Curry-Howard isomorphism: programs as proofs. *[Girard, Lafont, and Taylor: 1,2; van Dalen: 5.1]*

February 29: Strong normalization of proofs and proof search. *[Girard, Lafont, and Taylor: 4; class notes]*

March 2*: Semantics: Kripke models for propositional intuitionistic logic. *[van Dalen: 5.3; Troelstra and van Dalen 2.5, 2.6]*

March 14*: "Double negation" embedding of classical logic in intuitionistic logic. *[van Dalen: 5.2, pp. 172ff.]*

March 16: Continuations, double-negation embeddings, and continuation-passing style. *[Murthy; class notes]*

March 21*: PSPACE-completeness of intuitionistic theorem proving. *[Statman]*

March 23*: PTIME-completeness of type inference for simply-typed λ-calculus. *[Class notes]*

Higher-order logic *[4 lecture]*

March 28: Higher-order typed λ-calculi: systems F_1, F_2, F_3, ... Representing logical connectives. *[Girard, Lafont, and Taylor: 11;van Dalen: 4; class notes]*

March 30*: Contracting proofs to programs. *[Leivant]*

April 4: Metamathematics: proving termination and totality. A version of the Gödel incompleteness theorem. *[Leivant; class notes]*

April 6*: Decision problem for higher-order logic: deciding equivalence of simply-typed λ-terms. *[Mairson]*

Linear logic *[7 lectures]*

April 11: Introduction to the linear connectives: multiplicatives, additives, exponentials. *[Girard]*

April 13*: Undecidability of derivability. *[Troelstra; class notes]*

April 18: Coding linear logic in proofnets. *[Lafont; class notes]*

April 20: Coding λ-calculus as proofnets. Call-by-value and call-by-name translations. Proofnets for control operators. *[Class notes]*

April 25: Semantics: geometry of interaction. Optimal reduction of λ-terms. *[Gonthier Abadi and Lévy; class notes]*

April 27: Complexity of optimal reduction. *[Asperti and Mairson]*

May 2: Complexity of bookkeeping and box maintenance. *[Class notes]*

References

Books:

Chin-Liang Chang and Richard Char-Tung Lee. **Symbolic Logic and Mechanical Theorem Proving.** Academic Press, 1973. *(chapters 2–5)*

Dirk van Dalen. **Logic and Structure.** Springer-Verlag, 1983.

Jean-Yves Girard. **Proof Theory and Logical Complexity.** Bibliopolis, 1987.

Jean-Yves Girard, Yves Lafont, Paul Taylor. **Proofs and Types.** Cambridge University Press, 1989. *(chapters 1–3, 5, 13)*

John Hopcroft and Jeffrey Ullman. **Introduction to Automata Theory, Languages, and Computation.** Addison-Wesley, 1979. *(chapter 13)*

Chetan Murthy. **Extracting Computational Content from Classical Proofs.** PhD Thesis, Cornell, 1990.

Christos Papadimitriou. **Computational Complexity.** Addison-Wesley, 1993.

Anne Troelstra. **Notes on Linear Logic.** CSLI Press, 1990.

Anne Troelstra and Dirk van Dalen. **Constructivism in Mathematics, Volume I.** North-Holland, 1988. *(chapter 2)*

Articles:

Andrea Asperti and Harry Mairson. *Parallel β-reduction is not elementary recursive.* **Information and Computation**, to appear.

Jean-Yves Girard. *Linear logic: its syntax and semantics.* In **Advances in Linear Logic** (ed. J.-Y. Girard, Y. Lafont, and L. Regnier) Cambridge University Press (1995), pp. 1–42.

Georges Gonthier, Martin Abadi, and Jean-Jacques Lévy. *The geometry of optimal lambda reduction.* 1992 POPL Conference.

Gerd Hillebrand, Paris Kanellakis, Harry Mairson, and Moshe Vardi. *Undecidable boundedness problems for database logic programs.* **J. Logic Programming** 25:2 (1995), pp. 163–190.

Neil Immerman. *Nondeterministic space is closed under complementation.* **SIAM J. Computing** 17 (1988), pp. 935–938.

Yves Lafont. *From proof nets to interaction nets.* In **Advances in Linear Logic** (ed. J.-Y. Girard, Y. Lafont, and L. Regnier) Cambridge University Press (1995), pp. 225–248.

Harry Mairson. *A simple proof of a theorem of Statman.* **Theoretical Computer Science** 103 (1992), pp. 387–394.

Chetan Murthy. *Classical proofs as programs: how, when, and why.* 1991 Constructivity in Computer Science Symposium.

Richard Statman. *Intuitionistic propositional logic is polynomial-space complete.* **Theoretical Computer Science 9** (1979), pp. 67–72.