

Carl Eastlund

Northeastern University
College of Computer and Information Science
West Village H 330
Boston, MA 02115

Phone: 617-877-0184
cce@ccs.neu.edu
ccs.neu.edu/~cce

Education

- 2012 PH.D. in Computer Science, Northeastern University
Thesis title: *Modular Proof Development in ACL2*
Adviser: Matthias Felleisen
- 2005 M.S. in Computer Science, Northeastern University
- 2001 B.S. in Computer Science with University Honors, Carnegie Mellon University
In-Major GPA: 4.0

Experience

- 2013 Postdoc, Northeastern University, Boston, MA
Continued research related to ACL2 modules and macros. Extended Racket libraries for generic interfaces, dictionary datatypes, and set datatypes.
- 2003-2012 Research/Teaching Assistant, Northeastern University, Boston, MA
Research extending the ACL2 theorem prover with macros from Racket and modules from ML. Developed Dracula, an IDE for automated theorem proving with ACL2, including features for automated testing and graphical programs. Contributed libraries to the Racket language, including tools for custom runtime software contracts.
- 2005 Intern, Sun Microsystems, Burlington, MA
Developed the prototype interpreter for the Fortress language, written in OCaml.
- 2004 Intern, IBM, Austin, TX
Designed and implemented type system for ML-based hardware description language.
- 2003 Software Engineer, Lockheed Martin, Gaithersburg, MD
Worked on software to transfer operations between GPS satellite ground control stations.
- 2001-2002 Software Engineer, Green Hills Software, Santa Barbara, CA
Implemented drop-in replacement compatibility with ARM vendor assembler.

Interests

I am interested in the design and construction of programming and verification tools.

Publications

BOOK CHAPTERS

- 2008 Modular ACL2, in *Semantics Engineering with PLT Redex* by Matthias Felleisen, Robert Bruce Findler, and Matthew Flatt, editors.

CONFERENCES

- 2012 Run Your Research: On the Effectiveness of Lightweight Mechanization, with Casey Klein, John Clements, Christos Dimoulas, Matthias Felleisen, Matthew Flatt, Jay A. McCarthy, Jon Rafkind, Sam Tobin-Hochstadt, and Robert Bruce Findler, *Proc. 39th ACM Symposium on Principles of Programming Languages (POPL)*.
- 2010 Hygienic Macros for ACL2, with Matthias Felleisen, *Proc. 11th International Symposium on Trends in Functional Programming (TFP)*.
- 2009 Making Induction Manifest in Modular ACL2, with Matthias Felleisen, *Proc. 11th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming (PPDP)*.
- 2009 Toward a Practical Module System for ACL2, with Matthias Felleisen, *Eleventh International Symposium on Practical Aspects of Declarative Languages (PADL)*.

WORKSHOPS

- 2009 Sequence Traces for Object-Oriented Executions, with Matthias Felleisen, *Scheme and Functional Programming 2009*.
- 2009 Automatic Verification for Interactive Graphical Programs, with Matthias Felleisen, *Eighth International Workshop On The ACL2 Theorem Prover and Its Applications*.
- 2009 DoubleCheck Your Theorems, *Eighth International Workshop On The ACL2 Theorem Prover and Its Applications*.
- 2008 Functional Programming and Theorem Proving for Undergraduates: A Progress Report, with Rex Page and Matthias Felleisen, *Functional and Declarative Programming in Education (FDPE) 2008*.
- 2007 ACL2 for Freshmen: First Experiences, with Dale Vaillancourt and Matthias Felleisen, *Seventh International Workshop on the ACL2 Theorem Prover and its Applications*.

Professional Activities

- 2013 International Workshop on the ACL2 Theorem Prover and its Applications, Program Committee

Research Statement

My research interests include programming language design, software engineering, and formal verification, with a focus on expressive and reliable programming tools. I am especially interested in module systems, metaprogramming, and automated verification. The language I created in my dissertation combines the formal verification of ACL2, the module system features of ML, and the metaprogramming tools of Racket. The result is a programming language that facilitates top-down specification and verification, generic reusable components, and flexible domain-specific sublanguages.

Teaching Statement

As a teacher, I am interested in the topics of design-based programming, the semantics of programming languages, data structures, and algorithm design techniques. My approach to teaching is informed by my experience with the Program by Design curriculum, exemplified by the “How to Design Programs” textbook and the “TeachScheme!” series of workshops. A combination of lightweight specifications, data definitions, and unit tests help students focus on design rather than low-level details of implementation or syntax. The Dracula and DoubleCheck tools I have produced extend these student-targetted design tools with

random testing and theorem proving, which allows students to build gradually from concrete counterexamples to symbolic reasoning.

Teaching

2012-2013 Instructor, Northeastern University, Algorithms and Data
2008 Instructor, Teaching Software Correctness Workshop, University of Oklahoma
2008 Teaching Assistant, TeachScheme!/ReachJava Workshop, Northeastern University
2003-2007 Teaching Assistant, Northeastern University, Fundamentals of Computer Science I/II

References

Matthias Felleisen

Trustee Professor of Computer Science
Northeastern University
360 Huntington Avenue
Boston, MA 02115
matthias@ccs.neu.edu

Daniel P. Friedman

Professor of Computer Science
Indiana University
150 S. Woodlawn Ave.
Bloomington, IN 47405
dfried@cs.indiana.edu

Rex Page

Professor of Computer Science
University of Oklahoma
110 West Boyd Street
Norman, OK 73019
page@ou.edu

Ruben Gamboa

Associate Professor of Computer Science
University of Wyoming
1000 E. University Avenue
Laramie, WY 82071
ruben@uwyo.edu