William E. Byrd

http://webyrd.net/

Scientist
Department of Computer Science
University of Alabama at Birmingham

Work
115A Campbell Hall
1300 University Boulevard
Birmingham, AL 35294-1170
webyrd@uab.edu

Residence
1508 4th Ave S
Apt 223
Birmingham, AL 35233
(812) 320-8505 (mobile)
webyrd@gmail.com

EDUCATION

Doctor of Philosophy in Computer Science, September 2009
Indiana University, Bloomington, IN
Specialty Area: Programming languages, with an emphasis on functional and logic programming.
Committee: Daniel P. Friedman (Advisor), Amr Sabry, Christopher T. Haynes, Lawrence S. Moss.

Bachelor of Science in Computer Science, *cum laude*, May 1999
University of Maryland, Baltimore County, Baltimore, MD

Bachelor of Science in Special Education, *magna cum laude*, December 1994
College of Charleston, Charleston, SC

RESEARCH INTERESTS

Programming Languages and Program Synthesis
For over ten years I have been working on miniKanren, an embedded domain-specific language for constraint logic programming. I am especially interested in using miniKanren to write programs as *relations*, without distinguishing between input and output arguments. Over the past decade my colleagues and I have developed relational interpreters, term reducers, type inferencers, and theorem provers, all of which are capable of program synthesis, or of otherwise running "backwards." miniKanren is described in *The Reasoned Schemer* (MIT Press, 2005).

To explore how relational programming can be used for program synthesis we have created a prototype interactive editor, Barliman, that can synthesize higher-order, recursive Scheme functions from example inputs and outputs. In addition to investigating software engineering applications of Barliman, I am working with Molly Feldman and others in Erik Andersen's lab at Cornell on adapting Barliman for educational uses, including automated tutoring.

In addition to relational programming, I am very interested in functional programming, especially in Scheme, Racket, Clojure, and other Lisps. I am a member of the Scheme and Functional Programming Workshop steering committee, organized the 2013 Scheme and Functional Programming Workshop, and have served on the program committees for the International Conference on Functional Programming (2015), the Scheme and Functional Programming Workshop (2013 and 2014), the International Workshop on Functional and (Constraint) Logic Programming (2014), and the International Lisp Conference (2014). I have given invited talks and tutorials at a variety of functional programming conferences and workshops.

Biology and Medicine
Recently I have become extremely interested in biology, especially synthetic biology. I took a intense two-week wet-lab synthetic biology course at Cold Spring Harbor Laboratory in the summer of 2015.

Since then I have been helping the Bertozzi lab at Stanford and the Woo lab at Harvard on improving their IsoStamp software for glycoprotein analysis using tandem mass spectrometry. I have begun helping the Bild lab at Utah with low-cost wet-lab automation, and have been helping the Myers lab at Utah with their work on the Synthetic Biology Open Language (SBOL) and their libSBOLj library.

I am especially interested in applying programming languages technology, including program synthesis, to problems in biology and medicine. My colleague Greg Rosenblatt and I have begun exploring how to use Barliman and miniKanren to reason about, and to automatically generate, programs in Kappa, a high-level rules-based language for modeling protein interaction networks. I am also interested in how declarative programming can be used to automate laboratory procedures, especially using microfluidic, "lab on a chip," and bio-MEMS systems.

PUBLICATIONS

Books

Daniel P. Friedman, **William E. Byrd**, and Oleg Kiselyov. *The Reasoned Schemer*. The MIT Press, Cambridge, MA, 2005. An introduction to relational (pure logic) programming using the miniKanren language. Includes the complete implementation of miniKanren.

In Preparation

The Reasoned Schemer, 2nd Edition (with Daniel P. Friedman, Oleg Kiselyov, and Jason Hemann). Streamlines the miniKanren logic programming language from the first edition, and better explains how to write programs as relations.

Theses

William E. Byrd. *Relational Programming in miniKanren: Techniques, Applications, and Implementations*. Indiana University, Bloomington, IN, September 30, 2009.

Refereed Papers

William E. Byrd, Daniel P. Friedman, Ramana Kumar, and Joseph P. Near. A Shallow Scheme Embedding of $\perp$-Avoiding Streams. To appear in a special issue of *Higher-Order and Symbolic Computation*, in honor of Mitchell Wand's 60th birthday.

William E. Byrd, Michael Ballantyne, Greg Rosenblatt, and Matthew Might. Functional Pearl: A Unified Approach to Solving Seven Programming Problems. To appear in *Proceedings of the ACM on Programming Languages (PACMPL)*, volume 1, ICFP, article 8, September 2017.

Jason Hemann, Dan Friedman, **William E. Byrd**, and Matthew Might. A Small Embedding of Logic Programming with a Simple Complete Search. In *Proceedings of the Dynamic Languages Symposium 2016 (DLS '16)*, Amsterdam, 2016.

Dakota Fisher, Matthew Hammer, **William E. Byrd**, and Matthew Might. miniAdapton: A Minimal Implementation of Incremental Computation in Scheme. In *Proceedings of the 2016 Workshop on Scheme and Functional Programming*, Nara, Japan, 2016.

Steven Lyde, Matthew Might, and **William E. Byrd**. Control-Flow Analysis of Dynamic Languages via Pointer Analysis. In *Proceedings of the Dynamic Languages Symposium 2015 (DLS '15)*, Pittsburgh, PA, 2015.

William E. Byrd, Eric Holk, and Daniel P. Friedman. miniKanren, Live and Untagged: Quine Generation via Relational Interpreters (Programming Pearl). In *Proceedings of the 2012 Workshop on Scheme and Functional Programming*, Copenhagen, Denmark, 2012.

Claire E. Alvis, Jeremiah J. Willcock, Kyle M. Carter, **William E. Byrd**, and Daniel P. Friedman. cKanren: miniKanren with Constraints. In *Proceedings of the 2011 Workshop on Scheme and Functional Programming (Scheme '11)*, Portland, OR, 2011.

Eric Holk, **William E. Byrd**, Nilesh Mahajan, Jeremiah Willcock, Arun Chauhan, and Andrew Lumsdaine. Declarative Parallel Programming for GPUs. In *Proceedings of the International Conference on Parallel Computing (ParCo)*, Ghent, Belgium, 2011.

Eric Holk, **William E. Byrd**, Jeremiah Willcock, Torsten Hoefler, Arun Chauhan, and Andrew Lumsdaine. Kanor: A Declarative Language for Explicit Communication. In *Proceedings of the Thirteenth International Symposium on the Practical Aspects of Declarative Languages (PADL)*, Austin, TX, pp. 190–204, 2011.

Andrew W. Keep, Michael D. Adams, Lindsey Kuper, **William E. Byrd**, and Daniel P. Friedman. A Pattern-matcher for miniKanren -or- How to Get into Trouble with CPS Macros.
In *Proceedings of the 2009 Workshop on Scheme and Functional Programming*, Cal Poly Technical Report CPSLO-CSC-09-03, pp. 37–45, 2009.

Joseph P. Near, **William E. Byrd**, and Daniel P. Friedman. αleanTAP: A Declarative Theorem Prover for First-Order Classical Logic. In *Proceedings of the 2008 International Conference on Logic Programming (ICLP)*, LNCS vol. 5366, Springer-Verlag, Heidelberg, pp. 238–252, 2008.

Oleg Kiselyov, **William E. Byrd**, Daniel Friedman, and Chung-chieh Shan. Pure, Declarative, and Constructive Arithmetic Relations (Declarative Pearl). In *Proceedings of the 2008 International Symposium on Functional and Logic Programming (FLOPS)*, LNCS vol. 4989, Springer-Verlag, Heidelberg, pp. 64–80, 2008.

William E. Byrd and Daniel P. Friedman. αKanren: A Fresh Name in Nominal Logic Programming. In *Proceedings of the 2007 Workshop on Scheme and Functional Programming*, Université Laval Technical Report DIUL-RT-0701, pp. 79–90, 2007.

William E. Byrd and Daniel P. Friedman. From Variadic Functions to Variadic Relations: A miniKanren Perspective. In *Proceedings of the 2006 Scheme and Functional Programming Workshop*, University of Chicago Technical Report TR-2006-06, pp. 105–117, 2006.

Alan T. Sherman, Brian O. Roberts, **William E. Byrd**, Matthew R. Baker, and John Simmons. Developing and Delivering Hands-on Information Assurance Exercises: Experiences with the Cyber Defense Lab at UMBC. In *Proceedings from the Fifth IEEE Systems, Man and Cybernetics Information Assurance Workshop*, West Point, NY, pp. 242–249, 2004.

Other Writings

William E. Byrd and Daniel P. Friedman.
Towers of Little Languages. (October 2013)
Foreword to *The Joy of Clojure, Second Edition* by Michael Fogus and Chris Houser.
`http://www.manning.com/fogus2/`

SCHOLARSHIPS

National Institute of General Medical Sciences (NIGMS) Scholarship (January, 2016)
Awarded $2,235 towards tuition for the Cold Spring Harbor Laboratory synthetic biology course I attended in the summer of 2015.

PROGRAMMING TOOLS DESIGNED

Barliman (in collaboration with Greg Rosenblatt)
Program editor that can perform synthesis of higher-order, recursive Scheme code.
`https://github.com/webyrd/Barliman`

PROGRAMMING LANGUAGES DESIGNED

miniKanren
A declarative applicative logic programming system.
`http://miniKanren.org/`
(in collaboration with Daniel P. Friedman, Oleg Kiselyov, and Chung-chieh Shan)
Basis for Clojure's `core.logic` standard library (`https://github.com/clojure/core.logic`)

cKanren
A constraint logic programming framework.
`https://github.com/calvis/cKanren`
(in collaboration with Daniel P. Friedman, Claire Alvis, Jeremiah Willcock, and Kyle Carter; derived from miniKanren)
Basis for Racket's `cKanren` library (`https://github.com/calvis/cKanren`)

αKanren
A nominal logic programming language.
`http://www.cs.indiana.edu/~webyrd/#alphaKanren`
(in collaboration with Daniel P. Friedman; derived from miniKanren, and inspired by James Cheney and Christian Urban's αProlog)
Basis for Clojure's `core.logic.nominal` standard library
(`https://github.com/clojure/core.logic/wiki/core.logic.nominal`)

Harlan
An experimental functional programming language for the GPU.
`https://github.com/eholk/harlan`
(in collaboration with Eric Holk, Claire Alvis, Jeremiah Willcock, Nilesh Mahajan, Ryan Newton, Arun Chauhan, and Andrew Lumsdaine)

Kanor
A declarative language for explicit communication.
(in collaboration with Eric Holk, Jeremiah Willcock, Nilesh Mahajan, Aaron Hsu, Arun Chauhan, and Andrew Lumsdaine)

STEERING COMMITTEES

Scheme and Functional Programming Workshop Steering Committee (September 2015–present)

PROGRAM COMMITTEES

20th ACM SIGPLAN International Conference on Functional Programming (ICFP 2015)
August 31–September 2, 2015, Vancouver, Canada.

2014 Scheme and Functional Programming Workshop
November 19, 2014, Washington, DC. Co-located with Clojure/conj.

International Workshop on Functional and (Constraint) Logic Programming (WFLP 2014)
September 15–17, 2014, Lutherstadt Wittenberg, Germany.

International Lisp Conference (ILC 2014)
August 14–17, 2014, Université de Montréal, Montréal, Canada.

2013 Scheme and Functional Programming Workshop (Program Chair)
November 13, 2013, Alexandria, VA. Co-located with Clojure/conj.

ACADEMIC WORKSHOPS ORGANIZED

2013 Scheme and Functional Programming Workshop
November 13, 2013, Alexandria, VA. Co-located with Clojure/conj.
`http://webyrd.net/scheme-2013/`

INVITED COURSES

Summer School on Relational Programming (sponsored by JetBrains, courtesy of Dmitri Boulytchev)
St. Petersburg, Russia, August 24–28, 2015.
Taught a one-week, 40-hour course on relational programming in miniKanren to 13 students from universities in St. Petersburg and from JetBrains.
Topics included: introduction to Scheme and Racket, introduction to relational programming, translating functions to relations, environment-passing interpeters in Scheme, relational interpreters in miniKanren, and implementing logic programming and constraint logic programming systems.

INVITED TALKS

Program Synthesis with miniKanren
Code Mesh 2017, London, England, November 8, 2017.
`http://www.codemesh.io/`

The Reasoned Racketeers (keynote, with Daniel P. Friedman)
RacketCon, Seattle, WA, October 7, 2017.
`http://con.racket-lang.org/`

The Most Beautiful Program Ever Written
Papers We Love, New York, NY, April 12, 2017.
`https://www.meetup.com/papers-we-love/events/238718664/`

Synthesis of Lisp Programs
LispNYC, New York, NY, April 11, 2017.
`https://www.meetup.com/LispNYC/events/236903739/`

Relational programming
KatsConf 2, Dublin, February 18, 2017.
`http://www.katsconf.com/#nav-speakers`

The Promise of Relational Programming
Partial Evaluation and Semantic-Based Program Manipulation (PEPM 2016), San Diego, CA, January 18, 2016.
`http://conf.researchr.org/event/POPL-2016/pepm-2016-invited-invited-talk-william-byrd`

A Vision for Relational Programming in miniKanren
Code Mesh 2015, London, England, November 4, 2015.
`https://vimeo.com/146117469`

The Promise of Relational Programming (keynote)
PolyConf, Poznań, Poland, July 2, 2015.
`https://www.youtube.com/watch?v=eQL48qYDwp4`

Relational Programming in miniKanren
The Computer Laboratory, University of Cambridge, Cambridge, England, December 6, 2013.
`http://talks.cam.ac.uk/talk/index/49257`

Meta-Programming in Logic Programming (with Nada Amin)
Code Mesh 2013, London, England, December 5, 2013.
`http://codemesh.io/#william-byrd`

Fun with Relational Interpreters in miniKanren (keynote, with Daniel P. Friedman)
flatMap, Oslo, Norway, May 14, 2013.
`http://2013.flatmap.no/danwill.html`

The Unreasonable Schemers (guest of honor, with Daniel P. Friedman)
miniKanren *Conf°* (associated with Clojure/West), Portland, OR, March 19, 2013.
`http://clojurewest.org/schedule/`

Relational Programming in miniKanren (with Daniel P. Friedman)
Strange Loop, St. Louis, MO, September 24, 2012.
`http://www.infoq.com/presentations/miniKanren`

3D Printing, Open Hardware, and the DIY Revolution
CATALYST, Informatics and Computing Student Association, Indiana University, Bloomington, IN, February 18, 2012.

3D Printing, Open Hardware, and the DIY Revolution
Technology Management Club, Kelley School of Business, Indiana University, Bloomington, IN, February 7, 2012.

3D Printing, Open Hardware, and the DIY Revolution
Mechanical Engineering Technology (MET) Department, Purdue University, October 12, 2011. Talk led Purdue's MET Department to create an undergraduate digital fabrication club, and to purchase of five 3D printers and two CNC routers for the club.

TALKS

Barliman: Trying the Halting Problem Backwards, Blindfolded (with Greg Rosenblatt)
Clojure/conj, Austin, TX, December 3, 2016.
`https://www.youtube.com/watch?v=er_lLvkklsk`

New Tools and Practices for Online Collaboration in Teaching, Learning, and Research of Programming Languages
Off-the-Beaten Track (co-located with POPL 2016), St. Petersburg, FL, January 23, 2016.
`http://conf.researchr.org/getImage/OBT-2016/orig/OBT_2016_paper_14.pdf`

miniKanren, a Language for Relational Programming (CS242 guest lecture)
Stanford University, Stanford, CA, November 30, 2015.

Concatenative Programming (with Rob Martin)
Lambda Lounge Utah, Sandy, UT, May 12, 2015.

We Need Real Tools for Generating Type Inferencers (with Nada Amin)
Off-the-Beaten Track (co-located with POPL 2014), San Diego, CA, January 25, 2014.

From Greek to Clojure! (with Nada Amin)
Clojure/conj, Alexandria, VA, November 14, 2013.
`https://www.youtube.com/watch?v=7kPMFkNm2dw`

A Relational Exploration of the Chomsky Hierarchy (with Daniel P. Friedman)
Strange Loop, St. Louis, MO, September 18–20, 2013.
`http://www.infoq.com/presentations/chomsky-hierarchy`

miniKanren Philosophy (with Daniel P. Friedman)
Clojure/conj, Raleigh, NC, November 16, 2012.
`http://www.youtube.com/watch?v=fHK-uS-Iedc`

miniKanren, Live and Untagged: Quine Generation via Relational Interpreters
Scheme and Functional Programming Workshop, Copenhagen, Denmark, September 9, 2012.

Declarative Parallel Programming
Computer Science Department, Colorado State University, Ft. Collins, CO, February 29, 2012.

Generating Quines Using a Relational Interpreter
Indiana University Logic Seminar, Bloomington, IN, February 15, 2012.
`http://www.math.indiana.edu/seminars/event.phtml?id=3355`

miniKanren ("Untalk", with Daniel P. Friedman)
Clojure/conj, Raleigh, NC, November 10, 2011.
`http://blip.tv/clojure/dan-friedman-and-william-byrd-minikanren-5936333`

Animatronic Kittehs, 3D Printers, and Rogue Game Studios –or– Why Computer Science Education Should be Just-in-Time, not Just-in-Case, and Why (Epic) Failure Should Always be an Option
Indiana University Programming Languages Group, Bloomington, IN, April 15, 2011.
`http://vimeo.com/23254348`

From Variadic Functions to Variadic Relations: A miniKanren Perspective
Scheme and Functional Programming Workshop, Portland, OR, September 17, 2006.

INVITED TUTORIALS

Thinking About Recursion (with Nada Amin)
KatsConf 2, Dublin, February 18, 2017.
`http://www.katsconf.com/#nav-speakers`

Interpreting Scheme procedures as logic programs using miniKanren (with Michael Ballantyne)
Scheme and Functional Programming Workshop, Vancouver, Canada, September 4, 2015.
http://andykeep.com/SchemeWorkshop2015/

Write a Relational Scheme Interpreter in miniKanren
PolyConf, Poznań, Poland, July 2, 2015.
http://polyconf.com/

Logic Night
Lambda Lounge Utah, Sandy, UT, May 13, 2014.
http://www.meetup.com/Lambda-Lounge-Utah/events/164368252/

miniKanren Tutorial (with Daniel P. Friedman)
Practical Aspects of Declarative Languages (PADL 2014), San Diego, CA, January 23, 2014.
http://www.ist.unomaha.edu/padl2014/#invited

Electronic Voodoo Dolls (e-Textiles Tutorial)
ZOOM—Examining the Future of Craft, Indiana University, Bloomington, IN, October 19, 2013.
http://www.iub.edu/~zoom/

TUTORIALS

Concatenative Programming (with Rob Martin)
Code Mesh 2015, London, England, November 2, 2015.
http://www.codemesh.io/codemesh2015/william-e-byrd

Program Synthesis Using miniKanren (with Daniel P. Friedman)
Strange Loop, St. Louis, MO, September 17, 2014.
https://thestrangeloop.com/sessions/program-synthesis-using-minikanren

miniKanren Summer School (with Daniel P. Friedman and Jason Hemann)
University of Utah, Salt Lake City, UT, May 27 and 29, 2014.

Program Transformations (with Nada Amin)
Lambda Jam, Chicago, IL, July 9, 2013.
http://lambdajam.com/sessions#amin

ONLINE TUTORIALS (at https://www.youtube.com/user/WilliamEByrd)

Weekly miniKanren "uncourse" on Google+ Hangouts
https://www.youtube.com/watch?v=iCuVTGWNU3s&index=1&list=PLO4TbomOdn2cks2n5PvifialL8kQwt0aW

Older miniKanren Hangouts
https://www.youtube.com/watch?v=vRrgaibcTYs&list=PLO4TbomOdn2eGFBHHHZpwo2o82e8Tv6gI

Google+ Functional & Logic Programming Tutorials
https://www.youtube.com/watch?v=2GfFlfToBCo&list=PLO4TbomOdn2dD5HsavVOhlZzZ37o73P3F

INTERVIEWS

Mostly Erlang Episode 050 – miniKanren With William Byrd
Recorded by Zachary Kessin, Thursday, November 13, 2014.
http://mostlyerlang.com/2014/11/26/050-minikanren-with-william-byrd/

Functional Geekery Episode 9 – William E. Byrd
Recorded by Proctor, Monday, April 21, 2014.
http://www.functionalgeekery.com/episode-9-william-e-byrd/

William Byrd on Logic and Relational Programming, miniKanren
Recorded by InfoQ at Code Mesh 2013, London, England, December, 2013.
http://www.infoq.com/interviews/byrd-relational-programming-minikanren

UNDERGRADUATE PROJECTS SUPERVISED

Neurosymbolic computation in miniKanren (2014)
Student: Tevyn Bell
(Inspired by Pascal Hitzler's work on combining neural networks and first-order logic.) Tevyn and I implemented a prototype relational simulator for feedforward neural networks.

Weaver: A system for developing interactive fiction based on rules and defeasible logic (2013)
Student: Brittany Ann Moore
(Inspired by previous work with Andrew "Zarf" Plotkin on rule-based interactive fiction.)

"Seg-bot" Arduino-controlled Rideable Segway Clone (2011–2013)
Student: Maria Khokhar
(Inspired by the Seg-bot project: http://prototyperobotics.com/projects/the-seg-bot.) Maria and I took a welding class, sourced the parts, welded the steel frame, finished mechanical assembly, and wired the electronics.

Grammar-based Music Generation (summer 2012)
Students: Tevyn Bell, Emilie Mitchell, Brittany Ann Moore
(Inspired by Robert M. Keller's Impro-Visor project: http://www.cs.hmc.edu/~keller/jazz/improvisor/.) An experiment in performing research with first-year undergraduates (Tevyn and Emilie). We were able to implement most, but not all, of the grammar-based parsing algorithm used in Impro-Visor. We spent much of our time implementing the languages and machines from Michael Sipser's *Introduction to the Theory of Computation*, in both Scheme and miniKanren. We also discussed and implemented Scheme interpreters, memoization extensions to Scheme, CYK parsing, and grammar-related syntactic abstractions using hygienic macros.

RESIDENCIES

Hacker in Residence, Hacker School, New York City
November 18–22, 2013
https://www.hackerschool.com/residents#will-byrd
Gave multiple talks and tutorials on relational programming. Worked with students on their projects, including a relational solver for propositional logic and an artificial intelligence system to play the real-time strategy game *StarCraft: Brood War*.

EMPLOYMENT

UNIVERSITY OF ALABAMA AT BIRMINGHAM, Birmingham, AL
Scientist, Department of Computer Science,
August 2017–present
https://www.uab.edu/cas/computerscience/
Currently working with Matt Might and other members of the Hugh Kaul Personalized Medicine Institute on applying relational programming, interpreters, program synthesis, and static analysis to biology and medicine.

UNIVERSITY OF UTAH, Salt Lake City, UT
Research Assistant Professor, U Combinator Research Group,
February 2017–July 2017
`http://www.ucombinator.org/`
Research Associate, U Combinator Research Group,
October–December 2012 (remote), January 2013–February 2017 (on-site)
`http://www.ucombinator.org/`
Currently working with Matt Might and other members of the U Combinator programming languages research group on relational programming, interpreters, program synthesis, and static analysis, and applications of computer science to biology and medicine.

INDIANA UNIVERSITY, Bloomington, IN
Postdoctoral Researcher, Center for Research in Extreme Scale Technologies (CREST)
(formerly Open Systems Lab), March 2010–December 2012
`http://www.crest.iu.edu/`
Worked with Eric Holk, Andrew Lumsdaine, Arun Chauhan, and other members of the NSF-funded Declarative Parallel Programming project on the design, implementation, and assessment of two declarative programming languages for High-Performance Computing: *Kanor*, an embedded Domain-specific Language for message passing; and *Harlan*, a high-level language for general-purpose Graphics Processing Unit (GPGPU) programming. Worked with Nicole Jacquard, Andrew Lumsdaine, Craig Stewart, and Kylie Peppler towards the creation of an IU digital fabrication laboratory (Fab Lab).

Course Instructor, A290, Spring 2012
Course designer and inaugural instructor for A290, Tools for Computing: Arduino Development. Asked by the School of Informatics and Computing to design and teach a new 8-week course on physical computing using the Arduino microcontroller, based on previous success integrating Arduino into H211. Taught students how to solder, design and debug circuits, and program Arduino. Challenged students to design and implement their own unique physical computing capstone projects.

Course Instructor, H211, Fall 2010, Fall 2011
`http://www.cs.indiana.edu/classes/h211/`
Course instructor for H211, Honors Introduction to Computer Science. Redesigned course to incorporate Arduino microcontroller/electronics art projects, 3D modeling, fractals, procedural music generation, and game design and implementation. Incorporated smart materials and Arduino-controlled electronic piano workshops from visiting researcher Catarina Mota. Started popular undergraduate Arduino and 3D Printing clubs at the request of former H211 students.

Visiting Researcher, August 2009–January 2010
Refined a novel distributed architecture for data privacy in smart rooms. Implemented and benchmarked a prototype system in Scheme, using OpenSSL and a custom version of the *ThreshSig* Java threshold cryptography library.

Associate Instructor, C311 and B521, September 2003–May 2009
`http://www.cs.indiana.edu/classes/c311/` `http://www.cs.indiana.edu/classes/b521/`
Co-taught IU's graduate and undergraduate introductory programming languages courses. Co-wrote *The Reasoned Schemer* (MIT Press, 2005), one of two textbooks used in C311 and B521. Co-designed the miniKanren logic programming language used in both courses. Wrote and delivered lectures on every topic in the course, including: environment-passing, continuation-passing, and store-passing interpreters; continuation-passing style, trampolining, registerization, and other correctness-preserving transformations; hygienic macro expansion; types and type inference; and functional and logic programming paradigms. Nominated for the department's *Associate Instructor of the Year* award.

CAMP GREENTOP, Sabillasville, MD
Director, Summer 2002
Responsible for the safety and well-being of 200 campers and 65 staff at the nation's oldest residential summer camp for children and adults with physical and multiple disabilities. Interviewed and hired staff members, evaluated staff performance, and directly supervised the business manager, unit leaders, activities coordinator and nurses. Coordinated staff training, wrote the staff manual and created the camp web site. Ultimately responsible for all programs, activities, policies and procedures at camp. Guided camp through successful State of Maryland and American Camping Association accreditation visits. Previous positions held at Camp Greentop include:
Activities Coordinator, Summer 2000
Assistant Director, Summer 1999
Unit Leader, Summer 1995, Summer 1996
Cabin Leader, Summer 1992, Summer 1993, Summer 1994
Counselor, Summer 1991

THURMONT MIDDLE SCHOOL, Thurmont, MD
Special Education Teacher, August 1995–June 1997
Taught math and English to 7th and 8th grade students with learning disabilities and behavioral disorders. Created extra-curricular clubs for computer programming, chess, and American Sign Language. Co-wrote the Frederick County continuing education curriculum on basic and intermediate computer use in the classroom. Co-wrote Thurmont Middle School's technology grant proposal, resulting in the award of $50,000 for computer and networking hardware. Designed and taught seminars on Internet use for middle school teachers.

TEACHING CERTIFICATION

Maryland Educator Certificate (valid 7/1/1995 to 6/30/1998)
Educator ID 4464, Standard Professional I/Generic Special Education 1–8

CLUBS AND ACTIVITIES FOUNDED

University of Utah

Computer Science Department Weekly Grad Student Night, September 2013–present
Fridays, 6pm–10pm, Large Conference Room, Merrill Engineering Building. As of March 2014, sponsored and funded by the School of Computing.

My friend Andy Keep and I started these new groups at the University of Utah to encourage more interaction between students, and to help grow interest in programming languages research.

Indiana University

Indiana University Arduino Electronics Club, December 2010–November 2014
Fridays, 6pm–midnight, Lindley Hall 008

Indiana University 3D Printing Club, January 2011–November 2014
Sundays, 2pm–10pm, Lindley Hall 008

Created undergraduate Arduino/physical computing and 3D Printing/digital fabrication clubs at the request of former H211 students. Advised students on dozens of electronics and 3D printing projects. Supervised student assembly of three kit 3D printers and a 3D printed "EggBot" plotter. Developed Arduino projects for H211 and A290 courses.

OTHER COMPUTING-RELATED EDUCATIONAL ACTIVITIES

The CS Education Zoo
Computer Science education interview series, with co-host Steve Wolfman
`http://webyrd.net/zoo.html`

Volunteer, Indiana Celebration of Women in Computing (InWIC)
February 5/6, 2010 & February 11, 2012

Organizer, Teen Animatronics Club, July–August, 2010
Boys & Girls Club of Bloomington, IN

Volunteer, Indiana University Tech Week, February, 2010
Boys & Girls Club of Bloomington, IN

Organizer, TeenSIG Programming Club, October 2002–May 2003
Washington Apple Pi Users Group, Rockville, MD

Organizer, Programming Club, Spring 1997
Thurmont Middle School, Thurmont, MD

3D PRINTING DEMOS

Indiana University booth, SuperComputing '12, Salt Lake City, UT, November 10–14, 2012.

Arlington Heights Elementary School Science Night, Bloomington, IN, October 23, 2012.

Minority Engineering Advancement Program (MEAP), IUPUI, Indianapolis, IN, June 22, 2012.

Hoosier Guitar-Building Workshop, Bloomington, IN, May 12, 2012.

Farmer's Market, Bloomington, IN, April 21, 2012.

Pervasive Technology Institute, Indiana University, Bloomington, IN, April 5, 2012.

CATALYST, Informatics and Computing Student Association, Indiana University, Bloomington, IN, February 18, 2012.

Summit Elementary School Science Night, Bloomington, IN, February 10, 2012.

Technology Management Club, Kelley School of Business, Indiana University, Bloomington, IN, February 7, 2012.

Mechanical Engineering Technology (MET) Department, Purdue University, October 12, 2011.

3rd Grade Class, Cumberland Elementary School, West Lafayette, IN, October 12, 2011.

World Maker Faire, New York Hall of Science, Queens, NY, September 17–18, 2011

Bloomington Boys & Girls Club, Bloomington, IN, early 2011.

AWARDS

Benefitfocus.com "Medal of Honor"
Benefitfocus.com, 2001

Inaugural recipient of Benefitfocus.com's highest award, in recognition of exemplary work in porting the Benefitfocus application to the WebLogic J2EE application server. Award included a $3,000 bonus.

AppNet Excellence Award
AppNet, 2000
Awarded for developing, within a strict deadline, a complex XML Document Type Definition and BladeRunner configuration file that formed the foundation of a major document management system.

Outstanding Senior in Computer Science and Electrical Engineering
University of Maryland, Baltimore County, 1999
One of a handful of seniors to receive this award for "highest academic achievement of graduating seniors majoring in computer science and electrical engineering."

Director's Award for Outstanding Staff Person
Camp Greentop, 1993
Inaugural recipient of Camp Greentop's highest award, given to the staff member who "most epitomizes Camp Greentop and performs above and beyond all reasonable expectations."

PROFESSIONAL PROGRAMMING EXPERIENCE

Lyvegyde.com, Bloomington, IN/Garrett Park, MD
Co-founder and Lead Developer, August 2008–May 2009
Co-designed and developed *lyvegyde*, a guide to live online events. Responsible for site architecture (Ruby on Rails, AJAX, MySQL, nginx web server, Mongrel application server, Ubuntu Linux), middleware and backend development, page caching and other optimizations, and site deployment (using the Capistrano deployment tool).

Benefitfocus.com, Charleston, SC
Software Engineer, February 2001–June 2002
Collaborated with other software engineers to port a multi-million dollar online benefits enrollment application from the Enhydra Java application server to BEA's WebLogic Java 2 Enterprise Edition (J2EE) application server. Implemented the supplemental insurance, password authentication/account lockout and W4 subsystems of the application. Configured and tuned the production cluster of WebLogic J2EE application servers. Worked with other engineers to streamline and automate the application build process and integrate the JUnit Java unit testing framework into the application. Designed and implemented utility programs, including several generations of log parsers and a suite of Perl scripts to repair defects in Java Server Pages (JSP) and HTML source files.

Commerce One/AppNet/Century Computing, Laurel, MD
Software Engineer, August 1999–February 2001
Intern, October 1998–June 1999
Responsible for the XML-based form subsystem of ProcureZone.com, a multi-million dollar online procurement application for the construction industry. Designed and implemented custom software to validate the logic, layout and content of several thousand Extensible Form Description Language (XFDL) forms and trained subcontractors in the use of this validation software. Designed and implemented custom software to perform sophisticated transformations on over a million lines of XFDL code. Created the Document Type Definition for ProcureZone structured XML documents. Wrote a non-trivial context-free grammar for Interleaf's BladeRunner software to allow users to convert ProcureZone documents between Microsoft Word and XML formats.

Interactive Software Engineering (now Eiffel Software), Goleta, CA
Intern, Summer 1998
Worked with Bertrand Meyer, creator of the Eiffel programming language, on the implementation of

Simple Concurrent Object-Oriented Programming (SCOOP), an experimental concurrency mechanism for Eiffel. Presented an overview of SCOOP to a group of ISE's most valued customers at the Techniques of Object-Oriented Languages and Systems (TOOLS USA) conference.