Dustin Jamner

Boston, MA
jamner.d@husky.neu.edu (949) 573-4721
github.com/DIJamner

Education

Northeastern University. Boston, MA May 2020 (Expected)
College of Computer and Information Science
Candidate for Bachelor of Science in Computer Science
Minor in Mathematics
GPA/Honors: 3.8/4.0, Dean's Scholarship (half-tuition), Honors Program, Dean's List (all semesters)

Oregon Programming Languages Summer School. Eugene, OR July 2017
An intensive two-week lecture series on foundational concepts and research in programming languages

Experience

Northeastern University. Boston, MA.
Research Assistant (Full-Time) Summer 2016, January - July 2017, January - June 2019
Coauthored work on parametric polymorphism and gradual typing, including developing proofs of parametricity for gradually typed languages. Reworked and simplified a compositional compiler and proofs of its correctness.

Teaching Assistant, Logic and Computation January - April 2019
Led students' lab sections reviewing course material and supervised other teaching assistants.

Teaching Assistant, Programming Languages September - December 2018
Held office hours, graded homework assignments and exams, and proctored exams.

Tutor, Logic and Computation September 2016 - June 2017, September - December 2017
Led students' lab sections reviewing course material and supervised other teaching assistants. Created homework assignments and proofread the instructor's assignments.

The Charles Stark Draper Laboratory, Inc. Cambridge, MA.
Formal Methods Developer January - July 2018
Implemented a value-set static analysis for binaries (`https://github.com/draperlaboratory/cbat_tools`).
Proved a disassembly target language type-safe in the Coq theorem-prover.

Promenade Software. Irvine, CA.
Software Development Intern July - August 2014, July - August 2016
Implemented a Python scripting system within a web interface for medical devices in the Parlay software package (`https://promenadesoftware.com/parlaytm`).

Papers

1. Max New, Dustin Jamner, and Amal Ahmed.
   Graduality and Parametricity: Together Again for the First Time.
   To appear in *47th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL '20)*. New Orleans, Louisiana, United States. January 2020.

2. Chris Casinghino, Michael Dixon, Jt Paasch, Cody Roux, John Altidor and Dustin Jamner.
   Using Binary Analysis Frameworks: The Case for BAP and angr.
   To appear in *11th Annual NASA Formal Methods Symposium (NFM 2019)*. Houston, Texas, USA. May 2019.

3. Amal Ahmed, Dustin Jamner, Jeremy Siek, and Philip Wadler.
   Theorems for Free for Free: Parametricity With and Without Types.
   In *22nd ACM SIGPLAN International Conference on Functional Programming (ICFP '17)*, Oxford, UK, September 2017.

References provided upon request

Awards

Provost's Advanced Research/Creative Endeavor Award Northeastern University, May 2016
Abstracting Gradual Polymorphism – $2100

Summer Scholars Independent Research Fellowship Northeastern University, July - August 2019
Extension-Proof Compiler Verification – $4700

Invited Talks

Introduction to Category Theory Sage Hill School, January 2018
Guest Lecture, Advanced Topics in Mathematics
Presented an introductory lecture on category theory for students studying basic group theory.

Relational Parametricity for the Polymorphic Blame Calculus Northeastern University, June 2017
Northeastern University Programming Language Seminar
Presented research on proving parametricity for a gradually typed language with polymorphism.

Abstract Interpretation via Galois Connections Sage Hill School, March 2017
Guest Lecture, Advanced Topics in Mathematics
Presented Galois connections and their use in soundly approximating uncomputable properties.

Introduction to Constructive Logic and Type Theory Sage Hill School, March 2016
Guest Lecture, Advanced Topics in Mathematics
Presented introductory material on constructive logic and basic type theory.

Computer Knowledge

Programming Languages: *Proficient in:* Java, Haskell, Python, OCAML, Typed Racket, Racket, JavaScript/HTML/CSS, MIPS and NASM Assembly, Coq *familiar with:* ACL2, C, Rust, Go

References provided upon request