Talia Lily Ringer

http://tlringer.github.io/

EDUCATION

University of Washington *2015 – Present*
Ph.D. in Computer Science
M.S. in Computer Science *2017*
Advisor: Dan Grossman
Programming Languages & Software Engineering

University of Maryland, College Park *2008 – 2012*
B.S. in Mathematics and Computer Science
Advisor: Lawrence Washington
Honors Thesis: An Elliptic Curve Threshold Key Establishment Scheme

PUBLICATIONS

Talia Ringer, Karl Palmskog, Ilya Sergey, Milos Gligoric, and Zachary Tatlock.
QED at Large: A Survey of Engineering of Formally Verified Software.
Accepted with Minor Revisions to FTPL. Project website.

Talia Ringer, Nathaniel Yazdani, John Leo, and Dan Grossman.
Adapting Proof Automation to Adapt Proofs.
CPP 2018. Talk video, PUMPKIN PATCH tool repository.

Talia Ringer, Dan Grossman, Daniel Schwartz-Narbonne, and Serdar Tasiran.
A Solver-Aided Language for Test Input Generation.
OOPSLA 2017. Talk Video.

Talia Ringer, Dan Grossman, and Franziska Roesner.
AUDACIOUS: User-Driven Access Control with Unmodified Operating Systems.
CCS 2016. Talk Video.

UNDER SUBMISSION

Talia Ringer, Nathaniel Yazdani, John Leo, and Dan Grossman.
Ornaments for Proof Reuse in Coq.
Under Submission to ITP 2019. DEVOID tool repository.

RESEARCH VISION

My vision is a future of **verification** and **proof engineering** that makes interactive theorem provers accessible to any programmer. My research applies techniques from **programming by example**, **program evolution**, and **proof reuse** to a view of **proof automation** that considers how verification projects change over time.

CURRENT RESEARCH

Coq Change Analytics
with Alex-Sanchez Stern, Sorin Lerner, and Dan Grossman
` In recent years, verification efforts using interactive theorem provers like Coq have reached large, critical software projects. With this scale comes an interest in new proof engineering principles and technologies. In spite of this, there is little data on the development processes of proof engineers in the wild. We are developing and deploying a Coq plugin to a group of proof engineers that collects data on the changes that proof engineers make in development, then analyzing the data and using it to inform the next generation of proof automation tooling.

STUDENTS ADVISED

Jasper Hugunin.
Constructing Inductive-Inductive Types in Cubical Type Theory.
To appear at FOSSACS 2019.

HONORS & AWARDS

NSF GRFP Fellow *University of Washington*
Graduated with Honors in Computer Science *University of Maryland*
Graduation Speech Finalist *University of Maryland*
Corporate Scholar *University of Maryland*
Scholar Athlete *University of Maryland*

MENTORSHIP, DIVERSITY, & OUTREACH

UW CSE Care Committee *2019 – Present*
Lead organizer of a support network for graduate students in times of need.

Jewish Family Services *2017 – Present*
ESL tutor and friendly visitor for an elderly refugee.

UW CSE *2015 – Present*
Mentor for undergraduate women and graduate students in computer science.

UW QMP *2016 – Present*
Mentor for LGBT students from any major.

The Identity Function *2016 – Present*
Author of a blog interview series about LGBT computer science researchers.

TUNE House *2015 – 2016*
Mentor for undergraduate women in computer science.

Amazon *2012 – 2015*
Technical and career mentor for software engineers.

SERVICE

CAV Artifact Evaluation Committee	2019
CoqPL Program Committee	2019
POPL Artifact Evaluation Committee	2018, 2019
ITP Sub-Reviewer	2018
University of Washington Graduate Admissions Committee	2018

TEACHING

University of Washington *Fall 2018*
Teaching Assistant for Concepts of Programming Languages

University of Washington *Winter 2016*
Teaching Assistant for Compilers

University of Maryland, College Park *Spring 2012*
Teaching Assistant for Computer and Network Security

University of Maryland, College Park *2010*
Mathematics and Computer Science Tutor for Student-Athletes

INDUSTRY

Amazon *Summer 2016*
Research Scientist Intern
Worked with the Automated Reasoning Group on automatic test generation. Developed a solver-aided domain-specific language to generate test inputs.

Amazon *2012 – 2015*
Software Development Engineer
Worked with a team to develop the AmazonSupply website. Wrote and deployed code used company-wide and loaded hundreds of thousands of times per day. Developed a data flow analysis tool. Launched Amazon Business.

Amazon *Summer 2011*
Software Development Engineer Intern
Developed an internal web application to generate metadata for the AmazonSupply website in a safe and user-friendly manner. Enabled version control and staging for the metadata.

Carr Astronautics *2010 – 2011*
Corporate Scholars Program – Software Intern
Assisted in the development of a parallel image mosaicing application. Wrote code in C, MATLAB, and Java to read, alter, and write TIFF images with associated geographic data. Awarded a scholarship through the University of Maryland's Corporate Scholars Program.

INTERESTS

Other academic interests of mine include **domain-specific languages**, **program analysis**, **type systems**, **category theory**, **algebra**, **computer security**, and **cryptology**.

I enjoy writing **Coq plugins** and have implemented several tutorial plugins to help other plugin developers. I have also extended Rosette to handle strings.

My favorite programming languages are **Coq**, **OCaml**, and **Rosette**.

I compete for **Club Northwest**, a top distance running club. I served on the board of Club Northwest from 2015 to 2016. My role was to promote our top runners through social media and writing. I ran **NCAA Division I Cross-Country** in 2009.

I also enjoy **logic and number puzzles**, **chocolatiering**, **writing poetry**, **singing**, **studying Russian**, **making bagels**, and **composing music for the piano.**