BRADFORD LARSEN, COMPUTER SCIENTIST

brad.larsen@gmail.com
http://bradford-larsen.net

SKILLS

Compiler & interpreter development, software security, static program analysis, abstract interpretation, information visualization, continuous delivery, software as a service, multi-platform software development, general-purpose GPU programming, fuzz testing, property-based testing

EXPERIENCE

Senior Software Engineer **July 2017 – Present**
Lexumo Inc, Burlington MA

- Implemented an Amazon S3-based storage backend for a crawler of open-source software

Software Developer **January 2016 – July 2017**
Ab Initio Software, Lexington MA

- Forward-ported 30 internal modifications to CPython 2.5 onto CPython 2.7; ported CPython to z/OS
- Found and fixed a bug in SQLite 3.11+ that prevented it from working on EBCDIC-based systems
- Found and fixed over 100 memory corruption and undefined behavior errors in Ab Initio base code with the help of several C++ static analysis tools
- Improved developer productivity company-wide by implementing custom GDB extensions and custom pretty-printers for types in Ab Initio base code

Senior Software Engineer, Static Analysis **May 2012 – December 2015**
Veracode, Inc, Burlington MA

- Increased processing speed of the static security scanner — Veracode's flagship product — by 4x through improved job scheduling algorithms and coaching others on use of sampling profilers
- Saved more than 160 hours per week by overhauling our legacy QA infrastructure, implementing automated test failure triage and analysis, and improving build times
- Saved 20 hours per week in manual buffer overflow vulnerability validation by implementing an automated path-sensitive analysis postprocessing step using the Z3 SMT solver
- Enabled security rule authors to work more quickly and accurately by creating a linter for Veracode's domain-specific language for security rules; fixed 1000 existing errors using this linter
- Submitted 2 invention disclosures in areas of program analysis and machine learning-based runtime prediction of static analyses; both in patent process
- Identified dozens of latent defects in Veracode's flagship product by collecting metrics and analyzing them using R, ggplot2, and SQL; delegated repair of these defects
- Fixed 50 memory corruption errors by refining core data types — *interfaces really matter!*

Research Assistant **2010 – 2012**
Tufts University Computer Science, Medford MA

- Identified interpreter design decisions that have significant affect on runtime performance by implementing, testing, benchmarking, and profiling 924 interpreter variants for Lua in OCaml
- Formalized part of an idealized distributed revision control system using the Coq proof assistant
- Implemented an efficient binary decision diagram library in ANSI C, along with Haskell bindings

Research Assistant **2007 – 2010**
UNH Computer Science, Durham, NH

- Designed Barracuda, a statically typed, array-based language specifically for numeric problems
- Implemented an optimizing compiler for Barracuda, targeting GPUs via NVIDIA's CUDA; generated code was better than than NVIDIA's BLAS library for certain operations
- Designed and implemented Switchback, a shortest path algorithm that runs up to 10 times faster than earlier algorithms and requires no explicit heuristic function
- Improved processing time by 100x in space weather data conversion tools by reducing I/O
- Made software builds repeatable by replacing a legacy ad-hoc build system with GNU Autotools

Researcher **2007**
Friedrich-Alexander University, Erlangen, Germany

- Improved performance by a factor of 3 in a software model checker and a graph isomorphism checker by implementing object caching in a distributed shared-memory Java virtual machine

Performance Algorithms Group Intern **2005 – 2006**
Mercury Computers, Inc., Chelmsford, MA

- Fixed bugs and added new functions & tests to Mercury's Vector Signal Image Processing Library
- Maintained an Eclipse plugin for remote debugging of programs running on a Cell processor

EDUCATION

2010 – 2012	Ph.D. Computer Science (discontinued), Tufts University	3.70/4.00 GPA
2010	M.S. Computer Science, University of New Hampshire	3.57/4.00 GPA
2009	B.A. Philosophy, University of New Hampshire	3.73/4.00 GPA
2008	B.S. Computer Science, University of New Hampshire	3.72/4.00 GPA

PUBLICATIONS

Jan Midtgaard, Norman Ramsey, and Bradford Larsen. Engineering definitional interpreters. PPDP 2013. ACM, September 2013.

Bradford Larsen. Simple optimizations for an applicative array language for graphics processors. DAMP '11. ACM, January 2011.

Bradford Larsen, Ethan Burns, Wheeler Ruml, and Robert Holte. Searching without a heuristic: efficient use of abstraction. AAAI-10. AAAI Press, July 2010.

Ronald Veldema, Bradford Larsen, and Michael Philippsen. A DSM protocol aware of both thread migration and memory constraints. PDCS '08. ACTA Press, November 2008.

AWARDS AND HONORS

2010 – 2012	Dean's Fellowship, Tufts University, School of Engineering
2009	University of New Hampshire Barlow Prize (Honorable Mention)
2008 – 2009	NASA New Hampshire Space Grant Consortium Graduate Fellowship
2007	University of New Hampshire International Research Opportunities Program Fellow
2003	Eagle Scout, Boy Scouts of America