Curriculum Vitae

Daniel Donenfeld

277 Putnam Ave, Apt 3 ⋄ Cambridge MA, 02139
(917) 881-4701 ⋄ DanielDonenfeld@gmail.com

I am interested in compilers and abstractions for high performance computing. I have participated in research across disciplines and developed compilers in both academic and industry environments.

Education

Massachusetts Institute of Technology, *Cambridge, MA.* August 2020 - February 2023
Master of Science, Computer Science

Cornell University, College of Engineering, *Ithaca, NY.* August 2013 - May 2017
Bachelor of Science in Computer Science, *Summa Cum Laude* Overall GPA: 4.02
Dean's List

Professional Experience

Software Engineer II Fall 2017 - Fall 2020
Microsoft C++ Team

Worked on the Microsoft C++ compiler backend on language conformance, security, and optimization.

- Implemented the C++17 order of evaluation in the intermediate language reader to contribute to making the Microsoft C++ compiler C++17 conformant.
- Implemented compiler optimizations and improved existing optimizations. Examples include optimizing pointer subtractions, and improving an analysis on predicates in the compiler to better estimate known bits.
- Investigated and fixed bugs throughout the compiler.
- Facilitated the C++ backend release process for Visual Studio 16.1 and 16.2 by communicating deadlines and testing procedures, running validation testing, and publishing release notes

Research Experience

MIT Fall 2020 - Current
Research Assistant - Prof. Amarasinghe

Developed new abstractions within the Tensor Algebra Compiler (TACO) to support losslessly compressed tensors including variants of run length encoding and Lempel Ziv compression. We represent compression as a generalized form of sparsity with with repetitions of values represented by non-scalar, dynamic fill values. By unifying lossless compression with sparse tensor algebra, this technique is able to generate code that computes with both losslessly compressed data and sparse data, as well as generate code that computes directly on compressed data without needing to first decompress it.

Developed infrastructure to run the LLVM compiler in a distributed environment using serverless compute resources within AWS. Implemented support for distributing optimizations within the compiler. Developed techniques to effectively run global analysis using distributed resources.

Cornell Department of Computer Science Summer 2016 - Spring 2017
Research Assistant - Prof. Myers

Developed a new compiler backend for the Polyglot compiler to translate the Java language to LLVM IR. This project was created to run languages developed using Polyglot on devices with no JVM using the LLVM code generator, and to leverage the extensive suite of optimizations in LLVM. I implemented AST lowering compiler passes to remove syntactic sugar from the Java AST, implemented translations for many of the core Java language semantics to LLVM, and implemented runtime support for language features, including exception handling and threading.

University of Central Florida Center for Research in Computer Vision Summer 2015
Research Experience for Undergrads (REU) Participant - Prof. Shah

Developed and implemented a method for the localization of organs in 3D medical scans. The algorithm worked on a supervoxel segmentation to localize the heart, liver, and kidneys. Using a supervoxel segmentation reduced the search space significantly, and increased the perceptual meaning, as the supervoxels adhered to edges and were more uniform. The supervoxels were classified with handcrafted features using a random forest. The algorithm was effective at determining the locations of the organs and provided promising classification results.

Publications

Daniel Donenfeld, Stephen Chou, Saman Amarasinghe. *Unified Compilation for Lossless Compression and Sparse Computing*. Proceedings of the 2022 IEEE/ACM International Symposium on Code Generation and Optimization, 2022.

Willow Ahrens, **Daniel Donenfeld**, Fredrik Kjolstad, Saman Amarasinghe. *Looplets: A Language for Structured Coiteration*. Proceedings of the 2023 IEEE/ACM International Symposium on Code Generation and Optimization, 2023.

Teaching Experience

Teaching Assistant, 6.106: Performance Engineering of Software Systems Fall 2022
MIT

Led recitations on performance engineering which included lecturing and going over example problems. Held weekly office hours, graded assignments and exams.

Teaching Assistant, CS 3110: Data Structures and Functional Programming Fall 2015, 2016
Cornell University

Led recitations on data structures and functional programming in OCaml by going over example problems and answering questions on course material. Held weekly office hours and assisted in grading assignments and exams.

Course Staff, CS 1110: Introduction to Computing using Python Spring 2014 - Spring 2015
Cornell University

Assisted during labs by answering questions. Held office hours to answer any questions on course material. Graded assignments and exams.