jpriest@mit.edu
(210) 323 8821
github.com/JustAPerson

Jason Priest

For more information on my projects and experience, visit https://jpriest.me/

Education

Massachusetts Institute of Technology **2019 – 2020**
Masters of Engineering in EECS

Massachusetts Institute of Technology **2015 – 2019**
Bachelors of Engineering in EECS

- Technical GPA: 4.54
- Coursework
  - 6.828 (Operating Systems) 6.046 (Algorithms) 6.035 (Compilers)
  - 6.851 (Advanced Data Structures) 6.886 (Graph Algorithms)
  - 6.172 (Performance Engineering) 6.830 (Databases)
  - 6.UAT (Technical Communication) 6.033 (Computer Systems)
- SaveTFP – President of Student Community Service Club 2016 – Now
- Varsity Athlete – Men's Lightweight Crew Team 2015 – 2016

Experience

Broadway Technology **Summer 2018**
- Developed backend for new financial instrument on a platform connecting different types of financial institutions
- Profiled, benchmarked, and tested optimizations in C++ codebase

Ab Initio **Summer 2017**
- Added new frontend to data flow graph compiler
- Developed new optimizations within an existing compiler framework
- Developed, found bugs, and extended tests in large C++ codebase

MIT Lincoln Laboratory **Fall 2016**
- Performed research in reverse engineering and simulating PowerPC Linux wireless access point

Touchplan.io **Summer 2016**
- Implemented high-availability, online upgrading, and improved cooperative features for a Java and PostgreSQL backend server with Apache Mesos and ZooKeeper
- Helped introduce continuous delivery systems using Atlassian Pipelines
- Performed data analytics using PostgreSQL and Domo.com

Projects

RADS github.com/JustAPerson/rads **Rust 2017**
- Implementation of advanced data structures and algorithms for 6.851
- Vector with $O(\lg^2 n)$ insertion anywhere rather than O(n)
- Strict Fibonacci heap with non-amortized bounds
- Cache-oblivious sorting algorithm

MITscript **Rust 2017**
- Implementation of dynamic language for 6.035
- Mark-sweep GC and JIT compiler using LLVM

Denuos github.com/JustAPerson/denuos **Rust 2016**
- Toy x86-64 operating system learning experiment
- Basic virtual memory, interrupt, and syscall interface

LBI github.com/JustAPerson/lbi **Lua 2013**
- Basic implementation of the Lua virtual machine
- Accurately emulates nearly all valid Lua bytecode sequences

MODS github.com/JustAPerson/mods **Lua 2011**
- Assembler for Lua bytecode format
- Syntax permits labels and instruction-like macros

Skills

Programming Experience
- Languages: C, C++, Python, Java, PostgreSQL, Rust, Lua, x86 Assembly
- Tools: git, perforce, svn, make/cmake, gcc/clang, ld, gdb, vim/emacs, valgrind, perf
- Operating Systems: Windows, Mac OS X, Ubuntu, Arch Linux