

Vedant Kumar

vk@vedantk.com

Experience

- Apple, 2015-present, Compiler team (LLVM, Clang, and Swift development)
 - Improved optimized code debugging. I fixed bugs causing incomplete backtraces, incorrect line tables, or missing variable locations. I worked on two frontends, LLVM's target-independent passes, parts of the LLVM backend, and the debugger. I created tooling to isolate debug info quality regressions and mentored a [GSoC student](#) working with it.
 - Contributed new diagnostics and bugfixes to Undefined Behavior Sanitizer (UBSan), a widely-deployed run-time bug detection tool. I integrated UBSan into Xcode's build system and source editor, bringing UBSan to a large new audience. I presented this feature in a [WWDC '17 talk](#), contributed [documentation](#), and helped teams within Apple eliminate UB in their projects.
 - Maintained the source-based code coverage implementations in Swift and Clang. I improved the precision and quality of coverage reporting, [documented the feature](#), and set up continuous integration testing for it. Members of the LLVM community rely on [our bot's reports](#).
 - Helped stabilize several versions of Clang. These compilers built Apple's entire software stack (firmware, kernel, frameworks etc.) and were shipped to millions of developers.
- Apple, 2013-2014, Filesystems team (Summer Intern, 2x)
 - Worked on the XNU kernel, the CoreStorage data store, and the HFS+ filesystem.
 - Wrote randomized testers and LLDB extensions to debug race conditions in XNU.
 - Top-five finalist in the Apple-wide software engineering intern competition (2013).
- UC Berkeley, 2015, Programming Languages and Compilers (Teaching Assistant, [class website](#))
- UC Berkeley, 2014, Advanced Operating Systems (Teaching Assistant, [class website](#))
 - Co-taught a discussion section, wrote labs, and wrote exam questions.
 - Wrote an EDF scheduler, a small filesystem, and a device driver for Linux (in a team).
- HP Fortify, 2012, Static Analysis Team (Summer Intern)
 - Designed, implemented, tested, and shipped an interprocedural constant propagator.
- Personal projects (C++, Python), 2009-2014. All source code is available [here](#).
 - Data structures: skiplist, heaps, trie, B+ tree, kd-tree, open-addressed hash table
 - *quotient-filter*: compact approximate membership filter, supports merging and deletions
 - *53otron*: primitive Lisp \rightarrow LLVM compiler, used to vectorize equations to draw 3d shapes
 - *auto-diagonalize*: optimization pass which converts linearizable loops into $\Theta(\log n)$ processes ¹
 - Graphics: a fast ray-tracer (*radiate*), a model controlled with inverse kinematics (*spike*)
 - Networks: an [epoll-based server](#) (serves [vedantk.com](#)), a multi-threaded server

¹I demoed the optimization pass at the 2014 LLVM developer meeting: [slides](#), [longer writeup](#).

Education

- UC Berkeley (2015): B.S in Electrical and Computer Engineering.

At Berkeley, I served as a TA twice and participated in lightweight crew for a semester. I volunteered to teach several practical programming workshops, including a workshop on Unix tools and one on LLVM development. I took Advanced Operating Systems, Intro and Graduate CS Theory, Computer Architecture and Engineering, Discrete Math and Probability, Intro to Computer Graphics, and Intro to Computational Biology.

- duPont Manual High School (2011): Graduated from the MST program, a STEM-focused track.

At Manual, I participated in the science fair and was fortunate enough to compete at the international level (ISEF) twice. I was deeply inspired by the students I met and was honored to receive an award for my project.