SAMEERAN Joshi

# **Contact**

Salt Lake City, Utah, USA

[Joshisameeran17@gmail.com |](mailto:Joshisameeran17@gmail.com%20|)

<https://sameeranjoshi.github.io/sameeran.me//> | [www.linkedin.com/in/sameeran-joshi-b8b1b9144](http://www.linkedin.com/in/sameeran-joshi-b8b1b9144)

# **Research interest**

Compilers, Computer Architecture, Programming Languages, Compiler Optimizations, LLVM, Hardware-Software Codesign, Modern C++, HPC systems

# **Education**

**School of Computing, University of Utah**

PhD Student in Computer Science | Aug 2022 – Currently Enrolled

**Pune University, India**

*Bachelor’s In Computer Engineering | Aug 2015 – May 2019*

GPA: 8.29/10

# **Work experience**

|  | **Advanced Micro Devices (AMD)**  *CPU Compiler Engineer | June 2019 – Present*  **Software System Designer 2,** *April 2020 to Present*  **Parallel compilation in clang-driver**   * Parallelized the compilation phase for building applications with AOCC by adding compiler flag * Achieved huge gains in build times of large HPC applications like WRF from 10min to 1min and CAM4 from 38sec to 10 sec with AOCC clang driver   [**Binary Level Static Performance Analyzer Tool**](https://github.com/facebookincubator/BOLT)   * Extended [LLVM BOLT](https://github.com/facebookincubator/BOLT) to compare statically 2 binaries to report performance difference * Implemented python utilities for analyzing the data, plotting results and graphs to aid in reporting issues * Addedcount of basic blocks, vector, loops, scalars, loads & stores, spills & reloads, etc. to report issues based on various metrics * Reported performance issues and suggested missing optimizations in AOCC for SPEC CPU 2017, polybench and HPC workloads compared to ICC * Presented paper in internal conference (13% acceptance rate) at AMD   [**LLVM Flang Group**](https://github.com/llvm/llvm-project/tree/main/flang)   * 50+ commits to Fortran language compiler frontend in LLVM including new features, bug fixes, infrastructure changes * Added parsing and semantic support for OpenMP 4.5/5.0 and Fortran 2018 language features in LLVM Flang * Reviewed voluntarily OpenMP, OpenACC, Flang driver patches from community members   **Software System Designer 1,** *July 2019 to March 2020* |
| --- | --- |
|  | **Compiler Validation Group:**   * Implemented from scratch regression, unit tests for 12+ Fortran 2008 language standard to fuzz AOCC compiler * Reported Internal Compiler Errors, segmentation faults, mis compilations in AOCC Flang source * Focused on compiler validation, automation, CI/CD frameworks, debug testing to verify AOCC |

# **other PROJECTS**

|  | [**GCC - GNU Compiler Collection**](https://gcc.gnu.org/)  [Google Summer Of Code](https://summerofcode.withgoogle.com/archive/)  [*Extending Csmith for GCC C-Language Extensions*](https://github.com/Sameeranjoshi/csmith/tree/gcc-extensions)*, June 2018 – April 2019*  Mentor: [Andi Kleen](http://halobates.de/)   * Added ~15 GNU C language extensions to [Csmith](https://embed.cs.utah.edu/csmith/) and found unexplored bugs (ICE’s, seg faults, crashes) in GCC compiler * Found 12 critical bugs, 11 were fixed by GCC community * Increased the fuzzing code coverage of csmith on GCC by – line coverage: 5%, function coverage: 7%, branch coverage: 4% * Presented work at [Pune Kernel meetup](https://reserved-bit.com/2017/02/05/pune-kernel-meetup-the-beginning/) on work done in Csmith   [**Git Statistics Excel Generator**](https://github.com/Sameeranjoshi/git-log-stats)**,** *AMD, India*  Manager: Hariharan Parasuraman   * Designed small utility to collect, visualize, create reports of individual contributions in open-source projects for managerial audits |
| --- | --- |
|  |  |

# **AWARdS**

* TFWS Scholarship (awarded to 5% students of baccalaureate class) *2015-2019*
* AMD Spotlight award for performance *2020*

# **activities and interests**

* Volunteered at [CppOnSea’21](https://cpponsea.uk/), [CppCon’21](https://cppcon.org/)
* [2021 LLVM developers meeting](https://llvm.org/devmtg/2021-11/) PC member
* Co-founded [bitSimplify](https://bitsimplify.github.io/bitsimplify/index.html): RISC-V based LLVM toolchain startup