

SHRIRAM CHANDRAN

☎ +41-762069158 | ✉ shriramchandran253@gmail.com | 🌐 shriramch | 📍 Zürich, Switzerland
🌐 linkedin.com/in/shriram-ch | 🐙 github.com/shriramch

EDUCATION

ETH Zürich

Master of Science *ETH in Computer Science*

since Sep. 2022
Zürich, Switzerland

- **Focus:** Secure & Reliable Systems and Theoretical Computer Science.

Indian Institute of Technology Madras

B. Tech. *Computer Science and Engineering*

Jul. 2018 – Jun. 2022
Chennai, India

- **GPA:** 9.04/10
- **Thesis:** Program Analysis - StarPlat DSL: Grade 9/10

PATENTS AND PUBLICATIONS

M. Kabić, S. Chandran, G. Alonso. "Maximus: A Modular Accelerated Query Engine for Data Analytics on Heterogeneous Systems". *SIGMOD 2025*. (to appear).

Rupesh Nasre et al. "System and method for automatic parallel code generation for graph algorithms for multiple target architectures". *Patent 432922*. 14 November 2022. India

EXPERIENCE

Master Thesis with Prof. Torsten Hoefler

Higher-Order Graph Database Engine

since Oct. 2024
ETH Zürich

- Designing and implementing a **Higher-Order Graph Database Engine** with OLTP and OLAP capabilities.
- Designed algorithms to project higher-order graphs as plain graphs, performance models and proofs of correctness.
- Implemented a higher-order database layer with **CRUD** capabilities on top of **Neo4j**.
- Future plans include designing and implementing higher-order OLAP workloads and benchmarking.

Practical Work with Prof. Gustavo Alonso

GPU-Acceleration of the Maximus Database Engine

Oct. 2023 – July. 2024
ETH Zürich

- Implemented **GPU support** to Maximus, a distributed, scalable and heterogeneous database engine.
- Designed Arrow-compatible data structures to store databases on GPUs for GPU processing on various architectures.
- Integrated GPU-accelerated operators with the existing Maximus interfaces.
- Benchmarked Maximus-GPU on the complete TPC-H query set and demonstrated massive performance gains.

Bachelor Project with Prof. Rupesh Nasre

Program Analysis - StarPlat DSL

Jan. 2022 – Jun. 2022
IIT Madras

- Implemented compiler level optimizations for **StarPlat**, a **graph processing DSL**.
- Analyzed and implemented multiple optimizations for the DSL compiler using AST-level analyses.
- Conducted successful testing of the implemented optimizations on multiple large graphs across different algorithms.
- Produced positive performance gains for generated code.

Undergraduate Research with Prof. Raghavendra Rao

Stability Analysis

Aug. 2021 – Nov. 2021
IIT Madras

- Conducted a comprehensive literature review of previous research on **stability** of instances for various popular algorithms, in particular **Bilu-Linial** stability amongst other definitions.

- Analyzed stable instances of [Christofides' algorithm](#) under various stability conditions.
- Identified improved approximation bounds for stable instances of metric TSP.
- Developed new approximation algorithms for the vertex cover problem for almost-stable instances in random graphs.

Rubrik

May 2021 – Jul. 2021

Software Engineering Intern

Bangalore, India

- Developed a statistics system with the [Blobstore](#) team for analyzing backup metadata job performance.
- Designed a functional model for storing statistics within their respective databases.
- Enabled access to histograms and statistical data to aid in debugging and data analytics.
- Successfully tested the model on live clusters, demonstrating integrability with the main codebase.

Godot Media

Nov. 2020 – Jan. 2021

Software Engineering Intern

Bangalore, India

- Designed and constructed an automatic [plagiarism checker](#) to search text against documents on the internet.
- Implemented heuristics and parallelization to increase accuracy and reduce algorithm latency.
- Developed a graphical UI to accept data from a user and display results with highlighted matching text.
- Achieved highly accurate plagiarism detection on articles of up to [2000 words](#) in just a few seconds.

SKILLS

Programming: ([Proficient](#)) C/C++, Python, Scala, Haskell, ([Competent](#)) OCaml, Java, C#, RISC-V, x86, SQL, Cypher

Frameworks/Tools: Javascript, Django, HTML / CSS, \LaTeX , Git, Coq

TEACHING

Coordinator

Programming Club, [CFI](#) IIT Madras, Mar. 2019 – Jun. 2020

- Organized educational sessions on data structures and algorithms, fostering a culture of programming excellence in the institute. Organized regular collegiate programming contests to enhance students' programming abilities.

Content Developer

Melvano, Dec. 2018 – Mar. 2019

- Mentored JEE aspirants by creating Q&A content and addressing doubts through [Melvano](#), an indigenous startup building an educational social platform. Helped with the app's initial publicity with UI and design suggestions that enhanced its user appeal.

SELECTED COURSE PROJECTS

GPU-Acceleration of an Algorithm

Design of Parallel & High-Performance Computing, Sep. 2023 – Dec. 2023

- Implemented a banded matrix inversion algorithm, and performed parallelization using [MPI](#) and [CUDA](#) to improve the performance of the algorithm.

Building a Program Verifier

Program Verification, Apr. 2023 – Jun. 2023

- Designed and implemented a program verification system to translate programs from [EVT](#), an event-based research language, to [Viper](#), a verification backend language, allowing users to attach specifications with EVT programs.

System Level Optimization of an Algorithm

Advanced Systems Lab, Mar. 2023 – Jun. 2023

- Implemented the *PolyVest* approximation algorithm for volume estimation of a convex polytope and performed system optimizations including vectorization and cache performance improvement to get more than a [23x](#) speedup.

Approximation Algorithms for b-MST

Topics in Design & Analysis of Algorithms, Feb. 2021 – Jul. 2021

- Designed various approximation algorithms for the [b-MST](#) problem. Analyzed smoothed performance across multiple custom graph datasets with random perturbations.

Nand2Tetris

Computer System Design, Aug. 2019 – Nov. 2019

- Designed a Hack micro-architecture using the Hack Hardware Simulator on [Nand2Tetris](#) and implemented an assembler and a compiler for the Jack language.