

DARSHAN DINESH KUMAR

+1 (551) 362 9928 [✉ dd3888@nyu.edu](mailto:dd3888@nyu.edu) [🐙 Github](#) [🌐 Website](#) [🌐 LinkedIn](#) [📍 New York, USA](#)

EDUCATION

New York University (Courant Institute of Mathematical Sciences) Sept 2024 – May 2026
MS in Computer Science; **CGPA: 4.00/4.00**; Joann & Stanley Benson Master's Fellowship Recipient New York, USA
Concentration: Systems and Security; Advisor: [Prof. Sam Westrick](#); Research Lab: [ParCour](#)

PES University July 2018 – May 2022
B.Tech in Computer Science & Engineering (Honors); **CGPA: 9.54/10**; CNR Rao Merit Scholarship Recipient Bangalore, India
Specialization: Systems and Core Computing; Advisor: [Prof. N S Kumar](#)

EXPERIENCE

Samsung R&D Institute India - Bangalore July 2022 – Aug 2024
Platform & Software Research Team, 6G Lab | C, C++, SIMD, Vectorization, Compiler Passes, LLVM Bangalore, India
Engineer (July 2022 – March 2024); **Senior Engineer** (March 2024 – August 2024)

- Designed and developed the S/W stack for an end-to-end test bed that achieved **100 Gbps** in the data plane
- Implemented **SIMD** and **Vectorization C++ libraries** for Intel, ARM & AMD architectures, realizing **20%** speedup
- Developed custom **LLVM Compiler passes** for struct-reordering optimizations, achieving **10%** speedup and cache efficiency
- Received the **Spot Award** and **Super Tech Excellence Award** for impactful contributions

Samsung R&D Institute India - Bangalore Jan 2022 – July 2022
Research Intern, Vision Research Team | Python, Unreal Engine, Computer Vision & Graphics Bangalore, India

- Developed a **Photorealistic Data Engine** for generating diverse photorealistic synthetic data for Depth Estimation
- Supported data generation for various complex 3D Environments, forming the basis for Depth Estimation Models

Samsung R&D Institute India - Bangalore May 2021 – July 2021
Research Intern, IoT Platform & Connectivity Team | Android Studio, IoT, Smart Things Bangalore, India

- Designed a **Smart Things IoT App** solution to improve user experience
- Supported the identification and control of nearby Smart Devices as an overlay during a video call

PROJECTS

Master's Research | Advisor: [Prof. Sam Westrick](#) | [\[Thesis Report\]](#) [\[Paper\]](#) [\[Code\]](#) | (Jan 2025 - Present)

- Ongoing research on topics under *Elastic Scheduling, Work Stealing, Parallel Computing, Compilers & Programming Languages*
- Thesis: Performance Analysis and Debugging of Parallel Programs within Lightweight Scheduling Frameworks**
- Open Source contribution to ParlayLib, a C++ library for developing efficient Parallel Algorithms: [PR Link](#)

Lock-Free RSM Database | Advisor: [Prof. Dennis Shasha](#) | *Badger, Go, Databases, Distributed Systems* | (Sept 2025 - Present)

- Ongoing Research on designing and implementing a Lock-Free Replicated State Machine Database
- Proposes a highly parallel transaction design without a single point of failure, but with the same guarantees of an RSM

Parallel Relational Database | *C++, Parallel Algorithms, Databases* | [\[Details\]](#) [\[Report\]](#) [\[Presentation\]](#) [\[Code\]](#)

- Designed and implemented a Parallel Relational Database from scratch, supporting typical CRUD operations
- Implemented B-Tree based indexing with **20x** speedup for Select queries, parallel merge sort with 2.2x speedup for Order By queries, & various aggregate queries using parallel primitives such as filter, tabulate, & reduce, achieving up to **8.3x** speedup

Deque Data Structure Library for GPUs | *CUDA, GPU, Concurrent Data Structures* | [\[Report\]](#) [\[Presentation\]](#) [\[Code\]](#)

- Implemented 3 variants, global lock-based, global lock-free (atomic counter), & a shared memory-based deque for GPUs
- Built functional & performance benchmarks; achieved up to **36%** speedup over the open-source stdgpu deque

Automated Parallelization | *C++, Pthreads, OpenMP* | [\[Details\]](#) [\[Video Demo\]](#) [\[Report\]](#) [\[Presentation\]](#) [\[Code\]](#)

- Aimed at automatically converting a sequential program to its parallel equivalent to ensure efficient hardware usage
- Implemented **Intra-Function Parallelism** using program comprehension and **Inter-Function Parallelism** using a novel thread scheduling algorithm, achieving a substantial **performance gain of up to 500x** for large data

Memory Allocator for OpenMP Programs | *C++, OpenMP, Python* | [\[Details\]](#) [\[Report\]](#) [\[Presentation\]](#) [\[Code\]](#)

- Implemented a **scalable, concurrent memory allocator** with per-thread heaps & memory ownership for OpenMP
- Achieved **substantial performance gains** as compared to Hoard & Malloc, with super-linear speedup for Scalability tests; showing best performance for Active & Passive False Sharing tests, & for Fragmentation tests by a factor of 10000

Generic Programming in C | *C, Design Patterns* | [\[Details\]](#) [\[Video Demo\]](#) [\[Report\]](#) [\[Readme\]](#) [\[Code\]](#)

- Implemented Generic Programming features as a design pattern in C, using preprocessor directives
- Implemented **Generic Containers** - list, vector, hashmap, stack, queue; **Iterators** for each of these containers; and **Generic Algorithms** - find, find_if, count, count_if, min, max, and accumulate that make use of the Iterators

Mini-Compiler for Python | *Lex, Yacc, C++, Python, Compiler Design* | [\[Details\]](#) [\[Video Demo\]](#) [\[Report\]](#) [\[Code\]](#)

- Implemented a mini compiler for the Python programming language
- Implemented the **compiler phases** - lexical, syntax, & semantic analyses, intermediate code generation & optimization
- Supported various programming constructs like if, if-else, for; different operators; error detection & handling mechanisms

SKILLS

Programming Languages: C, C++, Python, Java, Bash, SQL, Racket, OCaml, Go, Elixir

Systems & Parallel Computing: CUDA, OpenMP, MPI, SIMD/Vectorization, LLVM (custom Optimization Passes)

Tools & Platforms: Git/GitHub, Linux, Docker, Kubernetes, Jenkins, AWS, Android Studio, Unreal Engine, LaTeX

Other Technical Skills: *Programming & Paradigms:* OOP, Functional Programming, Generic Programming, Design Patterns

Software Engineering: Software Engineering, System Design, Performance Engineering, SDLC, Unit Testing, Agile/Scrum, Open Source Contributions, CI/CD Pipelines, DevOps Practices, Microservices Architecture, REST APIs

Systems & Computing: Parallel Computing, Programming Languages, GPU Programming, Compiler Construction & Optimizations, Data Structures & Algorithms, Database Systems, Operating Systems, Distributed Systems, Embedded Systems

Applications: Android App Development, Computer Graphics

Industry tools: Jira, Confluence

Soft Skills: Teamwork & Collaboration, Effective Communication, Problem Solving, Critical Thinking, Research acumen

PUBLICATIONS AND PATENTS

Scheduler Augmentation: A Lightweight, Customizable, Low-Cost Profiling Technique for Fork-Join Parallel Programs **SPAA 2026**

Sam Westrick*, Darshan Dinesh Kumar*, and Seong-Heon Jung*

[\[Paper\]](#)

ACM Symposium on Parallelism in Algorithms and Architectures (SPAA), July 2026, London, UK

* Equal Contribution

Method and System for AI Compute Networking in Virtual RAN **2024**

Patent - Provisional & Complete Specifications Filed, App. No. 202341032551

Method and System for Efficient TB Preparation in Real-Time **2024**

Patent - Provisional & Complete Specifications Filed, App. No. 202341037879

Method and System for Efficient Memory Management During HARQ **2024**

Patent - Provisional & Complete Specifications Filed, App. No. 202341037877