Siddharth Nayak

siddharth97nayak@gmail.com | siddharth1297.github.io | linkedin.com/in/siddharth1297 | github.com/siddharth1297 | **New Delhi**

Education

Indraprastha Institute of Information Technology, Delhi — Aug. 2022 – June 2024
M.Tech in Computer Science and Engineering — *CGPA: 9.0/10*

Institute of Technical Education and Research, Bhubaneswar — Aug. 2015 – May 2019
B.Tech in Computer Science and Engineering — *CGPA: 9.3/10*

Skills

Areas of Interest: Operating Systems, Networking, Languages and Runtimes
Languages: C/C++, Go, Python, Java, CPython, Shell Scripting, JavaScript, SQL, P4, Dafny
Tools: Git/GitHub, gdb, Valgrind, clang-tools, Docker, Kubernetes, eBPF
Frameworks: LLVM, DPDK, Django, Flask, C++ QT, gRPC
Databases: MySQL, Redis
Cloud Platforms: AWS, GCP

Experience

Open Futures, New Delhi | *Software Developer* — Aug. 2019 – Sep. 2021

Designed and delivered micro-second scale features and trading algorithms for in-house low-latency trading system using C++ and Python.

- **Increased profit potential by 10%** for *high-frequency automated arbitrage trading* algorithms by revamping trade execution algorithms (in C++ and Python) in collaboration with a team of 2.
- **Reduced app startup time to *1/3rd*** by porting sequential C++ code to *multithreaded* code.
- Independently, built a web-based *real-time* risk monitoring system that **slashed traders' decision-making time by 95%** using Django, WebSocket, and Redis. Wrote *asynchronous* Python HTTP and WebSocket clients for multiple crypto exchanges (**Full ownership**).

Projects

Serialization Performance Optimisation | *Systems Programming* — May 2023 – May 2024

- Led a group of 4 to design and implement a new serialization library that **reduced serialization latency by 1/6th** by leveraging *Linux scatter-gather I/O* in a microservices environment.

Fault Tolerant Distributed Key-Value Store | *Distributed Systems* — March 2024

- Built a distributed key-value store from scratch using **Python and gRPC**, deployed over **Google Cloud Platform**. It achieved a **significantly low latency of sub-1ms for reads and sub-150ms for writes**, utilising the *Raft consensus algorithm and leader-lease technique.*

Kanva: Lock Free Search | *Concurrent Data Structures* — Jan. 2023 – May 2023

- Significantly extended Kanva, a non-blocking linearizable learned lock-free search data structure written in C++, by implementing a *linearizable lock-free range search* that offers a **throughput of 12MOPS/128 cores**, using a memory-efficient *constant-time snapshot algorithm.*

Argolib: A Parallel Runtime | *Parallel Programming* — Sept. 2022 – Dec. 2022

- Developed a *Fork-Join style parallel programming library and runtime* for C/C++ programs, offering a variety of work-stealing scheduling algorithms. Additionally, **minimised runtime performance overhead up to 30%** by implementing *trace and replay* mechanism.

SafeC | *Compilers* — Sept. 2022 – Dec. 2022

- **Enhanced memory safety of C programs** by writing an *LLVM* pass to catch null pointer access and an **automatic memory manager** with a **conservative garbage collector** using the *mark-and-sweep* algorithm.

Publication

Learned Lock-free Search Data Structures [preprint]
Gaurav Bhardwaj, Bapi Chatterjee, Abhinav Sharma, Sathya Peri, and **Siddharth Nayak**
To appear in 53rd International Conference on Parallel Processing – 2024 (ICPP '24)

Relevant Courses/Certifications

Compilers, Parallel Runtimes for Modern Processors, Concurrent and Learned Data Structures, Programmable Networking, Distributed Systems, Systems for AI