

# Siddharth Nayak

[siddharth22128@iiitd.ac.in](mailto:siddharth22128@iiitd.ac.in) | [linkedin.com/in/siddharth1297](https://www.linkedin.com/in/siddharth1297) | [github.com/siddharth1297](https://github.com/siddharth1297) | New Delhi

## Education

**Indraprastha Institute of Information Technology, Delhi**

*M.Tech in Computer Science and Engineering*

Aug. 2022 – May 2024 (Expected)

*CGPA: 8.63/10 (Till 3<sup>rd</sup> Semester)*

**Institute of Technical Education and Research, Bhubaneswar**

*B.Tech in Computer Science and Engineering*

Aug. 2015 – May 2019

*CGPA: 9.3/10*

## Skills

**Areas of Interest:** Operating Systems, Networking, Cloud Computing, Backend Engineering

**Languages:** C/C++, Python, Go, Java, CPython, HTML/CSS, JavaScript, JQuery, Ajax, P4

**Tools:** Git/GitHub, Shell Scripting, gdb, LLVM, DPDK, Docker, Kubernetes

**Frameworks:** Django, Flask, C++ QT

**Databases:** PostgreSQL, Redis

## Experience

**Open Futures, New Delhi** | *Software Developer*

Aug. 2019 – Sep. 2021

*Responsible for adding and maintaining features to in-house low-latency trading system and implementing micro-second scale trading algorithms in C++.*

- Developed and implemented trade execution algorithms for micro-second scale automated trading strategies.
- Reduced app startup time to  $1/3^{\text{rd}}$  by porting sequential C++ code to *multithreaded* code.
- Built a web-based *real-time* risk monitoring system using Django, WebSocket, and Redis. Wrote *asynchronous* Python HTTP and WebSocket clients for multiple crypto exchanges (Full ownership).

**Centroxy, Bhubaneswar** | *Software Engineer Intern*

June 2017 – Aug. 2017

- Developed Front-end and REST API client libraries for Python (Flask) application for Open source software Gluu.

## Projects

**Serialization Performance Optimisation | (Systems Programming)** | *Guide: Dr. Rinku Shah*

May 2023 – Present

- As part of my M.Tech thesis, I am improving the application's end-to-end TCP and UDP network communication performance by reducing the serialization library overheads. Leveraging advanced Linux I/O techniques such as *scatter-gather*, and *zero-copy* to improve the *latency*, *CPU utilization*, and *memory consumption* in a microservice architecture.

**Kanva: Lock Free Search | (Concurrent Data Structure)** | *Guide: Dr. Bapi Chatterjee*

Jan. 2023 – May 2023

- Implemented a *strong consistent(Linearizable)* lock-free range search using a memory efficient constant-time snapshot algorithm for **Kanva**, a *Non-blocking Linearizable learned lock-free* search data structure, with *multicore scalability* and *progress guarantee*.

**Argolib: A Parallel Runtime | (Parallel Programming)** | *Guide: Dr. Vivek Kumar*

Sept. 2022 – Dec. 2022

- Developed a *Fork-Join style parallel programming library and runtime* for C/C++ programs using **Argobots** threading library. Experimented *multicore scalability* of different *work-stealing* algorithms. Implemented *trace and replay* mechanisms for minimizing runtime performance overheads. Also, implemented *dynamic concurrency throttling* for energy efficiency.

**SafeC | (Compiler)** | *Guide: Dr. Piyus Kedia*

Sept. 2022 – Dec. 2022

- Implemented *data flow analysis* using *LLVM* for a subset of C programs to avoid NULL pointer access. Also, implemented a *conservative garbage collection* 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**

*Under review at 50th International Conference on Very Large Databases (VLDB) - 2024*

## Relevant Courses

Compilers, Parallel Runtimes for Modern Processors, Concurrent and Learned Data Structures, Programmable Networking, Decision Procedures

## Certification

Machine Learning, Coursera