Last Updated: Dec 19th 2025

Parth Parikh

parthparikh1999p@gmail.com • github.com/pncnmnp • (+1) (984) 810-9749

Education

North Carolina State University - Masters in Computer Science (GPA: 4/4) *Aug 2021 - May 2023*

University of Mumbai - Bachelor in Computer Engineering (GPA: 9.19/10) *Aug 2017 - June 2021*

Technical Experience

Software Engineer – Microsoft *Redmond, WA | Jan 2025 – Present*

- Member of the Azure Performance Team in Azure Compute's Core OS group.
- Architected an automated performance pipeline for Windows Server Azure Edition guest VMs to detect CPU, storage, and networking regressions. Owned Windows Server guest performance sign-off for the past year.
- Investigated an Arm64 networking regression with up to 500% throughput variance on a single node; root-caused it to a Windows kernel inefficiency and drove mitigation. Also root-caused remote write workload failures on newer x64 hardware to an NVMe driver issue.
- Validated performance for an OpenHCL release and built automation to measure container startup times for OSGuard and AzLinux 3.0 across key container images on AKS.

Software Engineer - Neurelo *Remote | July 2023 - Jan 2025*

- Enhanced Neurelo's Rust-based query engine by consolidating multiple queries into a single read query and optimizing the read API performance for improved efficiency across multiple databases (Postgres, MySQL, and MongoDB).
- Designed and implemented an AI-driven mock data generator pipeline, intelligently simulating data across diverse databases with comprehensive schema specifications.
- Researched and engineered AI models for text-to-SQL, schema generation, and REST/GQL API generation, integrating them into Neurelo's product suite.
- Contributed to the development of Neurelo's performance and integration testing pipeline tailored for the query engine.

Production Engineering Intern - Meta *Bellevue, WA | May 2022 - Aug 2022*

- Built performance measuring and stress testing tools at Meta's *Warm Storage team.*
- Used Fio to benchmark Meta's clusters with different kinds of workloads to test performance. The benchmarks involved designing the IO behavior of the file systems - like tuning the block size, file size, total IO, bandwidth rate limit, the ratio of reads/writes, number of threads, etc.
- Architected a tool to perform A/B testing on OS metrics such as IOPS, memory bandwidth, CPU bandwidth, and network latency. Integrated this tool into Meta's CI/CD pipeline to gain routine information on such metrics and catch regressions.

Research Assistant - NC State's Theoretical Computer Science Lab *Raleigh, NC | Aug 2021 - May 2023*
Advisor: Prof. Donald Sheehy

- Analyzed a new metric tree data structure called *Greedy Tree* that utilizes the farthest point sampling (or greedy permutations) for its incremental construction. This research focused on the intersection of *Computational Geometry* and *Data Structures.*
- Surveyed the landscape of heuristics and algorithmic ideas that have been employed in Hausdorff distance algorithms.

MLH Fellow - Major League Hacking *Remote | Sept 2020 - Dec 2020*

Research Intern - Indian Institute of Information Technology, Allahabad *Remote | May 2020 - July 2020*

- Designed models to predict and classify emotions of all the passages in popular books from Project Gutenberg's collection.

Publications

Proximity Search in the Greedy Tree SIAM Symposium on Simplicity in Algorithms, 2023

Greedy Permutations and Finite Voronoi Diagrams Symposium on Computational Geometry, 2023

Linear-time Approximate Hausdorff Distance The 30th Fall Workshop on Computational Geometry, 2022

Spectral Bloom Filters for Client Side Search The 11th IEEE Annual IEMCON, 2020

Projects and Personal Research

Author of "The Secret World of Data Structures and Algorithms" *Jan 2023 - Present*

- Book on lesser-known data structures and algorithms, with in-depth explanations drawn from research papers.
- Topics covered include KHyperLogLog for privacy risk estimation, Odd Sketches for space-efficient Jaccard similarity estimation, GADDAG for optimally generating Scrabble moves, a survey on Bitmap compression techniques, and Timing Wheels for low-resolution software timers.

Last Updated: Dec 19th 2025

Shimmey *Oct 2022*

- Designed an end-to-end privacy model for *link shimming* using *Private Information Retrieval* (*PIR*).
- Engineered a 2-server PIR architecture with $O(n^{0.5})$ communication cost and observed a **96.2% decrease in runtime cost** over the state-of-the-art technique that relied upon Homomorphic Encryption.

Reversing the 20 Questions Game *Sept 2021 - Nov 2021*

- Engineered a transformer-based boolean question-answering model wherein the model chooses an entity at random and the human aims to guess this entity by asking natural language queries with an **accuracy of 78.7%**.

LuaNLP - Natural Language Processing Library for Lua *Feb 2021 - April 2021*

- Currently, it stands as **one of the most extensive native Lua libraries for statistical NLP**, featuring 14 implemented modules encompassing tokenization, lemmatization, stemming, parts-of-speech tagging, sentiment analysis, keyword extraction, named-entity recognition, and text summarization.

Sthir - Spectral Bloom Filters for Client-Side Search *June 2020 - Oct 2020*

- Pioneered a memory-efficient client-side search library utilizing *Spectral Bloom Filters*, delivering search rankings comparable to Lunr.js while **reducing memory usage by 85%**.

Detecting air pollution hotspots and identifying their source trajectories *Jan 2020 - Feb 2020*

- Architected two models using satellite data from ERA5 and Sentinel-5P and submitted them to the *Indian Space Research Organization* for predicting ground pollutant concentration using a geographically weighted regression model.

Crossword Solver to solve mini New York Times crosswords *Dec 2019*

- Probabilistically solved mini New York Times crosswords (**in under 2 minutes**) by guessing clues and positioning them on the grid (an NP-complete problem) using the Z3 Theorem Prover.
- This research was used by Rumshisky et al. (UMass Lowell) to introduce crossword-solving as a new NLP benchmark.

Branch Target Buffer *April 2019*

- Designed and implemented a *branch prediction buffer* and a *branch target buffer* using C, incorporating a 2-bit saturating counter and an LRU replacement policy to improve performance.

Technical Skills

Programming Languages	C, C#, Python, Bash, PowerShell (current); Rust, C++, Lua, JavaScript (prior)
Relevant Courses	Parallel Systems, Compiler Construction, Operating Systems, Databases, Graph Theory, Natural Language Processing, Privacy in the Digital Age, Software Engineering, Computer Networks, Computer Graphics
Operating Systems	Familiar with Windows NT and Linux internals

Teaching

- Maintaining a technical blog for the past six years on research and engineering topics in systems, data structures, databases, NLP, and computing history.
- Teaching Assistant for NC State's CSC 442 - *Introduction to Data Science* course during Fall '22.
- Guest lectured on the topic of Support Vector Machines in my undergraduate Machine Learning class.