

Chetan Goenka

Berkeley, CA | 512-363-2290 | cgoenka@berkeley.edu | cgoenka.me | linkedin.com/in/cgoenka | github.com/c-goenka

EDUCATION

University of California, Berkeley

Berkeley, CA

Master of Science in Electrical Engineering and Computer Science

May 2026

Bachelor of Arts in Computer Science – GPA: 3.90/4.0, Graduated with Distinction

2024

Relevant Coursework: Machine Learning, Artificial Intelligence, Efficient Algorithms, Data Structures, Human-AI Interaction, Computational Models of Language, Computer Architecture, Internet Architecture & Protocols

TECHNICAL SKILLS

Programming Languages: Python, JavaScript/TypeScript, Java, SQL, R, C/C++, HTML/CSS, RISC-V

Frameworks & Tools: PyTorch, TensorFlow, scikit-learn, Pandas, NumPy, React, Next.js, Git, OpenAI API, Tailwind

EXPERIENCE

Graduate Research Engineer

June 2024 – Present

UC Berkeley Electrical Engineering and Computer Science

Berkeley, CA

- Built AI-powered code explanation system using React, CodeMirror, and OpenAI API, implementing 3 hierarchical explanation modes (line, block, data-flow) with 700+ lines of custom editor extensions
- Engineered custom CodeMirror editor extensions to render interactive DOM-based overlays and widgets, utilizing Zod schemas to enforce structured metadata extraction and type safety for dynamic line mapping
- Driven product iteration cycles by integrating user feedback into feature development, deploying function-level summarization tools that improved code comprehension for developer workflows and problem framing

Undergraduate Course Staff

Aug. 2023 – May 2024

UC Berkeley College of Computing, Data Science, and Society - Data 100

Berkeley, CA

- Debugged Python and SQL implementations for 25+ students weekly, systematically resolving logic errors in pandas, NumPy, matplotlib, scikit-learn, and SQL across data wrangling, visualization, and ML projects
- Evaluated 5,000+ programming assignments over 2 semesters, delivering structured feedback on code quality, data pipeline efficiency, and statistical implementation using custom grading scripts and rubrics
- Conducted 4 hours of weekly office hours and supported discussion forum, creating FAQ documentation and debugging resources to resolve common challenges in DataFrame operations, data cleaning, and visualization

Data Science & Engineering Intern

June 2022 – Aug. 2022

Oracle Corporation

Austin, TX

- Programmed an automated ML scoring pipeline in Python and SQL to rank enterprise prospects across 4 products, replacing manual account prioritization with algorithmic predictive modeling for 500+ prospects
- Deployed robust ETL scripts to ingest and process marketing campaign data, programmatically calculating attribution and win rates for high-value strategic accounts using Pandas and NumPy
- Optimized data querying workflows by translating complex business requirements into efficient SQL logic, enabling executive teams to visualize revenue attribution and conversion metrics to guide targeting strategy

PROJECTS

Memory Bottle - TEI 2026 Student Design Challenge | *Python, C++, Flask*

Aug. 2025 – Present

- Engineered a distributed IoT architecture for multi-node audio synchronization, implementing a custom WiFi file transfer protocol and reliable USB serial communication between heterogeneous hardware nodes
- Architected embedded state machine on ESP32 controlling 5 sensors (microphone, color sensor, tilt, capacitive touch, potentiometer) with real-time feedback via NeoPixel LEDs, managing SD card I/O for local audio buffering
- Developed a Flask-based control server to manage the data pipeline, parsing color metadata to synchronize high-fidelity Pygame audio playback with Arduino LED arrays for immersive user interaction and responsiveness

Python Compiler | *Python, Java, RISC-V, JFlex, CUP*

Jan. 2024 – May 2024

- Built full-stack compiler in Java for statically-typed Python dialect, implementing JFlex lexer with stack-based indentation handling, CUP parser generating type-annotated ASTs, and 4-pass semantic analyzer performing class hierarchy resolution, symbol table construction, and type inference across 172 test cases
- Designed RISC-V code generator with optimization pipeline applying constant propagation, dead code elimination, and register allocation, reducing instruction count by 40% through dataflow analysis and algebraic simplification
- Developed Spring Boot web application providing compiler IDE with real-time visualization of compilation phases, enabling users to inspect tokens, AST structure, semantic errors, and optimized assembly output