



SOUMENDRA GANGULY

📍 Gøteborg Allé 6, 7.52, 8200 Aarhus, Denmark

✉️ soumendraganguly@gmail.com

☎️ +45 50 36 19 83

🌐 soumendraganguly.com

🌐 linkedin.com/in/soumendraganguly

🔗 github.com/8vasu

📄 arXiv.org

CORE SKILLS

- C (systems, POSIX)
- Python (core, NumPy, SymPy)
- HTML, CSS, JavaScript (ES)
- Bash, POSIX shell
- Go (distributed systems)
- Rust (systems, WebAssembly)
- Git, Docker, GitHub Actions
- Linux, FreeBSD (sys/net admin)
- GNU Emacs Lisp, Lua
- \LaTeX , PGF/TikZ

CORE CONTRIBUTIONS

- Official Python stdlib
8vasu.me/cpython-pty
- Official FreeBSD C stdlib
8vasu.me/freebsd-winsize
8vasu.me/freebsd-winsize-man
- Official Linux, BSD *script(1)*
8vasu.me/util-linux-script
8vasu.me/freebsd-script
8vasu.me/netbsd-script
- Official SymPy
8vasu.me/sympy-matrix

PACKAGES

- Python (PyPI)
 - pypi.org/project/stty
 - github.com/8vasu/stty.py
- Lua \TeX package on CTAN
 - ctan.org/pkg/fretplot
 - github.com/8vasu/fretplot
 - github.com/8vasu/fretplot-mcp
- GNU Emacs Lisp
 - github.com/8vasu/compuTeX
 - github.com/8vasu/2windows.el

AI/ML, DATA SCIENCE

- Geometric Graph Neural Networks
github.com/8vasu/gnn.py
- Local text-to-image generation
github.com/8vasu/sg-diffusion
- Time series electricity pricing
github.com/8vasu/power-opsd-dk1

PARSERS, COMPILERS, DSL

- Lisp interpreter in Rust
github.com/8vasu/best
- WebAssembly regex visualizer
8vasu.me/regex
github.com/8vasu/wasm-regex-tree
- WebAssembly (Pyodide) CAS
8vasu.me/plotcat

WEB (FULL STACK)

- Geometric GNNs: Flask back end
github.com/8vasu/gnn.py
- Cmdline interface for arXiv.org
github.com/8vasu/paper.py

PROFILE

Software engineer and mathematician with **11 years production experience in C and Python**, contributing to CPython (180M+ downloads/month), FreeBSD (1M+ servers), and util-linux (ships with every major Linux distribution).

PROFESSIONAL EXPERIENCE

Software Engineer II, Uber Denmark

Jan 2026–Mar 2026

- Worked on the scheduler component of **Odin**, Uber's distributed stateful platform, contributing to core infrastructure for large-scale workload scheduling:
 - **Reduced runtime** of a prediction pipeline by **13%**.
 - Set up **staging and production runtime environments** for a platform service with full documentation.
 - Designed and implemented **scheduling and synchronization algorithms** for distributed platform components.
 - Leveraged **AI-assisted development** extensively using Claude Code with custom **skills** and **MCP server integration**. Currently exploring **parallel independent Claude Code sessions across Git worktrees** for simultaneous multi-feature development, as well as **Claude Code subagents and agent teams** for coordinated multi-agent workflows.
 - Used GitHub Copilot for real-time code suggestions and boilerplate generation, significantly accelerating development speed.
 - **Stack:** Go, Bazel, Starlark, gRPC, Protocol Buffers, **Cadence**, **Fx**, **Grail/YQL**, **M3**, Redis, Grafana, Sourcegraph, Claude Code, GitHub Copilot, *tmux(1)*

Researcher, Aarhus University

Sep 2023–Dec 2025 (fixed-term contract)

- Led a team of 9 for a year to organize a complex, multi-stakeholder project: a 2-week international conference and summer school with close to 100 participants, including 30 speakers. Was responsible for end-to-end delivery:
 - **Cut conference costs by 47%** (from 840,000 DKK to 450,000 DKK) through vendor negotiations and logistics improvement.
 - Managed daily catering, accommodation of guests, reimbursement of transportation costs, excursions, and social events.
 - Wrote the conference website conferences.au.dk/aaf1, implementing responsive design.
- Directed a multi-year, self-driven project, decomposing complex objectives into structured milestones, and adapted the methodology with innovative solutions to **guarantee timely delivery and verifiable results**.
- Presented complex technical findings to expert and non-expert audiences, **adapting communication for diverse stakeholder groups**.
- Taught a master's math course (spring 2025), leveraging 9 years of teaching 60–100 undergraduate students/semester (4.5/5 mean rating).

Software Engineer, Open Source and Independent Projects

2014–

- Contributed low-level terminal control functionality to **major projects used by millions worldwide**:
 - Authored 7 functions for the *os*, *termios*, *tty*, and *pty* modules of the **Python standard library** with comprehensive **unit tests** across **20 pull requests over 5+ years**, collaborating asynchronously with CPython core developers across **US/Europe time zones** via GitHub PRs and code reviews.
 - Implemented 2 **POSIX** terminal control functions for the **FreeBSD standard C library** with full **manual page documentation** following BSD conventions.
 - Improved *script(1)* and *scriptreplay(1)* utilities across **util-linux**, **FreeBSD**, and **NetBSD**, demonstrating **cross-platform** compatibility.
 - Published package *stty.py* to **PyPI** via automated **GitHub Actions CI/CD workflows**, achieving **100% test coverage** before release, for robust, Pythonic *stty(1)*-style terminal manipulation.
 - **Stack:** C, Python, POSIX/BSD/GNU termios and pseudoterminal APIs, Git, GNU Autotools, *nroff(1)*, GitHub Actions
- Developed research-grade **image processing** and **data analysis** tools using **deep learning** techniques:
 - Trained **GPU-accelerated, Euclidean and non-Euclidean geometry-aware Graph Neural Networks** (GNNs) and packaged as a **web app with a dashboard** providing interactive visualizations of graph embeddings.
 - Engineered **local deep learning text-to-image** generation pipeline, **optimizing inference performance**.

LANGUAGES

- English, Hindi, Bangla ■■■■■■
- Danish (DU3, module 4) ■■■■

OTHER SKILLS, EXPERIENCES

- Guitar, breakdance, fine arts
- History, linguistics enthusiast
- Skydiving, paragliding experience

- Wrote *tailor.py*, an AI-powered CV tailoring tool: given a job description, generates a tailored \LaTeX CV with custom profile, reordered sections, and keyword-injected bullets designed to pass ATS screening. Uses the **OpenAI API** with curated prompts, **spaCy NLP** for keyword extraction, and **ESCO**, **Lightcast (from Databricks)**, and **O*NET** occupational databases for phrase injection. Automatically **bolds action verbs**, maintains correct tense, and filters out cliché phrases.
- Built a **multi-agent pipeline** in Python that converted **Jenkinsfile CI** definitions to **GitHub Actions workflow YAML** using an **Anthropic SDK**-powered converter agent that produced or repaired YAML based on feedback from a reviewer agent that was capable of tool usage, including *actionlint* and a deterministic transpiler written in Groovy which served as a source of truth. Performed ephemeral prompt caching of system prompts for speed and cost-effectiveness. Generated detailed HTML reports of the convert-review loop using a **Jinja2** HTML template for observability.
- Built simple **time series forecasting system** for Danish electricity pricing, performed **quadratic programming-based optimization** for cost minimization.
- **Stack:** PyTorch, NumPy, Matplotlib, pandas, PostgreSQL, TimescaleDB, CVXPY, Stable Diffusion, NVIDIA CUDA, cuDNN, Docker, Flask, Flask-SocketIO, HTML, CSS, JavaScript, OpenAI API, spaCy, Anthropic SDK, Jinja2, PyYAML, Groovy, Gradle, SnakeYAML, *actionlint*
- Designed **grammar, parser, and compilers for languages**, performed **lexical analysis**, **AST construction**, **semantic validation**, and vector graphics generation:
 - Crafted **EBNF (extended Backus-Naur form) grammar and parser** for \LaTeX matrix and complex number expressions for the **official SymPy project**, wrote **150+ test cases** covering edge cases, plus GNU Emacs interface *compuTeX* for interactive development.
 - Architected *fretplot* Lua \TeX package implementing **domain-specific language** called **fretplot** for automatic plotting of guitar scale diagrams based on scale formulae. Published on **CTAN** and **GitHub** with parser and compiler for meta-language translation. Package featured in [La Lettre GUTenberg](#). Wrote MCP (Model Context Protocol) server in Go with capability to generate fretplot code from natural language descriptions of scales and chord voicings.
 - Designed and implemented *best*, a **new statically typed Lisp**, defining its grammar in **POSIX *lex(1)*** and *yacc(1)*, a **LALR(1) parser**, a pre-evaluation **static type-checker** pass, and a **tree-walking evaluator** in Rust, bridging the generated C parser to the Rust runtime via **FFI**.
 - Devised *wasm-regex-tree*, a web browser-based **visualizer and SVG generator** for **syntactic and semantic tree** representations of Rust regular expressions. Wrote in Rust, compiled to WebAssembly.
 - Adapted a **WebAssembly/Emscripten (Pyodide)**-based Python interpreter into a web browser-based interactive command-line **computer algebra system** using *xterm.js* and developed a **line editor** interface.
 - **Stack:** Python, SymPy, Lark, ANTLR 4, Lua, \LaTeX , PGF/TikZ, Go, official Go SDK for MCP, Rust, *bison(1) --yacc*, *flex(1) --posix*, *wasm-pack* (WebAssembly), HTML, CSS, JavaScript, Pyodide, xterm.js

EDUCATION

2019–2023
PhD, Mathematics
Texas A&M University
USA

2017–2019
MS, Mathematical Sciences
Clemson University
USA

MSc, Mathematics, 2014–2016
BSc, Math and Computer Science, 2011–2014
Chennai Mathematical Institute
India