Ayaan Afridi

Schmalkalden, Germany | ayaanafridi97@gmail.com | linkedin.com/in/ayaan-afridi | udp.cx

Education

Hochschule Schmalkalden — Schmalkalden, Germany
B.Eng. Mechatronics and Robotics — *Apr. 2026 - 2029 (expected)*

Benchmark College — Islamabad, Pakistan
Cambridge International A Levels (Computer Science, Mathematics, Physics) — *2023 - 2025*

Experience

Freelance Python Developer, Algorithmic Trading — Apr. 2022 - Present
Fiverr and direct clients — *Remote*

- Delivered 30+ client trading systems covering strategy backtesting, live execution, and portfolio management.
- Automated order handling across crypto exchanges through CCXT and webhooks, with risk controls including dynamic position sizing, stop-loss rules, rebalancing, and drawdown limits.
- Built dashboards for positions, trade history, and performance metrics used for day-to-day monitoring.

Technical Projects

VFOB: Very Fast Order Book | C++23, CMake — Write-up

- Built a header-only, order-by-order (L3) limit order book that processes up to 98.5M real Binance updates per second on one core, at 13-16 ns median and under 120 ns p99 latency per message.
- Took throughput from an 8.3M ops/s std::map baseline to 87M with pooled 24-byte nodes, tagged open-addressing hash tables, and an indexed 4-ary heap, A/B testing every change on a pinned core.
- Rejected designs on measured data (a sorted price ladder fell to 3.4M ops/s on deep real books) and verified correctness with randomized differential tests against a brute-force model.

Pyngine: Chess Engine | Python, UCI, Tkinter — Write-up

- Wrote a dependency-free engine with legal move generation, alpha-beta search, a transposition table, quiescence search, and an exact king-and-pawn endgame bitbase; perft is exact through depth 6 (119M nodes).
- Made move generation 7x faster and cut search nodes by 55-87% through selective pruning. Against Stockfish limited to 1320 Elo it scored 9 wins, 3 draws, and 8 losses. Covered by 161 automated tests.

Veyra: Desktop AI Assistant | C#, .NET 10, WPF, Docker — Write-up

- Built a Windows assistant that captures a chosen window, transcribes both sides of a live call in real time over WebSockets, and streams answers into a transparent overlay.
- Started drafting answers after 0.7 s of settled speech, added correlation-based echo reduction, and ran generated code in a locked-down Docker sandbox (no network, 256 MB, 10 s limit).

Koma Comics | Kotlin, Jetpack Compose, Room — Google Play

- Published a native Android comic reader with 600+ Google Play downloads, featuring reading-progress tracking and OPDS library integration, with unit and emulator tests run in GitHub Actions.

AHL API SDK | Python, REST, pytest, GitHub Actions

- Reverse engineered a brokerage's Android app into an installable Python SDK for portfolio access, PSX market data, and orders, with 147 tests (87% coverage) across Windows, Linux, and three Python versions.

Skills

Programming: Python, C++ (C++23), C#, Kotlin; foundational TypeScript and C
Performance: Benchmarking, profiling, latency percentiles, A/B testing, cache-friendly data structures
Algorithms and Trading: Order books, market data replay, game-tree search, backtesting, risk controls
Tools: Git, Docker, CMake, GitHub Actions, pytest, Gradle, REST and WebSocket APIs, CCXT
Security: CompTIA Security+, eJPT; preparing for OSCP
Robotics: Introductory ROS2 concepts including nodes, topics, publishers, and subscribers
Languages: English (C1), German (A2) | **Interests:** Chess, poker