Hang Li

Sydney, Australia hang.dev@outlook.com 0493738013 hang-li ImHangLi

Education

University of New South Wales *Feb 2024 – Mar 2027*
BS in Computer Science

- Overall WAM: 88.5/100
- **Coursework:** Computer Systems, Data Structures and Algorithms, Software Engineering

Experience

Intern – Agentic Solutions *Remote*
Nethermind *Jan 2025 – Now*

- Engaged in the design and development of a decentralized open-sourced audit platform for AI agents, defining the proof-of-concept and crafting detailed user stories to capture stakeholder requirements.
- Implemented robust backend systems and integrated smart contracts to drive secure, autonomous audits.
- Developed granular access control mechanisms and versioning systems to ensure data integrity and traceability.
- Tech stack: Solidity, Hardhat, Typescript, Python

Undergraduate Research Scholar *Sydney, Australia*
Sydney Quantum Academy *Nov 2024 - Now*

- Awarded the Undergraduate Research Scholarship by SQA and working under Distinguished Prof. Mingsheng Ying.
- Researched formal verification of quantum programs using Quantum Hoare Logic with classical variables.
- Implemented a verification tool Quave using Lean 4 to automate formal proofs of quantum program correctness.
- Tech stack: Lean, Formal Verification

Projects

Verkle Tree Implementation *2024*

- Implemented a primal efficient blockchain data structure with IPA based polynomial commitment in rust, offering efficient proofs, reduced proof sizes, and improved scalability for distributed systems.
- Tools used: Rust

Real-Time Quiz Platform *2024*

- Developed a real-time quiz platform backend that enables quiz management for admins and user participation. Implemented CI/CD pipelines for automated testing and deployment.
- Tools used: Node.js, TypeScript, Express.js, Vercel KV, Git

File Synchronization Tool 2024

- Developed an rsync-inspired utility that optimizes data transfer by transmitting only changes. Implemented efficient metadata processing, block comparison, and file reconstruction for synchronization.
- Tools used: C

Technologies

Languages: Typescript, Rust, Solidity, Python, Java, C

Technologies: CI/CD, Docker, Git, Test-Driven Development