

Andre Zhang

azhang4@utexas.edu | 832-306-4246 | [linkedin.com/in/andre-zhang-91bb88234](https://www.linkedin.com/in/andre-zhang-91bb88234) | github.com/An-Fruit

Education

The University of Texas at Austin, GPA: 3.95
Bachelor of Science, Computer Science

May 2027

Relevant Coursework: Multivariable Calculus, Discrete Math, Probability & Statistics, Linear Algebra, Differential Equations, Algorithms, Data Structures, Computer Architecture, Operating Systems, Computer Graphics, Energy Efficient Computing

Work Experience

HP Personal Systems AI/ML Intern (June - August 2024)

- Created a text-to-action LLM agent using python and python frameworks.
 - Utilized the Langchain framework to provide consistent interface for structured output/tool calling
 - Utilized Chainlit framework to handle application logic and provide GUI
 - Utilized MS Graphs + ChromaDB to both enable Retrieval Augmented Generation (RAG) and provide a way for LLM Agent tool-calling to have effects on a cloud service.

Skills

Technical Skills:

Languages: Java, Python, C/C++, ARM64, SWI-Prolog, HTML/CSS, TypeScript

Frameworks/Libraries: Chainlit, Langchain, Microsoft Graphs, ChromaDB, Pandas, OpenGL.

Tools: Git, Make, CMake

Languages: Chinese (Native/Bilingual), English (Native/Bilingual)

Certifications:

- Adobe Certified Professional**
 - Web Authoring using Adobe Dreamweaver 2019
- Microsoft Office Specialist**
 - Microsoft Word (Office 2016)
 - Microsoft PowerPoint (Office 2016)

Research Experience

UTD ALPS Lab Research Internship (June - August 2022)

- Utilized FOLD-R++ and FOLD-RM Algorithms to analyze datasets in order to explore automated reasoning
- Used Prolog to build logic programs modeling legislation in order to explore structural similarities between logical computing and the linguistic structure of legal documents

Projects

AArch64 CPU Simulator

- Implemented a simulation of a CPU that could run a limited portion of the ARM64 ISA
 - Includes simulated L1 cache w/ adjustable cache line size/associativity
 - 5 stage pipelining process to help with instruction processing throughput
 - Includes hazard handling/control to decrease unused CPU cycles

PintOS

- Extended the PintOS kernel to support user-level programs, adding system call handling and process isolation.
- Implemented virtual memory with demand paging, supplemental page tables, and stack growth for dynamic user processes
- Built a basic file system with support for file creation, deletion, reading/writing, and file descriptor management

3D Interactive Fluid Simulation

- Built a 3D interactive fluid simulation using TypeScript and WebGL, implementing the PIC/FLIP method for an optimal combination of stability and detail.
 - Used a staggered MAC grid and solved the Poisson pressure equation with Gauss-Seidel relaxation to enforce incompressibility.
- Allowed for user to apply forces and move the fluid using the mouse

Whitted-Style Ray Tracer

- Developed a Whitted-style recursive ray tracer in C++ with support for reflections, refractions, and shadows.
- Implemented supersampling techniques (both stochastic and regular) to improve image quality and reduce aliasing.
- Added support for textures, normal mapping, and a cube mapping for realistic surface detail and lighting effects.
- Optimized performance using a Bounding Volume Hierarchy (BVH) and adaptive ray termination to accelerate ray-scene intersection.