

# Paris Zhou

Software Developer  
14123 NW Lakeview Dr. Portland, OR. 97229

P: 503-806-7269  
E: zhou.paris00@gmail.com

## Professional Summary

Software Developer with a strong foundation in computer science and proficiency in a diverse range of programming languages including Python, Java, and C++; skilled in creating secure, testable, and maintainable code, and experienced in utilizing automation frameworks and software-testing methodologies. Adept at collaborating in team environments to solve complex problems efficiently, with a vision to leverage innovative solutions in software-development projects.

## Skills

**Python** (Experienced),  
**C** (Skillful),  
**Git** (Skillful),  
**C++** (Skillful),  
**SQL** (Skillful),  
**Kali** (Novice),  
**Jira** (Beginner),  
**Java** (Beginner),  
**Slack** (Beginner),  
**MASM** (Beginner),  
**React** (Beginner),  
**Node.js** (Beginner),  
**VSCode**  
(Experienced),  
**JavaScript** (Beginner),  
**Wireshark** (Novice),  
**TCP Dump** (Novice),  
**Data Analysis**  
(Beginner),  
**Cybersecurity**  
(Novice),  
**Machine Learning**  
(Novice),  
**Software Testing**  
(Skillful),  
**API Development**  
(Skillful),  
**Agile Methodology**  
(Skillful),  
**Technical Writing**  
(Expert),  
**Software**  
**Architecture**  
(Experienced),

## Links

github.com/zhouparris  
linkedin.com/in/paris  
-y-zhou

## Employment History

### Medical Scribe

*OHSU, Beaverton* | Jun 2022 – Jan 2024

- Documented 100+ patient encounters per week in a digital-record system, maintaining 99% accuracy and attention to detail under time constraints.
- Collaborated with physicians in real time to produce thorough medical reports, demonstrating strong communication skills and quick adaptability to new software tools.

## Education

### Bachelor of Science in Computer Science - GPA: 3.60

*Oregon State University, Corvallis* | Jan 2023 – Jun 2025

Relevant Coursework: Computer Architecture & Assembly Language · Web Development · Introduction to Databases · Computer Networks · Operating Systems I · Software Engineering II · Networking & Security

### Bachelor of Science in Biology, Minor in Chemistry - GPA: 3.91

*Oregon State University, Corvallis* | Sept 2018 – 2021

Graduated Summa Cum Laude,

## Additional Information

### Artificial Life Simulation (Senior Capstone)

The **A-Life Challenge Simulation** is an artificial life ecosystem I'm developing with teammates in **Python** to study the emergence of complex behaviors from simple biological principles. Using **NumPy** for vectorized computations and **PyGame** for real-time visualization, the simulation models a dynamic environment where virtual organisms evolve, interact, and adapt to ecological pressures. Designed to enable large-scale parallel computation, supporting thousands of simultaneous agents within the ecosystem.

Organisms are defined by a robust genomic structure—including **morphological, metabolic, reproductive, and locomotion traits**—that dictate their survival strategies. Evolutionary adaptation is driven by controlled **probabilistic mutation**, resulting in **genetic drift and speciation** over time. I implemented efficient spatial indexing with cKDTree for rapid neighbor detection and built a **lineage tracker** to map parent-child relationships and visualize evolutionary paths. This project highlights my expertise in **Python, genetic modeling, and high-performance computing**.

## Database-Driven Website

- Created a **Node.js/Express** website powered by a **MySQL database** for the mock business 'AwesomeCarDealer'.
- Designed a fully normalized MySQL schema splitting composite keys to streamline warranty handling.
- Authored DDL scripts with auto-increment keys, precise data types, and foreign-key constraints to preserve referential integrity during deletes and updates .
- Integrated **dynamic lookup endpoints** for front-end dropdowns (e.g., fetching customers by name, vehicle VINs), eliminating manual ID entry and improving UX .
- Compiled a polished PDF report featuring an executive summary of schema iterations, updated ERD/schema diagrams, and annotated **CRUD** screenshots

## Small Shell (Command-Line Interpreter)

- **Custom Unix Shell** (zpshe): Designed and implemented a command-line interpreter in C with support for parsing up to 2,048-character inputs and 512 arguments, handling comments/blank lines, and executing both built-in (*exit*, *cd*, *status*) and external commands via *fork()*, *exec()*, and *waitpid()*.
- **I/O Redirection & Job Control**: Enabled `</>` redirection, background execution with `&`, `/dev/null` fallbacks, and real-time notifications of background job completions.
- **Signal Management**: Integrated robust signal handling to ignore SIGINT in the shell, deliver SIGINT to foreground child processes, and toggle “foreground-only” mode on SIGTSTP (Ctrl-Z) with user-friendly status messages.

## Campus Network Security Architecture

- Ongoing Threat Detection: Designed a multi-layer defense blueprint for all campuses, blending network segmentation, DDoS mitigation, and controlled research/partner connectivity to thwart unauthorized scans and intrusions.
- Designed an intelligent monitoring framework featuring selective SSL/TLS decryption, finely tuned IDS/IPS rules, honeypots, and a centralized SIEM to surface covert channels and beacon traffic.
- Strong Access Controls: Designed two-step login for all users, staff education on avoiding common scams, protected stored passwords, and active session monitoring to shut down any takeovers.
- Clear Documentation & Risk Plan: Created easy-to-follow network maps, an executive summary, and a risk-acceptance chart that explains why each security measure was chosen.

## Data Structures & Algorithms Collection

- Developed a comprehensive GitHub collection of Python programs showcasing implementations of core data structures (linked lists, stacks, queues, binary search trees, heaps, graphs) and algorithms (DFS, BFS, merge sort, quick sort, Dijkstra’s shortest path), demonstrating deep understanding of algorithmic problem-solving and complexity trade-offs. Authored thorough unit tests and clear documentation to ensure code robustness, readability, and maintainability, adhering to industry best practices in software design and version control.