Irene Wang

(248) 324-8139 • irenewang@rice.edu • https://irene-a-w.github.io/irene-w/

EDUCATION

Rice University, *Houston, TX.* May 2025
Bachelor of Arts in Computer Science
GPA: 3.67 / 4.0
Relevant Coursework: Computational Thinking, Algorithmic Thinking, Introduction to Programming Design, Reasoning About Algorithms, Principles of Parallel Programming, Introduction to Computer Systems

TECHNICAL SKILLS

Languages: Python, Java, C, SQL, HTML/CSS
Frameworks/Tools: Git, JUnit, NumPy, pandas, React, Linux, Snowflake, VS Code, IntelliJ, PyCharm,

EXPERIENCE

IT Intern | Caterpillar | Peoria, IL May 2023—Present

- Develop Python packages that create a GUI framework that connects and communicates with Snowflake
- Utilize object oriented programming to create a Python GUI that allows users to retrieve and modify data by automatically sending SQL queries to database
- Enhance database view by rewriting SQL queries to reduce the search time on a given database object
- Design test cases to ensure accuracy of Snowflake ETL pipeline and generate reports identifying discrepancies

Research Assistant | Treangen Lab, Rice University | Houston, TX May 2022—August 2022

- Reduced the memory consumption by 40% of the program Emu, a relative abundance estimator for 16S genomic sequences, by enhancing the data structures of the program through utilizing NumPy
- Improved the runtime by over 60% for creating a custom lineage database by removing data storage bottlenecks caused by Pandas Dataframes
- Developed scripts that parsed SILVA specific files, allowing Emu to be compatible with the SILVA database
- Documented the pre-existing codebase to ensure maintainability and used Git to collaborate with lab members

PROJECTS

Dynamic Memory Allocator | C March 2023

- Implemented a dynamic memory allocator for C programs that aims to utilize the allocated heap space through effectively allocating, freeing, and reallocating memory
- Employed a circular doubly linked linked list of segregated free list to efficiently store data and organize and track heap space as it is made available
- Designed a heap checker that scans heap to ensure there are no inconsistencies and all available memory is accurately tracked by the appropriate segregated free list

Feedback & Evaluation via Automated Tests | *Java* September 2022—November 2022

- Developed a program that automatically generates a set of test cases used to test student implementation of assigned Python functions and outputs results
- Implemented a recursive tree-like structure with nodes representing non-iterable and iterable Python objects that contains the necessary information to generate the corresponding Python object
- Employed the hitting set greedy algorithm to find a concise set of test cases that will cover a maximal amount of incorrect implementations
- Parsed JSON files describing specifications to build template for generating base set of test cases
- Utilized JUnit to create and execute testing suites to ensure accuracy of code

Star Cat | *Python, PyGame* August 2022—September 2022

- Employed object oriented programming to develop a platform game
- Formulated a point system that allows player to interact with randomly generated stars to collect points
- Configured game physics to have gravity, allowing for fluid movement of the player's character from platform to platform and triggering game ending when the player falls off the screen