Zibai (Matthew) Wang

zw737@cornell.edu (preferred contact) | (+1) 607-279-1958
https://github.com/Masasukam | https://www.linkedin.com/in/matthew-wang-9847331b7/

EDUCATION

Cornell University, **Ithaca, New York** Jan 2024 - Dec 2024
Master of Engineering: Computer Science; Flexible in Relocation
Relevant Course: Parallel Computing, Computer Vision, 3D Reconstruction, Computer Networks, Operating Systems

University of British Columbia, Vancouver, BC
Bachelor of Science: Computer Science And Mathematics; With Distinction Sep 2019 - May 2023
Relevant Course: Test-Driven Development, Applied AI & ML, Databases, Data Structures & Algorithms, Object-Oriented Programming, System design, Linux System, Linear Algebra

SKILLS

Languages	Java, C, C++, Python, C#, JavaScript, TypeScript, Kotlin, PHP, HTML/CSS, R, Bash
DataBase & Cloud	MySQL, MongoDB, Oracle, NoSQL, DynamoDB, AWS (Cloud Practitioner)
Tools/Libraries	PyTorch, TensorFlow, Cuda, MPI, OpenMP, Flask, React, Spring, Node.js, Scrapy, Git, REST API

WORK EXPERIENCES

Software Research & Development Intern Sep 2021 - April 2022
INTEL Corporation, Vancouver, BC

- Developed and implemented ***C***-based optimization settings for quality-speed tradeoffs during low-delay streaming and video compression in the widely-used ***SVT-AV1*** encoder, decreased ~10% bitrate loss and increased ~8% speed.
- Developed testing scripts to run on ***AWS EC2 Linux*** instances using ***Python*** for evaluating bitrate/speed tradeoffs for existing ***SVT-AV1*** features; Collaborated with teams to perform comparative analysis among video encoders in the market using data obtained from tests; addressed performance and stability issues through debugging.
- Designed and implemented an optimized video decoder program using ***C***. Simplified the 5 decoding levels to a more maintainable 2-level system by evaluating the decoder speed against existing solutions in the market. Implemented unit tests using ***Check framework*** , achieved test coverage of 95%+ and packaged the program using ***CMake***.

Software Developer Intern July 2019 - Aug 2019
Tencent Holdings Ltd, **Shenzhen, China**

- Built an event-driven notification system using ***Python*** and ***Flask framework*** to keep track of keywords and feedback given by users on stock forums.
- Extracted dynamically generated content from ***JavaScript***-based stock forums by integrating ***Python Scrapy*** library with ***Splash***, enabling server-side ***JavaScript*** execution and rendering for full ***HTML*** access. Utilized dynamic ***IP***s and controlled crawling rate to avoid throttling.
- Persisted users post data into ***MySQL*** databases consisting of >5G user data for further analysis and relational database management. Crafted schema to encapsulate essential attributes and employed strategic indexing on crucial attributes that are frequently used in search for efficient data retrieval.

PROJECTS

Java-Based Clinic Management System Sep 2019 - Dec 2019

- Designed and implemented a full-stack clinic management system using ***Java***, ***MySQL*** and ***Spring framework*** to help clinic managers to better track patient appointments.
- Utilized the ***Spring RestController*** to build ***Restful APIs*** for patient data retrieval and update. Created ***GUI*** using Java ***Swing Framework*** for user interaction and employed ***Maven*** for dependency management.
- Applied ***OOP principles*** and ***Observer pattern*** in order to make the connections between the objects of doctors and patients more transparent while avoiding tight coupling between the two objects.

Domain-Specific Language May 2022 - Jun 2022

- Designed and implemented a full-stack website generator using ***Java*** and ***JavaScript*** for users to design, generate, and maintain their own webpages with human-familiar language.
- Utilized ***Antlr*** library to tokenize and parse user inputs according to grammar, then implemented dynamic webpage rendering using ***React*** framework. Utilized ***Axios*** to asynchronously fetch remote resources through ***HTTP*** requests.