ZHEJIAN JIN

(917) 609-4100 | New York, 11101, NY | [Linkedin](#) | [Github](#) | [Website](#)

EDUCATION

Columbia University, New York, NY
M.S. in Electrical Engineering Dec 2022
- Coursework: Design using C++, Database System Implementation, Cloud Computing, Distributed Storage Systems, Blockchain, Heterogeneous Computing, Practical Deep Learning Systems Performance, Big Data Analytics, Stream Processing

Shanghai Jiao Tong University, Shanghai, CN
B.S. in Electrical and Computer Engineering , Honor Student Aug 2021
- Coursework: Data Structures and Algorithms, Operating System, Computer Networks, Artificial Intelligence.

WORK EXPERIENCE

Jupiter Research Capital (Minghong Investment), New York, NY Jun 2022 - Sep 2022 & Apr 2023 - now
Quantitative C++ Developer Intern
- Independently built a real-time low-latency asynchronous **streaming data system** and **GUI** from brokers/exchanges for **trading activity monitoring**, real-time risk calculation, alerting and other features using C++, thread-safe data structures, QT and QML.
- Built **colocated servers** and **PB-level storage systems** in the data center, set up **system monitors** using Grafana and Netdata.
- Independently wrote **memory-optimized multi-core** and **multi-thread** C++ tools and python scripts dealing with PB-level exchange tick data, including sequence-level **UDP network Pcap integrators**, **data feed parsers** and **searching index**, covering exchange multicast data feed products including China A-shares, Nasdaq, NYSE, CBOE, CTA and UTP.
- Built a low-latency and high-throughput asynchronous **FIX protocol parser** with database insertion supporting 30k+ inserts/second both for live trading messages and historical log files from **DropCopy**.
- Accelerated matrix multiplication on **CPU** by using **SIMD** instructions and also on **GPU** by using common libraries.

Columbia University, New York, NY Jan 2022 - May 2022
Course Assistant, CSEE4121 Computer Systems for Data Science
- Designed programming projects using Apache Spark, Spark Streaming, HDFS in Google Cloud Dataproc. ([link](#))
- Held office hours weekly and answered questions on Piazza daily to answer students' questions from homework and class content, took teaching-team meetings and graded homework with other 9 CAs for the class with 300+ students.

East Money Information Co., Ltd., Shanghai, CN Dec 2019 - Mar 2020
Back-End Software Developer Intern
- Implemented network framework with **Python** to develop websites for large-scale market data in **MongoDB** visualization.
- Utilized **Apache Kafka** to maintain the distributed publish-subscribe messaging system.
- Developed the Data Quality Detection System in **Golang** with alarming function.

SELECTED PROJECTS

Columbia University, Design Using C++ by Bjarne Stroustrup, New York, NY Sep 2022 - Dec 2022
Simple Order Matching Engine
- Finished a **parser** parsing Nasdaq ITCH data and a **book builder** building central order books, supporting operations such as adding to the order book, adding order, deleting order and executing stop orders; finished the **order matching** function.
- Tested the order matching engine by running different scenarios using **Google Test**.
- Composed documents including design doc, manual and tutorial. (link: [github.com/Jack-Kin/Order-Matching-Engine](#))

Columbia University, New York, NY Oct 2021 - Dec 2021
Evaluation of Disaggregated Persistent Memory System(PM)
- Benchmarked **latency** and **throughput** for both local and remote devices including Hard Disk, Ramdisk, DRAM, PM (Persistent Memory), remote DRAM and remote PM, where remote devices are accessed through **RDMA** over **InfiniBand network adapter**.
- Integration tested the performance by manually limiting the memory with C++ and ensured 99% availability.
- Designed a disaggregated integral test framework, generated workload by **Memaslap** and measured performance.

Columbia University, Heterogeneous Computing, New York, NY Sep 2021 - Dec 2021
GPU Acceleration of K-Means Clustering
- Proposed multiple **GPU parallelization** ways to speed up the naive K-means, including using shared memory to calculate distance for each data point and using shared memory and parallel scan to sum up each centroid.
- Implemented the algorithms on NVIDIA Tesla T4 in GCP. Beaten the Sklearn K-means algorithm by speeding up 5 times when the number of data points comes to 10^6 with dimension of 8 and the cluster number is set to 5.

PROGRAMMING SKILLS

Languages: C, C++, Python, Go, MySQL, CUDA, MATLAB, Verilog
Framework: Kafka, Hadoop, Spark, Airflow, ZeroMQ, QT, PyTorch, Flask, Django, Pandas
Tools: AWS, GCP, Docker, Cmake, Linux, Windows