Yingtong Yu

✉ Email: yingtongyujobs@gmail.com
Portfolio: JARVISHHH.github.io Github: https://github.com/JARVISHHH

Education

Brown University **Sept. 2022 – May 2025 (Expected)**
Master of Science in Computer Science - GPA: 4.0/4.0 *Providence, RI*

Courses taken: Advanced Computer Graphics, 3D/2D Game Engines

Nankai University **Sept. 2018 – May 2022**
Bachelor of Engineering in Computer Science - GPA: 90.12/100.00 *Tianjin, China*

Work Experience

Nvidia **Apr. 2024 – Present**
Graphics C++ Software Engineer Intern | C++ *Shanghai, China*

- Implemented plugins for a Windows/Linux profiling tool that can correlate Vulkan commands with GPU methods.
- Created new threads, while Command Buffers are submitted to queues, use the newly created threads to output all Vulkan commands and their related information into a static .csv file.
- While recording commands, insert patches into Command Buffers before each command to indicate places to insert annotations. While submitting, insert annotations into GPU's PushBuffer through the Callback function.
- Synchronized threads using semaphores to ensure all steps have been finished and all information has been retrieved before we output information into the .csv file.

Tencent **Sept. 2023 – Mar. 2024**
Game Engine Development Engineer Intern | C++, Python *Shenzhen, China*

- Implemented a PCG node that can be used to generate large-scale landscapes based on the Procedural Content Generation Framework(PCG) feature in Unreal Engine(UE). For a 1009 x 1009 scale landscape, it can generate an island with valleys, ridges, hills, lakes and rivers in less than 1s.
- Generated mesh of landscape and elevation of each vertices using Poisson Disk Sampling, Dual Mesh, noises and Voronoi Graph, and interpolated to get the height map. Calculated direction and flux of river using HalfEdge structure based on elevation and rainfall.
- Implemented manual editing feature. Under editor mode, users can adjust the elevation of the generated landscape in real-time, and landforms like lakes and rivers will change adaptively based on elevation.
- Implemented plugins using Python and Blueprint that can make rough models and categorize them with one click.
- Researched interior furniture placement algorithms and landscape texture automatic generation algorithms.

eBay **Jun. 2023 – Aug. 2023**
Full-stack Software Development Engineer Intern | Scala *San Jose, CA*

- Worked as a full-stack software engineer and implemented a new feature called Multiple Vacations.
- For the back end, implemented several new APIs and Classes using **Scala**, including APIs that get data from the database and pack them up for the front end, APIs that validate newly scheduled vacations, etc.
- For the front end, modified code in **Node.js** and **Marko** to show the list of vacations and a newly added button.
- Implemented a new batch using **Spring Batch** to manage events-producing for partner services.

ByteDance(TikTok) **Apr. 2022 – Jul. 2022**
Back-end Software Development Engineer Intern | Golang *Beijing, China*

- Reduced the latency of the packing part of the Suggestion Middle Page **from 160ms to 10ms** by reducing the number of RPC calls and parallelizing different processes, and increased the speed by about **1500%**.
- Refactored an entire API service, making it more readable and extensible.
- Added metrics and AB test, built **Grafana** dashboards to visualize performance.
- Implemented Pinyin fuzzy search and supported the proximity-based filtering with **Elasticsearch**.
- Integrated the new version of the recommendation engine and provided more informative search bar options.

Projects – Portfolio: JARVISHHH.github.io

Escape - A Game Produced by Self-made 3D Game Engine | *C++* **Feb. 2023 - May 2023**

- Designed and implemented a 3D game engine, and produced a dungeon escape rogue-like game using the engine.
- Implemented cylinder collision for dynamic game objects and Ellipsoid-Triangle collision for static game objects. Optimized collision checking with bounding volume hierarchy(BVH) and Hierarchical Grid.
- Implemented the A* algorithm based on the navmesh for pathfinding and behavior trees for decision-making.
- Implemented several graphics features based on OpenGL, like particle system, bump mapping, shadows, etc.

Technical Skills

Languages: C/C++, Java, Golang, Scala, Python, Shell, SQL
Developer Tools: VS Code, Visual Studio, Goland, IntelliJ, Anaconda, Virtual Box, Vim, Grafana, Qt Creator
Technologies/Frameworks: Git, Unreal Engine, Vulkan, OpenGL, Eigen, GLM, Linux, Elasticsearch