# TONGJIE WANG

[tongjiew@uci.edu](mailto:tongjiew@uci.edu) | linkedin.com/in/tongjiew/ | github.com/WANGJIEKE

## EDUCATION

**University of Southern California**, Los Angeles CA            **August 2021 – May 2023**

- M.S. in Computer Science. GPA 3.75/4.
- Courses: Multimedia Processing (C++); Game Engine Development (C++); Database (Java).

**University of California, Irvine**, Irvine CA            **August 2017 – June 2021**

- B.S. in Computer Science. GPA 3.925/4; Dean's List; Magna Cum Laude.
- Courses: Programming Languages (Java); Operating System (C); Network (Python)

## WORK, PROJECTS, AND RESEARCH EXPERIENCE

**Apple:** Software Engineer, San Diego CA            **10/2023 – Now**

- Building cool software for Audio and Media Technologies team at Apple.
- Working on improving existing functionalities for Apple AirPlay using C, C++, Objective-C and Swift.
- Frequently playing with various video codec like H.264/AVC and H.265/HEVC
- Familiar with network protocols like mDNS, HTTP, and HLS
- Developing tests for quality assurance using scripting languages like Python and TypeScript.

**Swift Programming Language:** Contributor of C++ Interop            **05/2022 – 02/2023**

- Contributed as a student open-source contributor and a member of Swift-C++ interop workgroup.
- Added feature allowing using Swift enums directly from C++: discuss the design of Swift enum user model in C++ with other contributors; implement functionality of using Swift enums with associated values, including switching over the enum, extracting payload from associated values, and creating enums from C++.
- Implementations are reviewed and merged by engineers from the Developer Tools team at Apple.
- The new C++ Interop is introduced at WWDC 23.

**PyNose:** Student Researcher and Author, UC Irvine            **09/2020 – 08/2021**

- Created PyNose, a PyCharm plugin written in Java, to detect test smells (bad test code) for Python projects.
- Adapted the test smell detection technique (e.g., traversing the test code's AST to check if test smell pattern exists in the test code's structure) from previous literatures for Java into Python.
- Enabled the tool to produce detailed JSON output including the file path, position, and types of the smell, andimplemented the GUI to display the detection result on PyCharm while the users write code.
- Wrote a command line tool in Python to automatically discover all test classes in a project, run PyCharm and PyNose, and generate statistics such as smell counts in JSON and CSV files.
- Achieved 96% precision, 97.1% recall, and an overall 96.5% F1-score. Wrote the accompanying paper which is accepted by ASE 2021 as the first author.

## SKILLS

- Programming Languages: C, C++, Objective-C, Swift, TypeScript, Python, Java.
- Frameworks and Tools: Git, LLVM, LLDB, SQL Server, Spring Boot, Memcached.