# **Richard Willie**

richardw@u.nus.edu
richwill.dev

🗘 richwill28

in richard-willie



# Education

 2020 – 2024 B.Eng. Computer Engineering, National University of Singapore. Research-focused Pathway (RfP). Thesis title: Detecting Atomicity Violations in Compressed Traces.

# Experience

2024 – 2024	<ul> <li>Teaching Assistant. School of Computing, National University of Singapore.</li> <li>Parallel and Concurrent Programming (CS3211), AY23/24 SEM 2.</li> <li>Feedback from students:         <ul> <li>Richard makes tutorials very engaging, quite frankly one of the best tutorial experiences I've had.</li> <li>Richard genuinely cares about student learning outcomes.</li> <li>Richard is very clear when teaching and goes at a good pace, while also clearly putting in a lot of effort into his slides.</li> </ul> </li> </ul>
2023 - 2024	<ul> <li>Research Assistant. School of Computing, National University of Singapore.</li> <li>Contributed to a state-of-the-art research on algorithmic techniques for analysis of concurrent software.</li> <li>The work culminated in a novel algorithm for detecting certain concurrency bugs from compressed traces of programs.</li> <li>Developed an experimental program analysis tool (over 5000 lines of codes) with C++, Python, and Java.</li> <li>Keywords: Program Analysis, Formal Verification, Automata Theory.</li> </ul>
2023 - 2023	<ul> <li>Teaching Assistant. School of Computing, National University of Singapore.</li> <li>Parallel Computing (CS3210), AY23/24 SEM 1.</li> <li>Feedback from students:         <ul> <li>Richard's class is the only class I find worth traveling to school for.</li> <li>Richard is extremely knowledgeable. He makes learning interactive and fun.</li> <li>Data Structures and Algorithms (CS2040C), AY23/24 SEM 1.</li> </ul> </li> </ul>
2022 - 2023	<ul> <li>Research Assistant. School of Computing, National University of Singapore.</li> <li>Contributed to a state-of-the-art research on 3D volumetric video streaming.</li> <li>Worked on a project with over 500 thousand lines of C++ codes.</li> <li>Keywords: Algorithms, Computer Graphics, Computer Vision.</li> </ul>
2022 – 2022	<ul> <li>Teaching Assistant. School of Computing, National University of Singapore.</li> <li>Data Structures and Algorithms (TIC2001), AY22/23 SEM 1.         <ul> <li>Taught a class of "lifelong" learners (aged 24 to over 60).</li> <li>Feedback from students:                 <ul></ul></li></ul></li></ul>

\* Richard cares about students.

## **Experience (continued)**

- **Software Engineer.** Blugraph Technologies.
  - Developed a full-stack web application with React, Node.js, and SQL.
  - Worked on IoT (Internet of Things) projects with Python, C, and MATLAB.

### 2021 - 2021

- **Teaching Assistant.** School of Computing, National University of Singapore.
  - Software Engineering & Object-Oriented Programming (CS2113), AY21/22 SEM 2.
    - Feedback from students:
      - \* Richard is great tutor who knows his content very well and is open to questions.
  - Data Structures and Algorithms (TIC2001), AY21/22 SEM 1.
    - Taught a class of "lifelong" learners (aged 24 to over 60).
    - Feedback from students:
      - \* Richard is very patient and explain his thoughts clearly.

## Projects

Sharded Reconfigurable Key-Value Service with Distributed Transactions

- A **fault-tolerant linearizable** key-value storage system that shards the keys over a set of replica groups and handles cross-group transactions.
- This is a highly technical **distributed systems** project. Fault-tolerance is achieved with an implementation of **Paxos**. The linearizability of the system is achieved with **two-phase commit** and **locking**.
- The whole system was implemented in Java.

### 📕 Oat Compiler

- Developed an **Oat** compiler in **OCaml**.
- This compiler is designed with a two-phase compilation strategy. First, it compiles **Oat** to **LLVM**, the intermediate representation. Then, it compiles **LLVM** to **x86lite**.
- Implemented with various dataflow analysis techniques and optimizations.

#### SML Interpreter

- Developed an **SML** (Standard ML) interpreter in **TypeScript**.
- Capable of interpreting a non-trivial subset of the language. For example, inference of polymorphic types and higher-order functions.
- Designed with rigor and formal specifications in mind, this project led to a redevelopment of the Hindley-Milner type system from first principles.

### Exchange Matching Engine

• This is a high-throughput concurrent exchange matching engine developed with C++ and Go.

### Looney Troons – Train Network Simulation

• This is a train network simulation program developed with C++ and parallel programming frameworks such as **OpenMP** and **MPI**.

#### **GPU Virus Signature Scanner**

• This is a virus signature scanner that runs on the GPU, written in CUDA and C++.

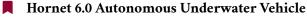
### Wearable Laser Tag System

- Led a team of five in developing a wearable laser tag system, equipped with AR (Augmented Reality).
- This is a major team effort involving **hardware**, **computer networks**, **machine learning**, and **game development**.
- Tools used: C++, Python, Unity, C#, vuforia, FPGA, Ultra96, Arduino.

### **Pathfinding Visualizer**

- A web application to visualize pathfinding and maze generation algorithms.
- The modern UI and animation were built with JavaScript, React, and Tailwind.

## **Projects (continued)**



- The architecture was developed using ROS (Robot Operating System), Python, and C++.
- Implemented a system of **Computer Vision** and **Optical Flow** using **OpenCV** to enable environmental tracking and autonomous movement.

### Autonomous Robotic Car

- Led a team of three in developing an autonomous robotic car, equipped with RTOS (Real-Time Operating System).
- The overall system was developed with FreeRTOS, C, C++, Assembly, and JavaScript.

#### Search Engine for Legal Cases

- Built a search engine for legal case retrieval with Python and NLTK.
- Capable of executing boolean and wildcard queries.
- Implemented with several index construction/compression, and query refinement techniques.

### Chess Engine

- Built a classical AI chess engine with **Python**.
- 📕 VisuTrader
  - A full-stack paper trading web application, developed using React, Django, and PostgreSQL.

### 📕 NUS Buddy

- Led a team of four in developing a Java application for NUS students to manage their tasks, lessons, and modules.
- Responsible for quality assurance (e.g., all PRs were reviewed; code were tested with JUnit).

### Skills

Languages	Fluent English, Fluent Bahasa Indonesia, Intermediate Japanese.
Coding	C++, Rust, Python, Go, OCaml, C, JavaScript, TypeScript, Java, C#, SML, Haskell, Coq.
Web Dev	HTML, CSS, JavaScript, TypeScript, React, Svelte, Sass, Tailwind.
Frameworks	Unity, OpenMP, CUDA, MPI.
Misc.	Academic research, teaching, consultation, Large typesetting and publishing.

### Miscellaneous

### Awards and Achievements

- 2022 **Top Students for Software Engineering & Object-Oriented Programming**, National University of Singapore.
- 2021 **Crbital Apollo 11 (Advanced)**, National University of Singapore.

### Certification

- 2024
- **Research-focused Pathway in Computer Engineering**, National University of Singapore.

RfP focuses on preparing students for the R&D sector. Thus, students are required to take graduate-level electives, conduct internship in Research Institutes/Laboratories and work on a research-focused final year project.

## Miscellaneous (continued)

### Hackathons

2022

### NUS Hackers Hack&Roll

- An annual 24-hour hackathon and the largest student-run hackathon in Singapore.
- Developed a monkeytype clone but with a little twist.
- Competed for "Most Annoying Hack".

### **Co-Curricular Activities**

2020 - 2021	NUS Bumblebee
	We design and build autonomous maritime vehicles capable of navigating underwater and on the water surface, performing complex tasks autonomously.
2021 - 2022	NUS Games Development Group
	We are a group dedicated to making games.
2022 - 2023	<b>NUS PINUS Tech</b> PINUS is a group of Indonesian Students in National University of Singapore (NUS) for- mally established in 2006. In Tech, we provide various software solutions to clients.
2023 - 2024	<b>NUS Comics &amp; Animation Society</b> We are a group of friends passionate about all sorts of Japanese Animation, Comics, Events and Games.

### **Notable Courses**

AY23/24
---------

### CS5469 Fundamentals of Logic in Computer Science

This is a graduate-level course taken by PhD students. It gives a formal and rigorous introduction to some fundamental results in logic from a computer science perspective, with particular emphasis on algorithmic and computational complexity components.

CS5223 Distributed Systems

This is a graduate-level course taken by master and PhD students.

- CS4212 Compiler Design
- CS3234 Logic for Proofs and Programs

The Curry-Howard isomorphism, i.e. the realization that mathematical proofs and computer programs are the same thing. This course serves as an introduction to the Coq Proof Assistant.

#### **CS3231** Theory of Computation

What is the P vs NP problem and why does it matter? How do we decide if a problem is easy or hard? This course is the bedrock of computer science.

CS3230 Design and Analysis of Algorithms

AY22/23

# CS4215 Programming Language Implementation

CS3211 Parallel and Concurrent Programming This course explores various concurrency paradigms through the lenses of different programming languages such as C++, Go, and Rust. Fun stuff.

#### **CS3210** Parallel Computing

This course provides an introduction to the field of parallel computing with hands-on parallel programming experience on real parallel machines. Fun stuff.

- CS2107 Introduction to Information Security
- CS2102 Database Systems
- AY21/22 **CS2106 Operating Systems** 
  - **CS2105** Computer Networks

# Miscellaneous (continued)

CG2271 Real-Time Operating Systems

### **Academic Interests**

- **Theoretical computer science**, particularly topics in computability and complexity theory.
- **The theory, design, and implementation of programming languages.**
- Program analysis techniques for detecting bugs in concurrent programs.
- **T**opics in algorithms and data structures.

### Hobbies

Books, Classical Guitar, Mathematics, Movies, Philosophy, Science Communication, Tea.