

John Wang

About me

Computational linguistics graduate with a strong interest in programming languages and functional programming. Comfortable working in both English and German, with experience in Rust, Haskell, and related tools.

Interests

- Functional Programming
- Formal Methods
- Math

Contact

✉ johnz.wang@outlook.com

☎ +49 017632280303

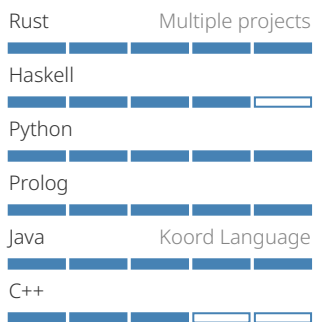
📍 Beim Kupferhammer 5/4,
Tübingen, Germany

Personal

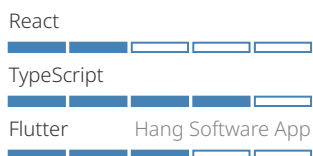
Nationality: American

Date of birth: 09.16.98

Programming Languages



Frontend



DevOps



EDUCATION

- 2022 – 2025 **M.Sc. in Computational Linguistics**
UNIVERSITY OF TÜBINGEN 📍 TÜBINGEN, GERMANY
- 2017 – 2021 **B.Sc. Computer Science & B.A. Linguistics (Minor in Mathematics)**
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN (UIUC) 📍 URBANA-CHAMPAIGN, IL, USA

EXPERIENCE

- Winter 2022, 2023 **Teaching Assistant**
UNIVERSITY OF TÜBINGEN 📍 TÜBINGEN, GERMANY
- Assisted in teaching **Data Structures and Algorithms**.
 - Provided student support and graded assignments.
- Spring 2021 – 2022 **Undergraduate Research Assistant**
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN 📍 CHAMPAIGN, IL, USA
- Developed a verifier for K Framework languages by compiling to Metamath.
- Spring 2020 – Spring 2021 **Software Developer**
HANG SOFTWARE 📍 CHAMPAIGN, IL, USA
- Built and deployed a cross-platform Flutter app
 - Contributed to frontend and backend features.
 - Project: findyourhang.app
- Summer 2020 **Software Engineering Intern**
EPIC SYSTEMS 📍 VERONA, WI, USA
- Designed and implemented a customizable hotkey UI using React and TypeScript.

PROJECTS

Koord Language

Designed and implemented a parser and compiler for the Koord language using ANTLR (Java).

rust-metamath

Created a Metamath theorem prover in Rust, applying formal verification principles.

Visual Semantics

Built a visual small-step semantics simulator in Rust.

Uwu Programming Language

Developed an anime-themed programming language with parallelism, compiling to an LLVM backend (C++).

rust-capr

Engineered Capr, a historical linguistics tool using finite-state transducers. Backend in Rust (WebAssembly), frontend in TypeScript.

PUBLICATIONS

- OOPSLA 2023 **Generating Proof Certificates for a Language-Agnostic Deductive Program Verifier**
Zhengyao Lin, Xiaohong Chen, Minh-Thai Trinh, John Wang, Grigore Roşu.

Tools

Git



Linux Daily Driver



Cargo



LaTeX



Coq Exposure



Isabelle Exposure



Languages

English Native



German C1 Speaking/Listening



Chinese Fluent

