

# William Chargin

[git.io/wc](https://git.io/wc) • [wchargin](https://wchargin.com)

Please visit my website at [git.io/wc](https://git.io/wc) for an interactive résumé, and more up-to-date and detailed project descriptions.

## Education and selected skills

---

**Carnegie Mellon University:** B.S. Computer Science with honors; minor in Mathematics. GPA 4.00.

**Honors thesis** in logical foundations of programming languages: *A general system of adjoint logic*. Related paper:

*Adjoint logic and its concurrent operational interpretation*, submitted to LICS 2018; [www.cs.cmu.edu/~fp/papers/adjoint18.pdf](http://www.cs.cmu.edu/~fp/papers/adjoint18.pdf)

**Proficient or better in:** Python, Java, C, Haskell; JavaScript, React; Google App Engine; Git;  $\text{\LaTeX}$ , TikZ, Blender 3D.

## Experience

---

### Google Brain

*Software Engineering Intern*

Mountain View, CA

May–August 2017

- Worked on the TensorFlow team developing TensorBoard, an open-source framework that helps developers and researchers to understand, debug, and improve their machine learning models.
- Designed and implemented a rich plugin system to enable all users to write data visualizations tailor-made for their purposes. This system is already used by external contributors. Work spans web frontend, Python backend, and core TensorFlow API design.

### Addison–Wesley Publishers

*Technical editor for Effective Java, 3rd edition*

Pittsburgh, PA

February–December 2017

- Contributed novel technical content, in addition to suggesting improvements to prose, grammar, typography, etc.

### Khan Academy

*Software Developer Intern (infrastructure)*

Mountain View, CA

May–August 2016

- Extended the site's core content system to enable creating and curating content separately for different languages and locales, as opposed to simply translating existing content.
- Conducted extensive testing of correctness, performance, memory, and cost to ensure a smooth transition upon launch.
- Improved tooling to help translators, content creators, and international teams work effectively with the new content system.

### Khan Academy

*Software Developer Intern (frontend and backend)*

Mountain View, CA

June–September 2015

- Frontend, backend: added CMS support for thumbnail upload, compositing, storage, and usage; implemented streaks (à la Duolingo).
- Backend: implemented hot loading of JSX and CSS/Less for development; sped up internal content publish process by 57%.

## Selected projects

---

### Assorted interesting course projects

Java

- A data visualization framework supporting arbitrary data plugins (e.g., web scrapers) and visualization plugins (e.g., histograms).
- An expanded version of *Scrabble*, including well-tested core logic and a polished GUI.
- More details (technical descriptions, documentation, screenshots, videos) at [wchargin.github.io/projects/cmu-15-214](https://wchargin.github.io/projects/cmu-15-214).

### Microcomputer assembler and simulator

JavaScript

- Web-based interactive simulator for the LC-3, a teaching microcomputer, to supersede the standard Windows application.
- Simulator includes accurate instruction cycle, debugging tools, polling-based and interrupt-driven I/O, and file upload/download.
- Released as free software at [wchargin.github.io/lc3web](https://wchargin.github.io/lc3web). Used by more than 2,000 students of Cal Poly's CPE 225 course to date.

### Model United Nations debate moderation system

Java

- Created and deployed an application system that unifies the tools that chairs need to aptly moderate debates.
- Implemented networking across multiple computers to maximize efficiency; separate modes for head chair, director, and rapporteur.
- Deployed system at multiple conferences; system used by dozens of chairs and hundreds of delegates.
- Released as free software; available at [wchargin.github.io/kiosk](https://wchargin.github.io/kiosk).

### Automated grading system

Bash, Java

- Automatically tests and grades student work for style and correctness, according to customizable and extensible grading modules.
- Grades and archives all student work at assignment due dates, and immediately emails students with helpful feedback.
- Includes tool to efficiently manually investigate failing submissions, to ensure that all grades are accurate.