

# Wesley Wei

617-275-9957 ◇ Wesley@WeiWesley.com ◇ U.S. Citizen

## TECHNICAL STRENGTHS

---

<b>Programming Languages</b>	C, C++, Python, Bash/Shell, MATLAB, HTML/CSS/Javascript, jQuery
<b>Technologies &amp; Tools</b>	GNU/Linux, Vim, Git, PyQt

## EDUCATION

---

**Tufts University, Medford, MA, *Class of 2018***  
Bachelor of Science, Computer Science  
Bachelor of Science, Mathematics

## EXPERIENCE

---

**Tufts University, Medford, MA** September 2015 - May 2018  
*Teaching Assistant - Department of Computer Science*

- Answered students' questions regarding assignments and topics covered in class.
- Graded design documents for class assignments.

**NSRRC, Hsinchu, Taiwan** June 2016 - August 2016  
*Temporary Assistant*

- Refactored a Python/PyQt based frontend for a Linux based image processing backend with the goal of easing future development and the addition of new features.
- Added the capability to operate on subsets of data, letting the application to load a fraction of thousands of images and operate much faster.

**NTHU, Hsinchu, Taiwan** June 2015 - November 2015  
*Temporary Assistant*

- Wrote a MATLAB script automating data analysis, replacing manual calculations involving ORIGIN and Microsoft Excel, cutting processing time from 20 minutes to a few seconds per data set.

## PROJECTS

---

**REPL** June 2018 - Present  
*Hosted on Github*

- REPL is a python framework for embedding a shell into an application, providing a simple way to bind python code to textual commands.
- REPL mimics some of the conveniences of a POSIX shell; it provides pipes, functions, flow control, aliases, and more.

**Composte** Fall 2017  
*Hosted on GitHub*

- Composte is written in Python and is a Linux based client-server application that facilitates real-time, collaborative editing of sheet music.
- Implemented network foundation using ZeroMQ, serverside storage using SQLite, REPL for user interaction and scripting, and encryption stubs.
- Composte includes a PyQt GUI developed by a team member, and a music backend based on the music21 package, developed by another team member.

**Questionable Battleship** Summer 2017 - Present  
*Hosted on GitLab*

- Multiplayer Battleship over websockets. Available at [questionablebattleship.com/simple](http://questionablebattleship.com/simple).
- Developed simple web-based UI using Javascript and jQuery.
- Developed backend server in C++, relying on the following libraries: crossguid, doctest, json, spdlog, tclap, websocketpp, cppzmq, libicu. The server uses worker processes to manage its state and ZeroMQ for IPC.