

Guanghao Ye

http://yeguanghai.xyz/

Last updated on August 31, 2019

Email : yeguanghai1999@gmail.com

Mobile : +1-206-228-7256

SKILLS

- Proficient: C/C++, Java, Kotlin, Python, SQL, Standard ML, Algorithm Design and Analysis
- Familiar: Golang, Racket, Haskell, Shell, Amazon AWS, Microsoft Azure, Apache Spark




EDUCATION

- **University of Washington** Seattle, WA
 - *Bachelor of Science: Computer Science, Mathematics;* *Aug. 2017 – Mar. 2020 (Expected)*
Selected Graduate Coursework: Theory of Convex Optimization, Randomized Algorithms, Probabilistic Combinatorics, Computational Complexity, Algorithms through Geometric Len.
 - *Master of Science: Computer Science* *Mar. 2020 – Jun. 2021 (Expected)*

EXPERIENCE

- **Airbnb, Inc.** Seattle, WA
 - Software Engineer Intern* *Jun. 2019 – Sept. 2019*
 - Worked on the Payment Tax team.
 - Designed and implemented a domain specific language for tax calculation.
 - Integrated the new service into current tax quotation process.
- **University of Washington** Seattle, WA
 - Undergraduate Researcher* *Mar. 2019 – Present*
I currently work with Prof. Yin Tat Lee on spectral graph theory.
- **University of Washington** Seattle, WA
 - Teaching Assistant* *Mar. 2018 – Present*
Assisted with course planning and development, graded students' homework, and held weekly office hours.
I have TAed CSE P521: Applied Algorithms and 3 × CSE 421: Introduction to Algorithm.

PROJECTS

- **File System Search Engine** *Jun. 2018*
Wrote a file system search engine using C++. It builds on-disk indices and has a POSIX multi-thread web server built on C++11 and Boost which support thread pool.  [Github](#)
- **Online Judge System** *Aug. 2017*
Wrote an Online Judge System based on Python, Django, and Docker to help students to test their programs of competitive programming problems and to help coaches to organize contests online.  [Github](#)
- **Top Trees C++ Implementation** *Aug. 2016*
Wrote a C++ implementation of *Tarjan's self-adjusting top trees* which can maintain information in a fully dynamic forest. Here fully dynamic means that edges may be both inserted and deleted.  [Github](#)

HONORS

- **Recipient:** Microsoft Endowed Scholarship *Sept. 2018 – Jun. 2019*
- **Rank 5:** International Collegiate Programming Contest, Pacific Northwest *Nov. 2017*
- **Rank 7:** Internet Problem Solving Contest 2017, High School Division *Jul. 2017*
- **Rank 1:** USA Computing Olympiad 2016-2017 First Contest *Dec. 2016*
- **Dean's List:** University of Washington Seattle Annual Dean's List *Sept. 2017 – Present*