# **Jonan Seeley**

Senior

#### **Personal Info**

Email: jseeley@cmu.edu Website: jseeley.xyz Github: jonanseeley

#### **Relevant Courses**

15-440 Distributed Systems

**15-410** Operating System Design and Implementation

**15-451** Algorithm Design and Analysis

**11-411** Natural Language Processing

**05-436** Usable Privacy and Security

**15-356** Introduction to Cryptography

**17-303** Cryptocurrencies, Blockchains, and Applications

#### **Awards**

#### **Deloitte StartUP**

3<sup>rd</sup> Place

Planned company based around bettering life in Pittsburgh in 72 hours

#### **Algorithms with a Purpose**

4th Place

Developed artificial intelligence to compete in a war-like scenario in 8 hours

#### Skills

Programming: C, Python, C++ Java, SML

Technologies: Django, NTLK, Tesseract OCR, Other: Unix, LaTeX, Vim

### **Education**

#### Carnegie Mellon University — Pittsburgh, PA

Bachelor of Science in Computer Science Concentration in Security and Privacy

QPA: 3.05/4.00 (Expected May 2021)

## **Experience**

#### **Software Development Intern** — Remote

Amazon

June 2020 - August 2020

- Integrated team's service with 2 sister team services to build foundation for further integration
- Updated frontend, backend, and interface to facilitate service integration

#### **Teaching Assistant** — Pittsburgh, PA

15-110 Principles of Computing

June 2018 - December 2020 (Head TA Summer '19)

- Led recitations and held office hours several times per week
- Managed autograding platform for 11 programming assignments and 2 lab exams

#### **DevOps Assistant** — Pittsburgh, PA

Pittsburgh Supercomputing Center June 2016 - May 2018

- Collaborated with the Systems team to maintain supercomputers
- Ensured that supercomputers utilization was monitored and above 75%

## **Projects**

## **Operating System Kernel**

- Developed preemptive multithreading operating system kernel
- Implemented over 20 unique POSIX-like system calls
- Integrated user-level threading library to enable multithreaded userland programs

#### **Boggle**

- Wrote multiplayer web-based implementation of classic board game
- Used websockets to allow for asynchronous communication