Noah Thompson

nst45@cornell.edu • (360) 932-4738

Education

Cornell University, College of Arts and Sciences Fall 2018 - Spring 2022

- B.A. Computer Science, Cumulative GPA: 3.823.
- Relevant Courses: (*5/6xxx = grad*) († = *TA*) (* = *currently enrolled*)

CS 5414 — Distributed Systems	CS 6820 — Graduate Algorithms	CS 6172 — Program Synthesis
CS 4120 — Compilers	CS 4820 — Analysis of Algorithms †	CS 5114 — Network PL
CS 6110 — Programming Languages	CS 3110 — Functional Programming †	CS 4410 — Operating Systems

Skills

Languages: Python, Java, Scala, Typescript, Purescript, OCaml, C, HTML/CSS, Unix scripting, Assembly (x86)
Technologies: Git, React, NumPy, OpenCV, SQL, Flask, Docker, writing parsers, CI/CD tooling

Work Experience

Microsoft Software Engineering Intern Summer 2021

- Started the Signup team's migration from static content to using a CMS and new A/B testing platform, leading regular meetings with stakeholders in localization, CMS devs, and internal partner PMs, and team members
- Used typescript and React to integrate CMS and A/B experimentation platform into existing code
- Ran A/B experiments to evaluate post-integration experience and proof-of-concept A/B tests through new CMS

Apple Software Engineering Intern Summer 2020

- Worked on build management service used by ~2000 employees for collaborating on chip specifications
- Architected a CI/CD-tool-agnostic interface in Scala, using it to better integrate our system with TeamCity
- Utilized new interface to add new user features for live build monitoring: remaining time estimates, early stopping

Projects

Paxos-enabled Distributed Key Value Store Fall 2020

- Build a linearizable, sharded key-value store with multi-key updates and dynamic load balancing
- Implemented various optimizations leveraging a data flow analysis including register allocation and propagations

Xi Compiler Spring 2020

- Designed custom optimizing compiler for Xi Language, wrote over 10k lines of Scala, Assembly in team of four
- Implemented tokenization, parsing, type-checking, intermediate representations and lowering to assembly
- Implemented various optimizations leveraging a data flow analysis including register allocation and propagations

APIP — Accountable and Private Internet Protocol Spring 2021

- Designed custom optimizing compiler for Xi Language, wrote over 10k lines of Scala, Assembly in team of four
- Implemented tokenization, parsing, type-checking, intermediate representations and lowering to assembly
- Implemented various optimizations leveraging a data flow analysis including register allocation and propagations

Cornell University Unmanned Air Systems Fall 2018 – Present

- Software engineer for top undergraduate air systems team in the nation that builds a custom autonomous aircraft
- Built a testing and metrics framework for flight path planning; implemented RRT and potential flow based algorithms, optimized by tuning parameters, adding spatial indexing, and leveraging numpy parallelizing/caching
- Created technical and cultural infrastructure for recruiting process in effort to reduce implicit bias in team recruiting

Network Program Synthesis Laboratory Fall 2020 - Spring 2021

- Developed in OCaml to synthesize simple network programs by searching program space using SAT solver
- Built cache system to store simple and partial program solutions, allowing synthesis for more complex programs

Robotic Personal Assistant Laboratory Summer 2019 - Spring 2020

- Developed an AI to play a cooperative card game, *Hanabi*, that could process human expressions and gestures
- Investigated vision based inverse kinematics through partner pose estimation with OpenCV, ROSpy and AprilTags.