Austin Miles

69 Brown St, Box 4845 | Providence, RI 02912 | Phone: (336) 200-0954 | E-Mail: austin_miles@brown.edu

EDUCATION

Brown University, *B.S.* 3.8/4.0 GPA — Providence, RI | **Expected Graduation May 2024**
Relevant Courses**:** Data Structures and Algorithms, Operating Systems, Computer Networks
North Carolina School of Science and Mathematics, 4.50/4.00 GPA — Durhman, NC | Class of 2020

WORK EXPERIENCE

Brown University Department of Computer Science
Head Teaching Assistant for CSCI0170 — Providence, RI | March 2022 – Present

- Lead a team of 17 teaching assistants (with 3 others) to support 150+ students in learning introductory coding concepts
- Redesign course projects and interact with project code to realign the material with ever-changing course goals
- Coordinate staff meetings to gather teaching assistant and student feedback, communicate next steps with the course, and organize our teaching methods once a week

Brown University Department of Computer Science
Systems, Programmer, Operator, Consultant (SPOC) — Providence, RI | May 2022 – Present

- Manage and maintain the computer science (CS) department systems during nights, holidays, and weekends for at least 5 hours a week
- Upgrade the CS department software and systems
- On-call for acute problems, like file system downtime, power outages, or network issues

Brown University Department of Computer Science
Teaching Assistant for CSCI0170 and CSCI0200 — Providence, RI | Sep 2021 – May 2022

- Held weekly lab and office hours to assist students with debugging and conceptual understanding
- Tested class project code and developed handouts to prepare for assignment releases
- Spent 4 hours each week offering students feedback and criticism on their assignments

PROJECTS

Weenix, *simple Operating System (Language: C)* — Jan 2022 – May 2022

- Developed a primitive thread scheduler with a one-to-one kernel thread to process model which allows processes to run
- Coded a tty driver with a functional line discipline which allows for visible terminal input along with 2 simple memory devices (/dev/null and /dev/zero)
- Added in a virtual file system which supports multiple system calls and a file-based abstraction of a disk drive
- Implemented a file system based off the original UNIX file system
- Integrated virtual memory into the codebase. This included developing data structures to represent virtual memory and adding in code to manage userspace processes

IP/TCP, *networking stack (Language: C)* — Feb 2022 – May 2022

- Constructed a link layer (layer 2 of OSI model) by encapsulating IP packets inside UDP packets
- Created a router like node for the network layer. This node implements RIP, packet forwarding, and packet delivering to allow a collection of nodes to talk to another by referencing RFC 791
- Coded the TCP protocol to allow 1 megabyte files to be sent reliably between two nodes by referencing RFC 793

Snowcast, *music server (Language: C++)* — Feb 2022

- Implemented a simulated radio station that streams out data (read from input files) to specific UDP ports in Linux
- Established a TCP server and client connection which both follow a reusable network protocol
- Built a multithreaded backend in which the server supports multiple “station” threads that send out data to clients

Shell, *computer terminal (Language: C)* — Oct 2021

- Included extensive error checking to prevent user input from crashing shell
- Supported various builtin commands: cd, cat, rm, ln, exit. While also providing use of non-builtins by using path name
- Controlled jobs and processes through the shell via signals and builtins like: fg and bg

SKILLS & INTERESTS

Programming Languages: C/C++ (proficient); Go, Python, Java, ReasonML (experienced)
Technical Skills: Linux, Git, Latex, Bash, G suite
Language: German (intermediate)