Andrew Kerr

B.S. Physics & B.Mus. Percussion Performance
kerrand@protonmail.com | (734) 834-9161 | andkerr.github.io

EDUCATION

UNIVERSITY OF MICHIGAN
September 2018 - May 2023

Physics & Percussion Performance, Computer Science Minor

- 3.9 / 4.0 GPA
- Coursework: Data Structures and Algorithms, Computer Organization, Discrete Math, Computational Physics, Quantum Mechanics, Statistical Mechanics, Math Methods of Theoretical Physics

EXPERIENCE

DISCRETE MATHEMATICS TEACHING ASSISTANT
August 2022 - Present

U of M College of Engineering

- Lead weekly office hours and discussion section to assist students in learning introductory discrete math topics.
- Collaborate with a team of ~30 instructors to develop course materials, including homework and exams.

SOFTWARE ENGINEERING INTERN
June 2022 - August 2022

Vouch Insurance

- Apply responsive design principles to high-traffic areas of a new client onboarding application, increasing usability.
- Develop appealing, functional user-facing web components in React and Typescript.

RESEARCH ASSISTANT
January 2022 - September 2022

U of M Center for Academic Innovation

- Aggregate, compute statistics, and plot figures for ~15 years of U of M student data using pandas and matplotlib.
- Maintain and deploy a minimal Node.js web application for displaying figures and interactive animations.

RESEARCH ASSISTANT
April 2021 - December 2021

Glotzer Research Group, U of M College of Engineering

- Contribute code updates, bug fixes, and new features to molecular dynamics simulation software.
- Implement unit tests with Pytest to verify the correctness of existing student-developed code.

PROJECTS

RAY TRACER
August 2022 - Present

- A C++ implementation of a basic 3D render that displays various shapes and textures. (source code)
- Work in progress: currently being extended to support a YAML file format for specifying scenes.

SIMPLESORTER
July 2022

- In-browser visualizations of canonical sorting algorithms. (source code)

LORENZ ATTRACTOR ANIMATIONS
October 2021

- A CLI that allows users to specify initial conditions for up to five simultaneous Lorenz Attractors. (source code)

SKILLS

- Programming Languages: Python, C, C++, Bash, LaTeX
- Scientific Computing: Numpy, Matplotlib, Pandas, Jupyter Notebook, conda
- Frameworks: Django, Flask, Bootstrap.js, React.js
- Developer Tools: VSCode, Vim / NeoVim, Makefile, Git