

Daniel Van Der Maden

✉ dvandermaden0@berkeley.edu ☎ (714) 251-4974

🌐 danielvandermaden.com

🐙 github.com/Daniel-VDM

in linkedin.com/in/Daniel-VDM

EDUCATION

University of California, Berkeley

Berkeley, CA

B.A. Computer Science – GPA: 3.68/4.00

December 2019 (Expected)

- *Relevant coursework:* Computer Security (IP Su), Database Systems (IP Fa), UI/UX Design (IP Fa), Efficient Algorithms, Artificial Intelligence, Natural Language Processing, Intro to Linguistics, Machine Structures, Data Structures, Structure & Interpretation of Programs, Designing Information Devices, Discrete Math & Probability Theory, Linear Algebra & Differential Equations.

EXPERIENCE

Microsemi Corporation

Aliso Viejo, CA

Software Engineering Intern – Frequency and Timing Division

May 2018 – August 2018

- Gained practical industry experience by developing embedded programs in C for a modern radio navigation and data system.
- Worked closely with other software engineers to rapidly implement multiple signal schemes and meet fixed live test deadlines.
- Improved transmission timers by extending the API (written in C) to allow for more versatility in the signal scheme implementation.
- Shortened scheme implementation time by creating development tools using Python. Also wrote thorough documentation.
- Presented and handed off the work to the rest of the project team in Boulder, CO. Live transmission tests were successful.

EECS Department, UC Berkeley

Berkeley, CA

Tutor – Self-Paced Center

January 2019 – May 2019

- Improved student understanding by holding office hours for students learning C, C++, Java, and Python.
- Gave code feedback that encouraged industry-ready practices. Also graded projects, quizzes, and final exams.

Academic Intern – Structure & Interpretation of Programs Course (Python)

August 2018 – December 2018

- Fostered interest in Computer Science fundamentals by providing guidance on homework and projects during office hours.
- Assisted course staff during lab sessions by conducting lab checkoffs and answering questions regarding course content.

Computer Science Mentors, UC Berkeley

Berkeley, CA

Peer Mentor – Designing Information Devices & Systems Course (Circuits / Linear Alg)

January 2019 – May 2019

- Helped students solidify core EECS concepts by leading a supplemental 1.5 hour discussion section each week.
- Mentored a cohort of students throughout the semester and helped develop supplemental worksheets and lesson plans.

J-Sei Community Center

Emeryville, CA

Technology Consultant for Seniors

September 2017 – December 2017

- Effectively communicate a verity of tech safety topics, such as internet privacy, scam detection, and malware protection.

NOTABLE PROJECTS

A detailed list (with repo links) can be found on my personal website

Easy Seq2Seq Chatbot – Personal Project

🔗 <https://tinyurl.com/y2tsst5x>

- A Python script that implements a chatbot and has easy ways of defining various parameters so that it can be used as a learning tool.
- It features a caching system, saving and loading models, interrupt recovery, memory efficient training, and multiple data filters.

Approximate Solver for a NP-Hard Problem – Efficient Algorithms Course Project

🔗 <https://tinyurl.com/y2dttmrv>

- A polynomial-time approximate solver (implemented in Python) that uses a greedy algorithm with various heuristics.
- Yielded solutions that were in the top 10% of all approximations in the course (which had ≈ 700 students).

Concurrent Cached File Server – Machine Structures Course Project

🔗 <https://tinyurl.com/y6rhmbpd>

- A file server (implemented in Go) with a cache that can efficiently handle thousands of concurrent file requests (on a laptop).

Coreference Annotator – Natural Language Processing Course Project

🔗 <https://tinyurl.com/y54l8mn6>

- A Python script that annotates the antecedent for each pronoun in a given dataset. Achieved 71% accuracy on test data.

Personal Website – Personal Project

🔗 <https://tinyurl.com/y2vwnvld>

- A mobile and ultra-wide friendly website that was created using Bootstrap and some JavaScript with jQuery.

61Ccc Compiler (to RISC-V) – Machine Structures Course Project

🔗 Link to repo upon request

- Compiles code that is written in the 61Ccc language (a made-up subset of C) into universally compatible RISC-V assembly.

SKILLS

Programming Languages: Python, Java, C++, C, Golang, Scheme, RISC-V/x86, SQL, HTML, CSS, JavaScript, LaTeX

Technologies: Scipy/Numpy/Pandas, NLTK, Keras, Scikit-learn, Hadoop, OpenMP, Intel AVX, Bootstrap, MySQL, AWS EC2

Spoken Languages: Native English speaker, fluent in French, basic understanding of Vietnamese.