Emaan Hariri

ehariri@berkeley.edu · emaan.me · +1 (949) 690-4052

Education

University of California, Berkeley — Berkeley, CA
Master of Science in Electrical Engineering and Computer Science; Advisor: Satish Rao — *August 2020 – May 2021*

University of California, Berkeley — Berkeley, CA
Bachelor of Arts in Computer Science (Focus in Statistics); GPA: 3.97, Major GPA: 4.00 — *August 2016 – May 2020*

Honors/Awards: Phi Beta Kappa, Dean's Honors, Upsilon Pi Epsilon (CS Honors), US Congress Cert. of Recognition

Relevant Coursework: Graduate Algorithms & Data Structures (CS 270)†, Machine Learning (CS 189)†, Operating Systems (CS 162)[IP], Stochastic Processes (STAT 150)†, Deep Reinforcement Learning (CS 285), Machine Structures (CS 61C), Discrete Mathematics (CS 70), Artificial Intelligence (CS 188), Computer Security (CS 161)†, Databases (CS 186)†, Data Science (STAT 133)†, Probability Theory (STAT 134)†, Linear Algebra & Differential Equations (MATH 54)†.

†Courses in which an A+ grade was received.

Experience

Salesforce, Inc. — San Francisco, CA
Software Engineering Intern — *June 2019 - August 2019*

- **Service Cloud**: Work with Embedded Service for Web team to implement the conversion of internal libraries to enable customers to adapt both the Lightning Web Component and Aura JS frameworks to embedded service snap-ins.
  - Modify core Java backend and implemented ES6 libraries and functional/integration tests written in Jest/JUnit.
  - Architect new internal/external library design to maintain Aura API while conforming with LWC interoperability.

UC Berkeley College of Engineering — Berkeley, CA
Undergraduate Student Instructor — *June 2018 - Present*

- **Efficient Algorithms and Intractable Problems (CS 170)**:
  - Teach topics and develop project involving divide-and-conquer, FFT, greedy algorithms, graph algorithms, dynamic programming, linear programming, approximation algorithms, streaming algorithms, and complexity theory.
  - *Project TA:* In charge of developing course project, which involves concepts of NP-Completeness and optimization.
- **Discrete Mathematics and Probability Theory (CS 70)**: Only student to TA two CS courses Summer 2018.
  - Teach topics in discrete mathematics and probability which include propositional logic, proofs, number theory, graph theory, polynomials, error correction, RSA, random variables, load balancing, bounding, and Markov chains.
- **Machine Structures and Computer Architecture (CS 61C)**:
  - Taught topics which include C programming, assembly/machine language, floating point, caches, parallelism, warehouse computing, MapReduce, virtual memory, dependability, RAID, I/O, disks, and networking.
  - Developed projects and laboratory assignments involving Logisim, OpenMP, Intel SSE, RISC-V and Python.

Mobile Developers of Berkeley — Berkeley, CA
Android and Web Developer — *September 2016 - Present*

- **Android Development**: Created Android statistics application StatWiz with team of three.
  - Implemented custom statistics library and screen designs using Java, Android SDK, and XML.
- **Web Development**: Developed web app PriceGuessr, a price guessing trivia game, in team of four.
  - Implemented front end of web app using AngularJS, Bootstrap and helped with backend on Firebase.

UC Berkeley Computer Science Theory Group — Berkeley, CA
Undergraduate Researcher — *September 2018 - January 2019*

- **Majority Dynamics**: Explore bounds on stable groups appearing in social networks as highlighted in this paper (link). Programmatically compute convergence bounds using C++, optimizing calculations using Eigen and OpenMP.

Projects & Research

StatWiz: Statistics utility, reference, and calculator Android app made for students in introductory statistics courses.

Framework for Coping with Slow Environments that have Fast Approximations: Experimental agent-training framework for reinforcement designed for functioning in serial and parallel (submitted for CS 285).‡

Community Detection in Networks: Final project for CS 270 on community structures in social networks.‡

‡ Links on personal website.

Programming Skills

Languages: Python, Java, C, R, SQL, HTML/CSS, JS **Technologies**: Git, Perforce, NumPy/SciPy, LaTeX