Stella Kaval

stellakaval@berkeley.edu | 650-400-0746 | Berkeley, CA | https://www.linkedin.com/in/stellakaval

EDUCATION

University of California, Berkeley **Berkeley, CA**
B.A. Computer Science *08/2021 – 05/2025*
Coursework: Data Structures, Algorithms, Computer Architecture, Discrete Mathematics and Probability Theory, Computer Security, Principles and Techniques of Data Science, Systems and Devices, Web Development
Awards: Tapia 2023 Scholarship, Grace Hopper 2022 Scholarship, NCWIT Campus Representative

SKILLS

Languages: Java, Python, C, SQL, JavaScript, TypeScript, HTML, CSS, Assembly
Technologies: Git, Node.js, React, Angular, Express.js, NumPy, scikit-learn, pandas, Matplotlib, Vim, Figma, Sass, Jira

EXPERIENCE

Oracle **Redwood City, CA**
Software Engineer Intern *05/2023 – 08/2023*
- Developed a full-stack UX feedback tool within Oracle Cloud Infrastructure for 43,000+ internal user guides and 492 teams
- Created a feedback system using agile practices to collect user attributes (ex. ID, category, region) from individual services, facilitating efficient distribution to teams and resulting in ~2x speed improvement to issue resolution
- Implemented automated testing with Jest and Enzyme in the React-based application, with a focus on rendering components, handling UI interactions, and validating post requests, achieving 100% test coverage

UC Berkeley Electrical Engineering and Computer Science Department **Berkeley, CA**
Teaching Assistant, CS61B: Data Structures *01/2023 – 05/2023*
- Instructed 24 students each week through a 1-hour discussion and 2-hour lab section, covering topics such as Object-Oriented Programing, Data Structures, Sorting Algorithms, Trees, Software Development, Recursion, and Asymptotic Analysis
- Developed course materials, authored portions of the textbook, created challenging exam problems, proctored midterms, graded assignments, curated video content, and provided support during 3-hour weekly office hours for 1500+ student class
- Earned teaching effectiveness rating of 6.9/7, surpassing the departmental average of 6.3

UC Berkeley Data Science Discovery **Berkeley, CA**
Research Assistant *08/2022 – 01/2023*
- Collaborated with 3 students to conduct feature engineering analysis of 20+ homeless service provider attributes in California, employing SQL queries and data modeling techniques to identify key trends in service diversity and funding
- Used Tableau's data blending, aggregation, reshaping, and visualization capabilities to create and present 4 dashboards

Code for Fun **Palo Alto, CA**
Programming Instructor *05/2022 – 08/2022*
- Educated K-9 students 40-hours/week through game development-centered lessons in Scratch, Python, JavaScript, and Java

LEADERSHIP

CS Kickstart **Berkeley, CA**
Lead Director *01/2022 – Present*
- Organize an annual 1-week bootcamp for computer science to 100 female-identifying freshmen at UC Berkeley, executing a program growth of 150% in the past 2 years and maintaining an average satisfaction rating of 4.8/5
- Supervise a team of 25+ volunteers, manage a $100K+ budget, and negotiate strategic partnerships with 8 industry sponsors, including Meta, SAP, Jane Street, Splunk, and Aurora, to support the program's expansion
- Achieved a 30% increase in underrepresented student participation through impactful annual outreach events

PROJECTS

NumC, Computer Architecture CS61C Project **Spring 2023**
- Built a simple version of Python's NumPy in C for mathematical and logical operations on arrays and matrices
- Achieved 1750x speedup for exponential operations and 110x speedup for general multiplications through parallelization techniques and algorithmic optimizations such as SIMD, OpenMP, and Intel AVX intrinsics
- Applied debugging tools such as GDB and Valgrind, leading to identification and resolution of issues

Gitlet Mini Version Control System, Data Structures CS61B Project **Spring 2022**
- Utilized Java, Git, and Markdown to develop a version control system, consisting of 1000+ lines of code and 6 Java classes
- Implemented data persistence through serialization, leveraging Java's Object Output/Input Stream libraries and the BFS algorithm to efficiently serialize Java objects into binary data for seamless storage and retrieval of version control data
- Conducted 30 integration tests using JUnit, covering various functionalities such as commit, merge, and rollback operations