## RANJODH SINGH

Sacramento, California · (916)-512-7203 · <u>joda844@gmail.com</u> · <u>https://www.linkedin.com/in/rsingh844/</u> · <u>https://rsingh84.github.io/Portfolio/</u>

## **Technical Skills**

Coding: Python, C/C++, Java, Javascript, HTML5, CSS

Frameworks: React.js

Tools: Git, Github Actions, Eclipse, VSCode

## **Technical Projects**

## **Bullet Journal**

- Designed a course organizer for UCSD students by saving text and images for each date entry
- Co-designed the navigation of pages and layout of journals using HTML, CSS, and JavaScript in order to create a unique and responsive user interface
- Used Firebase to enable user authentication for web page application in order to retain user information

### **AI Sudoku Solver**

- Developed AI Solver written in Python which makes a decision and propagates the decision until the puzzle is solved
- Al will keep track of constraints of the puzzle in order to backtrack if a wrong decision is made

## **Huffman File Compression**

- Developed lossless data compression algorithm written in C++ in order to compress files and encode file descriptions
- Decompresses files by reading file header to build min heap of frequencies in order to decode bits

## **EDUCATION**

University of California San Diego, San Diego, CA

2022

#### **BS Computer Science**

- 3.7 Overall **GPA**
- Coursework in Data Structures and Algorithms, Object-Oriented Programming, and Software Engineering

# **TRAINING/COURSES**

### **CSE 110 - Software Engineering**

Learn software development and engineering methods, including specification, design, implementation, testing, and process and an emphasis on team development, agile methods, and use of tools such as IDE's, version control, and test harnesses.

#### **CSE 101 - Design and Analysis of Algorithms**

Design and analyze efficient algorithms with emphasis on nonnumerical algorithms such as sorting, searching, pattern matching, and graph and network algorithms.

### **CSE 100 - Advanced Data Structures**

High-performance data structures and supporting algorithms. Theoretical and practical performance analysis, both average case and amortized.