

Khang Nguyen

615-939-3645
KhangN44345@gmail.com
LinkedIn
GitHub

EDUCATION

University Of Illinois Chicago

Bachelor of Arts in Computer Science

Chicago, Illinois

Expected: Spring 2027

- Relevant Courses: Data Structures & Algorithms, Introduction to C, Discrete Mathematics, Linear Algebra

TECHNICAL SKILLS

Languages: C/C++, Python, Java, HTML, CSS, JavaScript, TypeScript

Frameworks: React, Node.js, Tailwind CSS, Flask, Vite

Tools: Git, GitHub, VS Code, Docker

Concepts: Data Structures & Algorithms, Object-Oriented Programming, Web Development, REST APIs

PROJECTS

DSA Visualizer (in progress) | React, Node.js, TypeScript, Vite, Tailwind, Framer Motion

- Developing interactive platform to visualize data structures and algorithms with real-time state changes
- Building modular React components with TypeScript for algorithm step visualization and user controls
- Implementing smooth animations using Framer Motion and responsive design with Tailwind CSS
- Designing complexity metrics display and step-by-step breakdowns to support multiple learning approaches

Red Light Camera Data Analysis | C++, File I/O, Data Structures

- Processed over 20,000 rows of Chicago traffic violation data using C++ classes, vectors, and file I/O operations
- Built modular system to filter, aggregate, and generate reports on violation trends by location and time
- Implemented input validation and performance considerations for large dataset processing
- Created structured summary reports with frequency analysis and data pipeline components

HalfGammon with Undo Stack | C++, Mersenne Twister RNG, Linked List

- Implemented two-player board game engine with 16 spaces and complex move validation logic
- Created undo functionality using custom linked-list snapshot system for game state management
- Integrated Mersenne Twister random number generation for die rolls and developed bearing-off game mechanics
- Built move validation system including blocked spaces, forced moves, and turn-based game flow logic

Totalistic Cellular Automaton Simulator | C, Simulation

- Built 1-D cellular automata simulator with 65 cells, 49 generations, and configurable rules (0–2186)
- Implemented cell transition logic using neighborhood sums and ternary rule sets
- Added cell activity tracking and visualization across generations with user-configurable initial states
- Developed rule configuration system allowing users to explore different mathematical patterns

EXPERIENCE

SparkHacks Participant

24-hour Collaborative Hackathon

UIC, Chicago, IL

Spring 2025

- Collaborated with team to develop and present project solution under time constraints
- Applied problem-solving and programming skills in a competitive development environment
- Worked with team members on rapid prototyping and iterative development approaches