Yash Garg

yash.garg1803@gmail.com | 469-336-0208 | yashgarg1803 | y-garg

Education

University of Texas at Austin Aug 2021 - May 2025
Bachelor of Science, Computer Science (Turing Scholars Honors) GPA: 3.75
Bachelor of Science, Math

- **Relevant Coursework:** Operating Systems, Algorithms, Data Structures, Concurrency, Machine Learning, Probability, Statistics, Game Theory, Randomized Algorithms, Stochastic Processes, Linear Algebra
- **Activities:** Undergraduate Computational Finance, Programming Contests, Math Club, Poker Club, Ultimate Frisbee

Work Experience

Jane Street Capital *(Propietary trading firm)* May 2024 - Aug 2024
Incoming Quantitative Trading Intern

Tokyo Electron Ltd *(Semiconductor processing equipment manufacturer)* May 2023 - Aug 2023
Software Engineering Intern on Data Applications team

- Built and designed client-facing Windows desktop apps using C# and .NET framework for nodes for an automated data workflow, including an interpreter for mathematical and logical expressions.
- Developed web application component of a dashboard (using Angular and Node.js) that creates charts and reports with interactive options and filters from users.
- **Tech:** C#, Windows Forms, .NET framework, SQL, TypeScript, Angular, Node.js, RxJS, HTML/CSS

Indeed *(Employment website for job listings)* May 2022 - Aug 2022
Software Engineering Intern on Cloud Infrastructure team

- Developed Python scripts to automate cleanup of AWS EC2 resources, particularly Amazon Machine Images (AMIs).
- Identified ∼3,000 unused AMIs for deletion and saved over 100,000 GB of memory costs per month.
- Discovered a redundant AMI flavor, saving time and resources by reducing the number of necessary builds by 25%.
- Assisted debugging Ansible playbooks and shell scripts of failed builds in Jenkins (CI/CD).
- **Tech:** Python, Boto3, AWS EC2, Docker

Projects

DoomOS | *C++, C, x86 assembly*

- Individually developed operating system kernel in C++ (without stdlib) that implements preemptive multithreading, synchronization primitives, memory management, file system reads, virtual memory, caches, and system calls.
- Extended with team of ∼50 to run multiplayer Doom game; led users/login subteam, and assisted in file system writes.

Kalshi Options "Arbitrage" Bot | *Python, pandas, scikit-learn, NumPy, matplotlib*

- Constructed probability distribution of S&P price using options-like contracts on Kalshi, an event contracts exchange.
- Compared probability implied by NYSE SPY options data with Kalshi distribution as signal in a trading strategy.

Amazon Review Rating Bias App | *Java (Spring), Python (Flask), Selenium, DynamoDB, Javascript (React)*

- Applied a Kalman filter on the rating of scraped Amazon reviews to test for bias and estimate 'true rating'.
- Extended by creating a Flask web app that determines true rating and bias of inputted Amazon product link.

RISC Pipelined CPU | *Verilog, ARM assembly, GTKWave*

- Built processor using Verilog that supports pipelining, two cycle read/writes, and back-to-back data dependencies.
- Implemented a small subset of ARM assembly instructions; debugged by analyzing wave traces in GTKWave.

Awards

USA Math Olympiad Qualifier: Scored in top 100 of nation. USAMO − 15; AIME − 12; AMC 12 − 136.5

High School Salutatorian: Ranked 2nd among class of 1361 students

Skills

Languages: Java, Python, C++, C, C#, JavaScript, Go, R, Verilog, SQL, HTML/CSS
Frameworks & Libraries: pandas, NumPy, matplotlib, Flask, unittest; JUnit, Selenium; React, Node.js; .NET