

# Vasily Ilin

Blog: [vilin97.github.io](https://vilin97.github.io) | Github: [Vilin97](https://github.com/Vilin97) | email: [vasilin97@gmail.com](mailto:vasilin97@gmail.com)

## Education

---

### University of Washington | 2020 - 2025

PhD in Mathematics.

Earned the McKibben and Merner Endowed Fellowship – a competitive \$10,000 fellowship.

### Boston University | 2015 - 2020

Master of Science in Computer Science | GPA: 4.0

Master of Arts in Mathematics | GPA: 4.0

Bachelor of Arts in Mathematics (Magna Cum Laude). Thesis: “[Stochastic Simulation Algorithms and Benchmarks](#).”

Earned the Marvin Freedman Prize for outstanding undergraduate research in applied mathematics.

## Skills & Certifications

---

**Technical Skills:** Python, [Julia](#), Java, Git, Markdown, [Lean](#), Data Structures & Algorithms, Complexity Analysis & Optimization.

**Certifications:** [Neural Networks & Deep Learning](#) (Coursera), [Triplebyte Data Scientist](#) – top 10%.

**Languages & General Skills:** Russian (native), German (elementary), blog maintenance, quantitative thinking, patience.

## Projects

---

### Deep Reinforcement Learning in DeepRacer | Spring 2022 | [code](#)

- Placed in top 3% out of 1000+ participants in a virtual race hosted by AWS.
- trained 80+ different models, [presented](#) our findings, wrote a [report](#).

### Google Summer of Code (Julia) | Summer 2021 | [code](#)

- Implemented, tested, benchmarked, and optimized algorithms to simulate jump processes, e.g. epidemics.
- Added 3 algorithms, 3 tests, 1 tutorial, and 1 benchmarking script.
- Improved run-time of space simulations by 70%.
- Contributed over 7000 lines of code to [DiffEqJump](#) and maintained a [blog](#).

### Fully Dynamic Connectivity (Python) | Spring 2020 | [code](#)

- Benchmarked the data structure described in [[Henzinger and King, 1995](#)]. Wrote a 15-page [report](#).
- The algorithm performs exponentially faster than BFS.
- Found a heuristic that improves the run-time 10 times.

### Object-Oriented Projects (Java) | Spring 2020

- Quest game: playable in the terminal, 30+ classes, 3k+ LOC, [code](#).
- Bank terminal: used JSON files for data permanence and Java Swing for GUI, 70+ classes, 11k+ LOC, [code](#).

### Dense Low Rank Subgraph (Julia) | Summer 2020 | [code](#)

- Developed 2 approximate and 2 exact algorithms for the Low-rank-subset problem. Wrote a 7-page [write-up](#).
- Proved the Low-rank-subset problem is polynomial-solvable for constant dimension.

### Data Mining Competition (Python) | Fall 2019 | [code](#)

- Placed second out of a class of over 100 graduate students in a data mining competition on Kaggle.
- Obtained ROC curve area of 0.902.

## Work Experience

---

### Data Engineer at Google (FlumeJava, SQL) | Summer 2022

- Implemented an end-to-end pipeline, from collecting Android logs to the SQL tables with insightful metrics.
- Improved the load time of dashboards with metrics by precomputing aggregation of 12 million logs.

### Academic

- Teaching Assistant for math and data science | 2020 - present
- Ran the second-year [math circle](#) for highschoolers. Worked through “[Mathematics and Politics](#)” book | 2020-2021
- Graded classes in abstract algebra, mathematical statistics, and differential equations | 2016 - 2020

## **Data Science Intern at Orion Health (Python) | Fall 2017 | [slides](#)**

- Developed an anomaly classifier algorithm for an AWS server based on timeseries with 250 features.
- Used PCA, K-means, and leave-one-out for visualization and clustering.

## **Graduate Coursework**

---

### **Computer Science**

CSE 537 – Artificial Intelligence: search, reinforcement learning, Bayes' nets, deep neural networks.

CS 591 – Object Oriented Design in Java.

CS 565 – Data Mining: k-means, hierarchical clustering, spectral methods, PCA.

CS 530 – Data Structures and Algorithms: fast matrix multiplication, graph algorithms, linear programming.

### **Probability and Statistics**

MA 581, 582 – Mathematical Statistics: law of large numbers, central limit theorem, generating functions, estimation theory.

MA 779, 780 (audited) – Probability with measure theory.

### **Applied & Computational Mathematics**

MA 565, 579 – Analytic and numerical methods for ODEs and PDEs in mathematical biology.

CS 535 – Turing Machines, uncomputability, complexity, randomized algorithms, interactive proofs, PCP theorems.

CS 537 – Randomness in Computing.

CS 591 – Graph Mining: graph sparsifying, probability tail bounds.