

# Dr. Elena Rostova

Associate Professor of Computer Science & Data Analytics

Department of Computational Sciences · Apex University

✉ e.rostova@apex.edu  
🌐 elenarostova-academic.org  
📍 Turing Hall, Suite 402  
☎ +1 (555) 019-2834

## EXECUTIVE SUMMARY & TEACHING STATEMENT

My pedagogical mission centers on transforming abstract computational theory into empowering, real-world analytical capability. By bridging rigorous computer science fundamentals with inclusive studio-based learning, interactive live-coding, and ethically conscious software engineering, I cultivate students who are not merely proficient coders, but critical thinkers, collaborative problem-solvers, and resilient digital architects. Over nine years of university instruction across undergraduate and graduate levels, I have maintained an average overall teaching evaluation of **4.88 / 5.00**, recognized with the Apex University Excellence in Undergraduate Teaching Award (2024).

## 1. Core Pillars of Teaching Philosophy

My teaching methodology is structured around four interconnected pedagogical pillars designed to engage diverse learner profiles, instill deep conceptual mastery, and foster long-term professional self-efficacy:

### Pillar I: Active Learning & Inquiry-Driven Studio Synthesis

Lectures should never be passive transmissions of static information. I structure every course around the **"Studio-Lab" model**, replacing standard multi-hour slide presentations with targeted 15-minute conceptual overviews followed immediately by peer-based collaborative coding sprints and interactive live debugging sessions.

In *CS 201: Data Structures & Algorithms*, for instance, rather than simply writing pseudocode for graph traversal on a whiteboard, students construct visual graph algorithms in real time using custom web tools, deliberately introducing edge cases (e.g., disconnected subgraphs, cyclic loops) to observe algorithmic breakdown. This hands-on inquiry transforms abstract mathematical properties into concrete mental models.

### Pillar II: Universal Design for Learning (UDL) & Inclusive Pedagogy

Computational fields often suffer from high barrier perceptions that discourage non-traditional and underrepresented students. I embed Universal Design for Learning principles into all curriculum design by providing **multimodal learning pathways**:

- **Multiple Modes of Representation:** Core concepts are presented simultaneously through formal mathematical proofs, visual architectural diagrams, and runnable open-source codebases.
- **Flexible Assessment Frameworks:** Major project deliverables offer choice in application domain (e.g., climate data analysis, bio-informatics sequencing, or algorithmic music synthesis), allowing students to connect core algorithms to personal intellectual passions.
- **Low-Floor, High-Ceiling Scaffolding:** Introductory assignments begin with intuitive starter code and step-by-step diagnostic checks, while optional extension modules challenge advanced students with research-grade optimization problems.

### Pillar III: Industry-Aligned Engineering & Ethical Synthesis

Modern software engineering demands more than syntactical fluency; it requires software hygiene, system architecture awareness, and societal responsibility. In upper-division courses such as *CS 412: Applied Machine Learning*, students operate in simulated agile teams utilizing industry-standard Git version control, continuous integration pipelines, and peer code review workflows.

Crucially, technical assignments incorporate **Ethical Impact Audits**. When training classification models, students must quantify demographic parity metrics, evaluate dataset bias, and submit a formal societal impact statement alongside their technical benchmark logs.

Pillar IV: Mentorship, Growth Mindset, & Student Self-Efficacy

Errors in computer science are not failures; they are diagnostic data points. I actively cultivate a "growth mindset" classroom culture where bug hunting is celebrated as a core learning vector. Through structured rubric feedback, low-stakes iterative draft submissions, and dedicated office-hour mentoring, I empower students to build independent troubleshooting capabilities and confidence in handling ambiguous technical challenges.

2. Teaching Record & Course Portfolio

I have instructed a broad spectrum of courses ranging from large-enrollment introductory core requirements to specialized graduate research seminars. The table below summarizes my primary teaching assignments at Apex University:

| Code   | Course Title & Description                                                                                          | Level            | Terms Taught  | Avg Size | Eval (/5.0) |
|--------|---------------------------------------------------------------------------------------------------------------------|------------------|---------------|----------|-------------|
| CS 201 | <b>Data Structures &amp; Algorithms</b><br>Core memory layout, trees, graphs, sorting, complexity analysis.         | Undergrad (Core) | Fall & Spring | 180      | 4.86        |
| CS 412 | <b>Applied Machine Learning &amp; Data Mining</b><br>Supervised/unsupervised learning, deep networks, PyTorch.      | Upper Undergrad  | Fall & Spring | 85       | 4.91        |
| CS 635 | <b>Advanced Distributed Systems</b><br>Consensus algorithms, Paxos/Raft, microservices, fault tolerance.            | Graduate         | Fall Term     | 38       | 4.94        |
| DS 105 | <b>Foundations of Data Science for All</b><br>Interdisciplinary intro to data literacy, Python, and visualizations. | Gen Ed / Intro   | Spring & Fall | 140      | 4.82        |
| CS 490 | <b>Senior Software Engineering Capstone</b><br>Year-long client-sponsored software synthesis & team delivery.       | Capstone         | Annually      | 45       | 4.89        |

3. Quantitative Teaching Effectiveness & Student Metrics

Student evaluation data collected consistently over the past five academic years demonstrates high student satisfaction, strong perception of learning outcomes, and instructor accessibility.

4.88 / 5.0

Overall Instructor Rating

Dept Avg: 4.31

4.92 / 5.0

Course Organization

Dept Avg: 4.38

4.95 / 5.0

Helpfulness & Access

Dept Avg: 4.42

4.87 / 5.0

Intellectual Challenge

Dept Avg: 4.25

3.1 Detailed Multi-Year Metric Breakdown (Scale 1.00 – 5.00)

The comparative matrix below illustrates survey evaluations aggregated across 1,200+ student responses against departmental and college benchmarks:

| Evaluation Metric Domain                                 | Dr. Rostova | CS Dept Avg | College of Sci Avg | Percentile |
|----------------------------------------------------------|-------------|-------------|--------------------|------------|
| Clear Explanation of Complex Concepts                    | 4.91        | 4.30        | 4.22               | 98th       |
| Fairness & Transparency in Grading Rubrics               | 4.89        | 4.35        | 4.28               | 96th       |
| Encouragement of Inclusivity & Classroom Discussion      | 4.96        | 4.41        | 4.35               | 99th       |
| Timeliness & Constructiveness of Project Feedback        | 4.85        | 4.18        | 4.12               | 95th       |
| Fostering Critical Problem-Solving & Analytical Thinking | 4.93        | 4.39        | 4.30               | 97th       |
| Overall Learning Value of the Course                     | 4.90        | 4.32        | 4.25               | 98th       |

4. Representative Student Feedback & Narrative Testimonials

Selected qualitative comments from anonymous end-of-term evaluations reflect the impact of these pedagogical strategies:

“Data Structures used to terrify me because I didn’t see myself as a ‘hardcore programmer.’ Dr. Rostova’s studio-lab approach completely transformed my perception. She breaks down complex graph traversals into intuitive visual steps and never makes you feel silly for asking questions. Her office hours were the single most supportive academic space I experienced at Apex.”

— Student, CS 201: Data Structures & Algorithms (Fall 2024)

“CS 412 was by far the best course of my undergraduate career. Integrating bias audits into our machine learning projects opened my eyes to the real-world consequences of software engineering. Dr. Rostova doesn’t just teach code; she teaches you how to be an ethical, thorough scientist.”

— Senior Computer Science Major, CS 412 (Spring 2025)

“In CS 635, Dr. Rostova treats graduate students as junior colleagues. Her breakdown of Paxos consensus was masterclass-level pedagogy. She created an environment where high-level architectural debates were productive, rigorous, and deeply engaging.”

— M.S. Student, CS 635: Advanced Distributed Systems (Fall 2025)

5. Student Mentorship & Academic Development

Beyond classroom instruction, I am deeply committed to advising, research mentorship, and championing diversity, equity, and inclusion in computational fields.

### 5.1 Undergraduate & Graduate Research Supervision

- **Ph.D. Thesis Advisor:** Marcus Vance (Graduated 2025; now Assistant Professor at Meridian Institute). Dissertation: "*Fairness-Aware Distributed Consensus in Heterogeneous Edge Networks*."
- **M.S. Thesis Advisor:** 6 Master's students completed thesis research under my supervision, resulting in 4 peer-reviewed workshop publications at IEEE/ACM venues.
- **Undergraduate Research Mentorship (REU):** Supervised 14 undergraduate researchers over 5 years. 8 students co-authored publications; 11 transitioned into top-tier Ph.D. or M.S. programs (e.g., Vertex Tech, Apex Graduate School).

### 5.2 Diversity, Equity, & Inclusive Excellence Initiatives

- **Faculty Advisor, Apex ACM-W Chapter (2022–Present):** Expanded chapter membership by 140%, organized annual hackathons targeting non-traditional coders, and established a peer-mentoring network for first-year women in STEM.
- **Coordinator, Apex CodeBridge Workshop:** Designed a free 4-week summer bridge program in Python and data literacy for incoming first-generation college students, increasing retention in introductory CS core by 22%.

## 6. Curricular Leadership & Pedagogical Development

I continuously refine course offerings to reflect emerging technical paradigms and pedagogical best practices:

1. **Total Redesign of CS 201 Core (2023):** Led departmental committee to modernize data structures curriculum. Replaced legacy C++ templates with modern memory-safe paradigms, created 12 interactive Jupyter lab modules, and built an automated feedback autograder utilized department-wide.
2. **Creation of DS 105 (Interdisciplinary Gen-Ed):** Authored new syllabus designed to provide non-STEM majors (humanities, social sciences, business) with actionable data literacy, ethical AI awareness, and introductory Python skills. Course enrollment grew from 35 to 140 within three semesters.
3. **Pedagogy Workshops & Faculty Development:**
  - Invited Speaker, "*Active Learning Strategies in Large Computer Science Lecture Halls*," Apex Faculty Development Symposium (2024).
  - Participant, *National Effective Teaching Institute (NETI-1)* Workshop (2023).
  - Co-Principal Investigator, Apex Instructional Innovation Grant (\$25,000) for integrating AI-assisted debugging tutors into introductory programming workflows.

## 7. Future Pedagogical Horizons & Teaching Goals

As computational tools and student demographics continue to evolve, I aim to expand my teaching practice across three primary vectors:

### 1. Integrating Generative AI as a Pedagogical Partner (Not a Shortcut)

Rather than banning AI assistance tools, I am developing exercises where students analyze, critique, and refactor code generated by Large Language Models. Teaching students how to systematically audit machine-generated outputs for edge-case failures, security vulnerabilities, and inefficiency prepares them directly for future industry realities.

**2. Expanding Cross-Disciplinary Computational Labs**

I am collaborating with the Department of Bio-Engineering and the Department of Digital Humanities to launch joint studio modules. Allowing computer science students to collaborate directly with humanities and biology majors on shared datasets builds essential cross-disciplinary communication skills.

**3. Open-Access Educational Resource Creation**

I am currently drafting an open-access interactive digital textbook for *Applied Machine Learning & Data Ethics*, making high-quality, ethically grounded computer science materials accessible to institutions worldwide without cost barriers.