Teaching Statement - Mohamad Latifian

In this statement, I outline my teaching philosophy, experience and interests, and describe how I plan to contribute to teaching in the Department of Computer Science at the University of Oxford.

Approach to Teaching

For me, teaching is one of the most rewarding aspects of an academic career. I have been passionate about teaching since high school, and throughout the years I have consistently had positive and motivating experiences in this regard. As an early-career academic, I have substantial teaching experience with diverse student cohorts and at multiple levels. I have served as a teaching assistant for several courses at Sharif University of Technology and the University of Toronto, primarily for core courses in algorithms, data structures, and theoretical computer science. In these roles, I held tutorials, designed theoretical and programming assignments, and contributed to the design and marking of exams. I have also had the opportunity to deliver lectures at the University of Edinburgh in the Algorithms and Data Structures course.

Over the years, this sustained engagement with teaching has led me to reflect carefully on different instructional styles. I have closely observed a wide range of instructors and teaching approaches, often taking notes on how course material is presented and how students respond. One conclusion I have drawn from these observations is that there is no single best way to teach. Effective teaching requires a toolkit of complementary methods and the ability to adapt them to the topic, the level of the course, and the background of the students. I therefore approach each class as a new problem to be solved. At the same time, I hold myself to a small set of core teaching principles that guide my work in the classroom.

Finding the right pace. Finding the right pace is central to effective teaching. In lectures, I set the pace assuming that students have completed only the stated prerequisites, with the goal of keeping the material accessible to the full cohort. At the same time, I introduce occasional more advanced questions during lectures without resolving them immediately. These questions are intended to spark curiosity among more advanced students and to encourage deeper reflection beyond the core material. I complement this approach by making myself available after lectures and during office hours. This provides a chance for students who may be struggling to clarify points from the lecture, while also giving more advanced students the opportunity to discuss alternative perspectives, or related problems. In tutorials, where teaching is necessarily more interactive and individualized, I adapt the pace more dynamically. I use students' responses and written work to calibrate the depth and speed of discussion, ensuring that all students are challenged appropriately while remaining engaged.

Keep the class engaging for everyone I place a strong emphasis on active student engagement in my teaching. During lectures, I regularly ask questions and encourage the audience to reason through ideas before complete solutions are presented. Rather than immediately resolving a problem, I provide hints that guide students toward the key ideas. When students arrive at an answer themselves, they tend to better appreciate the underlying reasoning; when they do not, they are often more motivated to understand the solution once it is explained.

I am also mindful that classroom participation can be dominated by a small group of confident students. To ensure broader engagement, I use a range of strategies, including offering multiple options and asking students to vote, waiting a little longer for other students to chime in, asking different students to answer different parts of a question, and, when appropriate, using anonymous in-class response tools such as Poll Everywhere or Wooclap.

Answer the "What" and "Why" before the "How". When introducing a new concept, algorithm, or tool, I first dedicate some time to clarify what it is and why it is needed, before delving into technical details. I place particular emphasis on this point, especially when teaching core techniques

that students may become proficient in using, without fully understanding their underlying purpose. In my experience, students engage more deeply with the material when they understand the motivation behind it and the problem it is intended to solve. For example, when teaching topics such as data structures or databases, I begin by discussing the limitations of naïve approaches and asking students why more structured solutions are necessary, before introducing specific solutions. Once students appreciate the purpose and common structure of these methods, they are typically more willing and better prepared to follow the technical development of how individual algorithms work, whether this involves proofs, algorithmic design, or implementation details. I apply this principle consistently across different teaching settings, using it to support sustained engagement and deeper understanding.

See the material from the students' perspective. When teaching material that I have worked with for many years, I am mindful that concepts which now seem straightforward were once unfamiliar and nontrivial. I therefore make a conscious effort to see the material from the students' perspective, and build the arguments incrementally and introduce the concepts in small, coherent steps. This helps students develop understanding gradually rather than feeling overwhelmed by technical details.

Be an active observer in the classroom. I view teaching as a shared process that requires sustained attention from both the instructor and the students. In recent years, and especially with the widespread use of social media, it has become harder for students to maintain focus for long periods of time. As an instructor, I try to be attentive to what is happening in the classroom and to respond when I sense that students' attention is drifting.

This can involve briefly revisiting a key idea, changing the pace or tone of the lecture, or giving a short recap of what we have covered so far. I am particularly careful during technically dense parts of a course, such as detailed proofs or algorithmic analyses. In these moments, I make a point of stepping back and reminding students how the current discussion fits into the overall structure of the topic, so that they do not lose sight of the big picture while working through the details.

Teaching Experience

My teaching experience spans a wide range of levels and educational contexts. I began teaching during high school through mathematics and informatics workshops, where I introduced advanced concepts to younger students in an accessible and engaging way. After receiving a Silver Medal in the National Olympiad in Informatics, I taught students preparing for the same competition, working with groups that varied significantly in background and prior preparation. These early experiences required careful pacing and adaptability and played an important role in shaping my approach to teaching.

I served as a teaching assistant for several core courses in algorithms, data structures, and theoretical computer science. During my undergraduate and master's studies at Sharif University of Technology, I was very eager to be involved in teaching, and in these positions my responsibilities included holding tutorials, answering questions on online forums, designing and marking theoretical and programming assignments, and contributing to exam preparation. During my master's studies, I served as a lead teaching assistant, coordinating the work of other teaching assistants in collaboration with the course instructor. This role gave me valuable experience with course organization and assessment. I later served as a teaching assistant for several courses at the University of Toronto, where I held tutorials, conducted office hours, and marked assignments and exams.

More recently, I delivered a lecture on network flows as part of the Algorithms and Data Structures course taught by Aris Filos-Ratsikas at the University of Edinburgh during the 2025-2026 academic year. A crucial take-away from this experience was that you have to be very patient when you give lectures. You might need to go over something a few times with different wordings until it clicks in students' minds. I believe that Aris trusted me to give this lecture because he had seen my teaching skills. My teaching style is adaptable: for more theoretical topics I often prefer a

classic board based approach, while for algorithmic demonstrations I use carefully prepared slides to illustrate examples or pseudocode. I devote significant time to preparing teaching materials and consistently receive positive feedback on the clarity and structure of my presentations.

Supervision. Alongside my research, I have supervised eight undergraduate students across four research projects over the past few years. These students, from Sharif University of Technology, approached me to work on research problems related to my work. Although these were undergraduate projects, the research was conducted at a high level. One paper is scheduled to appear at AAAI 2026, another is currently under submission to The Web Conference with very positive feedback, and two additional papers are being prepared for submission to IJCAI. In supervising students, my primary goal is to teach the process of research. I encourage students to read and present relevant papers, develop their own understanding of the literature, and gradually take ownership of their ideas. I meet with students regularly, provide detailed feedback, and help them refine both their technical arguments and their writing. My aim is to give students a realistic and rewarding experience of research and, when possible, to guide them toward producing work that is suitable for submission to top venues.

For graduate students, where the supervisory relationship is longer, I believe it is important to create a supportive and comfortable environment while also encouraging intellectual independence. One of the central roles of an adviser, in my view, is to help students develop confidence in their own ideas, so that by the end of their doctoral studies they are able to pose meaningful research questions and pursue an independent research agenda.

Overall, I am aware of the influence an instructor can have on a student's intellectual development and long-term academic trajectory. I take the responsibility of teaching seriously and aim to foster curiosity, confidence, and sustained interest in the material, particularly in technically demanding subjects. Guided by the principles and experiences described above, I am committed to contributing positively to teaching and supervision at the University of Oxford.

Teaching Interests

I would be happy to contribute by teaching courses at both departmental and college level. I am well prepared to teach all of the modules offered in the undergraduate and MSc courses (with some additional preparation for advanced modules completely outside of my research area), drawing on my extensive hands-on teaching experience with students at different levels. While my background naturally inclines me toward modules with a strong theoretical component, I find teaching courses out of my research area both engaging and rewarding. My teaching preferences include core theory courses such as Discrete Mathematics, Design and Analysis of Algorithms, Algorithms and Data Structures, Computational Complexity, Computational Game Theory, and Algorithmic Foundations of Collective Decision Making.

Beyond contributing to existing modules, I am enthusiastic about the opportunity to help design or further develop courses, where needed. In particular, I believe there is strong value in offering advanced algorithmic modules that cover topics such as algorithms for large-scale data, sublinear methods and approximation and randomized algorithms, which are foundational for students aiming to pursue research careers in theoretical computer science.

I am also keen to contribute to, or help develop, a dedicated module on the mathematical foundations of algorithmic fairness. With the rapid development of large-scale machine learning systems and increased attention to fairness in algorithmic decision making, this area has become central to modern computer science, with direct relevance to artificial intelligence, multi-agent systems, and computational economics. I envision such a module covering topics including fair division, fairness in voting and participatory budgeting, and algorithmic approaches to fairness in machine learning. The course could be designed to be accessible to students from a range of programs, including advanced computer science, AI-focused degrees, and joint mathematics-computer science pathways.