Andrei Migunov

Teaching Statement

Teaching is an immense honor and a high priority for me. It goes without saying that teaching is important for disseminating knowledge, and this is a core responsibility, but it's also one of the only 'official' avenues by which a student can discover the 'scientific worldview'. A good teacher can capture the imagination of a student and help them come to see more in something they're studying than what is being described by the textbook. A good teacher can bring to life the earnest desire to uncover truth.

Teaching Experience

My first teaching experiences were in my undergraduate years, as a tutor in moral theory, introductory philosophy, and discrete mathematics. However, tutoring requires one to follow, or fill in, another's teaching style. It's a limited form of teaching. My first more proper teaching experiences were during the first three years of my Ph.D. program, all of which I spent as a teaching assistant for various professors teaching an undergraduate (and, for one semester, graduate) course in the theory of computation. These offerings all required grading homework assignments and exams, preparing review sessions and recitations, and meeting with students in office hours. In all semesters but one, I lectured in the professor's absence and wrote those series of lectures myself. On some occasions, I would lecture for a week or more at a time, covering topics like automata, computability, the Turing model, and analog computation. I've also lectured for spans of a week or more in a class on analog computation and randomness.

An implicit duty in both lecturing and any other interaction with students is to be able to help students feel somewhat at peace before they walk out of the door. Students will sometimes walk into the room panicked because they are overwhelmed in general, or by the course material specifically. I've had a lot of panicked students, and I always remind them that there *is* time. There is time at the beginning of an exam to allow oneself three minutes to panic, and then to face the exam calmly. There is time throughout the semester to pause, eat a meal, and then shore up the knowledge one has accumulated. We can always give a healthy dose of reassurance to students about their ability to succeed in the course, while treating them as equals who have their own skills (and lives, for that matter) and who we can reasonably expect to put in their own effort. This, more than anything, has paid off in terms of course outcomes for students I've worked with.

Students also have all sorts of different backgrounds and life circumstances, and advice that claims to be general enough for the entire student body requires careful consideration and is limited in scope. One can't overly encourage a student to spend more time studying material if they have kids and a full-time job. One cannot rely on culturally specific forms of communication to convey ideas clearly to everyone all at once. My policy is, in general, that I will not be the arbiter of good and bad reasons for late assignments or low scores. I want to be forgiving and understanding of others' circumstances, and part of that will be to avoid penalizing individuals for their circumstances or making arbitrary decisions about what is and is not out of their control in life and thereby justified from the standpoint of the instructor. Instead, I will be available a large share of the time for

discussion, homework help, and exam preparation, and I will extend deadlines as much as needed while encouraging a certain pace set by the course, and clearly articulating that homework is the ultimate means by which one truly learns the material and prepares for the exam.

Teaching Methodology

There's nothing magical about teachers at the level of lecturing, other than that they can premeditate the presentation of some information and, above all, tell a story. It isn't magic, but lectures do lay much of the terrain for students' thought. Getting to know one's students can make it clear what the right mode of presentation is for certain material, and is even more essential at the level of individual interactions and office hour help sessions. It's not just that physics students, for example, love demonstrations of explosions and fire. They also love demonstrations of a connection between an explosion and something deeper about the world they inhabit, something that shows them that the *bang* is *about* the world! Every person has a basic right to the exciting uncertainty of the scientific process. A big part of that is coming to see that this process was, and really does continue to be, determined by creative impulse. One should strive to set a stage for that with homework and projects. A good way to accomplish this – and to escape a sometimes paternalistic teacher-student relationship – is to give students a chance to try their hand at a problem that is as real for their teacher as it is for them.

Another aspect of methodology that I consider important is historical context. In an interview on pedagogy, late sociologist Moishe Postone says[1] that the reason why the humanities sometimes focus on primary works even in undergraduate courses is because that's how students can be exposed to the problem that someone *believed* they were solving, when the tool, method, or idea we teach today was conceived. The same is true in our discipline. For example, the Turing machine is the result of *Turing's* circumstances, not of its logical reconstruction in theory. In practice, it is both. It helps students to know the origins of the tools they use and ideas they think about, and I make sure to discuss this historical context when possible. Historicizing course material is not a concession to lack of rigor, it's a foundation for rigor.

Tentatively, I'd like to structure a course in the following way: Weekly problem sets consisting of a few questions on fresh, recent course material. I would include medium-term (monthly) problems on which students can collaborate in small groups, or by themselves, and one or two long term problems that would be genuine contributions to the state of thought about the subject (at least within the class), or a long-term project with a creative component in the case of programming courses. I would include exams as required by policy or depending on the nature of any followup courses in the program (that is, I do not want to send students to more advanced courses unprepared). However, I would in general aim to write one fairly humble and straightforward exam that every student can approach calmly, testing basic ability to work with the ideas and tools of the subject and nothing more. I want to design courses that can easily accommodate shifts to online teaching as needed, without disrupting the continuity of the learning experience, to which end I would plan to have not just online lectures but engaging visuals and demonstrations prepared.

Teaching Interests

[1] "A theorist is trying to explain a phenomenon. By reading theory, students become aware of the phenomenon that the particular theorist is trying to explain, how this particular theorist goes about trying to grapple with this phenomenon and why they develop the kinds of categories they do in order to make sense of it. If you read theory like this, the categories themselves become a) up for grabs, b) something students become self-aware of, and c) something that students are taught how to spot, which is a skill." (M. Postone, in General Education After the Canon Wars)

I love teaching the theory of computation. I really enjoy showing students the limits of computation because that provokes them to try to transcend those limits. I like discussing computational complexity with students because complexity is a whole worldview in some ways, one that is properly computational. It really shows students the depth of meaning of the *notion* of computation, and introduces them to problems that show that the world (even just the computational one) is rich and deep. It's an avenue to philosophy and mathematics as well, because it lets students ask questions about the structuredness of things, and these two disciplines share that interest. Automata theory also brings students closer to formal languages and linguistics. All of these connections bring students from other disciplines to computer science, as well.

I am qualified and prepared to teach both graduate and undergraduate courses in theoretical computer science, automata, algorithms, computability, complexity, and more. I would be especially thrilled to have the opportunity to teach my own research in molecular programming and/or algorithmic randomness and dimension. I would especially enjoy designing a course on analog computation, its prospects, and its open problems. Someday I would like to teach an applied course in these topics. I am also reasonably familiar with object-oriented programming and have industry experience as a software developer, having worked on a mobile applications team at a large company for two years. I would enjoy teaching an introductory programming class on occasion.

Advising Approach

A sensible goal of advising is to help students achieve independence in research. It's necessary to strike a balance between orienting them in a productive research direction and helping them establish a genuine interest. I think it's important for me to be able to expand my own interests for my students, within some reasonably broad domain of my own research area. I also think it's important to be able to show a student how to reason about the novelty and impact of a potential result or sub-field, where to find ideas and inspiration, and how to obtain the technical skills to work in certain areas. The best way to do this is to meet weekly to discuss new research, and ideally for the student to participate in a lab and be able to collaborate with other students.