Teaching Statement

Konstantinos Mamouras

www.cis.upenn.edu/~mamouras

I find teaching and student mentoring extremely enjoyable and rewarding, and the prospect of a career that includes these components has largely motivated my decision to pursue an academic position. I am qualified to teach courses on programming languages, compilers, program logics and verification, as well as the theory of computation.

Teaching and mentoring experience

My teaching experience comes primarily from serving as a teaching assistant (TA) for several undergraduate and graduate courses during my PhD studies at Cornell. My TA duties included running introductory programming labs, designing the material for and then giving weekly recitations, grading programming assignments and homework problems, and holding office hours and tutorials. I have also given several lectures as a substitute for the instructor. Being a TA has been a very formative and rewarding experience. The students have given me very positive reviews, and their feedback has taught me a great deal about how to be a more effective teacher.

I served as a TA for several offerings of freshman and sophomore programming courses based on Java, where I had to deal with the challenge of teaching programming to a group of students with diverse academic backgrounds and prior experience. This experience formed my teaching philosophy about the value of connecting computer science concepts to real-world problems that all students can relate to. It was impressive to see how engaged students become during recitations when they realized that the material they had to learn was not just dry academic knowledge, but a valuable tool they could apply in their future scientific and engineering careers.

My TA experience also includes a senior-level theoretical course, taught by John Hopcroft, on the mathematics underlying the analysis of data and complex networks. The course placed a large emphasis on writing mathematical proofs and understanding statistical concepts, and it helped me understand the struggles of students who have not been exposed to rigorous mathematical writing. I devoted a significant amount of extra time tutoring weaker students. This experience helped me realize that allocating additional tutorial sessions for struggling students helps them keep up with the pace of the course and prevents them from dropping the class.

I was also fortunate to be a TA for a graduate course, taught by Dexter Kozen, on the theoretical foundations of programming languages. During this teaching assistantship I taught four lectures on the semantics of imperative programs and on partial-correctness logics. This course had a strong mathematical focus, but it also convinced me of the usefulness of implementation exercises. The students were asked to implement some programming language features in OCaml, and this process really helped them consolidate their knowledge and understand subtle concepts.

During my postdoc at UPenn, I had the opportunity to give two lectures in the Marktoberdorf 2016 Summer School on StreamQRE, a domain-specific language for stream processing I created, to around 85 PhD students. I also organized two hands-on programming labs for the students and helped them complete the assigned stream processing exercises.

Finally, I mentored a team of three senior undergraduate students at UPenn for their senior-level project. This was a significant year-long research project, in which I guided the students through the development of an efficient multi-threaded stream processing engine. This was a very instructive experience regarding how to channel the enthusiasm of the students and keep them engaged and productive. Initially, I had to provide significant assistance in order to help them understand the relevant literature.
and indentify the key technical questions. After a few months, their effort became more focused and I provided direction at our weekly meetings without having to manage the details. Eventually, the project led to very good technical results.

Teaching interests

At the undergraduate level, I would be excited to teach courses on programming languages, applied logic for program verification, formal methods, and compilers. I would also be happy to contribute in teaching programming courses at all levels, discrete mathematics for computer science, data structures, algorithms, and theory of computation.

I would be especially interested in teaching undergraduate programming courses for a wide audience beyond computer science majors that puts a focus on data science applications. There is increasing need to have basic data analysis skills for students across all engineering and science fields. Therefore, I believe that courses that teach fundamental computational skills and widely-used data processing tools in the context of data science applications can be extremely engaging and rewarding.

At the graduate level, I would be very interested in teaching fundamental courses on the foundations of programming languages, program semantics and logics, decision procedures and software verification, as well as a seminar on the use of programming language techniques in the context of data processing.