

# Teaching Statement

Arie Gurfinkel

I enjoy teaching. I believe that a teacher must guide students through the material and help them build the necessary foundation to not simply memorize the concepts presented in class but also to be able to extend and augment them as necessary for a practical application. When preparing a lecture, I try to facilitate this by highlighting connections between different concepts, showing not only why something works but also why and when it does not, and how theoretical results lead to practical applications. When lecturing, I urge students to ask questions, and if appropriate, design lectures with student participation in mind. I feel that by preparing a lecture, designing interesting and challenging assignments, and answering students questions, I learn at least as much as my students do. I find this process very rewarding.

I first discovered that I enjoy teaching in my second year as an undergraduate student when I was hired to teach an introductory programming course in a Software Testing Lab. This experience taught me that preparing a course requires a lot of work, even on the material that I myself found trivial. However, it has also showed how rewarding a well-prepared lecture can be from the teacher's perspective.

During my graduate studies, I had a variety of teaching experiences including working as a Teaching Assistant for graduate and undergraduate courses, developing and lecturing in a graduate course on Automated Verification, mentoring both undergraduate and graduate students, and presenting my work at conferences.

In the Summer of 2003 I have worked with my adviser Prof. M. Chechik to update her graduate course on Automated Verification to reflect the current state-of-the-art in this field, and specifically, model-checking. I was closely involved in designing the syllabus, preparing lectures, and designing assignments. In the Fall of 2003 as a TA for this course, I have delivered 4 out of 13 three-hours lectures that covered the automata-theoretic approach to model-checking. I have also worked closely with the students to help them pick course projects relevant both to their research interests and to the course material. Finally, at the end of the course, I have organized a mini-conference in which these course projects were presented in a formal setting. I am TAing this course again this year. I believe that this experience has prepared me to teach a graduate course in my area of research.

As a TA for a cross-listed course on Compilers and Interpreters (Spring 2005 and Fall 2005), my interaction with the students was mostly limited to office hours and grading assignments and exams. However, I have worked on redesigning the six-part course project on constructing a compiler to utilize industry accepted build tools, a revision control system, and a formal testing framework. I was also closely involved in designing questions for mid-term and final exams. At the end of the course, I was invited to give a guest lecture on Software Model-Checking, in which I tried to show the students how the techniques they have learned in class can be applied to a rather different area of software verification instead of compilation.

A teaching activity that I enjoy most is mentoring. As a graduate student, I was fortunate to supervise several, typically first or second year, undergraduate students who joined our research group for a summer project. Although their main task was programming, I have tried to make their projects both interesting and educational. Our daily meetings often evolved into discussions about good software engineering practices and mini-tutorials on design patterns, algorithms, and data-structures. I found that this experience was rewarding for both of us and many of the students have continued working with me throughout the year. In particular, Kelvin Ku, who was in his third year when he started his project, has continued the work as a fourth-year independent project course (CSC494) under my supervision, and has recently joined our research group as a graduate student.

Being one of the most senior graduate students of the verification research group, I also mentored several starting graduate students. I often met with new members of our group to discuss how scientific research should be conducted and helped them explore solutions to problems they have. The highlights of this experience were my collaborations with Shiva Nejati, Ou Wei, and Mihaela Gheorghiu. With all of these students, I was closely involved in their first research project in automated verification, and acted essentially as their primary adviser during the preparation of the results for a scholarly publication and a conference presentation. I believe that this experience helped me develop the skills necessary to advise graduate students at the university level.

I look forward to teaching the following advanced courses: Automated Verification, Logic and Automata, Software Engineering, Compilers, Programming Languages, Data Structures and Algorithms, Theory of Computation, as well as a variety introductory Computer Science courses.