TEACHING STATEMENT **MICHAEL J. LEWIS**

As the son of a professor and a high school teacher, I have been witness both to the commitment necessary to be an effective teacher, and to the rewards that the commitment can bring. The promise of a career that includes teaching drove me to pursue a PhD in computer science. As a teaching assistant and substitute lecturer during my time as a graduate student, I have been involved in all aspects of running courses, and I have experienced first-hand the challenges and joys of teaching. In my first year at Penn State, I taught six classes per week of an introductory programming and software tools course. I was also responsible for designing lectures, assignments, and exams, for holding office hours and working with students on an individual basis, and for maintaining and assigning student grades. The experience was overwhelmingly positive, and confirmed and enhanced my desire to make teaching an important part of my career.

Receiving research assistantships in all subsequent semesters as a graduate student (thereby exempting me from teaching duties) has not stopped me from being involved with teaching. I volunteered to be the head TA for a 2nd-year course in software engineering here at Virginia; for that course, I lectured to over 100 students on several occasions, managed four other TAs and six undergraduate graders, designed and oversaw labs, and won the Outstanding Teaching Assistant Award in computer science at UVa for the 1995-96 academic year. I have also been a "guest lecturer" on numerous occasions, including in graduate-level courses, and I have attended several teaching workshops at the Teaching Resource Center at UVa.

I feel that it is important to engage and excite students about the subject matter during class. Computer science is about problem solving, and problem solving is fun; but course material is too often presented in an expository manner, and professors sometimes leave their passion for their work in their office, instead of bringing it to the classroom. As much as possible, students should understand and work on problems before being presented with solutions. This breeds intuition and interest, allowing students to better comprehend and retain information, and to seek more.

It is also important to challenge the best students without leaving the weaker ones behind. I've found that the abilities and experience of computer science students in the same class vary widely, as much or more than in any other discipline. Therefore, professors should develop open-ended assignments that have a substantial minimum set of requirements, but that challenge the more ambitious and talented students to explore further.

The UVa computer science department has pioneered the practice of including closed laboratories in undergraduate courses at all levels. I have used this technique, I have seen it work, and I believe in it strongly. I also share the department's philosophy that students generally learn more by using modern tools to tackle realistic problems, than by working on toy problems with defunct "teaching" languages.

The speed at which changes are taking place in all aspects of computing requires that professors continuously revise and extend their course material to reflect modern advances. This is especially true of introductory programming courses and of graduate level courses in systems areas. All courses should conclude with at least a week's worth of lectures on relevant modern research directions, and on industry trends and recent developments. Furthermore, students should read important journal articles as soon and as often as possible, as part of their coursework. It's never too soon to learn how research is accomplished and disseminated.

I look forward to continuing to develop my teaching skills throughout my career as a professor, and to educating and exciting students about computer science, both in classroom settings and in individual student-advisor relationships.