INTERACTIVE NUMERICAL ANALYSIS

I teach interactively, and have done so for the past nine years. By interactive, I mean that my students submit extensive written work as well as presenting material in class. Depending on the level and nature of the course, the amount of material presented by students at the board during class ranges from 30 to 90any teacher, my methods are influenced by the courses I took over the years, many of which were taught by proponents of the "Moore" method.*

Despite the fact that this course is not taught in the tradition of the "pure" Texas method, the following anecdote concerning two students of Moore's accurately describes the tone of the course.

A problem had been outstanding for two semesters and both students wanted desperately to prove and present the theorem. One evening the first student called the second and said, "I'm tired of working on these problems; let's go out and have a beer!" After a night of drinking the students returned to class and Moore called on the second student, who stated that he did not have the problem. Then Moore called on the first student, who smiled and proceeded to present the problem, having solved it on the previous day! I remember these competitions from graduate school, and while I never went to the extent that Moore's student did to assure presenting a problem, just knowing that a colleague might get it first was always sufficient motivation to keep me working on a problem late into the night. I see the same healthy competition between my students, and I believe it drives them to a fuller understanding of the material.

The course.

For the 1995-96 school year at Nicholls State University, I taught a two-semester course entitled "Numerical Methods," an upper-level undergraduate or first-year graduate course. Having taught a one-semester version utilizing *Mathematica* and the classic text *Numerical Analysis* by Burden and Faires at the University of North Texas, I wanted to make some changes. The students had been excited about the programming and the technology, but the mathematics was not catching their attention. This was a definite case where "more" really turned out to be "less." Thus, I wanted the new course to utilize technology in a way where "more" was really "more." I wanted the students to be excited about the technology and the mathematics. The course I taught this year accomplished this and more.

I started with three major goals. I wanted the students to prove theorems, to write programs based on mathematical models, and to prepare papers describing the mathematical models and programs they had developed. The course met once a week for two-and-a-half hours, a difficult schedule at best, so I divided every class meeting into two parts with a flexible break somewhere near the middle. During the first half, I lectured on the mathematical theory necessary to prepare the students for their next programming assignments, involving the students with discussion and questions. During the second half, students presented problems at the board.

While lecturing, I made two types of assignments. The first type was the programming assignment, due the next week, to be solved and written utilizing technology. I made it clear that I did not expect industry-standard presentations (briefings) from them in the beginning and that they would develop these skills as the course progressed. However, I did expect everything they wrote to be mathematically correct, grammatically correct without spelling errors, and completely prepared utilizing technology. A number of "A"s and "D"s were given, along with lots of encouragement and advice, at the beginning of the course. By the end of the second semester a majority of the assignments turned in were A-quality work. They had learned the axiom that I learned in industry: "Tell them what you are going to tell them, tell it to them, and then tell them what you told them." They had also learned the detail, accuracy, and mathematical rigor required to convey material that they already understood. These briefings always included the mathematical theory behind the model, a simple example of the model for the reader, a computer simulation to determine solutions to the model, the results of the simulation, and conclusions based on the results. Many of these models came from Burden and Faires, and many came from problems associated with my own interests in solving differential equations using descent methods based on Sobolev gradients. Often a programming assignment would be resubmitted as additional theory led to coding enhancements.

The second type of assignment was material to be presented in class during the following weeks. These were generally theorems that I used during my lectures, theorems that were needed for the future lectures, and problems to clarify mathematical concepts used in lectures or programs. The students went to the board and presented mathematically rigorous arguments for the problems assigned previously. Problems that were not solved remained outstanding until solved. These problems varied considerably in difficulty. Enough easy problems were assigned that we never wasted class time because students did not have anything to present. However, certain problems were sufficiently difficult that they remained outstanding for weeks or months. Each class period I encouraged students to work on the difficult problems by conveying that not every problem would be weighted equally in the final grading scale and by asking questions such as, "Does anyone have problem 32 yet?! I sure look forward to seeing that one."

As mentioned earlier, the original goals were mathematical rigor, programming ability, and technical writing. As the course progressed I realized that I was achieving a fourth goal. I was preparing them for graduate-level courses in functional analysis and differential equations. The programs were heavily biased toward solving differential equations and therefore we were constantly in need of tools from differential equations and finite dimensional vector space theory. Because I was generally reducing theory that I approached from a continuous function space setting, I was phrasing all the finite dimensional vector space material in ways such that the proofs and theorems would generalize to infinite dimensional spaces. Occasionally, due to students' questions, we delved into the infinite dimensional setting and discussed such concepts as Hilbert spaces, bounded operators, and Gatou derivatives.

Judging by the attitudes of the students, the work turned in by the students, and the level of difficulty of the problems worked at the board, the class was a success. Furthermore, the students really enjoyed the course, commenting on their enjoyment of working the problems in class. Given the choice of several other courses, all the students in the class signed up for the second semester and for a different course I taught, despite the fact that no "A"s were achieved in the first semester. I believe these students left the course with much greater ability to prove theorems, present material in person, and prepare technical reports. The written component prepares them for research papers or industry briefings, while the interactive component allows them to present their work to their fellow students, a talent all mathematicians need in industry or academia. Oh, I almost forgot, they also learned a tremendous amount of the mathematics behind the numerical algorithms and picked up quite a few programming skills!

*W. Ted Mahavier (math-wtm@nich-nsunet.nich.edu) is an assistant professor at Nicholls State University in Thibodaux, Louisiana. His father, W. S. Mahavier, was a student of R. L. Moore, and his advisor, J. W. Neuberger, was a student of H. S. Wall. A large part of his education was guided by Moore's, Wall's, and Etlinger's "children" and "grandchildren."