Using MATLAB to improve student understanding of vector calculus

I work in the field of atmospheric sciences. My research involves analysis of data from observations and running numerical models. I teach two courses are in the undergraduate curriculum. One is Atmospheric Fluid Dynamics, which is a standard third/fourth year course in most atmospheric science programs. This course requires students to be well versed in vector calculus and basic physics and have some experience with differential equations. Students are required to complete the calculus sequence and a computer programming class prior to taking this course. Yet, year after year, I found the same issues in student understanding and preparation: Difficulties in interpreting concepts that involved vector calculus; in relating mathematical expressions with their physical meanings; and in writing computer programs to visualize atmospheric fields. This appears to be a problem not just at my institution. Informal discussions with colleagues at other atmospheric science programs reveals the same experience.

In order to address this learning gap, I proposed a new course with the primary objective of bridging the mathematics and atmospheric science courses. The working hypothesis being that if we introduce vector calculus, wave theory and computer programming to students with context-based examples drawn from atmospheric sciences, learning outcomes will improve. A critical aspect of this new course is a dedicated lab section focused on visualizing data using MATLAB. Students take this course in their third semester, and prior to taking any major upper-division atmospheric science course.

In the MATLAB portion of this course, we begin from the basics and assume little prior knowledge. The lab is typically taught by a teaching assistant with input from me in designing lab exercises. The lab exercises include both derivation of equations and visualizing the results. By the end of the course, the students can work with loops, arrays, subprograms and can plot 1-D data as well as 2-D fields.

The decision to use MATLAB was made in consultation with other colleagues in my department. Previously, students had a choice between C, FORTRAN or java. However, students faced the same problem as in calculus. Taking a computer programming course in large classes with engineering and general science students meant fewer, if any, examples from atmospheric science. Standardizing our computer language requirement was one step in helping students improve their programming skills and creating a shared experience.

At the time of writing, this course, in its current form with a dedicated programming lab, is being offered for the second time. While it is too early to make concrete assessments, there are indications of progress as measured from their programming abilities and improved understanding in the fluid dynamics class. More importantly, students also report positive experience regarding programming.

Several challenges, however, remain to be addressed. Students still appear to have an initial reluctance to learn programming. Not surprisingly, students with negative feelings about math in general have similar outlook towards programming. In our department, we are also attempting to expand participation of faculty so that students continue to use MATLAB in more courses. We remain motivated by the fact that computer literacy and programming skills are essential to the careers options for our students in the present and the future.