Teaching a Computational Course to Students with Diverse Science Backgrounds

Geology majors take the introduction to geophysics predominantly in their senior year. In addition, about a third of the students are engineering or physics majors. The course is elective. The geology and physics students typically have no programming/numerical experience beyond basic Excel skills. The class meets once a week for three hours. The class meetings generally start with a lecture and continue in the computer lab.

During the lab, students are given a geophysical problem from the topic that was discussed in the lecture. They are asked to write MATLAB code that will help them solve the problem and visualize the solution. As an example, in one of the labs, students are required to visualize the effect of placing a gold deposit at different depths below the surface and also in different host rocks. They are asked to compute the gravity anomaly caused by this gold deposit and plot this anomaly as a function of distance from the gold deposit. In another example, they are asked to visualize the thickness of the oceanic lithosphere as a function of its age, or how temperature changes as a function of depth. After they obtain and visualize results, students are then asked to provide a short summary of their results. The aim is for the students to analyze the mathematical model used to describe the particular problem, analyze its caveats and appreciate all the approximations that are used to formulate this model. By varying parameters such as depth or density contrast of the deposit, the students are prompted to think whether the numerical value they computed is sensible, or perhaps indicates a problem with their code. This approach enables students to learn computational methods in the context of Earth science. Since the Earth is a complex system, and in most cases simple analytical solutions are not adequate to simulate its behavior, students also gain appreciation for numerical simulations as a powerful tool to model processes in the Earth. Despite all the obstacles and steep learning curve associated with coding students react positively to this format of the course and find it easier to learn some mathematical concepts through “real life” applications. They also often say that those numerical simulations they learn enable them to better see “a big picture”. So this course format of blending computational methods and theory is beneficial for both learning computational methods and expanding students’ understanding of the Earth system.

Challenges I faced so far assessing student learning: Lab portion of the course constitutes 35% of the final grade. To earn the full credit for each lab students need to solve the geophysical problem, visualize the solution and write a short discussion of their results. To successfully complete lab assignments students need to demonstrate the ability to use principles of mathematics, chemistry and physics to solve Earth Science problems and also solve numerical problems utilizing MATLAB. Without prior computational experience students tend to give up when faced difficulties in debugging their codes and turn in half-completed assignments which show “some result” without any consideration of the meaningfulness of this result. I overcome this in part by offering extensive discussion of an assignment at the beginning of the semester and also giving students a short guide of some basic concepts in MATLAB. They can always consult this guide in tackling their code errors.

The challenges I face teaching a computational skills course far are the following: Lack of a geophysics textbooks appropriate for an undergraduate level in general, and in particular textbooks that include computational approaches. This is a particular problem in a class where students have diverse backgrounds (geology and engineering), the class is relatively large (25 students), but there is no TA to help either with guidance for programming, or providing background. The second challenge is the lack of institutional support – despite all the evidence for the necessity of teaching students computational skills for their professional careers, there is almost no support to add computational skills to the curriculum (across all the science departments at my institution).