Step Zero is Thinking — Don't Skip Step Zero!
Eliza Richardson
Penn State University

The students I have taught during my career, be they undergraduates, graduate students, mid-career high school science teachers, or my own children, all are well-served by learning and practicing the habits of mind of a problem solver. When I teach a computation course I emphasize techniques for approaching problems as much as I try to teach programming / computation / mathematical skills. Frequently teaching how to approach problems boils down to some simple (but not always easy) advice: think before you do anything else.

I teach my department's Computation in the Geosciences course, which is effectively a MATLAB primer (no prior coding knowledge of any kind assumed) that makes use of geoscientific data and problems as the lens through which to view MATLAB functionalities. Since I draw from many types of data and subfields when I create the exercises, students need to be able to solve problems without necessarily being experts in that subfield. It's equally about the approach, the "MATLAB way," and the content.

Sometimes students push back initially because the approach seems too slow, especially at the beginning of a course when the exercises are simple and there are fewer different ways to go about setting up the same problem. Later on, the habit of thinking before doing anything else proves useful and students frequently claim the problems on the problem sets must have gotten easier because they haven't quite internalized that they've gotten better at solving problems.

I try to walk them through initial questions with every problem such as, "What is the end result going to look like? (Is it a number? A plot? A relationship? A range of possibilities?) What do I need to do? (Do I need data? If I need data, do I have it? Do I know where to get it? Do I need to estimate anything? Do I need to set up an equation? Is there an analytical solution or am I numerically approximating? Am I doing a best fit? Is there a "known" relationship I am trying to verify? Is the solution to this problem going to let me tell an interesting story, or is it just about practicing solving something?) I emphasize that MATLAB is good for the hard work you can't do in your head, but it can't read your mind and it can only solve the problem you set up, so you need to take the time to think about how to set it before you start typing into the command window.

The end goal for me is that when students exit my course they never again say, "I don't even know where to start," when faced with a problem they've never seen before.