I teach quantum mechanics, thermodynamics, kinetics and statistical mechanics. These courses typically have problems that are very challenging and computational methods would complement the simplistic analytical problems typically presented. I am only getting started at trying to include computational skills in the classroom. My biggest challenge so far is that the students typically have little to no experience working with computational software so if a serious inclusion of computational work is to be attempted, a significant amount of effort needs to first be dedicated to teaching pupils a software package. Furthermore, not every student owns a computer they can bring to class that is capable of performing these types of calculations. Finally, I also see a problem with assessing their knowledge with computational tools at their fingertips as it is not clear to me how to address the possibility of academic dishonesty.

To date, I haven't included computational based problems in assessments. I have flipped my classroom where I have produced online lecture videos and preparation exercises so that class time can be used for students to work on problems and concepts in teams. I have used, and pushed students to use computational tools in the classroom where I am focused on student understanding of the class' learning objectives. Working in teams also tends to eliminate problems around certain individuals not having the equipment to do the computational problems as they can share with a peer. My assessments include problems that can be solved analytically as well as conceptual problems that were explored computationally in class to develop the intuition required to successfully answer them.

I also manage a research group that uses quantum chemical software to model chemical systems. The software is written largely in Python meaning that to interact meaningfully with the software, users need to know how to program in Python. Students typically come in to the group without any prior programming experience. I have found a Python programming book that teaches the basics of object oriented programming (Python for the Absolute Beginner by Michael Dawson ISBN:1435455002). The difference between this book and many others is that each chapter prompts the reader to write a game in Python using the topic of focus of that chapter. At every group meeting, new members who are working through the book present their game and we all play it. This provides a good venue to orally assess a student's understanding of the topic they just covered in a fun and inclusive way.