Computational Scientific Thinking

By Rubin Landau, Department Editor

It's hard not to take notice when Carnegie Mellon University's computer science department—one of the country's premier CS departments—and Microsoft Research—the premier software company—start up an institute with the catchy title of Center for Computational Thinking (CCT; www.cs.cmu.edu/~CompThink/). With Jeanette Wing's paper on the subject (www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing06.pdf) seemingly referenced by every third person in the computational science community, and Purdue sponsoring a series of workshops (SECANT: Science Education in Computational Thinking; http://secant.cs.purdue.edu/) in which even physicists and biologists had views to contribute, I couldn't help but wonder if there might be something more here than just a catchy phrase (not to discount the importance of catchy phrases helping premier departments find success with grant proposals). I mean, isn't computational thinking what all of us reading this magazine have been doing for a living for years? Granted, after spending days debugging and formatting code, we might feel like we do more computation than thinking, but in the end, we do like to think that we are truly *Homo sapiens*.

According to the CCT, "Computational thinking is a way of solving problems, designing systems, and understanding human behavior that draws on concepts fundamental to computer science. To flourish in today's world, computational thinking has to be a fundamental part of the way people think and understand the world. Computational thinking means creating and making use of different levels of abstraction, to understand and solve problems more effectively; thinking algorithmically and with the ability to apply mathematical concepts such as induction to develop more efficient, fair, and secure solutions; understanding the consequences of scale, not only for reasons of efficiency but also for economic and social reasons."

Well, as someone who has been teaching computational physics and computational science for more than a decade, I can't say that I disagree with these views, but I also can't say that they encapsulate my views of computational thinking. Of course, as a basic researcher and educator, my values, goals, prejudices, and measures of success differ from those of a computer scientist and so might be more accurately described as "computational scientific thinking." In fact, as a consequence of contributing to the Microsoft Research e-Science Workshop (http://research.microsoft.com/en-us/events/escience2008/) and of planning an honors seminar on the subject, I've gathered some thoughts on the subject and present them here in the hopes of putting more science into computational thinking. I would say:

- Computational scientific thinking (CST) is using simulation and data processing to augment the scientific method's search for the truth and for the realities hidden within data and revealed by abstractions.
- Concretely, CST is providing a coherent view of a natural system as the integration of data, theory, algorithmic model, and software implementation.
- Pragmatically, CST is learning the multiple disciplines needed to solve a problem and understanding them more deeply and efficiently by understanding them in context. This entails learning the human and computer languages of multiple disciplines, respecting the values of these disciplines, and trading in good faith.
- CST practitioners gain control of their working environments by having the confidence to look at and understand the insides of computing black boxes and by having the courage to be nonexperts on some parts of a problem.
- Computational scientific thinkers understand that it's more important to have the correct answer than the fastest answer and are willing to take on the hard work needed to obtain the correct answer.
- Computational scientific thinkers recognize that there might be uncertainties and indeterminacies in computing the correct answer and that some mathematical colleagues might not think that a computed answer is an answer at all, yet the thinkers understand that moving beyond analytic solutions to approximate ones is often more realistic and accurate than elegant exact solutions.
- CST is the appeal of pursuing new science in complexity rather than developing different ways to view the same simple systems. It is including new subjects in science curricula, such as continuous media, nonlinear phenomena, space-time correlations, integral equations, wavelets, principle component analysis, (signal processing beyond-Fourier), many-body theories, molecular dynamics, and imbued visualizations, for which computation is essential.
- In educational practice, CST might mean reversing the egalitarian trend of trying to make hard subjects more accessible by deemphasizing the importance of mathematics and abstractions. CST requires additional abstractions to understand and contribute to subjects such things as multidimensional representations of physical quantities and of data, and parallel and cloud

computing languages.

That pretty much summarizes my thought on computational thinking. I would truly appreciate hearing your thoughts on the subject, both so I can have something for future columns, and to help improve my planned seminar on the subject. And if any of you are interested in starting an Institute for Scientific Computational Thinking (something the NSF CPATH program might support), please let me know.

Rubin H Landau
rubin@science.oregonstate.edu