Computer Scientist
The 21st Century Renaissance Man

Luca Aceto and Anna Ingolfsdottir
School of Computer Science
Reykjavik University
Kringlan 1, 103 Reykjavik, Iceland

Since the Renaissance, figures like Leonardo da Vinci, Isaac Newton and Albert Einstein, to name but a few, have played a fundamental role in the development of our civilization. So much so that those exceptional human beings are part of the collective memory of humankind, and make guest appearances in everyday conversations amongst people who normally have little or no concern for art or science. Who doesn't know about 'Mona Lisa' or the famous fall of an apple, which, according to scientific lore, inspired Newton to discover the law of gravitation? Einstein, for one, stares at us from the front of T-shirts all over the globe.

People like Leonardo da Vinci, Isaac Newton and Albert Einstein are often referred to as 'Renaissance men' or 'polymaths'. According to a typical dictionary, a Renaissance man is a person who knows a lot about many different subjects, and has many practical skills and abilities. There is no doubt that the above-mentioned people were prototypical Renaissance men, but can a truly Renaissance man exist today? Many observers would argue that all human endeavours have become so specialized that our modern age cannot breed polymaths anymore. Our thesis in this short essay is that broadly educated computer scientists may be considered the 21st century Renaissance men. Let us state, at the outset, that we are *not* claiming that a well-educated computer scientist necessarily knows a lot about many different subjects outside her/his range of expertise. The kind of polymath nature that computer scientists possess is not related to a specific knowledge per se; rather it is methodological, and rests on the mental tool that computer scientists use to understand their world—be it imaginary, natural or artificial.

To quote from Jeanette Wing's paper [7],

> Computer science is the study of computation: what can be computed and how to compute it.

According to Gerald Sussman [2, Chapter 9], the essence of computer science is 'procedural epistemology'—the study of structure of knowledge from an algorithmic viewpoint, as opposed to the more descriptive point of view taken by classic mathematics.

Computer science gives us a framework for describing 'how to do things', and for presenting our knowledge in terms of precisely-defined processes—the *algorithms* that computer scientists use in their daily work. The so-called algorithmic method has proven to have definite advantages over more classic, purely descriptive approaches, and it has a range of applications that increases by the day.

A lot of experience has been gained on the applicability of the algorithmic method since Richard Bellman, the prolific mathematician who was the father of *dynamic programming*, wrote:

> Again the intriguing thought: A solution is not merely a set of functions of time, or a set of numbers, but a rule telling the decision-maker what to do; a policy.

Today, we would say that the solution is an algorithm. In fact, experience has shown that expressing our knowledge in terms of algorithms sharpens our understanding of the problems we are tackling considerably. Indeed, as Donald Knuth, a true polymath of our times, has often stated, a person does not really understand a subject until he has tried to teach it to a computer—that is, until one has expressed it in terms of algorithms.

But, does the algorithmic way of thinking pervade only computer science? The answer to this question is a resounding 'no', and it is precisely when looking at other sciences with the algorithmic lens of a computer scientist that his polymath nature comes to the fore. (Read, for instance, the highly entertaining essay [1] or watch the videos of the Louis Clark Vanuxem Lectures delivered by Avi Wigderson at Princeton University [6].) Indeed, the list of disciplines that are being influenced by computer science is steadily growing.

- Machine learning algorithms have changed the way statisticians work.
- Computational biology is having a profound influence on the way biologists think about the large masses of data they have available. Most importantly, computational methods may help us understand how living organisms function in isolation and in cooperation.

- Quantum computing is having a deep impact on the way cosmologists and physicists think about the universe—see, for instance, the book [4], which proposes the view that the universe is actually a giant quantum computer, and that information is just as fundamental a physical quantity as, say, energy is.

- Our understanding of how the human brain may work is being increased by the application of the algorithmic method. For instance, in his groundbreaking book [5], Leslie G. Valiant proposed a promising new computational approach to studying the workings of the human brain; the predictions made using Valiant's model have later been vindicated by research in the neurosciences.

Even human sciences like philosophy and sociology are not immune from the algorithmic treatment. As an example of the influence that computer science can have on philosophy, consider the concept of randomness or chance. In order to understand phenomena having to do with the algorithmic generation and transmission of information, computer scientists have developed a formal theory of randomness. This theory has philosophical significance since it offers us a well-defined framework within which to propose answers to age-old questions like, 'When is a phenomenon random?' and 'Can random phenomena be simulated by (efficient) deterministic processes?'

In the setting of the social sciences, exactly the same models and algorithmic tools that have led computer scientists to analyze efficiently the structure of the World Wide Web and to design search engines like Google are being used to shed new light on our social networks and interactions—see, e.g., [3]. Perhaps in our lives we are much more predictable than we like to believe, and the novel algorithmic view of sociology might bring this and many other things to light.

Overall, it is becoming abundantly clear that, as all sciences are increasingly data driven, a computer scientist's viewpoint will give anybody a significant advantage in shaping and understanding the future developments not only in science and technology, but also in our everyday lives. Moreover, the algorithmic lens used by computer scientists to understand computation can help one take a fresh and clarifying look at some of the basic questions with which humankind has struggled for millennia.

Studying computer science and learning about the algorithmic method is one of the best ways not to miss the bus for the future, and to join the club of the 21st century Renaissance men—the individuals who will play a key role in shaping our society in this millennium.

References and Suggestions for Further Reading

[1] Bernard Chazelle. The algorithm: Idiom of modern science. Available from http://www.cs.princeton.edu/~chazelle/pubs/algorithm.html.

[2] Committee on the Fundamentals of Computer Science: Challenges and Opportunities, National Research Council. *Computer Science: Reflections on the Field, Reflections from the Field*. The National Academies Press, 2004. Available from http://www.nap.edu/catalog.php?record_id=11106#toc.

[3] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: Membership, growth, and evolution. In Proceedings 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006. Available from http://www.cs.cornell.edu/home/kleinber/kdd06-comm.pdf.

[4] Seth Lloyd. *Programming The Universe: A Quantum Computer Scientist Takes On the Cosmos*. Knopf, 2006.

[5] Leslie G. Valiant. *Circuits of the Mind*. Oxford University Press, 2000.

[6] Avi Wigderson. A world view through the computational lens. Louis Clark Vanuxem Lectures, three talks given at Princeton University (cosponsored by Princeton University Press). Videos are available from http://www.math.ias.edu/pages/people/faculty/survey-talks/avi-wigdersons-survey-talks.php.

[7] Jeannette M. Wing. Computational thinking. *Communications of the ACM*, 49(3):33–35, 2006. Available from http://www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing06.pdf.