Fabio A. Schreiber

The Cultural Roots of Computer Science

Communications of the ACM – Forum
vol. 48, n. 11, Nov. 2005
pp. 11-12

The Cultural Roots of Computer Science

In the August 2000 *Viewpoint* column of CACM, Arora and Chazelle propose some remedy to an apparently paradoxical situation in which, while computers are pervading our everyday lives, Computer Science is facing a deep crisis in the U.S.. They attribute it to " *... our collective failure as educators, researchers, and practitioners to articulate a cogent, compelling narrative about the science of computing (as opposed to just the technology)*" and blame the reduction of the field to programming and possibly its heterogeneous nature among the causes of the demotivation of young students in enrolling in Computer Science curricula and of a reduction in research funding.

While I agree with the authors on the benefits of integrating Computer Science with other disciplines (including the sociological fallout), I think that it should be also very important to go the other way round, i.e. to show, by an epistemological approach, how other disciplines lead to the development of a computer application. A first step toward this direction is constituted by R. Karp's Turing Lecture; the author, by examining the historical development of a part of Discrete Mathematics, shows how "Theoretical Computer Science" results from a set of mosaic tesserae combining like in a puzzle.

I think that such an approach has even a further consequence. C.P. Snow brought to evidence, in a famous lecture, the divergence which developed between the humanistic and the scientific culture during the XIX and the XX centuries; since the end of the XVIII century, when the philosopher Kant and the mathematician Laplace joined their names in a cosmological theory, humanistic and scientific disciplines evolved in such a way that, while any cultured person knows who Shakespeare was and can mention a few of his works, very few humanists could even know the existence of the 2nd principle of Thermodynamics. Showing how propositional and predicate calculus derive from the Aristotle's syllogism and the logic programming paradigm from them; how developments from electron physics and abstract algebra give rise to modern integrated circuits; how theoretical linguistics and again abstract algebra merge to implement computer language translators, is a formidable challenge and may implement the link between the two cultures and end the climate of diffidence and contempt that still exists among many scholars.

I think that work in this direction could be of use to attract brilliant young people to Computer Science; in the eighties, I used it with good results for introducing Computer Science and Engineering to teachers in Italian high schools (Liceo Classico).

Fabio A. Schreiber