

How experimental is computing science?

Several years before I came to Austin I paid a visit to Cornell University, where the people informed me about a US movement to push computing science as an experimental science. They asked me for my opinion, and I remember Juris Hartmanis nodding in emphatic agreement when I told them that it struck me as absurd. By the way: it still does.

American computing science never having been for me a source of inspiration, I did not pay much attention to it. Moreover, with my background in physics, I had a way of understanding the phenomenon. You know, there are such things as experimental physicists. In the physics community in which I grew up, in fact the majority of the students in physics were diverted into experimental physics and only a select minority was allowed to specialize in theoretical physics. But from these student ratios no one ever concluded that physics was primarily an experimental subject! Of course not, because we all knew the reasons: the physics laboratory could make good use of the cheap labour provided by graduate students and, more fundamentally, in physics --like elsewhere-- the fountain pen is harder to handle than the soldering iron. Experimental physics was very much kept alive to provide an outlet for the less able students. Another reason to keep experimental physics alive I learned later: it is a way of getting lots of money. At the time I assumed similar mechanisms at work in American computing science, and I think that I was not completely wrong.

Since I arrived in Austin I discovered that the notion of computing science being an experimental one is quite widespread, a fact with quite strange consequences: for instance, the NSF acts as if --and, I presume, honestly believes that-- it supports computing science by providing the funding for supercomputers on a number of selected(!) campuses, and that while it is totally unclear how in the world the vicinity of a supercomputer can contribute to the health and vigour of a CS Department. [It is much more readily an impediment, and we should consider ourselves fortunate in that UT's supercomputer is located at the other end of town.]

The circumstances invited me to reconsider my judgement of the appropriateness of the notion of computing being an experimental topic. I did so, and my judgement did not change.

Proponents of experimental computing science have an argument, which is as follows. The original von Neumann machine was an imperative of the technology at the time, as indeed it was. In those days it was absolutely necessary to partition the machine into two distinct components, each of them meeting its own challenge, viz. the active logical engine, which had to be as fast as possible, and the passive store, which had to be as big as possible. These technological constraints are no longer as strong as they were, and therefore we should investigate alternatives, such as radically different architectures and equally radically different programming languages. Technological constraints are, in fact, now so weak that we have a wealth of possibilities to explore. Grading them, however, is way beyond our analytical abilities and the only effective way of singling out the most promising opportunities is by trying them out. Hence the intrinsically experimental way of approaching these current issues in computing.

So far, the argument is flawless. The question, however, is, how much it has to do with computing science. And, once more, the analogy with physics may be illuminating.

Physical measurements, no matter how easy or difficult to perform, are taken for mainly two completely different reasons. The one reason comes from technology, the other one from curiosity and scientific concern. A technical application might need some material with properties within strict and perhaps unusual limits, and if our theoretical understanding of how these properties are determined is insufficient to predict them, we have to measure possibly large numbers of candidates. Such measurements are not considered to contribute to the intellectual discipline called physics. The other reason for measuring emerges in the crucial experiment that could refute a proposed theory: theoretical physicists propose "laws of nature", but the physical subculture owes its vigour and respect to the fact that making such proposals is not a free-for-all, as it has adopted the discipline of designing and carrying out the potentially refuting experiment. The main role of such experiments is to maintain the intellectual integrity of an otherwise only speculative community.

And now we have reached the heart of the dilemma: the quoted argument in favour of experiments being fully granted, has this any bearing on computing science? It does if you accept as its role to answer as many questions related to computing as possible. It does not, if you require from an academic discipline the internal (conceptual and methodological) coherence referred to by terms like "insight". The question raised in the title, how experimental computing science is, has little to do with computing, but everything with our conception of science.

A scientific discipline is a module of human knowledge and abilities. But, as I wrote in 1974, "of course, any odd collection of scraps of knowledge and an arbitrary bunch of abilities [...] do not constitute a scientific discipline: for the separation to be meaningful, we have also an internal and an external requirement. The internal requirement is one of coherence: the knowledge must support the abilities and the abilities must enable us to improve the knowledge. The external requirement is one of what I usually call "a narrow interface": the more self-supporting such an intellectual subuniverse, the less detailed the knowledge that its practitioners need about other areas of human endeavour, the greater its viability."

Any mathematician of some fecundity can cook up a combinatorial game, the statistical properties of which are utterly untractable and unrelated to anything else. Mathematicians are fully justified in ignoring such games when their study has no significance beyond the game studied: their study is a scientifically barren activity. Proposed experiments in computing, for instance to compare the efficiency of alternative implementation techniques, explore a number of just such untractable combinatorial games. The outcome of the experiment may be of external interest, from the point of view of computing science the experiment is a barren activity.

Once it has been pointed out that computing science is much more an intellectual discipline than a collection of experimentally established facts, a utilitarian society may believe that the facts suffice and may decide not to be interested in computer science after all. According to all experience gathered so far, however, such a society would make a grave mistake, because in the production of tangible results --the kind of results the dyed-in-the-wool utilitarian recognizes-- the introverted, seemingly disinterested scientist has no equals. The most effective way of being useful is not trying to be useful but intertwined perfection of your knowledge and honing of your abilities.

Many CS departments in this country were founded before clear visibility of the outlines of the intellectual discipline we now call computing science.

In that sense they were erected prematurely. For lack of anything better they were no more than a cocktail of presumably computer-related topics, but certainly not related to each other. Subsequently we had the years during which in the columns of the Communications of the ACM people that knew what science is would argue that no such thing like Computing Science existed, and from the perspective of their own department they were probably right. Today those departments face the challenge of freeing themselves from their inadequate past: they have to recognize that eventually the intellectually discipline emerged. This is a tough job, for new intellectual disciplines are no more welcome than an unwanted child. Pushing the view of computing science as an experimental discipline can, alas, be interpreted as an effort at postponing the painful recognition.

The NSF funded supercomputers in an effort to cure the relative insignificance of American computing science; it did, however, prescribe a remedy without a proper diagnosis having been made.

prof.dr.Edsger W.Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712 - 1188
United States of America

(The above was written in October 1986, the decision to give it an EWD-number was taken on 30 November 1991.)