DOI:10.1145/2688497

Thomas Haigh

Historical Reflections
The Tears of Donald Knuth

Has the history of computing taken a tragic turn?

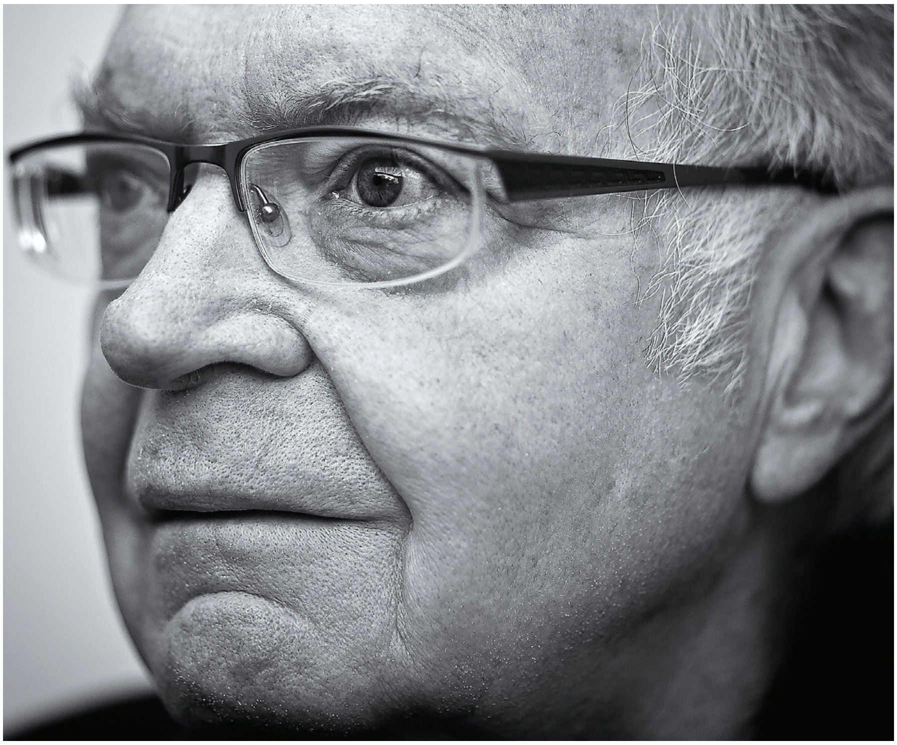
In this column I will be looking at the changing relationship between the discipline of computer science and the growing body of scholarly work on the history of computing, beginning with a recent plea made by renowned computer scientist Donald Knuth. This provides an opportunity to point you toward some interesting recent work on the history of computer science and to think more broadly about what the history of computing is, who is writing it, and for whom they are writing.

Last year historians of computing heard an odd rumor: that Knuth had given the Kailath lecture at Stanford University and spent the whole time talking about us. Its title, "Let's Not Dumb Down the History of Computer Science," was certainly intriguing, and its abstract confirmed that some forceful positions were being taken.[a] The online video eventually showed something remarkable: his lecture focused on a single paper, Martin Campbell-Kelly's 2007 "The History of the History of Software."[6,b] Reading it had deeply saddened Knuth, who "finished reading it only with great difficulty" through his tear-stained glasses.

What Knuth Said

Knuth began by announcing that, despite an aversion to confrontation, he would be "flaming" historians of computing. This, he worried "could turn out to be the biggest mistake of my life." The bout might nevertheless be seen as a mismatch. Knuth is among the world's most celebrated computer scientists, renowned for his ongoing project to classify and document families of algorithms in *The Art of Computer Programming* and for his creation of the TeX computerized typesetting system ubiquitous within computer science and mathematics. Campbell-Kelly has a similar prominence within the much smaller community of historians of computing but, even by Google Scholar's generous definitions, the paper that saddened Knuth has been cited only nine times.

Knuth then enumerated his motivations, as a computer scientist, to read the history of science. First, reading history helped him to understand the process of discovery. Second, understanding the difficulty and false starts experienced by brilliant historical scientists in making discoveries that specialists now find obvious helped him to



Donald Knuth.

PHOTOGRAPH BY RAJAN P. PARRIKAR

[a] See http://kailathlecture.stanford.edu/featured_speaker.html#abstract_bio.

[b] The video is posted at http://kailathlecture.stanford.edu/2014KailathLecture.html.

see what made concepts challenging to students and thus to become a "much better writer and teacher." Third, appreciating the historical contribution of non-Western scientists helped in "celebrating the contributions of many cultures." Fourth, history is the craft of telling stories, which is "the best way to teach, to explain something." Fifth, the biographies of scientists teach tactics for a successful and rewarding career. Sixth, history teaches how human experience has changed over time. As humans we should care about that.

Knuth also identified some special contributions to the history of science that professionally trained historians are uniquely well placed to make. We are good at "smoking out" primary sources and putting historical activities in the context of broader timelines. He also appreciates our ability to translate papers written in languages that he cannot himself read. He finds attempts at historical analysis "probably the least interesting" aspects of our papers but appreciates lengthy quotations from primary sources.

Things then headed in a less positive direction. Knuth explained that Campbell-Kelly had centered his paper on a table of important works related to the history of software published between 1967 and 2004. It coded the predominant approaches into four categories—one of which was technical—to demonstrate the technical approach had been dominant until about 1990, dwindling thereafter and vanishing altogether after 1997. Campbell-Kelly characterized this as an "evolution" away from "technical histories" of the "low-hanging-fruit variety" written by Knuth and other "outstanding technical experts" that were "constrained, excessively technical, and lacking in breadth of vision."

Knuth had previously viewed Campbell-Kelly as a kindred spirit but had now been granted a glimpse of "what historians say when they're talking to historians instead of when they're talking to people like me." Without pausing to dry his glasses he had written to Campbell-Kelly to accuse him of having "lost faith in the notion that computer science is actually scientific."

The shift described by Campbell-Kelly reflected a change in the population of scholars writing the history of computing. Many of the senior computing figures of the 1970s worked to preserve the history of the 1940s and early 1950s, starting with a number of "pioneer days" and workshops organized. The most important of these was held at Los Alamos National Laboratory in 1976.[15] Most of the 90 participants included in the group photograph of attendees were computer pioneers of the 1940s. Knuth himself contributed a detailed history of the first tools for "automatic programming" (assemblers and compilers). He was one of a handful of interested younger computer scientists who entered the field in the 1950s, which also included Edsger Dijkstra and Brian Randell, a systems programmer turned academic who had assembled an important collection of reprinted historical documents. At the conference were only a handful of trained historians. The editorial board of *Annals of the History of Computing*, which began in 1979 as a publication of AFIPS, a long-defunct umbrella group for professional computing societies, had a similar makeup. As graduate students in history and history of science programs began to write dissertations on computer-related topics they eventually inverted the ratio of trained historians to computer scientists, though the journal continues to publish a significant number of papers by computer scientists and technical experts.

So why is the history of computer science not being written in the volume it deserves, or the manner favored by Knuth?

In his lecture Knuth worried that a "dismal trend" in historical work meant that "all we get nowadays is dumbed down" through the elimination of technical detail. According to Knuth "historians of math have always faced the fact that they won't be able to please everybody." He feels that other historians of science have succumbed to "the delusion that ... an ordinary person can understand physics ..."

I am going to tell you why Knuth's tears were misguided, or at least misdirected, but first let me stress that historians of computing deeply appreciate his conviction that our mission is of profound importance. Indeed, one distinguished historian of computing recently asked me what he could do to get flamed by Knuth. Knuth has been engaged for decades with history. This is not one of his passionate interests outside computer science, such as his project reading verses 3:16 of different books of the Bible. Knuth's core work on computer programming reflects a historical sensibility, as he tracks down the origin and development of algorithms and reconstructs the development of thought in specific areas. For years advertisements for *IEEE Annals of the History of Computing*, where Campbell-Kelly's paper was published, relied on a quote from Knuth that it was the only publication he read from cover to cover. With the freedom to choose a vital topic for a distinguished lecture Knuth chose to focus on history rather than one of his better-known scientific enthusiasms such as literate programming or his progress with *The Art of Computer Programming*.

Computing vs. Computer Science

Here is where I part ways with Knuth's interpretation. Campbell-Kelly's article was "The History of the History of Software," not "The History of the History of Computer Science." Knuth's complaint that historians have been led astray by fads and pursuit of a mass audience into "dumbed down" history reflects an assumption that computer science is the whole of computing, or at least the only part in which historians can find important questions about software. This conflated the history of computing with the history of computer science. Distinguished computer scientists are prone to blur their own discipline, and in particular few dozen elite programs, with the much broader field of computing. The tools and ideas produced by computer scientists underpin all areas of IT and make possible the work carried out by network technicians, business analysts, help desk workers, and Excel programmers. That does not make those workers computer scientists.

The U.S. alone is estimated to have more than 10 million "information technology workers," which is about a hundred times more than the ACM's membership. Vint Cerf has warned in *Communications* that even the population of "professional programmers" dwarfs the association's membership.[7] ACM's share of the IT workforce has been in decline for a half-century, despite efforts begun back in the 1960s and 1970s by leaders such as Walter Carlson and Herb Grosch to broaden its appeal.

Computing is much bigger than computer science, and so the history of computing is much bigger than the history of computer science. Yet Knuth treated Campbell-Kelly's book on the business history of the software industry (accurately subtitled "a history of the software industry") and all the rest of the history of computing as part of "the history of computer science."[4] Others have written about the history of computer use in life insurance and other areas of business, the history of cybernetics, the history of the semiconductor industry, the history of punched card machines, the history of the IT workforce, the history of computer-producing companies such as IBM, the use and development of computers in particular countries, the history of the personal computer, and the history of computer usage in particular areas of scientific practice such as biomedicine. To call such work "dumbed down" history of computer science, rather than smart history of many other things, is to misunderstand both the intentions and the accomplishments of its authors.

The truth is that regrettably little history of computer science, whether dumb or deep, has been written by trained historians even though the history of computing literature as a whole has been expanding rapidly. Consider our output between 1990 and 2010. Michael Mahoney, a historian of science and mathematics at Princeton University, worked on a narrative history of theoretical computer science but ultimately produced only a set of provocative but schematic papers.[13] Mahoney was also interested in the history of software engineering, and several other historians have discussed the 1968 NATO Conference on Software Engineering at which that field was launched. Eminent sociologist of science Donald MacKenzie worked on the history of formal methods and its relationship to the development of computer technology.[11,12] Two books explored the history of DARPA and its role in shaping the development of computer science and technology, though Knuth would not approve of their institutional focus.[17,19] William Aspray wrote several papers on the history of NSF support for computing[2] and a book on John von Neumann.[1] A complete list would be longer, but not that much longer.

Historical Careers in Computer Science

So why is the history of computer science not being written in the volume it deserves, or the manner favored by Knuth? I am, at heart, a social historian of science and technology and so my analysis of the situation is grounded in disciplinary and institutional factors. Books of this kind would demand years of expert research and sell a few hundred copies. They would thus be authored by those not expected to support themselves with royalties, primarily academics.

Academic careers are profoundly shaped by the disciplinary communities in which they develop. Throughout their training, scholars are socialized into the culture of their field and pick up a wealth of tacit and explicit knowledge on what is expected of them. They learn how to select a research project, what kinds of work are noticed and which are ignored, what style to write in, how to structure a paper, which professors are respected, what search committees and grant review panels are looking for. This continues throughout their careers, as they aspire to prestigious awards, named chairs, or favors from the Dean. Whether they realize it or not, successful academics have internalized the rules of the game played in their particular field.

The history of computer science might be undertaken from two disciplinary base camps within academia: computer science and the history of science. Someone whose primary training is in history will naturally see the history of computing differently from someone whose disciplinary loyalty is to computer science. They will choose different topics and explore them in different ways for different audiences. For different reasons, outlined below, neither group has shown much interest in supporting work of the kind favored by Knuth. That is why it has rarely been written.

Prospects within the History of Science

The history of science is a kind of history, which is in turn part of the humanities. Some historians of science are specialists within broad history departments, and others work in specialized programs devoted to science studies or to the history of science, technology, or medicine. In both settings, historians judge the work of prospective colleagues by the standards of history, not those of computer science. There are no faculty jobs earmarked for scholars with doctoral training in the history of computing, still less in the history of computer science. The persistently brutal state of the humanities job market means that search committees can shortlist candidates precisely fitting whatever obscure combination of geographical area, time period, and methodological approaches are desired. So a bright young scholar aspiring to a career teaching and researching the history of computer science would need to appear to a humanities search committee as an exceptionally well qualified historian of the variety being sought (perhaps a specialist in gender studies or the history of capitalism) who happens to work on topics related to computing.

This, more than anything else, explains the rise of the broad and non-technical approaches decried by Knuth. Work in the history of computing has been seen by most in the humanities as dull and provincial, excessively technical and devoid of big historical ideas. Whereas fields such as environmental history have produced widely recognized classics that convince non-specialists of the scholarly potential, historians of computing are still inching toward broad acceptance of their relevance. The roles Knuth outlined for them would not serve them well as they were essentially those of the research assistant: gather primary materials, translate them if necessary,

Historical work on computing as a contribution to computer science versus as a contribution to a subfield of history or social science studies.

Historical work on computing framed primarily as a contribution to computer science.	Common in the 1970s and 1980s, less so today.	Almost impossible to accomplish (Campbell-Kelly's early work being an exception).
Historical work on computing framed primarily as a contribution to a subfield of history or science studies.	Fairly common, particularly for trained historians of computing working outside academia or in fields where history is not seen as central.	Increasingly common, with hopeful signs for further growth.
	Historical work on computing as a secondary interest or activity during retirement.	**Historical work on computing as the major focus of a scholarly career, for which one is hired or promoted.**

and make them available to computer scientists who will do the analysis.

Current enthusiasm for the "digital humanities" and the inescapable importance of computing to the modern world could provide opportunities. One day humanities search committees might even seek out historians of computing, but only those whose work engages with and appeals to scholars who themselves know nothing of computer science. In the meantime many scholars with doctorates in the history of computing have found work in museums or in academic employment outside both history and computer science, for example, in business schools, information schools, or specialist programs such as engineering education. These positions pose their own disciplinary challenges, but for obvious reasons provide few incentives to study the history of computer science.

Prospects within Computer Science

Thus the kind of historical work Knuth would like to read would have to be written by computer scientists themselves. Some disciplines support careers spent teaching history to their students and writing history for their practitioners. Knuth himself holds up the history of mathematics as an example of what the history of computing should be. It is possible to earn a Ph.D. within some mathematics departments by writing a historical thesis (euphemistically referred to as an "expository" approach). Such departments have also been known to hire, tenure, and promote scholars whose research is primarily historical. Likewise medical schools, law schools, and a few business schools have hired and trained historians. A friend involved in a history of medicine program recently told me that its Ph.D. students are helped to shape their work and market themselves differently depending on whether they are seeking jobs in medical schools or in history programs. In other words, some medical schools and mathematics departments have created a demand for scholars working on the history of their disciplines and in response a supply of such scholars has arisen.

As Knuth himself noted toward the end of his talk, computer science does not offer such possibilities. As far as I am aware no computer science department in the U.S. has ever hired as a faculty member someone who wrote a Ph.D. on a historical topic within computer science, still less someone with a Ph.D. in history. I am also not aware of anyone in the U.S. having been tenured or promoted within a computer science department on the basis of work on the history of computer science. Campbell-Kelly, now retired, did both things (earning his Ph.D. in computer science under Randell's direction) but he worked in England where reputable computer science departments have been more open to "fuzzy" topics than their American counterparts. Neither are the review processes and presentation formats at prestigious computer conferences well suited for the presentation of historical work. Nobody can reasonably expect to build a career within computer science by researching its history.

In its early days the history of computing was studied primarily by those who had already made their careers and could afford to indulge pursuing historical interests from tenured positions or to dabble after retirement. Despite some worthy initiatives, such as the efforts of the ACM History Committee to encourage historical projects, the impulse to write technical history has not spread widely among younger generations of distinguished and secure computer scientists.

To summarize, the upper-right quadrant in the accompanying table is essentially empty. It reflects historical work forming the backbone of a scholarly career and intended as a contribution to computer science. I share Knuth's regret that the technical history of computer science is greatly understudied. The main cause is that computer scientists have lost interest in preserving the intellectual heritage of their own discipline. It is not, as Knuth implies, that Campbell-Kelly is representative of a broader trend of individual researchers deciding to stop writing one kind of history and to devote a fixed pool of talent to writing another kind instead. There is no zero sum game here. More work by professionally trained historians on social, institutional, and cultural aspects of computing does not have to mean less work by computer scientists themselves. They cannot count on history departments to do this for them, and I hope Knuth's lament motivates a few to follow his lead in this area. Not simply because Knuth did it—few computer scientists have emulated him by procuring their own domestic pipe organs—but because his commitment to the intellectual history of computer science makes a powerful argument that historical knowledge of a particular kind is a prerequisite for deep technical understanding.

Reopening the Black Box

I will end on a positive note. In his paper, Campbell-Kelly offered a "biographical mea culpa" for his own early work that he now reads with a "mild flush of embarrassment." He came to see his erstwhile enthusiasm for technical history as a youthful indiscretion and his conversion to business history as an act of redemption, paralleling his own development and that of the field in a way that relied implicitly on a rather unfashionable conceptualization of history as progress along a fixed trajectory.

Contrary both to Knuth's despair and to Campbell-Kelly's story of a march of progress away from technical history, some scholars with formal training in history and philosophy have been turning to topics with more direct connections to computer science over the past few years. Liesbeth De Mol and Maarten Bullynck have been working to engage the history and philosophy of mathematics with issues raised by early computing practice and to bring computer scientists into more contact with historical work.[3] Working with like-minded colleagues, they helped to establish a new Commission for the History and Philosophy of Computing within the International Union of the History and Philosophy of Science. Edgar Daylight has been interviewing famous computer scientists, Knuth included, and weaving their remarks into fragments of a broader history of computer science.[8] Matti Tedre has been working on the historical shaping of computer science and its development as a discipline.[22] The history of Algol was a major focus of the recent European Science Foundation project *Software for Europe*. Algol, as its developers themselves have observed, was important not only for pioneering new capabilities such as recursive functions and block structures, but as a project bringing together a number of brilliant research-minded systems programmers from different countries at a time when computer science had yet to coalesce as a discipline.[c] Pierre Mounier-Kuhn has looked deeply into the institutional history of computer science in France and its relationship to the development of the computer industry.[16]

Stephanie Dick, who recently earned her Ph.D. from Harvard, has been exploring the history of artificial intelligence with close attention to technical aspects such as the development and significance of the linked list data structure.[d] Rebecca Slayton, another Harvard Ph.D., has written about the engagement of prominent computer scientists with the debate on the feasibility of the "Star Wars" missile defense system; her thesis has been published as an MIT Press book.[20] At Princeton, Ksenia Tatarchenko recently completed a dissertation on the USSR's flagship Akademgorodok Computer Center and its relationship to Western computer science.[21] British researcher Mark Priestley has written a deep and careful exploration of the history of computer architecture and its relationship to ideas about computation and logic.[18] I have worked with Priestly to explore the history of ENIAC, looking in great detail at the functioning and development of what we believe to be the first modern computer program ever executed.[9] Our research engaged with some of the earliest historical work on computing, including Knuth's own examination of John von Neumann's first sketch of a modern computer program[10] and Campbell-Kelly's technical papers on early programming techniques.[5]

The history of computer science retains an important place within the diverse and growing field of the history of computing.

Most of this new work is aimed primarily at historians, philosophers, or science studies specialists rather than computer scientists. However, it does not shy away from engagement with the specifics of computer technology or the detailed workings of the computer science community, reintroducing technical analysis along with continued attention to social, cultural, and institutional factors. Some of it may confirm Campbell-Kelly's prediction that the field will move toward "holistic" work integrating different approaches.

The history of computer science retains an important place within the diverse and growing field of the history of computing. Work of the particular kind preferred by Knuth will flourish only if his colleagues in computer science are willing to produce, reward, or commission it. I nevertheless hope he will continue to find much value in the work of historians and that we will rarely give him cause to reach for his handkerchief.

[c] *IEEE Annals of the History of Computing 36*, 4 (Oct.–Dec. 2014) is a special issue based on this work.

[d] Dick had earlier published "AfterMath: The Work of Proof in the Age of HumanMachine Collaboration," *Isis 102*, 3 (Sept. 2011), 494–505.

References

1. Aspray, W. *John von Neumann and the Origins of Modern Computing.* MIT Press, Cambridge, MA, 1990.
2. Aspray, W. and Williams, B.O. Arming American scientists: NSF and the provision of scientific computing facilities for universities, 1950–73. *IEEE Annals of the History of Computing 16*, 4 (Winter 1994), 60–74.
3. Bullynck, M. and De Mol, L. Setting-up early computer programs: D.H. Lehmer's ENIAC computation. *Archive of Mathematical Logic 49* (2010), 123–146.
4. Campbell-Kelly, M. *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry.* MIT Press, Cambridge, MA, 2003.
5. Campbell-Kelly, M. Programming the EDSAC: Early programming activity at the University of Cambridge. *Annals of the History of Computing 2*, 1 (Jan. 1980), 7–36.
6. Campbell-Kelly, M. The history of the history of software. *IEEE Annals of the History of Computing 29*, 4 (Oct.–Dec. 2007), 40–51.
7. Cerf, V. ACM and the professional programmer. *Commun. ACM 57*, 8 (Aug. 2014), 7.
8. Daylight, E.G. *The Dawn of Software Engineering: From Turing to Dijkstra.* Lonely Scholar, Heverlee, Belgium, 2012.
9. Haigh, T., Priestley, M., and Rope, C. Los Alamos bets on ENIAC: Nuclear Monte Carlo simulations, 1947–48. *IEEE Annals of the History of Computing 36*, 2 (Jan.–Mar. 2014), 42–63.
10. Knuth, D.E. Von Neumann's first computer program. *ACM Computing Surveys 2*, 4 (Dec. 1970), 247–260.
11. MacKenzie, D. *Knowing Machines.* MIT Press, Cambridge, MA, 1998.
12. MacKenzie, D. *Mechanizing Proof.* MIT Press, Cambridge, MA, 2001.
13. Mahoney, M.S. and Haigh, T., Eds. *Histories of Computing.* Harvard University Press, Cambridge, MA, 2011.
14. Matti, T. *The Science of Computing: Shaping a Discipline.* CRC Press/Taylor & Francis, 2014.
15. Metropolis, N., Howlett, J. and Rota, G.-C. Eds. *A History of Computing in the Twentieth Century: A Collection of Papers.* Academic Press, New York, 1980.
16. Mounier-Kuhn, P. Logic and computing in France: A late convergence. In *Proceedings of the Symposium on the History and Philosophy of Programming* (Birmingham, July 2012); http://bit.ly/15ZRPLx.
17. Norberg, A.L. and O'Neill, J.E. *Transforming Computer Technology: Information Processing for the Pentagon, 1962–1986.* Johns Hopkins University Press, Baltimore, MD, 1996.
18. Priestley, M. *A Science of Operations: Machines, Logic, and the Invention of Programming.* Springer, New York, 2011.
19. Roland, A. and Shiman, P. *Strategic Computing: DARPA and the Quest for Machine Intelligence.* MIT Press, Cambridge, MA, 2002.
20. Slayton, R. *Arguments that Count: Physics, Computing, and Missile Defense, 1949–2012.* MIT Press, Cambridge, MA, 2013.
21. Tatarchenko, K. *A House With the Window to the West: The Akademgorodok Computer Center (1958–1993).* Princeton, 2013.
22. Tedre, M. *The Science of Computing: Shaping a Discipline.* CRC Press/Taylor & Francis, 2014.

Thomas Haigh (thaigh@computer.org) is an associate professor of information studies at the University of Wisconsin, Milwaukee, and immediate past chair of the SIGCIS group for historians of computing.