RESEARCH ARTICLE | MAY 01 1995

Computer Scientist Addresses Grand Themes

David I. Lewin

Comput. Phys. 9, 248–249 (1995)
https://doi.org/10.1063/1.4823398

An Interview with

Donald Knuth

Computer Scientist Addresses Grand Themes

Donald Knuth's life work came to him while he was a graduate student. Now retired—at age 58—from the computer-science department at Stanford University in California, the 1979 National Medal of Science recipient expects to complete *The Art of Computer Programming*, his seven-volume (and growing) magnum opus, in the early part of the next century.

In 1962, when Knuth began thinking about the series, he was working on his doctorate in mathematics at the California Institute of Technology in Pasadena while supporting himself as a consultant to Burroughs Corporation. A publisher's representative proposed that he write a book on software creation. Recounting the incident in an essay in *Current Contents* in 1993, Knuth notes that he quickly sketched the outline of a 12-chapter book but was unable to put any significant time into the project until after he received his Ph.D. in June 1963.

"It was originally planned to be one book, then three, then seven," Knuth told *Computers in Physics*. By the late 1970s, he noted, over half the papers being published in computer science journals dealt with combinatorial algorithms. "Now Volume IV (*Combinatorial Algorithms*) has turned into three volumes itself." At the rate he was going, it would be at least 2025 before he completed the series; therefore, he decided that only by retiring early could he complete the task he had set himself.

With more than 475,000 copies of his books in print in English alone, Knuth felt himself financially secure. He officially retired from Stanford in 1993 but had moved away from active involvement in the computer-science department several years earlier.

Knuth still keeps an office at Stanford and has a fiercely protective secretary. A sociable person, the mathematician retains another tie to Stanford. "I hate the idea of a professor who just writes books, so I give a lecture about once a month to anybody who wants to listen on whatever subject I feel like," he said.

After growing up in Milwaukee, Wisconsin, Knuth attended Case Institute of Technology (now Case Western Reserve University) in Cleveland, Ohio. There, he enrolled with the expectation of majoring in physics but soon switched majors to mathematics. In the 1950s, people with mathematical aptitude were channeled into physics, he says, but the field was not for him. "In physics, it is too hard to know if your ideas are correct; astronomers can't go out and directly observe an exploding star," he said. "I loved the certainty you have in mathematics."

While an undergraduate, Knuth landed a part-time job sorting punched cards. Nearby was an early electronic computer, an IBM 650. Knuth was able to wangle access and began writing programs. At the time, there was little connection between mathematics and programming, though Knuth was later to be responsible for creating many of the links between the two fields. By the time he graduated from Case with simultaneous bachelor and master of science degrees, he was a skilled programmer.

He went off to Caltech to study pure mathematics with Marshall Hall, writing his thesis on finite projective geometry. Therein lies an irony, Knuth notes. "I've applied almost all of mathematics to computer programming *except* this field."

While working towards his doctorate, he married his college sweetheart and supported himself and his bride by his computer consulting work. "I wrote a compiler for Burroughs during the summer, and that paid for my honeymoon," Knuth said. Upon receipt of his doctorate, he joined the Caltech faculty as an assistant professor. Later he was promoted to associate professor.

In 1968, he found himself facing a major career decision. Caltech had offered him a promotion to full professor, and he had offers of full professorships in computer science at Harvard, Berkeley, and Stanford. Knuth chose Stanford because it had a well-established computer-science program that would allow him to concentrate on his work, rather than requiring him to take an active role in department-building.

At Stanford he continued to research and write *The Art of Computer Programming*, though the process of getting his book into print caused Knuth to undertake an 11-year detour, into electronic publishing and his other major contributions to computational science: TeX and Metafont.

The first volume of Knuth's magnum opus was published using traditional hot-type printing methods. By the time he was revising his manuscript for Volume II, the publishing industry had moved to optical (photo-) typesetting and was on its way to the digital typesetting that dominates today. Knuth was not pleased with the results—the printing industry had not given much attention to supporting mathematical typesetting during the transition from hot-type to photo-offset printing. To remedy this shortcoming, Knuth began work on a system to describe technical documents that would become TeX and a system for mathematically defining type called Metafont. "I thought it would take me a year, but it took me ten," Knuth said.

Asked how TeX differs from PostScript, now used by many laser printers and high-end typesetting devices, Knuth said, "PostScript is an engine that describes pages, while TEX is good for describing documents." The document description in TEX can be translated to PostScript to print out an individual page, he notes.

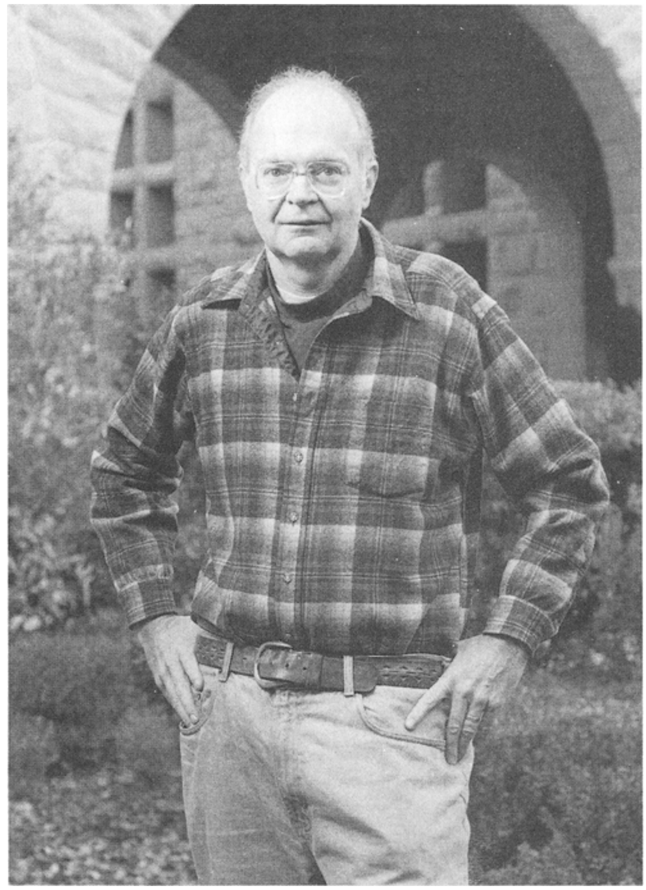
Originally, the TEX user would use the language to define the final appearance of a document, but he or she would not get to see the document's appearance until it was printed. Modern versions of the software allow the user to see the final appearance on the computer screen. Metafont, which was initially used for digital design of letters in a typeface, is a language for describing shapes generally, Knuth says. Though Metafont was originally aimed at letting mathematicians and scientists design specialized typefonts, Knuth soon realized that type design was best left in the hands of graphic-arts professionals. His own experience designing fonts for *The Art of Computer Programming* convinced him that it was no easy task to create fonts that were both readable and attractive. An extension to Metafont, called MetaPost, is a high-level language used for describing entire technical illustrations.

The document-description language TEX may be Knuth's biggest contribution to physics, he says, because 80% to 90% of all physics papers are written in TEX. Besides its utility, one of the program's great advantages is that Knuth placed it in the public domain. This openness helped speed the programming of TEX, he notes. "We had a tremendous amount of volunteer help because it was in the public domain."

"The power of making a beautiful book now is available to anyone...."

The software is among the most highly debugged available, with only two errors reported in the past two years. A small financial incentive helped this process: "I give a reward for anyone who finds an error in TEX," Knuth said. The reward started as a few cents per error, doubling each time until it reached some $327.68. At that point, Knuth stopped doubling the reward, calculating that it would soon increase to more money than was in circulation.

"Now that TEX is there and reliable, people can build on it," and they have done so, the mathematician says. "There are a lot of people whose livelihood is based on providing value-added to TEX," Knuth said. A number of companies now offer commercial versions of TEX, some with graphical user interfaces.



Putting TEX in the public domain was "absolutely necessary," Knuth says. "It struck me how the field of publishing was being held back by proprietary software," he said.

Placing his software in the public domain gave publishing power to people, including scholars from developing countries, willing to put in time rather than money. "The power of making a beautiful book now is available to anyone who has a beautiful idea," he said.

Such an idea—outside of mathematics and computer science—took the deeply religious Knuth on another detour from *The Art of Computer Programming*. In 1991 he published a thin volume, *3:16 Bible Texts Illuminated*, consisting of a verse (chapter 3, verse 16 wherever possible) from each of the books of the Bible, an approach he jokingly refers to as "The Way of the Cross-Section." He chose to focus on 3:16 verses because John 3:16—"For God so loved the world..."—is one of the most familiar Bible verses to Christians.

The book devotes four pages to each verse: one discusses the Biblical book it was taken from; another page contains the verse as interpreted visually by one of 59 internationally known calligraphers from 21 countries; and the last two pages survey what great theologians have said about the verse and its content. An exhibit based on the book—and its calligraphy of verses—has toured the world, Knuth says. "Putting this book together was a once-in-a-lifetime experience."

David I. Lewin