

## Love at First Byte

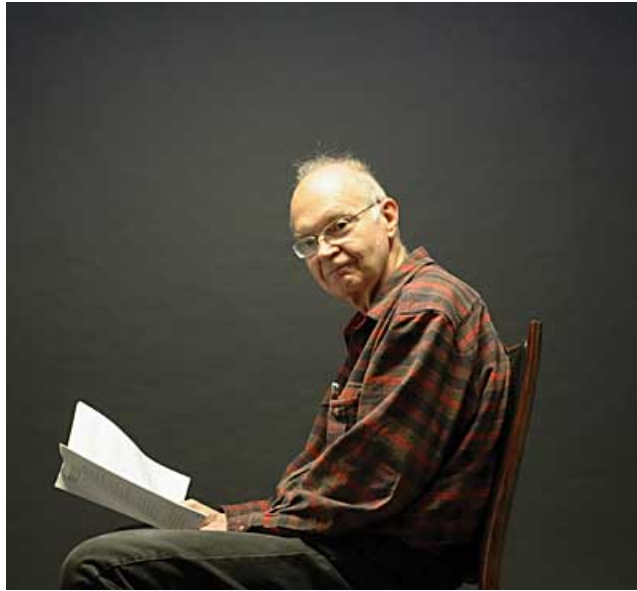
Among the many enduring passions of Donald Knuth, *The Art of Computer Programming* is only the one with the most pages.

BY KARA PLATONI

PHOTOGRAPHY BY TIMOTHY ARCHIBALD

---

IN 1957, a lanky, bespectacled college student named Donald Knuth caught a glimpse of a beautiful stranger through a window and fell deeply in love. The object of his affection blinked enticingly back at him. It was an IBM Type 650, one of the earliest mass-produced computers and the first Knuth had ever seen. Although computer science wasn't even really a science yet, Knuth was a goner.



As he would later muse in a memoir, “There was something special about the IBM 650, something that has provided the inspiration for much of my life’s work. Somehow this machine is powerful in spite of its severe limitations. Somehow it is friendly in spite of its primitive man-machine interface.” Knuth saw it as his passport to the new, man-made landscape of computer science, a world he would never tire of exploring.



Courtesy Donald Knuth

Retired after more than two decades of teaching at Stanford, Knuth, 68, lives on campus in a house he helped design. Most nights you can find him in his upstairs study, a cozy room lined with so many books and papers that he relies on a computerized index to tell him what goes where. Although he stands to work at his computers, Knuth writes longhand, reclining beneath the amber lamplight in a Dux chaise lounge whose upholstery is in its third incarnation and whose powers of comfort he deems “magical.” Usually he’s attending to his masterwork, a seven-volume series called *The Art of Computer Programming*, which, although only half-done, is widely regarded as the discipline’s ultimate reference guide.

In providing the literary backbone of computer science, Knuth, the Fletcher Jones Professor of Computer Science, emeritus, has racked up nearly every honor the discipline has to offer: the first Grace Murray Hopper Award in 1971, the Turing Award in 1974, membership in the National Academy of Sciences in 1975, the

National Medal of Science in 1979, and the Kyoto Prize in 1996. You would be hard-pressed to find a programmer who does not consider him the discipline's Obi-Wan Kenobi, the venerable Jedi master who is too humble to make a big deal out of being a big deal.



Charles Painter/News Service



NEVERENDING STORY: Knuth's office in 1984 (top) and his office today (below, right) show how the research for his computer series has expanded.

**BEFORE YOUNG DONALD** met the IBM 650, he was a physics major from Milwaukee at the Case Institute of Technology (now Case Western Reserve). His interest in physics was a departure from a long contemplated career in music: he played the organ, tuba, saxophone and sousaphone. Then one day, having missed the bus to band practice, he unraveled an extra-credit math problem so difficult that his professor had promised an automatic A in the course to anyone who solved it. By the year's end, he was the math major with the highest GPA in his class. Knuth published two scientific papers as an undergraduate, not counting his debut article that devised a system of weights and measures for *Mad* magazine. (Basic unit of force: the *whatmeworry*.)

Yet Knuth's switch to mathematics was nothing compared to the conversion inspired by the computer. Within a year of that "love at first sight" moment, he had written a user manual and instructional software for the IBM 650. The world at large first witnessed his programming skills in 1958, when the *CBS Evening News* covered a computerized "magic formula" Knuth had devised to crunch the statistics of Case's basketball players.

Knuth's next stop was the California Institute of Technology, where he completed his doctorate in mathematics and consulted for the Burroughs Corporation, which designed computers. He soon found himself at the forefront of a new discipline, albeit one then regarded somewhat warily by academics. "Mathematicians have a well-established field that's gone on for hundreds of years, and computer people were the new kids on the block," Knuth recalls. The young professor believed that computer science would enrich and reinvigorate math, the way physics had broadened it during the 19th century. Discovering that high-level math could be applied to computer science, he says, "was like discovering gold, or a new vein of stuff to be mined."

"He was basically the first great mathematician to take computer science seriously," recalls Christos Papadimitriou, a former Stanford colleague who now teaches at UC-Berkeley. "He did it by introducing rigor and elegance into programming."

Indeed, these have long been Knuth's watchwords. Rigor is providing mathematical proof that a program is correct. Where proof was a time-honored concept for math, Knuth says, in the early days of computer science people just wrote code, then fixed bugs wherever they popped up. Knuth wanted to verify that programs

worked in advance. Something is elegant, he says, if it is spare, memorable and pleasingly symmetrical; if it has the ease and immortal ring of an  $E = mc^2$ . Therein lies the art in computer programming.

**IN THE EARLY '60S,** publisher ADDISON-WESLEY invited Knuth to write a book on compiler design. Knuth eagerly drafted 3,000 pages by hand before someone at the publishing house informed him that would make an impossibly long book. The project was reconceived as the seven-volume *The Art of Computer Programming*.



Although Knuth has written other books in the interim, this would become his life's work. The first three volumes were published in 1968, 1969 and 1973. Volume 4 has been in the works nearly 30 years.

Its subject, combinatorial algorithms, or computational procedures that encompass vast numbers of possibilities, hardly existed when Knuth began the series. Now the topic grows faster than anyone could reasonably chronicle it. "He says if everyone else stopped doing work he would catch up better," deadpans Jill Knuth, his wife of nearly 45 years.

One of the reasons Volume 4 is so delayed is that Knuth slowed work to spend time with his family. "He started writing these books actually before we were married," Jill recalls. "He worked on it on our honeymoon. He had this idea that this project would be complete by the time our first child was born. Now our oldest son is 40." Their children, John and Jennifer, were born within a year and a half of each other, and trying to complete the series while caring for two babies was too much. "We had to sit down and say this is a lifetime project, and if we accept it that way it will go a lot better," she says.

Another reason: Knuth is the consummate perfectionist. "It's frustrating because I have high standards, he says wryly. "But the way I write, each page has at least 100 ways that it can be wrong. I don't just say something is 'faster,' I say it is '12 percent faster.' I'm always going out on a limb to give precision, and that takes a lot of time and a lot of cross-checking."

**ELEGANCE MAY BE** the ultimate reason for Volume 4's delay. Knuth adores beautifully designed textbooks, and in the late '70s, they were becoming an endangered species. As publishers were switching over from lead type to digital layout, or as Knuth once put it, "from metallurgy to bits," quality had slipped dramatically, particularly for technical publishing, which had long been derided by printers as penalty copy because it was so hard to do.

"The worst of it was the spacing, the way the letters would jam up against each other," Knuth says. "It was like if you took every letter and you wiggled it and made some of them go up and some of them go down. It wasn't random it was systematically bad." Because the letters in some words got smooshed together, it gave them the illusion of being darker than the others. The eye is naturally drawn toward dark spots, so the reader's focus would jump all over the page. By

1976, when it was time to print a second edition of Volume 2, Knuth could no longer stand to look at his own work, and he felt that other scientists were getting a similarly raw deal. “We didn’t want our papers just to be there, we wanted them to be beautiful,” he protests. “I wouldn’t have wanted to write *The Art of Computer Programming* if it was going to look ugly.”

Knuth took it upon himself to find a solution, even though his typesetting experience was limited to a high school job at a print shop and some experiments with the offset press his father had kept in the basement. Typesetting was no longer a manual craft. “It had changed into a problem of bits, zeroes and ones,” he says. “You put the one where you want ink on the page and zero where you don’t want ink. So I figured, okay, I’m good at zeroes and ones.”

In 1977, Knuth halted research on his books for what he expected to be a one-year hiatus. Instead, it took 10. Accompanied by Jill, Knuth took design classes from Stanford art professor Matthew Kahn. Knuth, trying to train his programmer’s brain to think like an artist’s, wanted to create a program that would understand why each stroke in a typeface would be pleasing to the eye. “I wanted to try to capture the intelligence of the design, not just the outcome of the design,” he says. For example, how do you insert line breaks into a paragraph so there isn’t too much space between words and so that most of the lines don’t end in hyphens? Although this seems like an aesthetic challenge to be solved by human taste, Knuth says, computers do it well. This is a combinatorial problem, he explains. “There might be a thousand ways to break a paragraph into lines and each way has a score.” His solution was to build a computer program capable of ranking the thousand options and picking the best one.

Usually a lone wolf, Knuth collaborated on his typography programs with some of the world’s best typographers and his students. He produced two software programs, the  $\text{\TeX}$  typesetting system and the METAFONT alphabet design system, which he released to the public domain. The programs are used for the bulk of scientific publishing today. “He made everybody’s life so much better and made the scholarly work so much more beautiful,” Papadimitriou says. “He has exported a lot of good will for computer science.”

Knuth was one of the first people to pronounce computer programming an art, believing that even the programs that crunch your taxes and produce your bank statements should be elegant. Like artists, he says, programmers have individual style and are propelled by intuitive leaps—Knuth knows he’s not really ready to solve a problem until he can think of it without using pencil and paper; his best ideas rise up from his subconscious while he swims. And like artists, he says, computer scientists are driven to produce. “Like a poet has to write poetry, I wake up in the morning and I have to write a computer program,” he says.

One of his most famous contributions to computer science is the idea of literate programming, that programs should be written so that people, not just computers, can understand them. They should be teaching instruments that explain themselves to subsequent programmers, making them more adaptable, less buggy and more enjoyable to read. An ideal program, he says, can be read by the fireside, like good prose.

Scott Kim, ’77, PhD ’88, one of Knuth’s doctoral students at Stanford who is now the brain behind Discover magazine’s Boggler puzzle, remembers how much Knuth emphasized enjoyment. “When it came to my dissertation defense, he said ‘This is a celebration, invite lots of your friends.’” Knuth is famous for his

playful touches: the clever epigrams at the head of his chapters, the practice of paying a hexadecimal dollar (that's 100 in base 16, or \$2.56 for us decimal-system folks) to people who spot mistakes in his work. The last time Knuth consulted his bankbook, he'd written out more than \$20,000 in checks for errata, although people rarely cash them—the checks are collectors' items. He likes to hide jokes in the index, as in Volume 3, where “royalties, use of” leads you to a page with an illustration of an organ-pipe array, a little wink to the 16-rank organ that dominates his home. He plays four-hands music with Jill, who swears that the neighbors tend to complain that the music emanating from their house is in fact not loud enough.

Most of all, Knuth, a devout Lutheran, is known for being kind. “By virtue of being near him, you feel you are smarter and a better person,” says Papadimitriou. However, people who bring this up in Knuth's presence tend to encounter what we will here dub the Knuth Modesty Paradox, in which the greatness of his achievements is inversely proportional to his willingness to admit them. He's made it clear that he considers himself less of an innovator than a cataloguer. “He once told me that his very talent is detail and he didn't consider himself all that brilliant otherwise,” Kim says. “His comment about playing the organ is you just hit the keys at the right time and the instrument plays itself.” No doubt he also believes that if you just hold the pencil long enough, the equations write themselves.

### HAS KNUTH'S LOVE AFFAIR

with computer science changed over the years? The man who essentially gave away two of his greatest creations,  $\text{\TeX}$  and  $\text{\METAFONT}$ , is troubled by the software industry's trend toward patenting everything, even the trivial. “This is a big detraction from doing creative work,” he says. “Imagine you are a writer and people have control over certain words in your vocabulary and you have to pay them a license to use their words!”

He wonders how long computer science, which is, after all, a man-made field of study, will continue to provide challenging problems that aren't too esoteric or derivative. “We're nowhere near the limits of interesting things to do—we're getting more every year than we had the previous year,” he says, yet muses, “How can we predict that that's going to continue, that we aren't going to have some kind of closure?”

But he's also very hopeful about the field's future—computers are more powerful



“ROYALTIES, USE OF”: What Knuth earned in computer science made possible a home pipe organ.

than ever, and the kind of cross-disciplinary cooperation he enjoyed with artists during his typography days is increasingly common. Just as he showed mathematicians that computer science would give them new room to explore, he hopes scholars from other fields will infuse computer science with fresh ideas. “The scientific world of the future will be pairs, or connections,” he says. “Everybody is going to be a bridge between specialties.” Fascinated by interdisciplinary overlap, Knuth is researching a book on how people computed before computers, finding the roots of binary thinking in places like the I *Ching* and ancient Greek poetry.

In retirement, he still writes several programs a week. He no longer advises students, but he hosts free public Computer Musings talks several times a year, drops in on graduate-level courses occasionally, and bikes to campus most days of the week to use the libraries or swim at the aquatic center. He famously regards email as a time sink and no longer reads it—except for correspondence (printed out by a department secretary) relating to errata or his current work. His primary agenda item is, after all, Volume 4, although every few months he’ll browse journals for items to squirrel away for Volumes 5 through 7. People inquire delicately about how he expects to complete a seven-volume project at his current pace, and Knuth concedes he’ll be content to finish the first five. Those are the core of his series, he says, and the last two are more specialized topics. If I get to the end of Number 5 and I see nobody’s yet written what I want to say about Number 6, I’ll continue. Despite the enormous changes in computer technology since he began his series, the English editions of each of the first three books still sell about 3,000 copies annually. The books have been translated into 10 languages, with more on the way.

Those around him also seem unfazed by the enormousness of his task, including his extraordinarily patient editor (“When I actually present him with something, he’s surprised,” says Knuth) and Jill, who recently tried to urge deadline compliance with the help of stickers on a chart. She gave up when her husband overshot every goal. “Because he is a perfectionist, he is often tempted to go off on a sidetrack,” she says. “The whole typeface thing was one of those branches, and that’s been beneficial to a lot of people. Nobody can say he shouldn’t have done it.”

For a guy who’s devoted to exploring, this is fitting. “The journey is more important than the destination that’s part of life,” he says. “If you only live for getting to the end, you’re almost always disappointed.” Perhaps, he adds, what seems like a liability is really a stroke of luck. “If I had been good at making estimates of how long something was going to take, I never would have started.”

His hair is whiter than when he began his journey, and his sideburns more brushy, but he is still the curious soul who has devoted his life to explaining humans and machines to one another. As the rest of the neighborhood sleeps, Knuth gets up from his magically comfortable seat and takes the first volume of *The Art of Computer Programming* from the shelf. He reads the dedication aloud: “This series of books is affectionately dedicated to the Type 650 computer once installed at Case Institute of Technology in remembrance of many pleasant evenings.” He closes the book, looking a bit misty. “It brings tears to the eye,” he says softly.