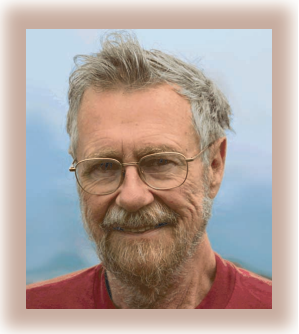
Senior Lead Editor: Dale Strok ■ software@computer.org

From Goto-less to Structured Programming: The Legacy of Edsger W. Dijkstra



Peter P. Chen

Did you know that Edsger Dijkstra just passed away?" When I first heard the news, I was shocked. "How could this happen?!" I asked myself. Last year (but it seemed just yesterday) when Edsger and I sat together on a small bus leaving the Software Pioneers Conference in downtown Bonn, Germany, to return to our hilltop hotel, he looked healthy. On that bus ride, we discussed the interesting difficulties he had faced early on in his career. Although he had run into obstacles, he could now laugh about them since he had discovered ways to overcome them and change history. Today as I recall his face and soft voice, I see the smile of a man who revolutionized software development.

A short biography

Edsger W. Dijkstra was born in the Netherlands in 1930 and died 6 August 2002. After receiving his PhD in computing science from the University of Amsterdam, he worked as a programmer at the Mathematical Centre in Amsterdam, a math professor at Eindhoven University of Technology, a research fellow at the Burroughs Corp., and the Schlumberger Centennial Professor of Computer Science at the University of Texas at Austin. He received the ACM Turing Award in 1972.

I am a "programmer!"

The first time I saw Dijkstra was when he delivered his Turing Award acceptance speech. I was one of the many people who stood for an hour to listen to that speech because all the seats were filled. It was worth it: he told us a story I will never forget.

When he applied for a wedding license in 1957, Dutch law required him to declare his profession. Filling in the form, Dijkstra stated he was a "programmer." The Amsterdam authorities claimed there was no such profession and rejected his initial application. As a result, his marriage certificate stated his profession as "theoretical physicist." What struck me 30 years ago and still resonates in my mind today is how Dijkstra was *proud* to be a programmer instead of a theoretical physicist. This is the kind of person software development needs; being proud of one's profession is one of the most crucial psychological steps toward better-quality work.

Major contributions

Dijkstra's most famous paper is probably "Goto Statement Considered Harmful" (*Comm. ACM*, Mar. 1968, pp. 147–148), which brought considerable attention to the problem of software developers' careless usage of the Goto statement. As a result, programmers today use it more carefully or not at all.

In 1972, Dijkstra published "Notes on Structured Programming" (*Structured Programming*, O.J. Dahl, E.W. Dijkstra, and C.A. Hoare, eds., Academic Press, 1972). This triggered the Structured Programming movement, which helped many of us improve our practices.

A survey of more than 1,000 college professors identified the 38 most influential papers in computer science (*Great Papers in Computer Science*, P. Laplante, ed., West Publishing Co., 1996; www.csc.lsu.edu/~chen/greatpapers.htm), and Dijkstra authored five of them. In June 2001, at the Software Pioneers Conference, about 1,200 software professionals saw Dijkstra speak for the last time. Fortunately, that speech is preserved in streaming video (www.sdm.de/conf2001/index_e.htm) and book/DVD (*Software Pioneers: Contributions to Software Engineering*, M. Broy and E. Denert, eds., Springer-Verlag, 2002) format.

Edsger Dijkstra is one of the most influential figures in computer science. His teachings (www.cs.utexas.edu/users/EWD) will resonate through the work of software developers for many years to come.

Peter P. Chen is the Foster Distinguished Professor at Louisiana State University and the inventor of the Entity-Relationship model. Contact him at pchen@lsu.edu.