

MEMORIAL RESOLUTION

ROBERT W. FLOYD

(1936-2001)

Robert W. "Bob" Floyd, Professor Emeritus of Computer Science and one of the all-time great pioneers in that field, died on 25 September 2001 after a long illness. He was 65 years old, mourned by his numerous colleagues, his mother, two ex-wives, a daughter, and three sons.

Although Bob was born in New York City on 8 June 1936, his parents moved more than a dozen times before he went to college. A child prodigy, he read every book he could get his hands on. His first grade teacher used him to teach fractions to the older children, in the small Southern schoolhouse where his education began. He built flying model airplanes, once carrying a large one on his lap during a very long train ride, so that it wouldn't be left behind in one of the frequent moves.

He received a scholarship to enter the University of Chicago at age 15, in an experimental program for gifted children, and received a Bachelor of Arts degree in 1953 at age 17.

Like almost every other child who has been in such accelerated programs, he soon lost his taste for school and continued to study only part time. Meanwhile he got a job at the Armour Research Foundation of Illinois Institute of Technology, first as a self-taught computer operator and then as a senior programmer and analyst. He received a B.S. degree in Physics from the University of Chicago in 1958, and published a conference paper about radio interference that same year. But it soon became clear that computing was his main love. Indeed, he began publishing articles in computer journals in 1959; and his first major paper, "A descriptive language for symbol manipulation" (1961), introduced an important concept that became known as "Floyd productions."

In 1962 he became a Senior Project Scientist at Computer Associates in Massachusetts, an early software firm that specialized in creating programs called compilers. Compilers translate from an algebraic language like FORTRAN into a computer's native machine language. He soon began making discoveries that revolutionized the field; for example, his paper of 1963, "Syntactic analysis and operator precedence," represented the birth of practical methods to derive language translators from specifications of grammar. His seminal paper of 1964, "The syntax of programming languages---A survey," not only brought order out of chaos by elegantly summarizing all previous work on compiler parsing but also introduced a significant new paradigm that led to the all-important notion of "object-oriented languages." When I drafted the material for a monograph on the subject of parsing in 1966, I came to the conclusion that only five really good papers about compilers had been written so far, and Bob had been the author of all five.

Another of his key contributions, "Assigning meanings to programs," was published in 1967. This paper represented the beginning of the theory of programming language semantics, and it was a radical break from the past.

Programmers had previously had no way to verify the correctness of their work, except by trial and error; in fact, programming was considered to be an inherently different sort of human activity, having no relation to mathematical rigor, so people like me didn't even realize that we lacked any way to prove that our programs were correct. As far as we knew, a programmer's task was to fiddle with programs until no more mistakes could be found, and to hope that nothing had been overlooked. Floyd's paper introduced so-called "invariant assertions," which opened our eyes and led directly to a long series of future developments.

Besides his theoretical work on languages, he also introduced several important combinatorial techniques, such as an efficient algorithm to find the shortest distances between all pairs of points in a network. He invented non-deterministic programming and he derived methods for selecting random permutations and combinations.

He joined the faculty of Carnegie Institute of Technology as an Associate Professor in 1965. Computer Science was a brand new subject in those days, and he played a large role in developing the curriculum. He immediately became a popular teacher at Carnegie, and began to serve on the editorial board of the major technical journal in computer science, the *Journal of the Association for Computing Machinery*. Stanford tempted him away from Pittsburgh in 1968, and when he was promoted to Full Professor in 1970 he was one of very few people judged to deserve such a rank after only five years as Associate Professor.

In 1971 he was the only computer scientist in the world to receive two independent nominations to give an invited plenary talk at the prestigious International Congress of Information Processing. By this time he had begun to make significant discoveries in yet another direction, known as the complexity analysis of algorithms: Given a problem to be solved, the task is to solve it in the fastest possible way. For example, he discovered that it is possible to compute the median of a set of data by making a number of comparisons that is at worst proportional to the size of the set; all of the obvious ways to solve that problem require substantially more work.

A few years later he discovered the best possible way to add two numbers together, if we are allowed to represent numbers in any convenient way. This result meant that he had to invent an addition procedure that was faster than any others heretofore known, and to prove that no faster method will ever be possible; both of these challenges were difficult.

Stanford's Computer Science department had been founded by George Forsythe, who built it up to be the best in the world. After Forsythe's death in 1972, the dean chose Bob to be our next chairman, and Bob served in this capacity from 1973 to 1976. He plunged into this new role with enthusiasm and made extremely good decisions that consolidated our position of leadership and prepared for a solid future in our rapidly expanding field. For example, this was the time when Computer Science was finally able to have its own "home" on the Stanford campus, with all faculty in the same building instead of scattered in four different locations. Bob worked tirelessly with the architects and was responsible for numerous critical decisions affecting the look and feel of our new quarters in Margaret Jacks Hall, a reconstructed building in the main quad.

At this time he also led the drive to help free Fernando Flores from prison in Chile. Flores, a former vice-rector of the Catholic University of Chile who had developed a computer-based information system for the entire Chilean economy and become a cabinet minister in the government of Salvador Allende, came to Stanford as a research associate in 1976 largely because of Bob's efforts, after having been held without charges for three years by the military junta headed by Augusto Pinochet. Other Stanford people, including Law School Dean Charles Meyers and President Richard Lyman, also contributed to this cause.

Together with Louis Steinberg, Bob published in 1976 a novel algorithm for rendering grayscale images with black and white dots. This method, now often called "Floyd-Steinberg dithering" although they called it "error diffusion," has become a standard technique, used millions of times every day in computer printing.

Floyd received Computer Science's highest honor, the ACM Turing Prize, at the end of 1978. He was cited for "helping to found the following important sub-fields of computer science: the theory of parsing, the semantics of programming languages, automatic program verification, automatic program synthesis, and the analysis of algorithms.

Your work has had a clear influence on methodologies for the creation of efficient and reliable software." The address he gave at the time, entitled "Paradigms of programming," summarized many significant concepts; some key elements of what has later become known as "literate programming" were among the points he raised.

One of Bob's courses, the Programming and Problem Seminar (CS204), became for many of our alumni the class they remember best, the class in which they learned the most. Bob supervised many of our top Ph.D. students, including Zohar Manna and Robert Tarjan, both of whom subsequently received tenure at Stanford. (Tarjan later became Professor at Princeton University and was the 1986 Turing laureate.) Ronald Rivest, another of Bob's students, received the Turing award in 2003.

Throughout all this time, Floyd had been teaching classes that involved a novel machine-based theory of complexity, rather different from the abstract complexity theory that had become popular elsewhere. His book on the subject, *The Language of Machines*—written with his former graduate student Richard Beigel, now Professor at Temple University—was published in 1994, and the approach soon became widely adopted. The book is now required reading on many campuses, and it has been translated into French and German.

Bob had many other interests: He was, for example, a connoisseur of fine food, and a world class backgammon player. He audited advanced courses at Stanford in German literature. He was a passionate hiker and rock climber who loved the wilderness country in the high Sierras. But in such matters he tended to be a very private person, and only a few of his closest friends knew much about those interests.

Floyd was a Fellow of the American Academy of Arts and Sciences, the American Association for the Advancement of Science, and the Association for Computing Machinery. He spent a year at the Naval Postgraduate School as their First Grace Murray Hopper Professor. He also received the IEEE Computer Pioneer Award in 1992.

The final years of Bob's life, alas, were lost to a rare ailment known as Pick's disease, a neurodegenerative illness that gradually began to affect his mind and his body. His intellect was so sharp that he could do brilliant work even when operating at half capacity; thus nobody knows how soon he began to be incapacitated. At the time of his retirement in 1994 he was still doing some interesting research, but certainly in a diminished way by comparison with his extraordinary trend-setting results of the 1960s and 1970s. By 1997 we knew that we had lost our colleague. We also know, of course, that his pioneering works will outlast all of us.

On a personal note, I would like to add that my professional life owes more to Bob than to any other person. He was the first designated reader for my books on {\sl The Art of Computer Programming}, and he is cited more often than anyone else in those volumes. We decided in 1967 that we wanted to work at the same university, and I'm extremely happy that we both chose Stanford.

Committee:

Donald Knuth, chair