

## Robert W Floyd (1936-2001)

The Mathematics Genealogy project is an ambitious venture which attempts to construct an academic family tree including all mathematicians on whom data is available. If one looks up the entry on Robert W Floyd, it says “Advisor unknown”. Yet among his PhD students are listed some of the most well-known computer scientists – Zohar Manna, an authority on the verification of software systems, Robert Tarjan, 1986 Turing Award winner, and Ron Rivest, a co-inventor of the RSA cryptosystem and a 1993 Turing Award winner. Floyd in fact, never ever got a PhD. He was made professor at Stanford at the age of 32. In the words of Donald Knuth, arguably the world’s best known living computer scientist, who wrote a letter of recommendation on behalf of Floyd in 1968 “... Floyd has never gone through the formalities of obtaining a PhD degree. I believe this was primarily due to the fact that he entered graduate school at the University of Chicago when he was only 16 or 17 years old, as part of an experimental accelerated education program; this was not a mature enough age to do graduate work ... . Certainly he has written at least a dozen papers by now, each of which is superior to any PhD thesis I have ever seen in computer science, so the mere fact that he has never formally received the degree should be quite irrelevant.”

Robert Floyd was born in New York city on June 8, 1936. His parents moved more than a dozen times before he went to college. A child prodigy, he read every book he could lay his hands on. 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 many unusual children who have been through accelerated programs, he soon lost his taste for conventional school and continued to study part time, beginning work at the Armour Research Foundation at the Illinois Institute of Technology. He began as a self taught computer operator, and then went on to become senior programmer and analyst. In 1958 he received a BS degree in Physics from the University of Chicago, and published a paper on radio interference the same year.

In 1962 he became a Senior Project Scientist at Computer Associates in Massachusetts, an early software firm that specialized in creating programs called compilers.

---

<sup>1</sup> Donald E. Knuth, Robert W Floyd, In Memoriam, *ACM Sigact News*, December 2003, Vol 34, No. 4.



#### ARTICLE-IN-A-BOX

Compilers translate from a formal language like FORTRAN into a computer's native machine language.

Floyd's first major paper entitled "A descriptive language for symbol manipulation" (1961) introduced a new notation for describing what is really going on in the process of translating a program from one language to another. Knuth, who had just completed writing a compiler for a subset of ALGOL, and who had painstakingly gone through the source listings of several other compilers, immediately recognized the importance of this work. It enabled the formal description of a compiler, so that a parser could be generated in a matter of hours. Floyd wrote a few more papers on the subject. Knuth later drafted the material for a monograph on the subject of parsing in 1966. He said that he came to the conclusion that only five really good papers about compilers had been written until then, and that Bob Floyd was the author of all five!

In 1965 Floyd joined the faculty of the Carnegie Institute of Technology as an associate professor. He was an extremely popular teacher and became a member of the editorial board of the most important technical journal in computer science at the time, the *Journal of the Association of Computing Machinery*. He moved to Stanford in 1968 and became Full Professor in 1970. He was Chairman of the Computer Science Department of Stanford University from 1973 to 1976.

Programming Language semantics was a key area Floyd made original contributions to. His paper entitled "Assigning meanings to programs" written in 1967 introduced the notion of *assertions* which provided programmers a way to verify the correctness of their work. Until then programming was considered to be a human activity having no relation to mathematical rigour, and a programmer's task was seen to be that of fiddling with programs, finding bugs and making patches until no more mistakes could be found, hoping for the best. Floyd's paper was the first in a long series of developments. Today verification is an important phase of safety critical programs like collision avoidance systems used at airports and avionics software.

Besides his theoretical work on languages, Floyd made important contributions to another area known as the complexity analysis of algorithms. He invented some very practical algorithms like finding shortest paths in a network, sorting numbers, finding medians in a set of data, and an error diffusion algorithm (with Louis Sternberg) for rendering grey scale images with binary pixels. According to Knuth, Floyd's course, the Programming and Problem Seminar, became for Stanford computer science



#### ARTICLE-IN-A-BOX

alumni, the course which they remember best and in which they learned the most.

The ultimate honour in the field of computer science, the Turing Award, was conferred on Floyd in 1978. The citation said “...for helping to found the following important subfields 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.”

His book entitled *The Language of Machines* written with his graduate student Richard Beigel, now Professor at Temple University was published in 1994 and introduced a rather novel machine based theory of complexity.

Many awards came his way, among them fellowships of the American Academy of Arts and Sciences, American Association for the Advancement of Science, and the Association for Computing Machinery. He received the IEEE Computer Pioneer award in 1992.

Floyd had a strong social conscience. He spent a significant amount of time to free Fernando Flores from prison in Chile, where he had been incarcerated for three years without charges under the orders of Augusto Pinochet. Flores came to Stanford as a research associate in 1976 largely because of his efforts. In 1970 when Richard Nixon ordered the invasion of Cambodia, Floyd and Knuth joined students in picketing Pine Hall (Stanford’s Computation Center) in protest.

Floyd enjoyed backgammon, playing against computer opponents at home and against human opponents in major tournaments. He was also a passionate hiker and mountain climber who loved the outdoors and the wilderness of the High Sierra.

Towards the end of his life, he was afflicted by a rare ailment known as Pick’s disease, a neurodegenerative illness that slowly began to affect his mind and body. Knuth observes that his intellect was so sharp, that few knew when the illness began to incapacitate him, as he was capable of brilliant work even when operating at half capacity. He passed away on September 25 2001 at the age of 65.

*Priti Shankar*

Department of Computer Science and Automation, IISc, Bangalore 560 012, India.

Email: [priti@csa.iisc.ernet.in](mailto:priti@csa.iisc.ernet.in)

