OBITUARY

# John Backus (1924–2007)

Inventor of science's most widespread programming language, Fortran.

John Backus, who died on 17 March, was a pioneer in the early development of computer programming languages, and was subsequently a leading researcher in so-called functional programming. He spent his entire career with IBM.

Backus was born on 3 December 1924 in Philadelphia, and was raised in Wilmington, Delaware. His father had been trained as a chemist, but changed careers to become a partner in a brokerage house, and the family became wealthy and socially prominent. As a child, Backus enjoyed mechanical tinkering and loved his chemistry set, but showed little scholastic interest or aptitude. He was sent to The Hill School, an exclusive private high school in Pottstown, Pennsylvania. His academic performance was so poor that he had to attend summer camps to catch up on his studies. He fared no better at the University of Virginia, where he enrolled for a degree in chemical engineering in 1942, and from which he was sent down for poor attendance.

Backus was drafted into the US Army in early 1943, where he was initially put on an anti-aircraft programme. Aptitude tests revealed his high intelligence, and he was sent on a specialist engineering course at the University of Pittsburgh, Pennsylvania. He also tried a pre-medical training course, but lost enthusiasm and dropped out. After leaving the army in May 1946, he enrolled at Columbia University, New York, initially as a probationary student because of his poor academic record. In mathematics, he at last found his calling, graduating with a BA in 1949 and obtaining a master's degree the following year.

In 1948, IBM had completed its first experimental electronic computer, known as the Selective Sequence Electronic Calculator (SSEC). It was a huge machine with 13,000 tubes and 23,000 relays. IBM displayed the machine in the showroom of its New York headquarters, where it was visible to passers-by and attracted much media attention. Backus went to see the SSEC, and persuaded IBM to hire him as a programmer. He spent the next year of his life calculating lunar positions, which he found a delight.

IBM introduced its first commercial computer product, the model 701, in 1952. The computer had to be programmed at a level very close to basic binary machine code, which was not intrinsically difficult but time-consuming and error-prone. Backus devised an automatic programming system called Speedcoding for the 701, which made the task much easier. Like similar systems being developed elsewhere around the same time, however, Speedcoding produced programs that were uneconomically slow.

Backus proposed to his manager the development of a system he called the Formula Translator — later contracted to Fortran — in autumn 1953 for the model 704 computer, which was soon to be launched. The unique feature of Fortran was that it would produce programs that were 90% as good as those written by a human programmer. Backus got the go-ahead, and was allocated a team of ten programmers for six months. Designing a translator that produced efficient programs turned out to be a huge challenge, and, by the time the system was launched in April 1957, six months had become three years. The Formula Translator consisted of 18,000 instructions, which was not especially long as programs went, but it embodied fiendishly complex algorithms for code optimization.

Fortran was eagerly taken up by users of the IBM 704, and other computer manufacturers also produced Fortran systems so that their machines' software would be compatible with IBM's. Fortran became the *lingua franca* of scientific computing, which it remains 50 years on — today, for example, the UK Meteorological Office's computer model for climate change consists of a one-million-line Fortran program. Fortran has often been criticized for its inelegance, but as Backus subsequently explained, when it was being designed efficiency was paramount and very little thought was given to the language itself. And of course, no one could have dreamed that the language would still be going strong half a century later. Because of the need for backward compatibility, Fortran has never escaped the path-dependency of those early decisions.

In the late 1950s Backus became a member of the Algol Committee, which was designing an international scientific programming language, named Algol 60. The language was specified in Backus–Naur form, a notation that Backus devised in collaboration with the Danish computer scientist Peter Naur. Algol 60 was an elegant language that was popular in Europe in the 1960s and 1970s. However, it was never able to replace Fortran, primarily because of the investment already made in the Fortran code. Nonetheless, Algol was hugely influential in programming-language design — most modern programming languages, such as C and Java, can trace their roots to the language, and the Backus–Naur form is part of the computer-science canon.

IBM/AP

After the pizzazz of the Fortran years, Backus's subsequent career was calmly scientific. In 1960 he became a staff member of IBM Research, Yorktown Heights, New York. In 1963 he was made an IBM fellow and spent the remainder of his career, until he retired in 1991, with IBM in both Yorktown Heights and San Jose, California.

At IBM Research, Backus led a research group in functional programming, a new programming paradigm that aimed at computation through the evaluation of mathematical functions. The idea was to bypass what Backus called the 'von Neumann bottleneck', after John von Neumann, one of the inventors of the computer. As Backus explained it, a computer consisted of a processor and a memory; the object of a program was to change the state of the memory, using the processor, but this had to be done painfully slowly, one instruction at a time. But although functional programming became (and remains) a major computer-science research topic worldwide, it has never broken through to the mainstream.

Fortran remained Backus's lasting contribution to computing, for which he was awarded the National Medal of Science in 1975 and the Turing Award of the Association of Computing Machinery in 1977 — computer science's highest honour. ■

**Martin Campbell-Kelly**
Martin Campbell-Kelly is in the Department of Computer Science, Warwick University, Coventry CV4 7AL, UK.
e-mail: m.campbell-kelly@warwick.ac.uk