

ARTICLE-IN-A-BOX

John Backus – Inventor of FORTRAN

V Rajaraman

In 1957 I attended my first class in computer programming at MIT USA. The instructor informed the class that the course was on assembly language programming for the IBM 704, then one of the largest computers in the world. He mentioned that there was a language called FORTRAN on the horizon but that it was at an experimental stage, and in any case too advanced for our class. In order to carry out my assignment which was to write a program to compute the Fourier transform of a function, I had to learn the nitty gritty details of the internal structure of the IBM 704. This was an educational experience in itself, though quite time consuming and sometimes exasperating. Anyone who has gone through the tedium of writing assembly level code, will appreciate the debt the programming community owes to John Backus, the inventor of FORTRAN. Though five decades have passed since FORTRAN was first introduced, it continues to rule as the language of choice for scientists working in the field of numerical computation

John Backus was born in Philadelphia, Pennsylvania on December 3, 1924. He went to a private school in Pottstown, Pennsylvania, but did not do well. However he completed high school and joined the University of Virginia, to major in Chemistry in accordance with the wishes of his father, a stockbroker. He disliked the course and stopped attending classes, as a result of which he was expelled by the University after the first year. He then joined the army where he had to undergo an aptitude test. His performance was so good that he was sponsored for a pre-engineering program at the University of Pittsburgh. However he discontinued the course and after a second aptitude test, was sponsored for a medical program in a school in New York. During the course he was diagnosed with a brain tumour which was successfully operated. He decided that this again was not the career he wanted to pursue, and dropped out a second time. He left the army in 1946. His love of music encouraged him to build a really good music system, and he enrolled in a radio technician's school. While at the course he had occasion to dabble in some mathematical computations which he greatly enjoyed. Thus he decided to join Columbia University for a degree in mathematics, and graduated in 1950 with a Masters degree.

Shortly before his graduation Backus walked into IBM looking for a job, and was offered a programmers job. This launched him into a long and distinguished career. Computer science was still in a nascent stage and words like compilers, hardware and software had not yet been coined. Backus was frustrated by his experience of “hand-to-hand combat with the machine” (the machine in question was the IBM 704). He began the task of developing a coding system



written in a notation easy for humans to learn and use, and a translator for generating machine code for execution on a target machine. To help him in this task, Backus recruited a team of about ten people, including a crystallographer, a cryptographer, a chess player, a researcher from MIT and a woman who had just graduated from Vassar.

Backus consulted several users of the IBM 704 for their opinion on a generally acceptable language. In 1954 Backus and his team published a report entitled “Preliminary Report, Specifications for IBM FORMula TRANslation System FORTRAN”. The actual system was ready for users in 1956. The remarkable feature of this translator was its efficiency—the quality of the code generated approached that of hand written assembly code. This prompted several scientists who would never have otherwise done so, to take up programming.

FORTTRAN was propriety to IBM, and as the international community wanted a language which could be implemented on any computer, Backus worked with a group of scientists, the most prominent of whom was Peter Naur, a Danish computer scientist, and designed ALGOL (ALGOrithmic Language) which would supplant FORTRAN. The language was defined in a notation now called Backus-Naur form. Many syntactic rules of the language were defined recursively. The effort to formalize the rules of the language was the first of its kind. The compilers for ALGOL were not particularly efficient and FORTRAN continued to be the most widely used language in the 60’s and 70’s.

Backus however was unhappy about the developments in computer languages. When he received the coveted Turing award in 1977, his Turing award lecture was on a new language paradigm called *functional programming*. An article^[1] based on this lecture argues persuasively about the need for functional programming languages. The Turing award citation was for “profound, influential and lasting contributions to the design of practical high level programming systems, notably through his work on FORTRAN, and for seminal publications of formal procedures for the specification of programming languages”. Besides the Turing award, he received the U.S Medal of Science in 1974, and the Charles Draper prize in 1993. He retired from IBM in 1991.

Backus died at his home in Ashland, Oregon on March 17, 2007.

*V Rajaraman, SERC, IISc, Bangalore
Email:rajaram@serc.iisc.ernet.in*

^[1] John Backus, Can Programming be Liberated from the von Neumann Style? A Functional Style and Its Algebra of Programs, Communications of the ACM, Vol.21, No.8, August 1978, pp.613–641.

