

CSc 480 — Questions about John Backus's Turing Award Lecture

John Backus was the chief designer of Fortran and a major designer of ALGOL 60. (He also invented BNF.) However, this paper criticizes these kinds of "von Neumann" languages and describes a very different programming style. The paper is both controversial and classic.

Please read Sections 1-10 (pp. 613–19) and Sections 15-16 (pp. 638–40), then prepare answers to the following questions.

1. What paragraph or section was the hardest for you to understand? What about it was confusing?
2. Section 5.2 shows all the steps in evaluating the inner product of two vectors. Develop a similar sequence of steps for evaluating the inner product of $\langle \langle 1, 2, 3, 4 \rangle \langle 5, 6, 7, 8 \rangle \rangle$. It is OK to refer to the imperative program in Section 5.1 to figure out what the final answer is supposed to be!!
3. Section 8 mentions the APL language, which was created by Kenneth Iverson, another Turing Award winner. Search for information on APL—either in a textbook or on the Web—and give (a) a short description of the language, and (b) a nontrivial example of an APL program. (The language is famous for its "one liners", namely the powerful programs that one could write with one—sometimes long—line of code.) Please list the sources you consulted.
4. Despite Backus's somewhat valid criticisms, von Neumann languages remain widely used; indeed they are the dominant languages in use today. Why do you think this is? What advice in Hoare's paper provides at least part of the answer?
5. List at least three applicative languages that are in use today.