

Postscript

Notation as a Tool of Thought: 1986

KENNETH E. IVERSON

The thesis of the present paper is that the advantages of executability and universality found in programming languages can be effectively combined, in a single coherent language, with the advantages offered by mathematical notation.

The executable language to be used is APL, a general-purpose language which originated in an attempt to provide clear and precise expression in writing and teaching, and which was implemented as a programming language only after several years of use and development.

The first of the foregoing passages from my 1980 paper states the case to be made for the use of an executable analytic notation, and the second states the particular vehicle to be used in developing it. The most obvious and important use of executable analytic notation is in teaching. The following comments summarize recent progress in this area.

Materials and Courses

A common theme in the materials mentioned here is the casual introduction of the necessary notation in context, in the manner familiar from the teaching of mathematics. A good example at a high-school level is the treatment of probability by Alvord [1]. In their treatment of circuit analysis, Spence and Burgess [2] make heavier use of APL as a means of implementing their system, and Hazony [3] makes combined use of graphic input and APL expressions to specify designs in an expert support system.

The direction of my own recent work is described in an ACM Forum letter [4], and drafts of two texts used in courses are currently available [5]. The Pesch and Berry paper on style and literacy [6] should be read by anyone interested in these matters.

Development of the Notation

A version of APL has recently been developed [7] which, while remaining within the bounds adopted in the ISO standard for the language, has both simplified its structure and increased its expressive power. It provides a significantly better basis for teaching than the notation used in my 1980 paper.

Availability of Implementations

Although APL has long been provided by central university computing services, it has been impracticable to use in teaching because of charging rates and lack of suitable terminals. The present availability of APL systems on microcomputers has changed this situation drastically.

The system provided for students here at the T. H. Twente is the one I find most satisfactory [8]; it does not yet incorporate such new functions as *nub*, *raze*, and *all* (a generalization of Cartesian product), but does provide the fundamental notions of *function rank*, the *box* function (for the general handling of representation or "structures"), and the *under* operator for the important mathematical notion of duality.

Moreover, the system handles complex numbers (with all of the mathematical functions suitably extended); provides the determinant ($- \cdot \times$), the permanent ($+ \cdot \times$), the test for a Latin square ($\vee \cdot \wedge$), and related func-

tions produced by the dot operator; generalizes the *or* and *and* functions to provide the greatest common divisor and least common multiple; and exploits the characteristics of the microcomputer and its screen display to provide a "union" keyboard in which most characters (such as the parentheses and the upper- and lower-case letters used in names) are in their normal typewriter positions.

References

1. Alvord, L. *Probability in APL*. APL Press, STSC Corp., Bethesda, Md.
2. Spence, R., and Burgess, J. *Circuit Analysis*. Prentice-Hall, Englewood Cliffs, N.J., 1986.
3. Hazony, Y. A brief report of his work at Boston University appears in a summary of a Minnowbrook Conference reported in *APL Quote-Quad* 16, 3 (1986).
4. Blaauw, G. A., et al. A curriculum proposal for computer science. *Commun. ACM*, Forum (Sept. 1985).
5. Iverson, K. E. *Mathematics and Programming* and *Applied Mathematics for Programmers*. (Drafts of both are available from I. P. Sharp Associates, Toronto, Ont., Canada.)
6. Pesch, R., and Berry, M. J. A. Style and literacy in APL. In *Proceedings of APL86*. ACM, New York, 1986.
7. Iverson, K. E. *A Dictionary of the APL Language*. Draft available from I. P. Sharp Associates, Toronto, Ont., Canada.
8. Sharp APL/PCX. Computer system for use on IBM AT/370 and XT/370 computers. Available from I. P. Sharp Associates, Toronto, Ont., Canada. The system also runs on a normal IBM PC or AT, much more slowly, but adequately for teaching purposes.