IBM and the Algol project:
Exploring the challenges and dilemmas of early computer science research

David Nofre, Independent Scholar
d.nofre@gmail.com

Rarely has the development of a programming language raised so many hopes and caused so much disappointment as the development of Algol. As Mark Priestley has shown, the process of definition of Algol, and particularly of the second version of the language, or Algol 60, contributed much to put computer programming on firmer theoretical grounds.[1] Hence, the task of programming came to be seen as a proper field of knowledge and pillar of the emergent discipline of computer science, not any more just a matter of getting the machine to work. This was, however, in sharp contrast with the adoption rate of any of the three versions of the language (Algol 58, Algol 60, and Algol 68), which was rather limited, particularly when considering the contemporary success of IBM Fortran, a language like Algol also conceived for scientific computation, yet initially only available for a couple of IBM computer models.[2]

Exact figures of real market share and implementation for programming languages are very difficult to give. But a rough estimation is that Algol 58 and Algol 60, which became the most successful versions, were, by the early 1960s, available in less than 10% of computers in the United States and Western Europe.[3] What were the reasons for such modest adoption? There were probably many factors that contributed to Algol's limited acceptance, but in this paper I would like to focus on the role that the strong research character of the Algol effort might have played in hindering the acceptance of the language. Specifically, I will try to show how the dynamics of open-ended research that characterized the development of Algol 58 and Algol 60 eventually came to conflict with the demands for a narrow and limited language development from computer manufacturers and user groups, most notably IBM and associated user group SHARE.

[1] Mark Priestley, *A science of operations: machines, logic and the invention of programming* (Springer, 2011).

[2] According to Jean Sammet, the first implementation of Fortran by a non-IBM firm was the first version of Fortran for the UNIVAC Solid State 80, which was running as early as January 1961. By 1963, virtually all manufacturers had either delivered or committed themselves to delivering some version of Fortran. Source: Jean Sammet, *Programming languages: History and fundamentals* (Englewood, 1969), p. 146-147.

[3] The interested reader could get the details of this estimation from the author upon request.

I will mostly focus on the period immediately after the first definition of Algol 60, roughly speaking between 1960 and 1965. For it was during these years that tensions between IBM top management, SHARE, and the Algol community in Western Europe ran high. This conflict would have critical consequences for the evolution of the Algol project in general, and in particular, for the successful adoption of Algol 60 as programming tool. More generally, the conflict between IBM and the Algol community reveals a fundamental tension in the nature of computer science research between instrumentality and intelligibility.

The publication of the Algol 60 report in the spring of 1960 came indeed as a surprise to many in the Algol community in the United States. This is an aspect that has received little attention in the historical literature on Algol, which so far has mostly focused on European developments.[4] The changes introduced in Algol 60 required to rewrite substantial parts of the compilers being developed for Algol 58. This was not a major issue in Europe, where, by early 1960, work on the development of compilers for Algol 58 had just started to take off. But in the United States the development of those compilers was in a more advanced stage than in Europe. Hence, many participants of the Algol project in the United States felt, including many in SHARE, that Algol 60 rather hindered than helped in the effort to produce a single translator, and that, thus, Algol 60 was a step backward in the realization of the goal of a universal computer language.

The disruptive character of Algol 60 increased IBM's doubts about a project over which the firm had little influence. During the early 1960s, the so-called "common languages", Algol and Cobol, seemed to threaten IBM's investment in its own languages, Fortran and Comtran. However, for reasons of corporate image, IBM did not want to appear as obstructing the development of either Algol or Cobol. During the years between 1960 and 1962, IBM kept a relatively balanced position, working diligently at keeping open the communication with the Algol community, while fearing that the US or European governments would enforce the use of Algol and/or Cobol. Yet, the full endorsement of Cobol by the US Defense Department in August 1961, the rising popularity of Algol in Europe, and fears of a domino effect in the United States, eventually convinced IBM top management of the need to push for the use of Fortran in Europe, a strategy that would prove successful.

[4] See, for example, the special issue edited by Gerard Alberts: "ALGOL Culture and Programming Styles", *IEEE Annals of the History of Computing* (Volume: 36, Issue: 4, Oct.-Dec. 2014).