Journal of Statistical Software

April 2012, Volume 47, Book Review 1. http://www.jstatsoft.org/

Reviewer: Jan de Leeuw
University of California, Los Angeles

Modern Fortran: Style and Usage

Norman S. Clerman and Walter Spector
Cambridge University Press, New York, NY, 2012.
ISBN 978-0-521-73052-5. 334 pp. USD 45.00 (P).
http://www.cambridge.org/us/knowledge/isbn/item6577536

The inside cover page of this excellent book is the title page of the 1954-11-10 report by John W. Backus *et al.* on the IBM Mathematical Formula Translation System, a.k.a. FORTRAN (International Business Machines Cooperation 1954). The report itself is still a delight to read. FORTRAN was intended to save about half the cost of program development, eliminating much of coding and debugging, because it only required about 5–20% of the number of characters needed by "direct coding" in IBM 704 assembly language.

It is truly remarkable that after half a century the language is still very much in use, although we must emphasize that it went through many, many revisions. Backus would probably have some difficulty recognizing modern Fortran, with its free format, derived types, dynamic arrays, parallel processing, and object-oriented coding, as an update of the language he originally designed. The Wikipedia (2012) article on Fortran gives a good overview of the development of the language over time.

Much of scientific legacy code, especially in numerical mathematics and the natural sciences, is still only available in FORTRAN, often only in FORTRAN 77. The standard distribution of R contains quite a bit of FORTRAN 77 code. Newer versions of the language, such as Fortran 90 and Fortran 95, are still used to develop modern statistical software (Lemmon and Schafer 2005).

The book by Clerman and Spector is perhaps most useful for readers with some Fortran experience. There are many code examples throughout the book. It concentrates on the Fortran 2003 revision, but it discusses more modern extensions, such as the coarrays of Fortran 2008. It is set up as a "best practices" book, which uses over 200 rules and guidelines to write optimal Fortran code and documentation. These include both style and programming guidelines, as well as various sets of useful conventions. These rules are used, however, to give a good introduction to programming in general, and Fortran programming in particular. In later chapters modern Fortran characteristics are used to introduce generic programming, object orientation, parallel processing, and interoperability with the C language.

For a statistician who has to program in a compiled language, for instance to optimize the speed of R functions, the question remains which language to choose. It is generally true

that C and its offspring are more popular and more widely distributed. Compilers are more readily available. Apple, for instance, has never included a Fortran compiler with its developer software. There is also more documentation on C and C++, and generally the differences between "old" and "new" C are much smaller and much more manageable than those between "old" and "new" Fortran. But what the Clerman and Spector book clearly shows is that modern Fortran is, in every respect, as powerful as anything the C family can offer.

References

International Business Machines Cooperation (1954). "Preliminary Report: Specifications for The IBM Mathematical FORmula TRANslating System FORTRAN." Available from http://www.computerhistory.org/collections/accession/102679231.

Lemmon DR, Schafer JL (2005). *Developing Statistical Software in Fortran 95*. Springer-Verlag.

Wikipedia (2012). "Fortran – Wikipedia, The Free Encyclopedia." URL http://en.wikipedia.org/wiki/Fortran, accessed 2012-04-11.

Reviewer:

Jan de Leeuw
University of California, Los Angeles
Department of Statistics
Los Angeles, California 90095-1554, United States of America
E-mail: deleeuw@stat.ucla.edu
URL: http://www.stat.ucla.edu/~deleeuw/