AB47.0 EDITOR'S NOTES.

First, my usual remarks about paucity of contributions, and hence the thinness of this issue. However, you do get your free plastic insert again. This time, it contains the complete Revised Report (together with the Commentaries collated from AB43.3.1 and AB44.3.1), the Sublanguage Report, and the Standard Hardware Representation. These are the three official documents, approved by I.F.I.P. Also included are the Partial Parametrization Proposal (AB39.3.1) and the Modules and Separate Compliation Proposal (AB43.3.2).

Standardization of ALGOL 68.

Following a suggestion from the ISO committee concerned with programming languages (ISO TC97/SC5), Working Group 2.1 decided, at its meeting in August 1980, to press for an International Standard on ALGOL 68. The Standard will be prepared jointly by IFIP and ISO, and I have been appointed to coordinate these activities.

The intention is to leave the Revised Report intact as the definition of the language. The Standard will refer to the Report, and will prescribe precise requirements for conforming programs, implementations and accompanying documentation. I hope to publish a working draft of the standard in a future edition of the ALGOL Bulletin.

In the meantime, there are many political problems to overcome. We have to persuade sufficient National Standards Organisations to vote, first to have the proposed standard even considered, and later to have it approved. The danger is that they may be unaware of the interest in ALGOL 68 within their respective countries, and that they will refuse to take it seriously. Here is where each of you, if you believe this to be a worthwhile endeavour, can help. Write to your National Standards Organization. Tell then that ALGOL 68 matters; that people really do use it; and that a Standard therefore ought to exist.

AB47.1 Announcements.

AB47.1.1 International Symposium on Algorithmic Languages.

This Symposium is organized by the Mathematical Centre under the auspices of IFIP TC2 as a tribute to Professor A. van Wijngaarden on the occasion of his retirement from the Mathematical Centre. Professor van Wijngaarden is well known for his contributions in the area of programming language design (ALGOL 60, ALGOL 68, two-level grammars). The Symposium is to be held from Oct. 26-29, 1981 at the Free University of Amsterdam, in The Netherlands.

Program:

H.Zemanek (IBM, Vienna): "The role of Professor van Wijngaarden in the early history of IFIP", (invited address).

A.I.Wasserman (University of California, San Francisco), R.P.van de Riet and M.L.Kersten (Free University, Amsterdam): "PLAIN, an algorithmic language for interactive information systems".

R.Schild (Landys & Gyr, Switzerland): "PORTAL, a process-oriented real-time algorithmic language".

J.D.Roberts (University of Reading): "Naming by colours: a graph-theoretic approach to distributed structure".

H.S.Warren, Jr. (IBM, Yorktown Heights): "Optimization of inductive assertions".

A.Bossavit and B.Meyer (Electricite de France, Clemart): "Methods for vector programming".

P.Klint (Mathematical Centre, Amsterdam): "Formal language definitions can be made practical".

J.Backus (IBM, San Jose): "Is computer science based on the wrong fundamental concept of "program"? An extended concept", (invited address).

L.G.L.T.Meertens (Mathematical Centre, Amsterdam): "Issues in the design of a beginners' programming language".

D.Grune (Mathematical Centre, Amsterdam): "From VW-grammar to ALEPH".

M.Broy, P.Pepper and M.Wirsing (Technical University of Munich): "On design principles for programming languages: An algebraic approach".

J.Darlington (Imperial College, London): "Structured descriptions of algorithm derivations", (invited address).

M.Sato and M.Hagiya (University of Tokyo): "HYPERLISP".

D.de Champeaux and J.de Bruin (University of Amsterdam): "Symbolic evaluation of LISP functions with side effects for verification".

P.Naur (University of Copenhagen): "Aad van Wijngaarden's contributions to ALGOL 60", (invited address).

M.M.Fokkinga (Technical University of Twente): "On the notion of strong typing".

H.B.M.Jonkers (Mathematical Centre, Amsterdam): "Abstract storage structures".

J.C.Reynolds (Syracuse University): "The essence of ALGOL", (invited address).

R.Kuiper (Mathematical Centre, Amsterdam): "An operational semantics for nondeterminism equivalent to a denotational one".

O.Grumberg, N.Francez, J.A.Makowsky (Technion, Haifa) and W.P. de Roever (University of Utrecht): "A proof rule for fair termination of guarded commands".

W.M.Turski (University of Warsaw): ALGOL 68 revisited twelve years later, or: from Aad to Ada", (invited address).

A full social program has also been arranged.

Full details concerning registration, fees and hotel accomodation may be obtained from:

Mrs.S.J.Kuipers,
Mathematical Centre,
Postbus 4079,
1009 AB Amsterdam,
The Netherlands.

AB47.1.2 ALGOL 68 Implementation - FLACC.

Some recent adjustments in the pricing structures for the FLACC system may be of interest.

These changes are primarily aimed at helping those interested in Algol 68 by making a high-quality implememtation generally available. We particularly wish to help people who want to learn and to use Algol 68, but who cannot justify the FLACC lease price.

We propose therefore, to distribute an unsupported version of FLACC. The frozen version is designated "FLACC V1.4U", and is available to universities and technical schools for a one-time charge of C$1500. This price does not include maintenance or update services, nor is there any provision for an acceptance period.

For complete licensing information, please write to:

Sigrid Fritz,
Chion Corporation,
Box 4942,
Edmonton, Alberta,
CANADA T6E 5G8.

AB47.1.3 International Conference on ALGOL 68 - Proceedings.

The Proceedings of the International Conference on ALGOL 68, held at Bochum on March 30-31 1981, under the sponsorship of the WG2.1 Subcommittee on ALGOL 68 Support and of the Rechenzentrum der Ruhr-Universitaet, can be obtained from:

Mathematical Centre,
Postbus 4079,
1009 AB Amsterdam,
The Netherlands.

Table of Contents:

On the design of an abstract machine for a portable ALGOL 68 compiler, by L.G.L.T.Meertens.

An implementation of modular compilation in ALGOL 68 (invited lecture), by G.J.Finnie & M.C.Thomas.

Programming languages for a course in data structures, by V.J.Rayward-Smith.

Context-free grammars and derivation trees in ALGOL 68, by V.Linnemann.

An ALGOL 68 prelude for the implementation of test generation algorithms, by S.D.Butland.

A programming system for interval arithmetic in ALGOL 68, by G.Guenther & G.Marquardt.

Teaching with ALGOL 68, in Manchester (invited lecture), by C.H.Lindsey.

The price of the proceedings, published as Mathematical Centre Tract 134, is Dfl 29.40 (including VAT). Foreign payments are subject to an additional Dfl 6.50 per remittance to cover bank, postal and handling charges. Forwarding of the publication(s) ordered from abroad will take place on receipt of remittance. Payment should be made in Dutch currency or its equivalent.

AB47.1.4 An Axiomatic Semantics for Expression Languages.

This is the Ph.D. Thesis of P.A.Pritchard, as submitted to the Australian National University. Subject to availability, copies may be obtained by writing to:

Professor P.A. Pritchard,
Department of Computer Science,
Cornell University,
Ithaca, New York 14853,
U.S.A.

The work is closely related to that on ALGOL 68 by Richard Schwartz (see AB44.1.5).

ABSTRACT

This thesis addresses the problem of giving Hoare-style axiomatic definition of the semantics of expression-oriented block-structured programming languages. This problem is tackled per medium of an exemplary expression language El which caters for the manipulation of both l-values and r-values.

A notational extension is presented which allows the effects of state-changing expressions to be naturally described, and a formal Hoare-style program logic D is then given which defines the partial correctness semantics of El.

Proofs of the consistency and completeness (in the sense of Cook) of D are obtained by a novel method involving a translation of El programs in an underlying statement-oriented language. This method enables clear comparisons to be made of the two styles of programming language.

It is shown that efficient syntax-driven program verification is possible

for El in both of the major styles, viz. backward substitution and symbolic execution, but that the latter style is more natural when l-values are manipulated.

Finally, the above mentioned work on El is related to and compared with other work in the literature concerned with "side-effects" in conventional languages, and Schwartz's closely related work on ALGOL 68 is examined in some detail.

AB47.1.5 An ALGOL 68 Indenter - Correction.

The following correction should be made to fix a bug in the ALGOL 68 indenter given in AB46.4.4. The bug only affects programs which contain an exit as a constituent of a serial-clause which is enclosed between brief delimiters.

AB46 p.34 "STATE=MIDDLER"+10

```
# THEN GAP => AND (CLAUSE<>EXIT) THEN GAP #
```

AB47.1.6 Book Review : Intermediate Language for Graphics.

by P.J.W.ten Hagen et al.
Mathematical Centre Tracts 130. ISBN: 90 6196 204 8

This text contains the definition of a special purpose data description language for pictures, the Intermediate Picture Language (ILP). The authors envisage ILP emdebbed in a high level programming language to provide, for example, variables, conditionals and loop constructs. All the parameters to ILP are constants. No details of this embedding are given. ILP is a high level plotfile and again no advice is offered on mapping ILP onto a plotfile. They do not envisage a FORTRAN implementation, and having eliminated the constraints such a language would imply has enabled the six authors to create a graphics language pleasant to read.

The essential construct of ILP is:

```
<picture>:: PICT (<dim>) <pname> <PE>
<PE>      :: {<picture element>} | {WITH <aname> DRAW <pname>}
```

The picture once defined and stored is subsequently referred to via its name <pname>. <dim> specifies the dimensions of the coordinates used in the picture elements.

Dimensions >4 seem to be unnecessary; (-7 is also syntactically correct)! The picture elements (lines, text, connected lines or curves) must all be of this dimension (unless a subspace has been selected). The authors do not indicate actions in the event of a conflict between the picture elements and the picture dimension. The simplest picture is a collection of picture elements. This simple picture may be invoked as part of another picture using the WITH <aname> DRAW <pname>. <aname> is the name of an attribute pack, a collection of attributes to be applied to <pname>. Attributes include line style, colour and width. More interesting attributes are the transformation control, specified either in basic operations such as translate, rotate, scale, or perspective projection or through a complete (affine or homogeneous) transformation matrix. The number of parameters required is determined by the current dimension specified as part of the picture header.

Since pictures invoke other pictures, mixing rules for the attributes are

necessary; Chapter 3 deals with attribute concatenation rules and semantics of ILP. Coordinate transformation matricies are multiplied together and, for example, for line style patterns the latest style invoked is used. The feature of ILP to be able to present the same picture on different output devices with differing attributes is ideal in a graphics environment.

The fourth and final chapter of the book is devoted to a discussion of the group's design goals and how far they have been able to achieve them. ILP was implemented in 1979 as a compiler and interpreter. The compiler checks the syntax and produces an efficient coding of the programs (a plotfile?) for the interpreter which drives the drawing machine. 60k bytes on a PDP11/45 for a plotfile spooler (be it so versatile) seems rather large!

Appendices one and two contain the syntax rules in BNF, though a book on graphics should have produced a graphic representation of them as for example in books on ALGOL68, and Pascal. The page numbers indicated in the index for the appendices are also one short! The last appendix is an example program, drawing a house and makes full use of ILP to produce a pythagoras tree for the curtains.

W.T.Hewitt,
Computer Graphics Unit, University of Manchester.

AB47.2 Letters to the Editor.

AB47.2.1 ALGOL 68 Syntax Chart on Microfiche.

135, East MAIN,
Apt. R6,
Westboro Ma. 01581,
U.S.A.

16th. Jan. 81.

Dr. Lindsey.

I recently received Algol Bulletin 46. I believe the following are errors in the syntax charts on the enclosed microfiche. Parallel and collateral clauses are missing, casts and format texts are not listed among the units, the possible suppression in a complex frame is not shown, pragments after the insertion of a boolean pattern are allowed, and only a single digit is allowed in a replicator. I hope this arrives in time to be helpful.

Keith Wright.

Reply by C.H.Lindsey:

I plead guilty on all counts. Also some further omissions in rowers were pointed out by Mr.G.Kaniuk. The casts and the format-texts were accidents during editing. The others are mostly features which I have never used, and certainly never teach. The diagrams below show how the offending items should have looked. I have prepared a corrected version of the microfiche. If any reader would like one, please send me a fiche-sized envelope addressed to yourself (readers in the U.K. are invited to supply a second-class stamp also).