

ALGOL 5

This specification – which is preliminary and may be subject to change – describes the Algol 5 compiler developed for the extended RC 4000 computer with drum and/or disc as backing store. The compiler translates an Algol 60 source program into a binary object program.

Program Configuration

The compiler and the object programs can only be executed under the control of the time-sharing system Monitor 1.

The compiler and the object programs can be loaded from the backing store by the basic operating system in the monitor. During execution they have no direct control over input/output, storage protection, and program interruption.

Algol programs can be translated simultaneously by copies of the compiler loaded from the same area on the backing store into different storage areas.

Source Language

The source language is Algol 60 with the exception of integer labels, value labels, value arrays, and own arrays. The language is extended with facilities for handling peripheral devices of all kinds (in block mode, record mode, or character mode) and for manipulating textstrings and bit patterns. The case concept, as proposed by Hoare, is also introduced.

The Algol program text can be input to the compiler from typewriter, paper tape, punched cards, magnetic tape, drum, disc, or a combination of these. Paper tape must be one-inch tape with 8 tracks punched in either the RC Flexowriter code or the ISO 7-bit character code. Punched cards must be 12-row, 80-column cards punched in the Hollerith code that is standard for the given installation. Input from drum, disc, or magnetic tape must be represented in the ISO 7-bit character code with 3 characters per word.

Procedures can be compiled separately, and are then available to all users like a standard procedure (e. g. sin, sqrt).

Object Program

The object program is stored on drum or disc as a sequence of relocatable segments of 256 words each. The program includes the automatic administration of transfers of segments to the core store at run time.

Variables of the real, integer, and boolean types are stored respectively in double words, words, and bytes. Buffer areas for peripheral devices are declared at block entry, and require core store space as specified by the programmer.

The execution time of a program depends on the amount of core store space available. The minimum requirements is 1,700 words plus the necessary space for variables. For long programs, 4,000 words plus space for variables is recommended.

Typical Execution Times in Microseconds

reference to subscripted variable,	
1 subscript	20
reference with check against bounds	29
integer + integer	7
real + real	18
go to statement, local label	8
for statement, each loop	30
call of empty procedure, no parameters	140
call of empty procedure, 6 parameters	270
reference to name parameter, simple	9
reference to name parameter, expression	130
call of sin, cos	450

Translation

The compiler requires a core store area of 4 K words and a working area on the backing store of sufficient size to hold the object program. If the backing store is a disc, a core store area of 8 K is recommended in order to obtain maximum compilation speed.

After a basic time of 0.4 seconds (drum) or 2.0 seconds (disc), the internal translation speed is about 1,200 characters/second equivalent to the generation of 300 instructions per second.

It is possible to translate Algol procedures separately on the backing store and use them as standard procedures within other Algol programs. The procedure code is then copied into the object program during translation.

The compiler itself is written in the assembly language Slang 3.