

DEC 3 1984

Distribution:

WM File ✓ 109.9  
WMEG  
NMSS  
REBrowning  
MJBell  
LHBarrett  
TCJohnson  
JVoglewede  
MRKnapp  
JOBunting  
LBHigginbotham

C2

JVSMITH1

- 1 -

MEMORANDUM FOR: J. Barry Badini  
System Support Branch  
Division of Automated Information Services, RM

FROM: John C. Voglewede  
Engineering Branch  
Division of Waste Management, NMSS

SUBJECT: APPLICATION OF DATA GENERAL FORTRAN 77 LANGUAGE

By this memorandum, your assistance is requested to resolve several problems involving application of the Data General FORTRAN 77 compiler on the NRC's MV/8000 computer. The issues of concern are (1) preservation of subprogram entities and (2) detection and correction of undefined variables. As discussed below, these two issues are somewhat related. Both appear to lie in the domain of compiler design rather than user or site-specific problems. It is expected that your support may simply involve obtaining additional information from Data General.

Background

The Division of Waste Management is primarily a reviewer, rather than a developer, of computer programs used nuclear waste management. Some of these computer programs exhibit poor programming techniques, which may lead to the problems identified above when run on certain computers. Rather than reprogramming (as developers), we must determine the adequacy of a given program as it was submitted (knowing our approval may require a restriction to a particular class of computers). This point should be kept in mind before dismissing the two problems as the result of poor programming practice.

Preservation of Subprogram Entities

As stated on page 7-2 of the FORTRAN 77 Environment Manual (Ref. 1), many non-DG FORTRAN compilers provide static (nonstack) storage of variables by default. This provides an implicit preservation of the values of local variables in subprograms from one CALL or function reference to the next. Implicit dependence upon this feature (as opposed to explicit use of the SAVE command or SAVEVARS compiler option) is clear a violation of the ANSI Standard (Ref. 2). Nevertheless, a large number of non-DG FORTRAN compilers provide this feature and it is often (incorrectly) used. I understand how to correct the problem (with the SAVEVARS compiler option), but I do not understand how to

WM Record File  
109.9

WM Project  
Docket No. 27  
PDR  
LPDR

DFC : WMEG	JW	: WMEG	: Distribution:	:	:	:
NAME : JVoglewede:gh:TCJohnson						
DATE : 12/03/84		: 12/3/84				
			8502280355 841203			
			PDR WASTE			
			WM-1		PDR	

JVSMITH1

- 2 -

detect it (at a local level in the code). Please note that the problem is not specific to non-DG FORTRAN programs imported to the MV/8000.

Detection of Uninitialized Variables

This concern is similar to that described above in that variables become undefined and are subsequently used or were never defined in the first place (uninitialized) before being used. Some loaders (e.g., CDC) offer the option of presetting all memory locations to some specific value before use. Other loaders (e.g., IBM) produced random results. In the case of presetting memory, it is possible to trap (i.e., detect) the use of uninitialized variables. I do not understand how this is accomplished with the Data General FORTRAN 77 compiler.

Correction of Uninitialized Variables

The FORTRAN 77 Reference Manual (Ref. 3) states that another effect of the SAVEVARS compiler switch is to initialize certain common blocks, local variables, and local arrays to zero (numeric), null (character), and .FALSE. (logical). It is not clear which variables are initialized (particularly when no SAVE statements are used) and what values are retained by local variables that are initialized with DATA statements but subsequently changed.

Summary

If possible, I would like a written response (from you or, if appropriate, Data General with copies to the ADP Information Technology Section) to the following questions: (1) How may uninitialized variables be detected in a Data General FORTRAN 77 environment? (2) How may variables that become undefined be detected in a Data General FORTRAN 77 environment? (3) What variables are initialized by the Data General SAVEVARS compiler switch, particularly when no SAVE statements are used? (4) What impact does the SAVE statement and SAVEVARS option have on variables initialized with a DATA statement (and subsequently changed by repeated calls to the subprogram).

---

DFC	:WMEG		:WMEG	:	:	:	:	:
NAME	:JVoglewede:gh:	TCJohnson	:	:	:	:	:	:
DATE	:11/ /84		:11/ /84	:	:	:	:	:

---

DEC 3 1984

JVSMITH1

- 3 -

Should you require additional information in order to respond to this request,  
please call me at 42-74275.

John C. Voglewede  
Materials Section  
Engineering Branch  
Division of Waste Management

cc: J. Shields  
James P. Smith  
F. Goldberg  
E. Robinson  
R. Codell

---

DFC	:WMEG	:WMEG	:	:	:	:	:
NAME	:JVoglewede:gh:TCJohnson	:	:	:	:	:	:
DATE	:11/ /84	:11/ /84	:	:	:	:	:

---

DEC 3 1984

JVSMITH1

- 4 -

REFERENCES

1. "FORTRAN 77 Environment Manual (AOS/VS)," Data General Corporation Report D93-000288-01, January 1984.
2. "American National Standard Programming Language FORTRAN," ANSI X3.9-1978 FORTRAN 77, American National Standards Institute, Inc.
3. "FORTRAN 77 Reference Manual," Data General Corporation Report 093-000162-02, October 1983.

---

DFC	:WMEG	:WMEG	:	:	:	:	:
NAME	:JVoglewede:gh:TCJohnson	:	:	:	:	:	:
DATE	:11/ /84	:11/ /84	:	:	:	:	:

---