

AD-A153 746

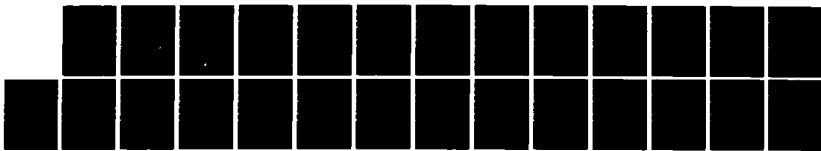
ADA (TRADEMARK) COMPILER VALIDATION SUMMARY REPORT
HONEYWELL INFORMATION. (U) FEDERAL SOFTWARE TESTING
CENTER FALLS CHURCH VA 14 DEC 84 OIT/FSTC-84/515

1/1

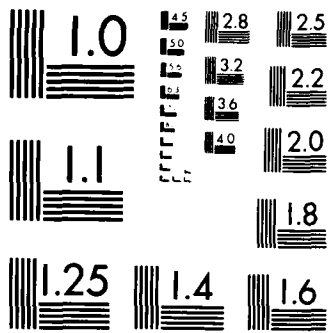
UNCLASSIFIED

F/G 9/2

NL



END
FILMED
DTC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION

REPORT

AD-A153 746

READ INSTRUCTIONS
BEFORE COMPLETING FORM
IN NO. 100-10
RECIPIENT'S CATALOG NUMBER

2

1. REPORT NUMBER
OIT/FSTC-84/515

4. TITLE (and Subtitle)
Ada Compiler Validation Summary Report
Honeywell Information Systems
GCOS6 Ada Version 1.1

5. TYPE OF REPORT & PERIOD COVERED
December 14, 1984
to December 14, 1985

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Federal Software Testing Center
Office of Software Development &
Information Technologies

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Federal Software Testing Center
Office of Software Development & Information
Technologies
Two Skyline Place, Suite 1100
5203 Leesburg Pike, Falls Church, VA 22041-3467

10. PROGRAM ELEMENT, PROJECT, TASK
AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Ada Validation Office
Institute for Defense Analyses
1801 N. Beauregard Street
Alexandria, VA 23311

12. REPORT DATE
December 14, 1985

13. NUMBER OF PAGES
22

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
Ada Joint Program Office
Room 3D-139 (1211 S. Fern, C-107)
The Pentagon
Washington, D.C. 20301-3081

15. SECURITY CLASS. (of this report)

15a. DECLASSIFICATION/DOWNGRADING
SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)
Unclassified

DTIC
SELECTED
MAY 15 1985
S A D

18. SUPPLEMENTARY NOTES
Final Version

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Ada Compiler Validation Summary Report, VSR, Ada Language, Validation Testing, ACVC,
Ada Compiler Validation Capability, FSTC, AVO Ada Validation Office, Ada Joint Program
Office, AJPO, Ada Validation Facility, AVF, Honeywell Information Systems, GCOS6
Ada Version 1.1, ANSI MIL/STD Ada, ACVC Version 1.4

20. ABSTRACT (Continue on reverse side if necessary and identify by block number).
The Honeywell Information Systems Compiler GCOS6 Ada version 1.1, for the DPS 6/95,
DPS 6/75, DPS 6/45, microSystem 6/20 and microSystem 6/10 using GCOS6 MOD400, was
tested with version 1.4 of the ACVC validation tests. Version 1.4 of the test suite
contained 2185 tests, of which 2008 were applicable to this implementation. Of the
applicable tests, 71 were withdrawn due to errors in the tests. Of the remaining
applicable correct tests 1937 passed, and no anomaly was discovered.

DTIC FILE COPY

OFFICE OF
SOFTWARE
DEVELOPMENT
AND
INFORMATION
TECHNOLOGY

FEDERAL
SOFTWARE
TESTING
CENTER



Prepared for:

HONEYWELL INFORMATION SYSTEMS, INC. "

Prepared by:

FEDERAL SOFTWARE TESTING CENTER
Office of Software Development
and Information Technology
Two Skyline Place, Suite 1100
5203 Leesburg Pike
Falls Church, VA 22041-3467

Report OIT/FSTC-84/515



Ada* COMPILER VALIDATION
SUMMARY REPORT

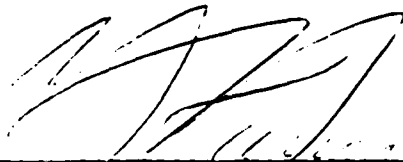
HONEYWELL INFORMATION
SYSTEMS, INC.
GCOS6 Ada Version 1.1
DPS 6/95, 6/75, 6/45,
microSystem 6/20, 6/10
using GCOS6 MOD400 Rel. 3.0

A-1

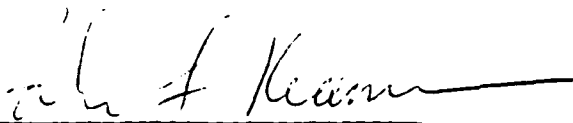
December 14, 1984

*Ada is a registered trademark of the U.S.
Government (Ada Joint Program Office)

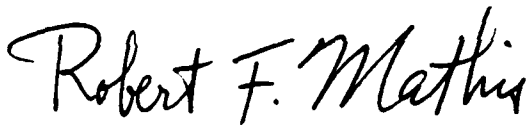
This report has been reviewed and is approved.



Richard G. Harrison, Acting Director
Federal Software Testing Center



Thomas H. Probert, Ph. D.
Institute for Defense Analyses



Robert F. Mathis
Director, AJPO

ABSTRACT

The Honeywell Information Systems Compiler GCOS6 Ada version 1.1, for the DPS 6/95, DPS 6/75, DPS 6/45, microSystem 6/20 and microSystem 6/10 using GCOS6 MOD400, was tested with version 1.4 of the ACVC validation tests. Version 1.4 of the test suite contained 2185 tests, of which 2008 were applicable to this implementation. Of the applicable tests, 71 were withdrawn due to errors in the tests. Of the remaining applicable correct tests 1937 passed, and no anomaly was discovered.

TABLE OF CONTENTS.

	Page
1. Introduction	1
1.1 Purpose of the Validation Summary Report	1
1.2 Use of the Validation Summary Report	2
1.3 References	3
1.4 Definitions of Terms	3
2. TEST ANALYSIS	6
2.1 Class A Testing	6
2.1.1 Class A Test Procedures	6
2.1.2 Class A Test Results	6
2.2 Class B Testing	8
2.2.1 Class B Test Procedures	8
2.2.2 Class B Test Results	8
2.3 Class C Testing	9
2.3.1 Class C Test Procedures	9
2.3.2 Class C Test Results	9
2.4 Class D Testing	9
2.4.1 Class D Test Procedures	9
2.4.2 Class D Test Results	9
2.5 Class E Testing	9
2.5.1 Class E Test Results	10
2.6 Class L Testing	10
2.6.1 Class L Test Procedures	10
2.6.2 Class L Test Results	10
3. COMPILER NONCONFORMANCES	11
4. ADDITIONAL INFORMATION	12
4.1 Compiler Parameters	12
4.2 Testing Information	13
4.2.1 Pre-Test Procedures	13
4.2.2 Control Files	13
4.2.3 Test Procedures	13
4.2.4 Test Analysis Procedures	14
4.2.5 Performance Information	14
4.2.6 Description of Errors in Withdrawn Tests	14
4.2.7 Description of Inapplicable Tests	17
4.2.8 Information derived from the Tests	18
4.2.9 Anomalous Test Results	21
5. SUMMARY AND CONCLUSIONS	22

LIST OF EXHIBITS

Page

1. Summary of Test Performance 7

1. Introduction

1.1 Purpose of the Validation Summary Report

This report describes the results of the validation effort for the following Ada compiler:

Host Machine: DPS 6/95, DPS 6/75, DPS 6/45,
and microSystem 6/20

Operating System: GCOS6 MOD400 Release 3.0 update 4

Host Disk System: DPS 6/95
MSU 9604, MSU 9607, MSU 9602, CDU 9639
DIU 9603, DIU 9608

DPS 6/75
MSU 9607, CDU 9639, DIU 9603, DIU 9608

DPS 6/45
CDU 9639, CDU 9652, DIU 9608

microSystem 6/20
2 built in units 40 Mbyte (CDU 9652)
1 built in 5 and one quarter inch
diskette drive

Target Machine: DPS 6/95, DPS 6/75, DPS 6/45,
microSystem 6/20

Operating System: GCOS6 MOD400 Release 3.0 update 4

Host Machine: DPS 6/95, DPS 6/75, DPS 6/45,
microSystem 6/20

Operating System: GCOS6 MOD400 Release 3.0 update 4

Target Machine: microSystem 6/10

Target Disk System: 2 built in 5 and one quarter inch
diskette drives
1 19.5 Mbyte hard disk

Operating System: GCOS6 MOD400 Release 3.0

Language Version: ANSI/MIL-STD-1815A Ada

Translator Name: GCOS6 Ada

Translator Version: 1.1

Validator Version: 1.4

Testing of this compiler was conducted by the Federal Software Testing Center under the supervision of the Ada Validation Office (AVO), at the direction of the Ada Joint Program Office. Testing was conducted from 84/11/12 through 84/11/15 at Billerica, Massachusetts in accordance with AVO policies and procedures.

The purpose of this report is to document the results of the testing performed on the compiler, and in particular, to:

- . identify any language constructs supported by the compiler that do not conform to the Ada standard.
- . identify any unsupported language constructs required by the Ada standard.
- . describe implementation-dependent behavior allowed by the standard.

1.2 Use of the Validation Summary Report

The Ada Validation Office may make full and free public disclosure of this report in accordance with the "Freedom of Information Act" (5 U.S.C. #552). The results of the validation are only for the purpose of satisfying United States Government requirements, and apply only to the computers, operating systems, and compiler version identified in this report.

The Ada Compiler Validation Capability is used to determine insofar as is practical, the degree to which the subject compiler conforms to the Ada standard. Thus, this report is necessarily discretionary and judgemental. The United States Government does not represent or warrant that the statements, or any one of them, set forth in this report are accurate or complete, nor that the subject compiler has no other nonconformances to the Ada standard. This report is not meant to be used for the purpose of publicizing the findings summarized therein.

Question regarding this report or the validation tests used for the validation should be sent to:

Validation Program Manager
Federal Software Testing Center
5203 Leesburg Pike, Suite 1100
Falls Church, VA 22041-3467

1.3 References

Reference Manual for the Ada Programming Language, ANSI/MIL-STD-1815A, February 1983.

Ada Validation Organization: Policies and Procedures, Mitre Corporation, June 1982, PB 83-110601.

Ada Compiler Validation Implementers' Guide, SofTech, Inc., October 1980.

The Ada Compiler Validation Capability, Computer, Vol. 14, No. 6, June 1981.

Using the ACVC Tests, SofTech, Inc. November 1981.

Ada Compiler Validation Plans and Procedures, SofTech, Inc. November 1981.

1.4 Definitions of Terms

Class A tests are passed if no errors are detected at compile time. Although these tests are constructed to be executable, no checks can be performed at run-time to see if the test objective has been met.

Class B tests are illegal programs. These tests pass if all the errors they contain are detected at compile-time (or link-time) and no legal statements are considered illegal by the compiler.

Class L tests consist of illegal programs whose errors cannot be detected until link time. These tests pass if errors are detected prior to beginning execution of the main program.

Class C tests consist of executable self-checking programs. These tests pass if they complete execution and do not report failure.

Class D tests are capacity tests. Since there are no firm criteria for the number of identifiers permitted in a compilation, number of units in a library, etc., a compiler may refuse to compile a class D test. However, if such a test is successfully compiled, it should execute without reporting a failure.

Class E tests provide information about an implementation's interpretation of the Standard. Each test has its own pass/fail criterion.

CUSTOMER: The agency requesting the validation (Honeywell Information Systems, Inc.)

HOST: The computer(s) on which the compiler executes is DPS 6/95, DPS 6/75, DPS 6/45 and microSystem 6/20.

FSTC: Federal Software Testing Center. In the context of this report the FSTC conducts Ada validations under contract to the AVO as a satellite facility.

ACVC: Acronym for the Ada Compiler Validation Capability.

RM: The Ada Language Reference Manual (ANSI/MIL-STD-1815A).

IG: ACVC Implementers' Guide.

AVO: The Ada Validation Office. In the context of this report the AVO is responsible for directing compiler validation.

TARGET: The computer(s) for which the compiler generates object code is DPS 6/95, DPS 6/75, DPS 6/45 microSystem 6/20 and microSystem 6/10.

VALIDATION: The process of validating a compiler. The term is used interchangeably with test or compiler test.

VALIDATION TESTS: The generic form used to refer to a set of test programs which evaluate how closely a compiler conforms to its language specification. In this report, the term will be used (unqualified) to mean the ACVC tests.

implementations (and the numeric literal might occur in an unexecutable portion of code). This test was rejected at compile time.

- . B26005A.ADA: This test contains all the ASCII control characters in string literals. GCOS6 Ada rejected all of these strings.
- . D29002K-B.ADA: This test declares 1000 identifiers and was passed by GCOS6 Ada.
- . D4A002A-AB.ADA, D4A002B.ADA, D4A004A-AB.ADA, and D4A004B.ADA: These tests contain universal integer calculations requiring 32 and 64 bits of accuracy, i.e., values that exceed SYSTEM.MAX_INT are used. An implementation is allowed to reject programs requiring such calculations. GCOS6 Ada passed test D4A002A-AB, and rejected tests D4A002B, D4A004A and D4A004B.
- . C52103X-B.ADA, E52103Y-B.ADA, C52104X-B.ADA, C52104Y-B.ADA: These tests declare BOOLEAN arrays with INTEGER'LAST+3 components. An implementation may raise NUMERIC_ERROR at the type declaration or STORAGE_ERROR when array objects of these types are declared, or it may accept the type and object declarations. GCOS6 Ada raised NUMERIC_ERROR.
- . A series of tests (D55A03*-AB.ADA) check to see what level of loop nesting is allowed by an implementation. Tests containing 65 or fewer nested loops passed.
- . D56001B-AB.ADA contains blocks nested 65 levels deep. This test was passed.
- . C94004A-B.ADA: This test checks to see what happens when a library unit initiates a task and a main program terminates without insuring that the library unit's task is terminated. An implementation is allowed to terminate the library unit task, or it is allowed to leave the task in execution. This test showed that such library tasks terminated execution when the main program terminated.
- . CA1012A4M-B.DEP: This test checks whether an implementation requires generic library unit bodies to be compiled in the same compilation as the generic declaration. GCOS6 Ada allows generic

- . CE2102F-B was inapplicable because the compiler supports INOUT_FILE.
- . CE2102G-B was inapplicable because the compiler supports RESET and DELETE.
- . CE2107C-B, CE2107D-B, CE2107E-B, CE2110B-B, CE2111D-B, CE3111B-B, CE3111C-B, CE3114B-B, and CE3115A-B were inapplicable because a file open for output cannot be associated with more than one internal file.
- . CE2401D-B was inapplicable because the compiler does not support I/O with unconstrained array and record types.
- . CE2108A-B, CE2108C-B, CE2108F-B and CE3112A-B were inapplicable because temporary files do not have names on this system. Therefore, calls to NAME for temporary files raise USE_ERROR.
- . CE2108B-B, CE2108D-B, CE2108F-B were inapplicable because the tests depend on files created by one of the above tests.
- . CE2202A-B was inapplicable because the file system restricts records to <16K characters.
- . CE3708A-B was inapplicable because the implicit conversion of the literal 36382 into INTEGER (due to ident_int) causes NUMERIC_ERROR to be raised (failed) because the implementation cannot represent the value as an INTEGER.

Results for inapplicable class D and E tests are given in Section 4.2.8.

4.2.8 Information derived from the Tests

Processing of the following tests indicated support as described below for a variety of implementation options examined by the tests.

- . E24101A-B.TST: If a based integer literal has a value exceeding SYSTEM.MAX_INT, an implementation may either reject the compilation unit at compile time or raise NUMERIC_ERROR at run-time. Raising NUMERIC_ERROR at run-time is preferred, since it makes programs compilable for a wider variety of

- . The exception handler in lines 87 to 89 does not reflect that exception INCOMPLETE is raised by inner exception handlers for USE_ERROR. These exceptions will be handled by the OTHERS choice (incorrectly) resulting in "failed". An additional exception handler "WHEN INCOMPLETE => RAISE;" should be added before line 88: CE3103A-B.

4.2.7 Description of Inapplicable Tests

The following tests in version 1.4 of the ACVC were considered inapplicable for this implementation for the reasons given below:

- . B86001DT-AB was inapplicable because the compiler does not support LONG_LONG_INTEGER.
- . C34001F-B, C35702A-AB and B86001CP-AB were inapplicable because the compiler does not support SHORT_FLOAT.
- . C24113L through C24113Y-B, C35705L through C35705Y-B, C35706L through C35706Y-B, C35707L through C35707Y-B, C35708L through C35708Y-B, C35802L through C35802Y-B, C45241L through C45241Y-B, C45421L through C45421Y-B, C45424L through C45424Y-B, and C45621L through C45621Z-B were inapplicable because they contain tests that exceed the machine capacity using DIGITS precision greater than MAX_DIGITS = 15.
- . C24113H through C24113K-B were inapplicable because these tests contains literals that exceed the maximum length of 120 characters supported by the GCOS6 Ada compiler.
- . C910AHA-B was inapplicable because the NATURAL variable SPYNUMB is increased from 0 up to 123456 (line 38) and this number is larger than NATURAL'LAST (= INTEGER'LAST) supported by the implementation.
- . CE2102D-B was inapplicable because the compiler supports IN_FILE.
- . CE2102E-B was inapplicable because the compiler supports CUT_FILE.

- . These tests attempt a comparison of a null string literal with a variable. Evaluation of the string literal correctly raises CONSTRAINT_ERROR because the lower bound of the literal is INTEGER'FIRST. The test does not include an exception handler for this exception, and execution terminates abnormally: C52104G-AB and C52104Q-AB.
- . In C52103X-B the slice assignment in lines 125 to 127 may raise NUMERIC_ERROR in the evaluation of the slices or the length test, prior to assignment. Hence, the check performed in lines 147 to 173 may fail because no values have been assigned to the four elements of ARR42 that are tested. The check in lines 147 to 173 should be performed only if no exceptions are raised during the slice assignment of the lines 125 to 127.
- . In C55B15A-B the CONSTRAINT_ERROR in line 89 should be changed to NUMERIC_ERROR.
- . In C87B04A-B an overloaded function call for the function "+" was ambiguous.
- . In C87B10A-B there are literal values that are outside an integer base type for some implementations.
- . In C87B26B-B at line 119, the attribute STORAGE_SIZE is applied to a prefix which is an object of an access type. This is illegal; STORAGE_SIZE is allowed only with a prefix that is an access type/subtype ("is", not "is appropriate for") or for a task type/subtype or object of a task type.
- . An array aggregate with named associations and an (others) choice must be qualified in an assignment context: C87B31A-B.
- . In C95008A it was possible for an entry call to call a terminated task, depending on the implementation.
- . In C95009A an unintended race condition in a tasking test allowed a null access value to be dereferenced before the access variable was assigned the access value of an allocated task.

- . The accept statement in B950BAA-B at line 38 is marked as being correct.
- . C37011A-B checks that sliding occurs for initialization of record components that are of an array type. However, 3.2.1(16) of the Ada standard is interpreted as restricting sliding (more precisely, an implicit array subtype conversion) to the initialization of a complete array object (that is, one declared by an "object declaration"; a record component is declared by a "component declaration" according to 3.7(2)).
- . An incomplete type with discriminants was constrained before its full declaration occurred. An implementation is allowed to reject such subtype indications because of an ambiguity in the language: C38104A-B.
- . The (model) interval used in the test of C (lines 151-152) is too narrow: C45321A,B,..Y-B.
- . The (model) interval used in the test of C (lines 181-182) is too narrow: C45521A,B,..Z-B.
- . The number 23.4 used in lines 28 and 33 is neither a model number of the float subtype FLT nor the anonymous type derived in line 15 (LRM 3.5.7(11)). A model number should have been used instead of 23.4 (e.g. 23.5): C52001B-AB.
- . In line 76, INTEGER'LAST is compared with SYSTEM.MAX_INT without allowing (by a special exception handler) the implicit conversion of SYSTEM.MAX_INT to INTEGER (before comparison) to raise NUMERIC_ERROR. This is an unintended omission in the test program. Line 136 may also (correctly) raise NUMERIC_ERROR when trying to implicitly convert W_LIT to INTEGER: C52007A-B.
- . A test assumed that when a constrained array type was declared using a range, e.g., 1..10, that the index subtype was INTEGER instead of INTEGER range 1..10: C52102A-AB and C52102B-AB.

4.2.4 Test Analysis Procedures

On completion of testing, all results were analyzed for failed class A, C, D, E, or L programs, and all class B compilation results were individually analyzed. Analysis procedures are described for each test class in chapter 2. Tests found to contain errors were withdrawn.

4.2.5 Performance Information

The real (i.e., wall clock hh:mm) time required for running was:

- . DPS 6/95 - 55:15 (standalone)
- . DPS 6/75 - 87:08 (standalone)
- . DPS 6/45 - 101:12 (standalone)
- . microSystem 6/20 - 220:24 (combined testing time on two standalone systems)
- . microSystem 6/10 - 04:10 (standalone execution only)

The above times for the DPS 6/95, 6/75, 6/45 and microSystem 6/20 reflect compilation and execution time. The DPS 6/95 was also used to compile 784 class B tests that were not run on the other host machines. The microSystem 6/10 was tested as a target system, therefore, the above time for the microSystem 6/10 reflects execution time only.

4.2.6 Description of Errors in Withdrawn Tests

The following tests in version 1.4 of the ACVC did not conform to the ANSI Ada standard and were withdrawn for the reasons given below:

- . In line 66 of B43201B-B, the 'others' choice in the aggregate which begins on the previous line is marked as being okay. (The first discrete range is static, but the second is not; this does not satisfy the requirements for a static index constraint in the Ada standard, 4.9(11)).
- . B43203B-B checks the legal use of an 'others' choice in array aggregates. In line 80, the test incorrectly indicates that the use of 'others' should be diagnosed in error.

- . TEXT_IO.FIELD'LAST: 16#7FFFFFFF#
- . illegal external file name1: BAD@CHAR[S]
- . illegal external file name2: "TOO_LONG_A_FILE_NAME"
- . SYSTEM.PRIORITY'FIRST: 0
- . SYSTEM.PRIORITY'LAST: 48

4.2 Testing Information

Tests were compiled/executed at Billerica, Massachusetts.

4.2.1 Pre-Test Procedures

Prior to testing, appropriate values for the compiler-dependent parameters were determined. These values were used to adapt tests that depend on the values. A magnetic tape containing the adapted tests [and split versions of some class B tests (see Section 2.2.2)] was prepared and brought to the testing site.

4.2.2 Control Files

Honeywell Information Systems provided command procedures that compiled and executed tests automatically.

4.2.3 Test Procedures

All files from the version 1.4 tape were read onto disk. The package REPORT and the procedure CHECK-FILE were first compiled and the corresponding library file saved. The tests checking the REPORT package and CHECK-FILE procedure were executed on all test systems. Then all executable tests were grouped into batch jobs by class and by chapter and compiled and executed on the DPS 6/95, DPS 6/75, DPS 6/45, microSystem 6/20. The class B tests were run on the DPS 6/95. The DPS 6/75 acted as a host for the microSystem 6/10. The object code from the executable tests were downloaded from the DPS 6/75 via floppy disks to the microsystem 6/10 and then executed. Additional testing was performed using the DPS 6/95, DPS 6/45 and microSystem 6/20 as host for the target microSystem 6/10.

4. ADDITIONAL INFORMATION

This section describes in more detail how the validation was conducted.

4.1 Compiler Parameters

Certain tests do not apply to all Ada compilers, e.g., compilers are not required to support several predefined floating point types, and so tests must be selected based on the predefined types an implementation actually supports. In addition, some tests are parameterized according to the maximum length allowed by an implementation for an identifier (or other lexical element; this is also the maximum line length), the maximum floating point precision supported, etc. The implementation dependent parameters used in performing this validation were:

- . maximum lexical element length: 120 characters
- . maximum digits value for floating point types: 15
- . SYSTEM.MIN_INT: -2**31
- . SYSTEM.MAX_INT: 16#7FFFFFFF#
- . predefined numeric types: INTEGER, SHORT_INTEGER, LONG_INTEGER, FLOAT, LONG_FLOAT
- . INTEGER'FIRST: -32768
- . INTEGER'LAST: 32767
- . source character set: ASCII
- . extended ascii chars: "ABCDEFGHJKLMNOPQRSTUVWXYZ!\$%?[@]^'{}"
- . non-ascii char type: GCOS6 Ada accepts only ASCII characters
- . TEXT_IO.COUNT'LAST: 16#7FFFFFFF#

3. COMPILER NONCONFORMANCES

For this implementation there were no nonconformances or anomalies noted during the official validation. Section 4.2.9 provides a definition of an anomaly and detailed information describing an anomaly if any were noted during the official validation.

2.5.1 Class E Test Results

Of the 7 class E tests, 6 passed and 1 was found to be inapplicable. See Sections 4.2.7 and 4.2.8 for further information.

2.6 Class L Testing

Class L tests check that incomplete or illegal Ada programs involving multiple separately compiled source files are detected at link time and are not allowed to execute. There were 60 test programs processed in this validation attempt.

2.6.1 Class L Test Procedures

Each Class L test is separately compiled and execution is attempted. The tests produce FAIL messages if executed. Any "failed" tests are individually checked to see if they are correct and if they are applicable to the implementation. Any tests that are inapplicable or that do not conform to the Ada standard are withdrawn.

2.6.2 Class L Test Results

All 60 class L tests passed.

2.3 Class C Testing

Class C tests check that legal Ada programs are correctly compiled and executed by an implementation. There were 1262 class C tests processed in this validation attempt.

2.3.1 Class C Test Procedures

Each Class C test is separately compiled and executed. The tests are self-checking and produce PASS/FAIL messages. Any 'failed' tests are individually checked to see if they are correct and if they are applicable to the implementation. Any tests that are inapplicable or that do not conform to the Ada Standard are withdrawn.

2.3.2 Class C Test Results

Of the 1262 class C tests, 1023 passed and 171 were found to be inapplicable to this implementation. Also 68 tests were withdrawn because of errors in the tests. See Sections 4.2.6 and 4.2.7 for further information.

2.4 Class D Testing

Class D tests are executable tests used to check an implementation's compilation and execution capacities. There were 14 class D tests used in this validation.

2.4.1 Class D Test Procedures

Each class D test is separately compiled and executed. The tests are self-checking and produce PASS/FAIL messages.

2.4.2 Class D Test Results

Of the 14 class D tests, 11 passed and 3 were found to be inapplicable to this implementation. None were withdrawn because of errors in the tests. See Section 4.2.7 for further information.

2.5 Class E Testing

Class E tests are executable tests that provide information about an implementation's interpretation of the Standard in areas where the Standard permits implementations to differ. Each test has its own pass/fail criterion. There were 7 class E tests used in this validation.

2.2 Class B Testing

Class B tests check the ability to recognize illegal language usage. There were 784 class B tests processed.

2.2.1 Class B Test Procedures

Each class B test was separately compiled. The resulting test compilation listings are manually examined to see whether every illegal construct in the test is detected. If all errors are not detected, a version of the test is created that contains only undetected illegal constructs. This revised version is recompiled and the results analyzed. If all errors are still not detected, the revision process is repeated until a revised test contains only a single illegal construct.

A B test is considered to fail only if a version of the test containing a single illegal construct is accepted by the compiler (i.e., an illegal construct is not detected) or a version containing no errors is rejected (i.e., a legal construct is rejected).

2.2.2 Class B Test Results

There were 784 class B tests presented to the compiler. Of these tests 2 were found to be inapplicable to this implementation (see Section 4.2.7); 3 tests were found to be incorrect (i.e., a conforming compiler would have failed each of these tests). All 779 remaining class B tests passed.

Because all errors were not detected when compiling the original tests, the following 14 tests were modified and divided into 39 separate units by removing the detected errors; the modified tests were then resubmitted to see if the remaining errors would be detected:

B22003A.ADA	B29001A-B.ADA	B33004A.ADA	B35101A.ADA
B37004A-B.ADA	B37301A-AB.ADA	B52002E-AB.ADA	B53009A-AB.ADA
B55A01H-AB.ADA	B67001A.ADA	B91001A-B.ADA	B97101A.ADA
B97101E.ADA	BC10AEA.ADA		

All illegal constructs were eventually detected except in some tests that were withdrawn because of errors in the tests (see Section 4.2.6).

Test Class	Pr	NA	Er	P	A	F	FE	FC	Ps
A	58	0	0	58	0	0	0	0	0
B	784	2	3	779	0	0	0	0	0
C	1262	171	68	1023	0	0	0	0	0
D	14	3	0	11	0	0	0	0	0
E	7	1	0	6	0	0	0	0	0
L	60	0	0	60	0	0	0	0	0
Total	2185	177	71	1937	0	0	0	0	0

EXHIBIT 1: Summary of Test Performance

2. TEST ANALYSIS

A summary of tests processed, by class, is given in Exhibit 1, where:

Pr = processed.
NA = found to be inapplicable for this implementation.
Er = found to be incorrect, and withdrawn from the validation.
P = passed.
A = anomalies.
F = failed.
FE = failed to execute to completion.
FC = failed to compile successfully.
Fs = total of all failures (i.e., F+FE+FC).

Exhibit 1 shows that the Honeywell Information Systems compiler passed all applicable correct tests.

There were 177 tests in the suite that were processed and found to be not applicable to the GCOS6 Ada compiler (see Section 4.2.7).

In addition, 71 tests were withdrawn from the test suite because they did not conform to the ANSI/MIL-STD-1815A Standard for the Ada Language (see Section 4.2.6 for details.).

2.1 Class A Testing

Class A tests check that legal Ada programs can be successfully compiled. These tests are executed but contain no executable self-checking capabilities. There were 58 class A test programs processed in this validation.

2.1.1 Class A Test Procedures

Each class A test was separately compiled and executed. However, the only purpose of execution is to produce a message indicating that the test passed.

2.1.2 Class A Test Results

Successful compilation and execution without any error messages indicates the tests passed. All 58 tests passed.

declarations and bodies to be compiled in completely separate compilations.

- . BC3204C*-B.ADA and BC3205D*-B.ADA: These tests contain a separately compiled generic declaration, some instantiations, and a body. An implementation must reject either the instantiations or the body. GCOS6 Ada rejected the body because of the instantiations.
- . CE2106A-B.DEP and CE3110A-B.DEP: These tests confirmed that dynamic creation and deletion of files is supported.
- . EE3102C-B.ADA: This test confirmed that an Ada program can open an existing file in OUT_FILE mode, and cannot create an existing file in either OUT_FILE or IN_FILE mode.
- . CE3111A-B.DEP: This test confirmed that more than one internal file could be associated with the same external file provided that both internal files were opened for reading.

4.2.9 Anomalous Test Results

An anomaly is test behavior that suggests the implementation does not conform to the Standard. The test behavior is not, however, considered to demonstrate nonconformance to the Standard for purposes of this validation attempt. The reasons for discounting an anomalous test result vary, depending on the anomaly. The existence of an anomaly sometimes means that new tests should be added to the suite to demonstrate the suspected nonconformance in a more convincing manner. Sometimes, when the anomaly is the result of procedural errors during the validation attempt, the existence of an anomaly requires revisions in validation procedures.

No anomalies were noted for this validation.

5. SUMMARY AND CONCLUSIONS

The Ada Validation Office identified 2185 of the ACVC version 1.4 tests as being being potentially applicable to the validation of the Honeywell Information Systems GCOS6 Ada compiler hosted on the DPS 6/95, DPS 6/75, DPS 6/45 and microSystem 6/20. Of these, 71 were withdrawn due to test errors, and 177 were determined to be inapplicable after they were processed. The compiler passed the remaining 1937 tests. These executable tests were also successfully run on the target microSystem 6/10.

The AVO considers these results to show acceptable compliance to the February 1983 ANSI Ada Reference Manual.

END

FILMED

6-85

DTIC