SC28-6865-1

Program Product

IBM OS FORTRAN IV (H Extended) Compiler and Library (Mod II) Messages

Program Numbers: 5734-FO3
5734-LM3

This publication contains a detailed description of the messages generated by the FORTRAN IV (H Extended) Compiler and the FORTRAN IV Library (Mod II), IBM Program Products that operate under the control of the IBM System/360 or System/370 Operating System.

PREFACE

This publication describes the messages produced by the FORTRAN IV (H Extended) Compiler and the FORTRAN IV Library (Mod II). The compiler issues both informative messages and diagnostic messages; the library issues diagnostic messages only. Informative messages are unnumbered and are listed in the sequence in which they appear on an output listing; diagnostic messages are numbered and are listed in numeric sequence.

INDUSTRY STANDARDS REFLECTED IN THIS PRODUCT

This product is designed according to the specifications of the American National Standard (ANS) FORTRAN, X3.9-1966 as understood and interpreted by IBM as of April 1971.

REFERENCE PUBLICATIONS

For information describing messages and codes produced by the operating system, the reader should consult the following publication:

IBM System/360 Operating System Messages and Codes
Order No. GC28-6631

OS/VS Message Library: VS1 System Messages Order No. GC38-1001

OS/VS Message Library: VS1 System Codes
Order No. GC38-1003

OS/VS Message Library: VS2 System Messages Order No. GC38-1002

OS/VS Message Library: VS2 System Codes
Order No. GC38-1008

For information on the form of a dump, and how to interpret the dump, the reader should see the following publications:

OS/MFT and OS/MVT Programmer's Guide to Debugging, Order No. GC28-6670

OS/VS1 Debugging Guide, Order No. GC24-5093

OS/VS2 Debugging Guide, Order No. GC28-0632

The FORTRAN programmer using the FORTRAN IV (H Extended) Compiler and the FORTRAN IV Library (Mod II) should be familiar with the information in the following publications:

IBM System/360 and System/370 FORTRAN IV Language
Order No. GC28-6515

OS FORTRAN IV Library Mathematical and Service Subprograms
Order No. GC28-6818

OS FORTRAN IV Mathematical and Service Suprograms Supplement for Mod I and Mod II Libraries
Order No. SC28-6864

OS FORTRAN IV (H Extended) Compiler Programmer's Guide
Order No. SC28-6852

Second Edition (November 1974)

This edition, as amended by Technical Newsletter SN20-9192, applies to Release 2.1 of IBM OS FORTRAN (H Extended) Compiler and FORTRAN IV Library (Mod II), Program Numbers 5734-FO3 and 5734-LM3, and to any subsequent releases of either system unless otherwise indicated in new editions or technical newsletters.

Significant system changes are summarized under "Summary of Amendments," following.

Information in this publication is subject to significant change. Any such changes will be published in new editions or technical newsletters. Before using the publication, consult the latest *IBM System/360 Bibliography*, GC20-0360, or *IBM System/370 Bibliography*, GC20-0001, and the technical newsletters that amend the particular bibliography, to learn which editions and technical newsletters are applicable and current.

Requests for copies of IBM publications should be made to the IBM branch office that serves you.

Forms for readers' comments are provided at the back of the publication. If the forms have been removed, comments may be addressed to IBM Corporation, P. O Box 50020, Programming Publishing, San Jose, California 95150. All comments and suggestions become the property of IBM.

© Copyright International Business Machines Corporation 1971, 1974

SUMMARY OF AMENDMENTS

Number 2

Date of Publication: January 1977

Form of Publication: TNL SN20-9192 to SC28-6865-1

Diagnostic Messages

Service Change

A number of diagnostic messages have been added or modified, and some message explanations have been clarified.

SUMMARY OF AMENDMENTS

Number 1

Date of Publication: November 1974

Form of Publication: Revision, SC28-6865-1

New Messages

New: Programming and Documentation

The following messages have been added: IFE042I, IFE118I, IFE119I, IFE201I, IFE261I, IFE262I, IFE263I, IFE264I, IFE281I, IFE309I, IFE311I, IFE321I, IFE362I, IFE432I, IFE433I, IFE434I, IFE435I, IFE436I, IHO243I

Deleted Messages

Maintenance: Programming and Documentation

The following messages have been deleted: IFE044I, IFE045I, IFE061I, IFE141I, IFE159I, IFE210I, IFE222I, IHO242I, IHO243I, IHO244I.

Clarification of Messages

Maintenance: Documentation Only

The discussion of many messages has been expanded to more fully describe error causes or standard corrective action.

Editorial changes having no significance are not noted here.

Specific changes to the text as of this publication date are indicated by a vertical bar to the left of the text. These bars will be deleted at any subsequent republication of the page affected.

CONTENTS

Figures

INTRODUCTION

The FORTRAN IV (H Extended) compiler produces messages describing conditions encountered during the compile job step. The FORTRAN IV library (Mod II) produces messages describing conditions encountered during the load module execution step.

This publication documents compiler and library messages, listing them in the following sequence:

1. The message number, if any, and the message text. Message text may contain variable information, that is, information that is specific to the part of the program causing the message to be generated, such as a statement number or a user-supplied name. The first occurrence of any variable information in message text is noted by an X or a string of XXXs; the second occurrence by a string of YYYs; and the third occurrence by a string of ZZZs.

2. An explanation of the message. The explanation identifies the variable information in the text and summarizes the possible conditions causing the message to be generated.

3. The condition code, where applicable, associated with the message. The condition code indicates the severity of the condition generating the message.

4. System action taken to correct the condition. System action is usually one of the following, depending on the severity of the condition:

   a. Ignore the condition and continue job processing.

   b. Continue processing but terminate the job at the end of the current job step (for example, complete the compile step but do not go on to the link edit step).

   c. Terminate processing immediately.

5. Programmer or operator response. The response suggests appropriate action which the user should take to correct the condition. Where no programmer action is required, as in the case of informative messages, the programmer response is stated as "None."

COMPILER MESSAGES

The FORTRAN IV (H Extended) compiler generates two types of messages:

- Informative messages
- Error/warning messages

Informative messages are unnumbered; they provide compiler status information, such as the options in effect and the amount of storage used during the compile step. Error/warning messages are numbered; they inform the programmer of conditions detected by the compiler during source program compilation.

INFORMATIVE MESSAGES

Informative messages appear before and after each compilation. No programmer response is needed for these messages.

```
LEVEL n  program-name  OS/360 FORTRAN H EXTENDED  DATE yy.ddd/hh.mm.ss
 PAGE nn
```

Explanation: This message is printed on each page of the compiler output listing and indicates the level number of the compiler, the name of the source module, and the date the job was run.

LEVEL n
indicates the release number of the compiler, and the month and year it was released.

program-name
is the name of the program or subprogram being compiled. This entry is not printed on the first page of the listing.

yy.ddd/hh.mm.ss
indicates when the job was run, specifying the year (yy), and the day in the year (ddd, as an entry from 01 to 365), followed by the precise time that the compilation began, specifying the hour (hh), the minute (mm), and the second (ss), based on a 24-hour clock. An example of this entry is:

71.031/22.35.37

This example states that the compilation was run in the year 1971, on the 31st day of the year, and began at 2200 hours (10 o'clock at night), 35 minutes past the hour, and 37 seconds past the minute.

nn
is the page number of the listing.

Programmer Response: None.

```
REQUESTED OPTIONS:  option[option...]
```

```
OPTIONS IN EFFECT:  option[option...]
```

Explanation: These messages are printed on the first page of every source listing. The first message prints those options explicitly requested in the PARM field of an EXEC statement (either the statement specifying cataloged procedures or the statement specifying the compiler) or in an *PROCESS card. The second message prints those options, including default options, that were in effect during compilation.

Programmer Response: None.

```
*OPTIONS IN EFFECT*  option[option...]

*OPTIONS IN EFFECT*  option[option...]

*STATISTICS*  SOURCE STATEMENTS= nn, PROGRAM SIZE= nnn, SUBPROGRAM
   NAME= name

*STATISTICS*  nnnn DIAGNOSTICS GENERATED, HIGHEST SEVERITY CODE IS n

          or

*STATISTICS*  NO DIAGNOSTICS GENERATED
```

Explanation: These messages appear on the last page of the compiler output listing, immediately before the End of Compilation message (detailed next in this list). They indicate the options in effect, the size of the source module and the object module, and whether any diagnostic messages were generated.

nn
is the number of source statements (excluding comments cards), expressed as a decimal number.

nnn
is the size of the object module, in bytes, expressed as a decimal number.

name
is the name of the program or subprogram being compiled.

nnnn
is the number of diagnostic messages issued for each compilation, expressed as a decimal number.

n
is the highest condition code issued for any message. Condition codes are summarized in the opening paragraphs of "Error/Warning Messages."

Programmer Response: None.

```
******END OF COMPILATION******     nnnnK BYTES OF CORE NOT USED
```

Explanation: This message is generated at the end of every compilation.

nnnnK
is the amount of storage requested but not used, expressed as a decimal number. This number indicates how much smaller the region size could be during compilation.

```
*STATISTICS*  NO DIAGNOSTICS THIS STEP

          or

*STATISTICS* nnnn DIAGNOSTICS THIS STEP, HIGHEST SEVERITY CODE IS n
```

Explanation: One of these messages is printed after the last compilation when there is more than one compilation in a job step. The first message is printed if there were no errors in any of the compilations. The second message is printed when errors were detected.

nnnn
is the cumulative number of error messages issued for all compilations.

n
is the highest completion code issued for any message.

Programmer Response: None

ERROR/WARNING MESSAGES

All error/warning messages produced are written in a group following the source module listing and object module name table. Figure 1 shows the format of each message as it is written in the data set specified by the SYSPRINT DD statement.

There are five types of error/warning messages:

- Unrecoverable error messages, which describe conditions that cause compilation to terminate prior to completion; condition code 16 returned.
- Severe error messages, which describe conditions that cannot be corrected by the compiler such that correct program execution can be guaranteed; condition code 12 returned. When a severe error is detected, processing of the erroneous source statement is discontinued. Therefore, subsequent coding errors in the same statement will not be found and diagnosed during the same compilation.
- Error messages, which describe conditions that can be corrected by the compiler with a high degree of confidence that the object program will be correct; condition code 8 returned.
- Warning messages, which describe possible program or machine errors; condition code 4 returned.
- Informational messages, which are notes to the programmer; condition code 0 returned.

xxxnnnI	ISN a LABEL b NAME c	yy(A)	message
xxx	is the 3-character program product identifier, IFE for the FORTRAN IV (H Extended) compiler, or IHO for the FORTRAN IV Library (Mod II)		
nnn	is a 3-digit message number		
a	is the internal statement number of either the statement in error or the statement following the last previous executable statement		
b	is a source label (statement number)		
c	is a variable name		
yy	is a number indicating the condition code of the error		
A	is one of the characters I, W, E, S, or U, indicating the level of the message. I indicates an informational message; W indicates a warning message; E indicates an error message; S indicates a severe error message; and U indicates an unrecoverable error message.		
message	is the message text		

Figure 1. Format of Error/Warning Messages

IFE000I IFEnnnI COMPILER ERROR. A MESSAGE DOES NOT EXIST FOR DETECTED ERROR.

Explanation: An error condition has occurred for which no message text has been developed. The number nnn indicates an error message number in the compiler message table having no corresponding text.

(Condition code - 8)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Review the source statements for possible errors and correct. If no errors are apparent, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified in the JOB statement.
- Have source and associated listing available.

IFE002I THE DO LOOPS ARE INCORRECTLY NESTED.

Explanation: A statement in the range of the inner DO is not also in the range of the outer DO.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Resubmit the job with all statements in the range of the inner DO also in the range of the outer DO. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that the extended range of a DO statement does not contain another DO statement that has an extended range if the second DO is within the same program unit as the first.
- Have source and associated listing available.

IFE003I THE EXPRESSION HAS AN INVALID LOGICAL OPERATOR.

Explanation: The compiler cannot recognize the logical operator.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check for keypunch errors and correct the invalid logical operator. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE004I AN END STATEMENT MAY NOT BE CONTINUED.

Explanation: The three letters of the END statement must be on the same card.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Correct the END statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE005I THE STATEMENT HAS AN INVALID USE OF PARENTHESES.

Explanation: A right or left parenthesis is not paired with its partner.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the parenthesis in question. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE006I THE STATEMENT HAS AN INVALID LABEL.

Explanation: A non-numeric character appears in the label field (columns 1-5).

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid character. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE007I THE EXPRESSION HAS AN INVALID DOUBLE DELIMITER.

Explanation: Double delimiters are used incorrectly, e.g., parentheses or quotation marks are not paired or are not paired properly. Double delimiters are the following:
( ) right/left parentheses
' ' quotation marks before and after a literal string
. . periods before and after a logical operator
/ / slashes before and after data values.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE008I THE EXPRESSION HAS A CONSTANT WHICH IS GREATER THAN THE ALLOWABLE MAGNITUDE.

Explanation: The allowable magnitude for an integer number is $2^{31}-1$; for a real number 16^{-65} through 16^{63}. The constant exceeds these limits.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the constant. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE009I THE EXPRESSION HAS A NON-NUMERIC CHARACTER IN A NUMERIC CONSTANT.

Explanation: A non-numeric character appears in a numeric constant.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the constant contains only numeric characters. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE010I THE EXPRESSION HAS A CONSTANT WITH AN INVALID EXPONENT.

Explanation: An unnecessary or invalid exponent is specified, or the base and exponent are not valid combinations. Valid combinations are the following:

Base	Exponent
Integer	Integer
Real	Real
Complex	Integer

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the invalid exponent. Make sure that the base and exponent are valid combinations of operand types. If the problem recurs, dc the following before calling IBM for programming support:

- Have source and associated listing available.

IFE011I THE ARITHMETIC OR LOGICAL EXPRESSION USES AN EXTERNAL FUNCTION NAME AS A VARIABLE NAME.

Explanation: An external function name is used as a variable name in the expression.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the expression so that the external function name is not employed as a variable. Resubmit the job using the MAP option to obtain indication of the use of each name. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE012I THE EXPRESSION HAS A COMPLEX CONSTANT WHICH IS NOT COMPOSED OF REAL CONSTANTS.

Explanation: Either the real or the imaginary part of a complex constant is not a valid real constant.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that both the real and imaginary parts of the complex number are valid real constants. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE013I AN INVALID CHARACTER IS USED AS A DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. The valid delimiters are the following:

(blank)
+
-
/
=
.
(
)
*
, (comma)
' (apostrophe)
&

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid character. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE014I THE STATEMENT HAS AN INVALID NON-INTEGER CONSTANT.

Explanation: An invalid or extraneous non-integer constant appears in the statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid non-integer constant. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE015I THE ARITHMETIC OR LOGICAL EXPRESSION USES A VARIABLE NAME AS AN EXTERNAL FUNCTION NAME.

Explanation: A variable name is also used as an external function name.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the variable is not employed as an external function. Resubmit the job using the MAP option to obtain indications of the use of each name. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE016I THE GO TO STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid character. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE017I THE ASSIGNED OR COMPUTED GO TO HAS AN INVALID ELEMENT IN ITS STATEMENT NUMBER LIST.

Explanation: The statement number list contains a non-executable statement label or, for assigned GO TO statements, the ASSIGN statement may be incorrect.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the statement number list contains executable statement numbers. If an assigned GO TO is in question, make sure that the ASSIGN statement is correct. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE018I THE STATEMENT HAS AN INVALID LABEL. LABEL IS IGNORED AND COMPILER PROCESSING OF THE STATEMENT CONTINUES.

Explanation: An invalid character (not a decimal digit) appears in the statement label.

| (Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid label. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE019I THE ASSIGNED GO TO HAS THE OPENING PARENTHESIS MISPLACED OR MISSING.

Explanation: The opening parenthesis of an assigned GO TO statement is missing or is invalid.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or insert parenthesis. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE020I THE ASSIGNED GO TO HAS AN INVALID DELIMITER FOLLOWING THE ASSIGNED VARIABLE.

Explanation: A comma must follow the assigned variable; the statement contains a delimiter other than a comma.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that a comma follows the assigned variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE021I THE COMPUTED GO TO HAS AN INVALID COMPUTED VARIABLE.

Explanation: The computed variable in a computed GO TO statement is either subscripted or is non-integer.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the referenced variable is integer and non-subscripted. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE022I THE VARIABLE IN THE ASSIGNED GO TO STATEMENT IS NOT INTEGRAL.

Explanation: A non-integral variable appears in an assigned GO TO statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the non-integral variable. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE023I THE DEFINE FILE STATEMENT HAS AN INVALID DATA SET REFERENCE NUMBER.

Explanation: The data set reference number in a DEFINE FILE statement is not an integer constant.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the data set reference number or size is an integer constant. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE024I THE DEFINE FILE STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE025I THE DEFINE FILE STATEMENT HAS AN INVALID INTEGER CONSTANT AS THE RECORD NUMBER OR SIZE.

Explanation: The record number or size specified in a DEFINE FILE statement is not a valid integer constant.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid integer. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE026I THE DEFINE FILE STATEMENT HAS AN INVALID FORMAT CONTROL CHARACTER.

Explanation: A format control character in the DEFINE FILE statement can be one of only the characters L, E, or U; the statement specifies a different character.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the format control character is either L, E, or U. If the problem recurs, do the following before calling IBM for programming support.
- Have source and associated listing available.

IFE027I THE ASSIGN STATEMENT HAS AN INVALID INTEGER VARIABLE.

Explanation: The statement contains a variable not an integer of length 4.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid integer variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE028I THE ASSIGN STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Determine the invalid delimiter and correct the statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE029I THE NUMBER OF CONTINUATION CARDS EXCEEDS 19. COMPILER PROCESSING OF THE STATEMENT CONTINUES.

Explanation: A statement contained more than nineteen continuation cards.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that the statement does not extend over more than nineteen continuation cards. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE030I THE DO STATEMENT HAS AN INVALID END OF RANGE STATEMENT NUMBER.

Explanation: The end of range statement in a DO statement is not an executable statement appearing after the DO statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the end of range statement number is an executable statement number appearing after the DO statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE031I THE DO STATEMENT OR IMPLIED DO HAS AN INVALID INITIAL VALUE.

Explanation: The initial value in a DO or implied DO statement is not an unsigned integer constant greater than zero or an unsigned nonsubscripted integer variable greater than zero.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check that the initial value is either an unsigned integer constant greater than zero, or an unsigned nonsubscripted integer variable greater than zero. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE032I A TYPE, GENERIC, EXTERNAL, DIMENSION, COMMON, EQUIVALENCE, OR NAMELIST STATEMENT APPEARS AFTER THE FIRST EXECUTABLE STATEMENT OF THE PROGRAM OR AFTER A STATEMENT FUNCTION DEFINITION.

Explanation: One of the non-executable statements was detected after the compiler processed an executable statement.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Insert the statement(s) before the first executable statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE033I THE STATEMENT MAY NOT APPEAR IN A BLOCK DATA SUBPROGRAM.

Explanation: The statement is invalid for a BLOCK DATA subprogram. BLOCK DATA subprograms may contain only DATA, COMMON, DIMENSION, EQUIVALENCE, and type statements; executable statements are not permitted. The invalid statement is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. If the subprogram is meant to be a BLOCK DATA subprogram remove the invalid statement. If the subprogram is meant to be executable, remove the BLOCK DATA statement.

IFE034I THE ASSIGNMENT STATEMENT BEGINS WITH A NON-VARIABLE.

Explanation: A non-variable appears as the first value in an assignment statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the statement. Resubmit the job with the MAP option to determine the nature of the non-variable in question, if possible. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE035I THE NUMBER OF CONTINUATION CARDS EXCEEDS 19. THE STATEMENT IS IGNORED.

Explanation: A statement contained more than nineteen continuation cards and the compiler cannot analyze the statement.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that the statement does not extend over more than nineteen continuation cards. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE036I THE STATEMENT CONTAINS INVALID SYNTAX. THE STATEMENT CANNOT BE CLASSIFIED.

Explanation: The compiler cannot analyze the statement because the keyword (e.g., DATA, FORMAT, READ) does not match a keyword in the compiler's keyword table.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE037I INVALID SYNTAX IN A GENERIC STATEMENT.

Explanation: An invalid character follows the word GENERIC.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the GENERIC statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE038I AN ASYNCHRONOUS I/O LIST ARRAY MAY NOT BE PADDED.

Explanation: A padded array appeared in an asynchronous I/O list.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the array so that padding does not occur. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE039I THE DEFINE FILE STATEMENT HAS AN INVALID ASSOCIATED VARIABLE.

Explanation: The associated variable is not a nonsubscripted integer variable.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check that the associated variable is nonsubscripted and integral. Make sure that the associated variable does not appear in the I/O list of a READ or WRITE for a data set associated with the DEFINE FILE statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE040I IT IS ILLEGAL TO HAVE A && STATEMENT NUMBER PARAMETER OUTSIDE A CALL STATEMENT.

Explanation: An && statement number may appear only in a CALL statement (e.g., CALL SUB(P,&20)); the compiler has detected an & statement number elsewhere.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete the & statement number. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE041I THE LIST DIRECTED I/O STATEMENT HAS A MISSING ASTERISK. PROCESSING CONTINUES.

Explanation: The required asterisk in a list-directed input/output statement is missing. The compiler assumes the presence of an asterisk.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Insert the missing asterisk. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE042I INTERNAL COMPILER TABLE FOR PARENTHESIS PROCESSING EXCEEDED.

Explanation: The number of parentheses in implied DO loops in an I/O statement exceeds 30.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Ensure that no single READ or WRITE statement contains more than 30 implied DO loops. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE046I THE EXPRESSION HAS AN INCORRECT PAIRING OF PARENTHESES. COMPILER INSERTS RIGHT PARENTHESES AT END OF STATEMENT TO PERMIT CONTINUED PROCESSING.

Explanation: Unbalanced parentheses occurred.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Add the correct number of right parentheses to give the statement the intended meaning. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE047I THE LITERAL HAS MORE THAN 255 CHARACTERS IN IT.

Explanation: A literal contains more than 255 characters, or a delimiter is missing, causing an excess to occur.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete the excessive characters, or make sure that the constant is properly delimited. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE048I MISSING END STATEMENT; END STATEMENT BEING PROVIDED.

Explanation: The source module contained no END statement.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Insert the missing statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE050I THE ARITHMETIC IF HAS INCORRECT BRANCH LABEL SYNTAX.

Explanation: The statement does not contain three executable statement numbers separated by commas.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that there are 3 executable statement numbers separated by commas. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE052I THE EXPRESSION HAS AN INCORRECT PAIRING OF PARENTHESES OR QUOTES.

Explanation: Unbalanced parentheses or apostrophes occurred in the expression.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check that there are as many left parentheses as there are right. If the statement is a FORMAT statement containing an H Format Code, check that w is large enough to accommodate the data and does not encompass the closing parenthesis. Make sure that there is an even number of apostrophes, and that any apostrophes within data are represented by two successive apostrophes. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE053I THE STATEMENT HAS A MISPLACED EQUAL SIGN.

Explanation: The statement contains an extraneous or invalid equal sign.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the equal sign indicated. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE056I THE FUNCTION STATEMENT MUST HAVE AT LEAST ONE ARGUMENT.

Explanation: No arguments appear in the FUNCTION statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the function statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE057I THE STATEMENT HAS A NON-VARIABLE SPECIFIED AS A SUBPROGRAM NAME.

Explanation: A subprogram name has been specified as other than a variable.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the non-variable subprogram name. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE058I THE SUBPROGRAM STATEMENT HAS AN INVALID ARGUMENT.

Explanation: An argument does not correspond in number, order, type, or length to the actual assignment to be included.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid argument. Use the MAP option to determine its nature, if necessary. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE059I THE FUNCTION STATEMENT HAS AN INVALID LENGTH SPECIFICATION.

Explanation: The length specification for the associated type is not permissible, or a type has not been specified, or DOUBLE PRECISION has been included.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the length specification is permissible for the associated type. Check that a type has been specified and that DOUBLE PRECISION has not been included. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE060I THE EXPRESSION HAS A STRUCTURED VARIABLE WITHOUT A SUBSCRIPT.

Explanation: A structured variable must have a subscript; the subscript is missing.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Include the subscript. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE062I THE EQUIVALENCE STATEMENT HAS AN ARRAY WITH AN INVALID NUMBER OF SUBSCRIPTS.

Explanation: An array has been specified that contains a number of elements not agreeing with the associated specification statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete the invalid subscripts or include those necessary for agreement with the associated specification statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE064I THE NAMELIST STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE065I THE NAMELIST STATEMENT HAS A NAMELIST NAME NOT BEGINNING WITH AN ALPHABETIC CHARACTER.

Explanation: The NAMELIST name begins with a non-alphabetic character.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid character. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE066I THE NAMELIST STATEMENT HAS A NON-UNIQUE NAMELIST NAME.

Explanation: A name which is not unique is used in a NAMELIST statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the NAMELIST name. Invoke the MAP option for indications of the use of each name, if necessary. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE067I THE NAMELIST STATEMENT HAS AN INVALID LIST ITEM.

Explanation: A list item in a NAMELIST statement is other than a variable or array name.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check that the list item is a variable or array name. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE069I THE COMMON STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE070I THE EQUIVALENCE STATEMENT HAS A MISSING OR MISPLACED DELIMITER.

Explanation: Unbalanced parentheses appear in the statement or a comma is missing.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that there are as many left parentheses as there are right parentheses. Check all commas. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE071I THE EQUIVALENCE STATEMENT DOES NOT SPECIFY AT LEAST TWO VARIABLES TO BE EQUIVALENCED.

Explanation: A delimiter is omitted or misplaced, or a variable is missing.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check delimiters and correct the invalid source. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE072I THE EQUIVALENCE STATEMENT HAS AN INVALID VARIABLE NAME.

Explanation: The variable name is incorrect. A dummy argument may not be specified.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the invalid variable name. Make sure that the variable in question is not a dummy argument. If necessary, invoke the MAP option for indications of the use of variable names. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE073I THE EQUIVALENCE STATEMENT HAS A SUBSCRIPT WHICH IS NOT AN INTEGER CONSTANT.

Explanation: A subscript has been specified which is not an integer constant.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid subscript. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE074I THE STATEMENT HAS A VARIABLE WITH MORE THAN SEVEN SUBSCRIPTS.

Explanation: More than seven subscripts have been specified; a comma may be misplaced causing the appearance of more than seven subscripts.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check that all commas are in correct position. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE075I THE COMMON STATEMENT REFERENCES A VARIABLE THAT HAS APPEARED PREVIOUSLY IN A COMMON STATEMENT.

Explanation: A variable has been specified on more than one COMMON statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the redundant entry. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE076I THE IMPLICIT STATEMENT IS NOT THE FIRST STATEMENT IN A MAIN PROGRAM OR THE SECOND STATEMENT IN A SUBPROGRAM.

Explanation: An IMPLICIT statement is misplaced in the sequence of program statements.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Place the IMPLICIT statement in correct sequence. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE077I THE IMPLICIT STATEMENT HAS A MISPLACED DELIMITER IN THE TYPE SPECIFICATION FIELD.

Explanation: A delimiter is misplaced, or a character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE078I THE IMPLICIT STATEMENT HAS AN INVALID TYPE.

Explanation: The statement specifies an invalid type or a valid type not having a corresponding valid length specification.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that for any type there is a corresponding valid standard or optional length specification. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE079I THE IMPLICIT STATEMENT HAS A MISSING LETTER SPECIFICATION.

Explanation: The character specifying one of the limits in a range is missing.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Insert the omitted specification. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE080I THE IMPLICIT STATEMENT HAS AN INVALID LETTER SPECIFICATION.

Explanation: A letter specification is not one of the valid characters A through Z and $.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid letter specification. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE081I THE IMPLICIT STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE082I THE IMPLICIT STATEMENT DOES NOT END WITH A RIGHT PARENTHESIS.

Explanation: The right parenthesis closing the statement is missing.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Insert the omitted parenthesis. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE083I THE IMPLICIT STATEMENT HAS A MISPLACED DELIMITER IN ITS PARAMETER FIELD.

Explanation: A delimiter is misplaced, or a character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the misplaced delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE084I THE IMPLICIT STATEMENT CONTAINS A LITERAL FIELD.

Explanation: The IMPLICIT statement contains invalid apostrophes or a wH specification.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that there are no apostrophes or wH specifications in the IMPLICIT statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE085I THE IMPLICIT STATEMENT HAS A RANGE OF LETTER SPECIFICATIONS NOT IN ASCENDING ALPHABETIC ORDER (A-Z). THE TYPING IS IGNORED.

Explanation: A range of letter specifications is not specified in ascending order.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the range of letters. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE086I THE COMMON STATEMENT SPECIFIES A NON-VARIABLE TO BE ENTERED.

Explanation: A non-variable or non-array name is specified for entry into a COMMON block, or a comma is misplaced causing invalid names.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that delimiters are in correct position. Check that only variable or array names are specified for entrance into a common block. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE087I THE COMMON STATEMENT SPECIFIES A NON-VARIABLE COMMON BLOCK NAME.

Explanation: The COMMON block name is not a variable.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid name. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE088I A DUMMY ARGUMENT IN A SUBPROGRAM STATEMENT MAY NOT BE IN COMMON.

Explanation: A dummy argument appears in both a subprogram statement and in COMMON.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that only variable or array names appear in the COMMON statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE090I THE EXTERNAL STATEMENT HAS A NON-VARIABLE DECLARED AS EXTERNAL.

Explanation: A non-variable appears in an EXTERNAL statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the non-variable name. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE091I THE EXTERNAL STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE092I THE TYPE STATEMENT MULTIPLY DEFINES THE VARIABLE.

Explanation: A variable is defined more than once in a type statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the variable in question. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE093I THE TYPE STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE094I THE TYPE STATEMENT HAS A NON-VARIABLE TO BE TYPED.

Explanation: A non-variable is specified to be typed.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the non-variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE095I THE TYPE STATEMENT HAS THE WRONG LENGTH FOR THE GIVEN TYPE.

Explanation: The length specified is not permissible for the associated type.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the specified length is permissible for the associated type. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE096I THE TYPE STATEMENT HAS A MISSING DELIMITER.

Explanation: The statement contains an invalid or extraneous delimiter, or is missing a necessary delimiter.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct, delete, or insert the delimiter in question. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE097I THE STRUCTURE STATEMENT HAS IMPROPER SYNTAX.

Explanation: The statement contains invalid delimiters or is missing necessary delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct, delete, or insert the delimiter(s). If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE098I THE STRUCTURE STATEMENT HAS A STRUCTURE WITH TOTAL DISPLACEMENTS WHICH ARE NOT EQUAL.

Explanation: The structure has unequal displacements.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Respecify the statement so that the displacements are made equal. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE099I THE STATEMENT SPECIFIES AN INVALID VARIABLE TO BE STRUCTURED.

Explanation: Other than a non-subscripted variable was specified.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE100I THE STRUCTURE STATEMENT HAS A STRUCTURE WITH IMPROPER WORD BOUNDARIES.

Explanation: The structure has improper alignment.

(Condition code - 12)

Programmer Response: Probable user error. Respecify the structure items in order of doubleword, fullword, and halfword items, or pad where necessary. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE101I THE DO STATEMENT OR IMPLIED DO HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE102I THE BACKSPACE/REWIND/END FILE STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE104I THE BACKSPACE/REWIND/END FILE STATEMENT HAS A DATA SET REFERENCE NUMBER THAT IS EITHER A NON-INTEGER OR AN ARRAY NAME.

Explanation: The data set reference number is either an integer not of length 4, or a signed integer constant.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the specified data set reference number is either an unsigned integer constant or an integer variable of length 4. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE109I THE PAUSE STATEMENT HAS A MISPLACED DELIMITER.

Explanation: A PAUSE stātement may contain only a literal constant not enclosed by a delimiter or enclosed only in apostrophes; a delimiter other than an apostrophe, or an apostrophe in an invalid position has been detected.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the PAUSE statement contains either no delimiter or a literal constant enclosed in apostrophes. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE110I THE PAUSE STATEMENT SPECIFIES A VALUE WHICH IS NEITHER A LITERAL NOR AN INTEGER CONSTANT.

Explanation: A PAUSE statement may specify only a literal or an integer constant.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the value. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE111I THE PAUSE STATEMENT HAS MORE THAN 255 CHARACTERS IN ITS LITERAL FIELD.

Explanation: The PAUSE statement specifies more than 255 characters. Continuation cards may be incorrectly indicated resulting in an overflow.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid characters. Make sure that continuation cards are correctly indicated. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE115I THE VARIABLE RETURN STATEMENT HAS NEITHER AN INTEGER CONSTANT NOR VARIABLE FOLLOWING THE KEYWORD.

Explanation: An integer constant is incorrect or is missing in a variable RETURN statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid constant or variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE116I THE DO STATEMENT OR IMPLIED DO HAS AN INVALID PARAMETER.

Explanation: The DO variable is a subscripted variable, or the initial value, test value, or increment value are not unsigned integer constants, or the test value exceeds the allowable magnitude.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the DO variable is a nonsubscripted integer variable. Check that the initial value, test value, and increment (if specified) are either unsigned integer constants greater than zero, or unsigned nonsubscripted integer variables whose value is greater than zero. Verify that the test value does not exceed the allowable magnitude. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE117I THE BLOCK DATA STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE118I THE VALUE OF A DO PARAMETER IS REDEFINED WITHIN THE RANGE OF THE DO BY THIS STATEMENT.

Explanation: Either the DO variable, initial value, test value, or increment of the DO statement is changed within the loop.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that neither the DO variable, initial value, test value, or increment of the DO statement is changed within the loop. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE119I THE VALUE OF AN IMPLIED DO PARAMETER IS REDEFINED WITHIN THE RANGE OF THE IMPLIED DO.

Explanation: Either the DO variable, initial value, test value or increment of an implied DO in a READ or WRITE statement is changed within the implied loop.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that neither the DO variable, initial value, test value, or increment of an implied DO is changed by another implied DO within the READ or WRITE statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE120I THE BLOCK DATA STATEMENT WAS NOT THE FIRST STATEMENT OF THE SUBPROGRAM.

Explanation: The BLOCK DATA statement is out of sequence.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Insert the statement at the beginning of the subprogram. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE121I THE DATA STATEMENT HAS A VARIABLE WHICH HAS A NON-ALPHABETIC FIRST CHARACTER.

Explanation: The first character in a variable name is not alphabetic. A misplaced delimiter may cause the error condition.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid variable, or correct any erroneous delimiters. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE122I THE DATA STATEMENT CONTAINS A SUBSCRIPTED VARIABLE WHICH HAS NOT BEEN DEFINED AS AN ARRAY.

Explanation: A subscripted variable has not been dimensioned.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Either define the variable as an array, or correct the variable in the DATA statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE123I THE DATA STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE124I THE DATA STATEMENT HAS A VARIABLE WITH AN INVALID INTEGER SUBSCRIPT.

Explanation: The subscript quantity for a variable in a DATA statement contains other than integer constants separated by commas.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the subscript quantity contains only integer constants separated by commas. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE125I THE DATA STATEMENT HAS A VARIABLE WITH A SUBSCRIPT THAT CONTAINS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE129I THE STATEMENT CONTAINS AN INVALID DATA CONSTANT.

Explanation: The data constant is not valid for its designated class or type or the first slash is immediately followed by a comma.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the constant is valid for its designated class and/or type or place comma correctly. Also make sure that any parentheses are not placed so that a complex data value is unintentionally indicated. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE132I THE DATA STATEMENT HAS AN INVALID DELIMITER IN ITS INITIALIZATION VALUES.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE133I THE DO STATEMENT CANNOT FOLLOW A LOGICAL IF STATEMENT.

Explanation: A DO statement follows a logical IF statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Change the DO to a GO TO n, where n is the statement label of the DO located elsewhere in the program unit. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE134I THE DO STATEMENT HAS AN INVALID INTEGER DO-VARIABLE.

Explanation: The DO-variable is not a non-subscripted integer variable.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the integer DO variable is a nonsubscripted integer variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE135I THE DO STATEMENT OR IMPLIED DO HAS AN INVALID TEST VALUE.

Explanation: The test value is not an unsigned integer constant or variable greater than zero, or the value exceeds $2^{31}-1$.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the test value is either an unsigned integer constant greater than zero, or an unsigned nonsubscripted integer variable greater than zero. Verify that the test value does not exceed $2^{31}-1$. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE136I THE NUMBER OF NESTED DO'S EXCEEDS THE COMPILER LIMIT.

Explanation: The maximum level of nesting is exceeded.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the maximum level of nesting for DO loops is not exceeded. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE137I THE DO STATEMENT OR IMPLIED DO HAS AN INVALID INCREMENT VALUE.

Explanation: The increment value is not an unsigned integer constant or variable greater than zero.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the increment value is either an unsigned integer constant greater than zero, or an unsigned nonsubscripted integer variable greater than zero. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE138I THE DO STATEMENT HAS A PREVIOUSLY DEFINED STATEMENT NUMBER SPECIFIED TO END THE DO RANGE.

Explanation: The statement number ending the DO range does not specify an executable statement appearing after the DO statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify that the statement number specified to end the DO range is an executable statement number appearing after the DO statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE139I A LOGICAL IF IS FOLLOWED BY ANOTHER LOGICAL IF OR A SPECIFICATION STATEMENT.

Explanation: Another logical IF statement, or a specification statement follows a logical IF statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Change the latter logical IF to a GO TO n, where n is the statement label of the logical IF located elsewhere in the program unit. Include the specification statement in the prescribed order prior to executable statements and statement function definitions, if any. If the problem recurs, do the following before calling IBM for programming support.
- Have source and associated listing available.

IFE140I THE IF STATEMENT BEGINS WITH AN INVALID CHARACTER.

Explanation: An incorrect or extraneous character appears in the word IF.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the misspelling. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE143I THE STATEMENT FUNCTION HAS AN ARGUMENT WHICH IS NOT A VARIABLE.

Explanation: An argument has been detected which is not a non-subscripted variable.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the arguments are nonsubscripted variables. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE144I THE STATEMENT FUNCTION HAS MORE THAN 20 ARGUMENTS.

Explanation: The maximum number of arguments that may be specified is 20; the argument list exceeds this limit.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Redefine the arguments so that the 20 argument limit is not exceeded. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE145I THE STATEMENT FUNCTION HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE146I THE STATEMENT FUNCTION HAS A MISPLACED EQUAL SIGN.

Explanation: An equal sign in the statement is extraneous or misplaced.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the misplaced equal sign. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE147I A STATEMENT FUNCTION DEFINITION MUST PRECEDE THE FIRST EXECUTABLE STATEMENT.

Explanation: A statement function definition does not follow a SUBPROGRAM, IMPLICIT, or other specification statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Resequence the definition so that it follows only SUBPROGRAM, IMPLICIT, or other specification statements. Include a DIMENSION statement if it has been omitted. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE148I THE DIMENSIONED ITEM HAS A NON-INTEGER SUBSCRIPT.

Explanation: The subscript value is not integer.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify that the subscript quantity is validly constructed. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE149I A VARIABLE TO BE DIMENSIONED USING ADJUSTABLE DIMENSIONS MUST HAVE BEEN PASSED AS AN ARGUMENT AND MUST NOT APPEAR IN COMMON.

Explanation: A variable to be dimensioned using adjustable dimensions either appears in COMMON or is not passed as an argument.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Remove the variable name from the COMMON statement, if it has been entered in a COMMON statement. Place the variable in the argument list, if not already there. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE150I THE DIMENSIONED ITEM HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE151I THE STATEMENT SPECIFIES A NON-VARIABLE TO BE DIMENSIONED.

Explanation: The statement specifies other than a variable to be dimensioned.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the non-variable name. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE152I THE SUBPROGRAM STATEMENT HAS AN INVALID DELIMITER IN THE ARGUMENT LIST.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Rəsponse: Probable user error. Correct or delete the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE153I THE STATEMENT HAS AN INVALID NAME SPECIFIED AS A FUNCTION REFERENCE.

Explanation: Either the function has been defined incorrectly or the type of the name used for the reference does not agree with the type of the name used in the definition.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the function has been correctly defined. Verify that the type of the name used for the reference agrees with the type of the name used in the definition. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE154I THE ASYNCRONOUS I/O STATEMENT HAS AN INVALID PARAMETER.

Explanation: The syntax of the parameter is incorrect.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify syntax of parameters. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE155I THE ASYNCHRONOUS I/O STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify punctuation in statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE156I THF I/O STATEMENT HAS AN INVALID NAME PRECEDING THE EQUAL SIGN.

Explanation: An invalid name precedes the equal sign.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid name. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE157I THE I/O STATEMENT HAS A NON-VARIABLE SPECIFIED AS A LIST ITEM.

Explanation: The I/O list specifies other than a variable name.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the I/O list contains variable names, subscripted or unsubscripted array names, or array names accompanied by indexing specifications in the form of an implied DO. Verify that no function references or arithmetic expressions appear in the I/O list. Use the MAP option to determine the use of names in the program unit. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE158I THE I/O STATEMENT HAS AN IMPROPER PAIRING OF PARENTHESES IN AN IMPLIED DO, OR A NON-INTEGRAL INDEX.

Explanation: Parentheses are unbalanced, or an index value (the initial value, the test value, or the increment) in the implied DO is not an integer.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that there are as many left parentheses as there are right parentheses. Correct any invalid index. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE160I THE I/O STATEMENT HAS AN INVALID DELIMITER IN THE PARAMETERS.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter, or a misplaced comma in the alternate form of a list-directed I/O statement may cause incorrect code to be generated. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid delimiter. For the alternate form of the list-directed I/O statement, ensure that a comma appears between the asterisk and the I/O list and that no trailing comma is coded. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE161I THE I/O STATEMENT HAS A DUPLICATE PARAMETER.

Explanation: A parameter has been specified more than once.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete the duplicate parameter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE162I THE ASYNCHRONOUS I/O STATEMENT HAS AN INCORRECTLY SPECIFIED LIST.

Explanation: The I/O list does not consist of references to a single array, or its elements are not in proper syntactical form.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify that the list consists of references to a single array or its elements in proper syntactical form. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE163I THE I/O STATEMENT HAS AN ARRAY WHICH IS NOT DIMENSIONED.

Explanation: A subscripted array name or an array name in the form of an implied DO has not been previously declared in a DIMENSION statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the subscripted array name or array name in the form of an implied DO has been previously declared in a DIMENSION statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE164I THE I/O STATEMENT HAS AN ARITHMETIC EXPRESSION OR A FUNCTION NAME SPECIFIED AS A LIST ITEM.

Explanation: A function reference or an arithmetic expression appears in an I/O list.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify that no function references or arithmetic expressions are contained in the I/O list. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE165I THE I/O STATEMENT HAS A PARAMETER WHICH IS NOT AN ARRAY AND NOT A NAMELIST NAME.

Explanation: A parameter is not either an array or a NAMELIST name.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid parameter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE166I THE I/O STATEMENT HAS A NON-INTEGER CONSTANT OR VARIABLE REPRESENTING THE DATA SET REFERENCE NUMBER.

Explanation: The data set reference number is not either an integer constant or an integer variable of length 4.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the data set reference number is either an unsigned integer constant or an integer variable of length 4. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE167I THE STATEMENT HAS AN INVALID USE OF A FUNCTION/SUBROUTINE NAME.

Explanation: A statement function name has not been previously defined, there is an invalid reference to it, or a subroutine has the same name as a labeled COMMON block.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify that the statement function name has been previously defined and that a COMMON block name does not duplicate a subroutine name. Make sure that a statement function name is not equivalenced. Correct any invalid references or duplicate names. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE168I THE STATEMENT SPECIFIES AS A SUBPROGRAM NAME A VARIABLE WHICH HAS BEEN PREVIOUSLY USED AS A NON-SUBPROGRAM NAME.

Explanation: A subprogram name has been specified which duplicates a variable name or COMMON block name.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. If the desired subprogram name duplicates a variable name, change the variable name and all references to it. If it duplicates a COMMON block label, change either the subroutine or COMMON name and all references to the changed name. Use the XREF option to determine where the variable occurs if the program unit contains many statements. If the subroutine name is in error, correct it. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the XREF option is specified in the PARM field of the EXEC statement, and that the necessary DD statement is included.

IFE169I THE DIRECT ACCESS I/O STATEMENT MAY NOT SPECIFY A NAMELIST NAME.

Explanation: The data set reference number in a direct access input/output statement refers to a sequential device incapable of handling direct access records, such as a magnetic tape or printer.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Change the data set reference number so that it refers to a valid direct access device. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE170I THE DIRECT ACCESS I/O STATEMENT HAS A NON-INTEGER SPECIFYING THE RECORD'S RELATIVE POSITION.

Explanation: Other than an integer is used to specify the relative record position.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid non-integer reference. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE171I THE NAME SPECIFIED FOR AN ENTRY POINT HAS ALREADY BEEN USED AS EITHER A VARIABLE, SUBROUTINE, OR FUNCTION NAME.

Explanation: An entry point name duplicates a variable name, a subroutine name, or a function name.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Change the entry point name and all references to it so that duplication is eliminated. Determine if the name was erroneously used previously and correct it. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the XREF option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.

IFE172I THE ASYNCHRONOUS I/O STATEMENT DOES NOT HAVE AN ID PARAMETER.

Explanation: The ID parameter is missing or is incorrectly punctuated.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure there is a correctly punctuated ID parameter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE173I COMPILER ERROR WHILE GENERATING SUBSCRIPTS FOR THE ASYNCHRONOUS I/O LIST.

Explanation: Invalid subscripts were specified for an array causing the compiler to generate invalid addresses.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Probable user error. Verify that subscripts are specified properly. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the XREF option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.

IFE176I THE I/O STATEMENT CONTAINS INVALID SYNTAX IN ITS IMPLIED DO.

Explanation: The indexing parameters for ar implied DO are specified incorrectly, or there are more than 20 implied DO's in the statement, or a character other than one of the 12 allowable characters is used as a delimiter.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that indexing parameters are correctly specified. Verify that there are no more than 20 implied DO's per statement. Correct any erroneous delimiters. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE177I THE FORMAT SPECIFICATION IN THE I/O STATEMENT IS INVALID.

Explanation: The statement specifies the format specification incorrectly, or specifies more than one format specification.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Make sure that one and only one format specification is used in the I/O statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE192I THE STATEMENT HAS A LABEL WHICH IS SPECIFIED AS BOTH THE LABEL OF A FORMAT STATEMENT AND THE OBJECT OF A BRANCH.

Explanation: A format label duplicates the label of an executable statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. If the branch has been specified erroneously to a FORMAT statement, correct it. Correct or delete any misplaced labels. Use the XREF option for listings of the internal statement number of the statements in which the label is defined and referenced. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the XREF option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.

IFE193I THE STATEMENT NUMBER HAS BEEN PREVIOUSLY DEFINED.

Explanation: A statement number duplicates a previously used number.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Change the statement number. Use the XREF option where many labels occur to determine which are unused. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the XREF option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.

IFE194I THE TYPE STATEMENT HAS A MISSING DELIMITER IN THE INITIALIZATION VALUES.

Explanation: An initialization value is missing a delimiter or a character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct any invalid delimiters. Supply the missing delimiter. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE197I THE STOP STATEMENT HAS A NON-INTEGER CONSTANT AFTER THE KEYWORD.

Explanation: The STOP statement specifies other than one to five decimal digits.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the constant following the keyword is a string of 1 to 5 decimal digits, inclusive. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE199I THE SUBROUTINE OR FUNCTION STATEMENT WAS NOT THE FIRST STATEMENT.

Explanation: A statement, other than a comment, appears before the SUBROUTINE or FUNCTION statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that no statements except comments occur prior to the SUBROUTINE or FUNCTION statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE200I QUOTE LITERALS MAY APPEAR ONLY IN CALL, DATA, AND FORMAT STATEMENTS.

Explanation: A literal enclosed in quotation marks occurs in other than a CALL, DATA, or FORMAT statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete the invalid reference to the quote literals. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE201I DO VARIABLE CONFLICT. THE SAME DO VARIABLE IS USED IN AN OUTER LOOP.

Explanation: Either a DO variable appears in more than one loop of a nest of DOs or an implied DO within a nest of DOs uses the same DO variable as an outer loop of the nest.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that the DO variable does not appear in another loop of the nest of DOs or that the implied DO variable (in the READ or WRITE statement) is not repeated in an outer loop of the nest in which it appears. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE202I THE STATEMENT HAS A VARIABLE WHICH HAS BEEN PREVIOUSLY DIMENSIONED. THE INITIAL DIMENSION FACTORS ARE USED.

Explanation: An unnecessary or erroneous dimension specification has occurred for a variable previously dimensioned.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Delete the unnecessary or erroneous dimension specification. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE203I AN ENTRY STATEMENT MUST NOT APPEAR IN A MAIN PROGRAM. THE STATEMENT IS IGNORED.

Explanation: An ENTRY statement appears in a main program.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Delete the ENTRY statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE204I THE STOP STATEMENT HAS AN INVALID DELIMITER.

Explanation: A character other than one of the 12 allowable characters is used as a delimiter. See the explanation of message IFE013 for a list of valid delimiters.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE205I THE ASSIGNED OR COMPUTED GO TO HAS AN INVALID ELEMENT FOLLOWING THE CLOSING PARENTHESIS.

Explanation: In an assigned GO TO statement, an element appears after the closing parenthesis; in a computed GO TO statement, an element other than a non-subscripted integer variable appears after the closing parenthesis.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct or delete the invalid element. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE206I THE STATEMENT HAS A NON-SUBSCRIPTED ARRAY ITEM.

Explanation: An array element not identified by a subscript appears in the statement. The first element of the array is substituted.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid array reference. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE207I THE CONTINUE STATEMENT DOES NOT END AFTER THE KEY WORD CONTINUE.

Explanation: Additional coding appears after the word CONTINUE, or a continuation has been inadvertently indicated on the next statement. The coding or the continuation is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Delete any code following the CONTINUE keyword. Make sure that a continuation has not been indicated on the statement immediately following the CONTINUE. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE208I THE CONTINUE STATEMENT DOES NOT HAVE A STATEMENT NUMBER.

Explanation: A label is missing from the CONTINUE statement.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Delete the CONTINUE statement if no related diagnostic appears with respect to undefined statement numbers. If there is an undefined statement number related to a label omission on the CONTINUE, correct the statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE209I THE STATEMENT HAS AN OCTAL CONSTANT SPECIFIED AS AN INITIAL VALUE. THE VALUE IS REPLACED BY ZERO.

Explanation: An actual constant has been specified, probably through the substitution of the letter O for the numeric 0 as the character of the value. The value is replaced by zero.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. If the value of the octal constant is necessary, convert it to the appropriate hexadecimal equivalent. Verify that a leading "O" has not been inadvertently specified for a leading "0" in an initialization statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE211I THE STATEMENT HAS A COMPLEX CONSTANT WHOSE REAL CONSTANTS DIFFER IN LENGTH.

Explanation: Both parts of a complex constant are not REAL*4 or REAL*8.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the constant so that both parts are either REAL*4 or REAL*8. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

| IFE224I THE STATEMENT AFTER AN ARITHMETIC IF, GO TO, STOP OR RETURN HAS NO LABEL.

Explanation: A required label is missing.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Insert any necessary labels. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE225I A LABEL APPEARS ON A NON-EXECUTABLE STATEMENT. THE LABEL IS IGNORED.

Explanation: An unnecessary label (statement number) appears on a non-executable statement.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Delete the label. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE226I THE STATEMENT HAS A VARIABLE WITH MORE THAN SIX CHARACTERS. THE RIGHTMOST CHARACTERS ARE TRUNCATED.

Explanation: A variable name having more than six characters has been specified.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Delete extraneous characters, or insert any missing delimiter. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE229I ALL THE ARGUMENTS OF AN ARITHMETIC STATEMENT FUNCTION ARE NOT USED IN THE DEFINITION.

Explanation: The arithmetic statement function contains extraneous arguments, or some arguments are omitted in the definition of the function.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. If the definition is correct, then delete extraneous arguments. If arguments were omitted in the definition, then include them. Verify that the expression on the right contains as many distinct variables as there are distinct dummy arguments. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE251I THE FORMAT STATEMENT SPECIFIES A FIELD WIDTH OF ZERO.

Explanation: The compiler expected to encounter a non-zero field width. The field is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Specify a field width for the FORMAT statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE252I THE FORMAT STATEMENT CONTAINS AN INVALID CHARACTER.

Explanation: A character used in an invalid context was specified in the FORMAT statement. The character is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Review and correct the FORMAT Statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE253I THE FORMAT STATEMENT HAS UNBALANCED PARENTHESES.

Explanation: A left or right parenthesis is missing its opposite partner. The condition is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the statement. Delete the unnecessary parentheses or insert the missing parentheses. Make sure wH specifications are correct. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE254I THE FORMAT STATEMENT HAS NO BEGINNING LEFT PARENTHESIS.

Explanation: The left parenthesis after the word FORMAT is missing. The condition is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the source statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE255I THE FORMAT STATEMENT SPECIFIES A COUNT OF ZERO FOR A LITERAL FIELD.

Explanation: The compiler detected adjacent quotes or a specification of 0X. The field is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the incorrectly specified count. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE256I THE FORMAT STATEMENT CONTAINS A MEANINGLESS NUMBER.

Explanation: The compiler found a number without a conversion letter. The number is ignored.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct or delete the invalid number. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE257I THE FORMAT STATEMENT HAS A MISSING DELIMITER.

Explanation: A second conversion letter has been encountered before a previous one has been delimited. The previous conversion is deleted.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Either correct or delete invalid delimiters, or insert the missing delimiters. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE258I THE FORMAT STATEMENT CONTAINS A NUMERIC SPECIFICATION GREATER THAN 255.

Explanation: A number greater than the maximum allowable was encountered. The compiler uses the number determined by n modulo 256.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid numeric specification. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE259I THE FORMAT STATEMENT CONTAINS A T WITH A NUMBER GREATER THAN 65,535.

Explanation: A number greater than the maximum allowable for T format code, 65,535, was encountered. The compiler used the number determined by n modulo 256.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid specification. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE260I THE FORMAT STATEMENT CONTAINS GROUP FORMAT SPECIFICATIONS NESTED TO A LEVEL GREATER THAN TWO.

Explanation: The compiler detected more than two left parentheses without intervening right parentheses, thereby indicating a nesting level exceeding two. The nesting is encoded as written.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid specification. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE261I THE FORMAT STATEMENT DOES NOT END WITH A RIGHT PARENTHESIS.

Explanation: The right parenthesis is missing or an extraneous delimiter is present.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct any invalid delimiters and include the right parenthesis. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE262I THE FORMAT STATEMENT DOES NOT HAVE A STATEMENT NUMBER.

Explanation: The statement number in the FORMAT statement is missing.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Insert the required statement number. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE263I THE FORMAT STATEMENT CONTAINS AN ILLEGAL DECIMAL SPECIFICATION.

Explanation: A decimal point was encountered, but the associated format code was not D, E, F, G, or Q.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid source. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE264I THE FORMAT STATEMENT IS MISSING A REQUIRED DECIMAL SPECIFICATION.

Explanation: No decimal point was encountered, but the format code was D, E, F, or Q.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the invalid source. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE280I UNEXPECTED END OF DATA SET ENCOUNTERED ON SYSUT2. XREF OPTION IGNORED.

Explanation: The end of the data set was encountered before a count of the blccks written could be satisfied.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Retry the program. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that DUMP is requested as a compiler option and that a SYSABEND DD card is supplied.
- Have source and associated listing available.

IFE281I XREF BUFFFR EXHAUSTED. XREF OPTION IGNORED.

Explanation: Due to the size of the program, the cross reference processing buffers were exhausted.

(Condition code - 4)

System Action: Cross reference processing is terminated. Job processing continues.

Programmer Response: Rerun the program without the XREF compiler option or reduce the size of the program. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE301I SOURCE PROGRAM TOO LARGE, TOO MANY COMPUTED GO TO DESTINATIONS.

Explanation: Because of too many computed GO TO statements, the compiler generated more object program branches than could be accommodated by the branch register (usually register 13).

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Subdivide the program. Move FORMAT statements to small subroutines. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE302I THE EQUIVALENCE STATEMENT HAS EXTENDED COMMON BACKWARDS.

Explanation: Equivalenced array elements have been added before the beginning of an established COMMON block. The array name that extends COMMON backwards appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that any implicit equivalencing or assignment statements involving arrays do not create an extension such that elements are added before the beginning of an established COMMON block. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE303I THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WHICH IS NOT DIMENSIONED.

Explanation: A subscripted variable that has not been previously defined as an array has been detected. The array name appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Include the necessary subscript quantity for the array name. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE304I THE EQUIVALENCE STATEMENT HAS LINKED BLOCKS OF COMMON TOGETHER.

Explanation: Equivalencing of variable names in different COMMON blocks has occurred. The name of the equivalencing variable appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that implicit equivalencing does not link COMMON blocks together. Use the MAP option to determine locations of variable names in the COMMON blocks in question. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the MAP option is specified in the PARM field of the EXEC statement.

IFE305I THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE.

Explanation: A subscript which is beyond the range of the array was specified. The name of the array appears before the message text.

(Condition code - 8)

System Action: No data is entered into the array element specifying the invalid subscript. Job processing continues.

Programmer Response: Probable user error. Correct the invalid subscript. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the 'XREF' option is specified in the PARM parameter of the EXEC statement and that the associated DD statement is included in the job stream.

IFE306I THE EQUIVALENCE STATEMENT HAS AN INCONSISTENCY.

Explanation: A variable being equivalenced contradicts itself or a previously established equivalencing. The variable name appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the EQUIVALENCE statement does not contradict itself or any previously established equivalences. Verify that implicit equivalencing, if it occurs, does not create inconsistencies. Make sure that two elements of the same array are not equivalenced to one another. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE307I THE DATA STATEMENT CONTAINS A VARIABLE THAT IS NOT REFERENCED.

Explanation: A variable appearing in a DATA statement has not been used later in an executable statement.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that the indicated variable has not inadvertently been omitted from a program unit. If not, delete the variable from the DATA statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE308I THE EQUIVALENCE STATEMENT HAS EQUIVALENCED TWO VARIABLES IN THE SAME COMMON BLOCK.

Explanation: A variable in a COMMON block has been equivalenced to another variable in the same COMMON block. The variable name appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that source is correct. If necessary and if possible, replace one of the invalid variables with a variable not in COMMON. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the MAP option is specified in the PARM field of the EXEC statement.

IFE309I THE EQUIVALENCE STATEMENT CONTAINS A NON-SUBSCRIPTED ARRAY ITEM. INCORRECT ADCONS MAY BE GENERATED.

Explanation: A non-subscripted array item appears in an EQUIVALENCE statement.

(Condition code - 8)

System Action: Job processing continues.

Programmr Response: Probable user error. Ensure that array item is dimensioned prior to being equivalenced. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE310I THE EQUIVALENCE STATEMENT HAS A VARIABLE WITH A VARIABLE DIMENSION.

Explanation: A variable has been detected which has a subscript quantity containing other than one through seven unsigned integer constants separated by commas.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the subscript quantity. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

FE311I THE EQUIVALENCE STATEMENT HAS AN ARRAY WITH AN INVALID NUMBER OF SUBSCRIPTS.

Explanation: There is a discrepancy between the subscript information in the equivalence statement and the dimension information specified for the array.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Ensure that all subscript and dimension information for the array is consistent. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE312I THE EQUIVALENCE STATEMENT CONTAINS AN EXTERNAL REFERENCE OR A DUMMY ARGUMENT.

Explanation: An EXTERNAL reference name or a dummy argument appears in an EQUIVALENCE group. The invalid reference name appears before the message text.
(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct the invalid externally referenced name or dummy argument in the EQUIVALENCE group. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE314I THE EQUIVALENCE STATEMENT MAY CAUSE INEFFICIENT ALIGNMENT.

Explanation: The displacement of a variable in an EQUIVALENCE group cannot be evenly divided by the reference number associated with the variable. Inefficient alignment may result. The variable name appears before the message text.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Arrange variables in fixed descending order according to length, or force proper alignment with dummy variables. Construct the group so that the displacement of each variable in the group can be evenly divided by the reference number associated with the variable. Use the MAP option for information on variables and relative addresses. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the MAP option is specified in the PARM field of the EXEC statement.

IFE315I THE EQUIVALENCE STATEMENT WILL CAUSE INEFFICIENT ALIGNMENT.

Explanation: The displacement of a variable in an EQUIVALENCE group cannot be evenly divided by the reference number associated with the variable. Inefficient alignment will result. The variable name appears before message text.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Arrange variables in fixed descending order according to length or force proper alignment with dummy variables. Construct the group so that the displacement of each variable in the group can be evenly divided by the reference number associated with the variable. Use the MAP option for information on variables and relative addresses. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE316I VARIABLES AND ARRAY ELEMENTS MAY NOT BE ASSIGNED INITIAL VALUES IN BLANK (UNNAMED) COMMON

Explanation: One or more variables or array elements in unnamed common are initialized in a BLOCK DATA subprogram by either an explicit specification statement or a DATA statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of the compilation.

Programmer Response: Probable user error. Name the common used when initializing data in a BLOCK DATA subprogram or initialize the variables via assignment statements in an executable main routine or subroutine. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE317I THE BLOCK DATA PROGRAM DOES NOT CONTAIN A COMMON STATEMENT.

Explanation: A BLOCK DATA subprogram does not have a COMMON statement listing all elements of a COMMON block in any main program or subprogram.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that all elements of a COMMON block in any main program or subprogram are listed in a COMMON statement in the BLOCK DATA subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE318I THE DATA STATEMENT IS USED TO ENTER DATA INTO COMMON OUTSIDE A BLOCK DATA SUBPROGRAM.

Explanation: Data is entered into a COMMON variable not in the BLOCK DATA subprogram.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete the invalid reference to the variable in COMMON. Delete the invalid reference or include the reference in a BLOCK DATA subprogram or in an assignment statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE319I DATA IS ENTERED INTO A LOCAL VARIABLE IN A BLOCK DATA PROGRAM.

Explanation: In a BLOCK DATA subprogram, DATA must be entered into a variable appearing in COMMON.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that the variable appears in COMMON. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE320I DATA MAY NOT BE ENTERED INTO A VARIABLE WHICH HAS BEEN PASSED AS AN ARGUMENT.

Explanation: A dummy argument appears in a DATA initialization list. The name of the dummy argument appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete any dummy arguments that appear in the data initialization list. Make sure source is correct. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE321I THE DATA INITIALIZATION VALUE TYPE DIFFERS FROM THE VARIABLE TYPE. THE VALUE IS CONVERTED TO THE TYPE OF THE VARIABLE.

Explanation: An initialization value in a DATA statement was detected that is of a different mode from that specified for the variable to which it is assigned. Examples of modes are real (floating point), complex, integer, logical and literal. The value is converted to the mode of the variable.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Ensure that each variable in a DATA statement is initialized to a value with the same mode as that specified for the variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE322I THE COMMON STATEMENT MAY CAUSE INEFFICIENT ALIGNMENT.

Explanation: The displacement of a variable in COMMON cannot be evenly divided by the reference number associated with the variable. Inefficient alignment may result. The variable name appears before the message text.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Arrange variables in fixed descending order according to length, or force proper alignment with dummy variables. Construct the block so that the displacement of each variable can be evenly divided by the reference number associated with the variable. Use the MAP option for information on the relative address of each variable in the block. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the MAP option is specified in the PARM field of the EXEC statement.

IFE323I THE COMMON STATEMENT WILL CAUSE AN INEFFICIENT ALIGNMENT.

Explanation: The displacement of a variable in COMMON cannot be evenly divided by the reference number associated with the variable. Inefficient alignment will result. The variable name appears before the message text.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Arrange variables in fixed descending order according to length, or force proper alignment with dummy variables. Construct the block so that the displacement of each variable can be evenly divided by the reference number associated with the variable. Use the MAP option for information on the relative address of each variable in the block. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that the MAP option is specified in the PARM field of the EXEC statement.

|IFE332I THE STATEMENT IS UNDEFINED. OPTIMIZATION IS DOWNGRADED.

Explanation: The statement does not reference a valid statement number. The undefined statement number appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the source so that a valid statement number is referenced. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE334I THE COMMON STATEMENT HAS A VARIABLE WITH A VARIABLE DIMENSION.

Explanation: A subscript quantity contains a value other than 1 through 7 unsigned integer constants separated by commas. The variable name appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that a subscript quantity contains only 1 through 7 unsigned integer constants separated by commas. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE350I THE DATA STATEMENT HAS A MISSING PARENTHESIS.

Explanation: The DATA statement contains invalid delimiters or unbalanced parentheses.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct any invalid delimiters and insert the appropriate parenthesis. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE351I THE DATA INITIALIZATION VALUE IS LONGER THAN THE VARIABLE OR ARRAY ELEMENT - TRUNCATION OR SPILL WILL OCCUR.

Explanation: An array or variable was initialized with a constant whose length was greater than the length of an array element. If the constant was specified as the first element in a non-subscripted array, part of the constant will spill over into the succeeding array element(s). If the constant was specified as other than the first element in a non-subscripted array, or if it was specified as any element in a subscripted array, the constant will be truncated.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. If spill is not desired, make sure that the length of a constant specified does not exceed the length of the element. If truncation is not desired, make sure that the length of any constant specified as a subsequent element in the array does not exceed the element length. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE352I THE DATA STATEMENT HAS TOO MANY OR TOO FEW INITIALIZATION VALUES.

Explanation: A one-to-one correspondence does not exist between the total number of constants and the number of elements in the data list.

(Condition code - 8)

System Action: The excess constants are ignored and job processing continues.

Programmer Response: Probable user error. Make sure that a one-to-one correspondence exists between the total number of elements specified or implied in the data list and the total number of constants specified by the corresponding list embedded in slashes. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE353I THE DIMENSIONED ITEM HAS A NON-INTEGER SUBSCRIPT.

Explanation: A non-integer subscript has been specified. The variable specifying the incorrect subscript appears before the message text.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Verify that all subscripts are integer. Make sure that only a combination of integer and real mixed mode expressions occurs, if mixed mode is present. Check that no subscript rules are violated. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE354I A VARIABLE TO BE DIMENSIONED USING ADJUSTABLE DIMENSIONS MUST HAVE BEEN PASSED AS AN ARGUMENT AND MUST NOT APPEAR IN COMMON.

Explanation: A variable to be dimensioned using adjustable dimensions was specified in COMMON or was not specified in an argument list. The variable name appears before the message text.

(Condition code - 12)

System_Action: Job processing is terminated at the end of compilation.

Programmer_Response: Probable user error. If the variable has been entered in a COMMON statement, remove it. If the variable is not already in the argument list, place it there. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE355I ADCON TABLE EXCEEDED.

Explanation: A program coded in a complex manner may cause the number of address constants (adcons) generated to exceed the maximum.

(Condition code - 16)

System_Action: Job processing is terminated immediately.

Programmer_Response: Restructure any complex statement into less involved statements. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE356I A PARAMETER CANNOT ALSO BE IN COMMON.

Explanation: A variable is used both as a parameter and in a COMMON statement. The variable name appears before the message text.

(Condition code - 12)

System_Action: Job processing is terminated at the end of compilation.

Programmer_Response: Probable user error. Correct the source and recompile. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE357I THE ARRAY HAS AN INCORRECT ADJUSTABLE DIMENSION.

Explanation: An invalid adjustable dimension was specified. The array name appears before the message text. Message IFE358I is also printed with this message.

(Condition code - 12)

System_Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid adjustable dimension. See message IFE358I for the incorrect message variable. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE358I THE ADJUSTABLE DIMENSION IS NOT PASSED AS AN ARGUMENT OR IN COMMON.

Explanation: The adjustable dimension was not specified in a COMMON statement or in an argument list. The invalid adjustable dimension name appears before the message text. Message IFE357I is also printed with this message.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Either include the adjustable dimension in an argument list, or place it in COMMON. See message IFE357I for the array name. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.
- Make sure that the 'XREF' option is specified in the PARM parameter and that the associated DD statement is included in the job stream.

IFE362I INCORRECTLY NESTED OR INCOMPLETE DO LOOP.

Explanation: The implied DO in the associated READ/WRITE statement is either incorrectly nested or incomplete.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation. The READ/WRITE statement will not be treated as partial short list, and the object module will probably be incorrect.

Programmer Response: Probable user error. Make sure the implied DO is nested correctly. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.
- Specify 'LIST, MAP, XREF, TRACE(8388672)' as options in the PARM field of the EXEC statement and include a SYSUT2 DD statement.

IFE401I COMPILATION DELETED. A MESSAGE FOR AN UNRECOVERABLE ERROR HAS BEEN ISSUED.

Explanation: The compiler phase in control requested immediate termination because of a condition that resulted in another level 16 error message. The accompanying message describes the error condition causing termination.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Investigate the accompanying level 16 message.

IFE402I COMPILATION DELETED. REGION SIZE OR SIZE OPTION TOO SMALL.

Explanation: The compiler could not continue processing because of storage size limitation. The size specified in the REGION parameter was too small, or the size specified in the SIZE option was too large to be accommodated by the operating system in its search for storage space.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Specify a larger size in the REGION parameter. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified in the JOB statement.
- Have source and associated listing available.

IFE411I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX1XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An invalid operation was requested, causing the operating system to terminate compiler processing. The string of characters XXXXXXX1XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 1 in the eighth position is the interrupt code for an invalid operation. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE412I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX2XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An invalid priviliged operation was requested, causing the operating system to terminate compiler processing. The string of characters XXXXXXX2XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 2 in the eighth position is the interrupt code for an invalid priviliged operation. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE413I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX3XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An invalid EXECUTE instruction was requested, causing the operating system to terminate compiler processing. The string of characters XXXXXXX3XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 3 in the eighth position is the interrupt code for an invalid EXECUTE operation. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE414I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX4XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An attempt was made to access protected storage, causing the operating system to terminate compiler processing. The string of characters XXXXXXX4XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 4 in the eighth position is the interrupt code for a protection exception. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE415I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX5XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An attempt was made to address a location outside available storage, causing the operating system to terminate compiler processing. The string of characters XXXXXXX5XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 5 in the eighth position is the interrupt code for an addressing exception. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE416I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX6XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An attempt was made to reference data not on an integral boundary, causing the operating system to terminate compiler processing. The string of characters XXXXXXX6XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 6 in the eighth position is the interrupt code for a specification exception. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE417I PROGRAM INTERRUPT - OLD PSW IS XXXXXXX7XXXXXXXX IFEAAA IS AT YYYYYY PHASE SWITCH IS ZZ

Explanation: An attempt was made to reference arithmetic data having an incorrect arithmetic sign or too many significant high-order digits, causing the operating system to terminate compiler processing. The string of characters XXXXXXX7XXXXXXXX indicates the PSW (program status word) present when the interrupt occurred; the character 7 in the eighth position is the interrupt code for a data exception. The characters YYY indicate the absolute location of the compiler module IFEAAA when processing stopped. The characters ZZZ indicate the switch name of the compiler phase in operation when processing stopped.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure the source module is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'DUMP,TRACE(64)' as options in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE420I COMPILER ERROR. AN ILLEGAL FORMAT CHARACTER WAS SPECIFIED.

Explanation: A compiler read or write operation used a non-supported format character. The programmer requested a compiler option (MAP or FORMAT) which caused the compiler to do a read or write operation in error.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Do not request the option causing the error. If the option is necessary, do the following before calling IBM for programming support:
- Have source and associated listing available.
- Specify DUMP as an option in the PARM field of the EXEC statement and include a SYSABEND DD statement.

| IFE421I COMPILER ERROR. I/O LIST EXCEEDS LOGICAL RECORD LENGTH.

Explanation: An input operation used in compiler processing tried to read a record that was too long. The programmer requested a compiler option (MAP or FORMAT) which caused the compiler to do a read operation in error.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Do not request the option causing the error. If the option is necessary, do the following before calling IBM for programming support:
- Have source and associated listing available.
- Specify DUMP as an option in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE422I MISSING DD CARD FOR SYSIN.

Explanation: No SYSIN DD statement is specified.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Probable user error. Include the missing DD statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE423I MISSING DD CARD FOR SYSLIN, NOOBJECT OPTION USED.

Explanation: No SYSLIN DD statement is specified. No object module is produced.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Include the missing DD statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE424I MISSING DD CARD FOR SYSPUNCH, NODECK OPTION USED.

Explanation: No SYSPUNCH DD statement is specified. No object module deck is produced.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Include the missing DD statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE425I MISSING DD CARD FOR SYSUT1, NOFORMAT OPTION USED.

Explanation: No SYSUT1 DD statement is specified. No structured source listing is produced.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Include the missing DD statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE426I MISSING DD CARD FOR SYSUT2, NOXREF OPTION USED.

Explanation: No SYSUT2 DD statement is specified. No compiler cross-reference listing is produced.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Include the missing DD statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE427I MISSING DD CARD FOR SYSPRINT.

Explanation: No SYSPRINT DD statement is specified. No printed output can be produced. This message prints on the same unit as system messages.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Probable user error. Include the missing DD statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE428I PERMANENT I/O ERROR, jobname, stepname, add, typ, ddname, op, err, block, accm

Explanation: A hardware error caused the operating system terminate compiler processing. This message prints on the same unit as system messages. In the message text, the variable information is the following:

Name	Meaning
jobname	The name of the job
stepname	The name of the job step
add	The address of the physical device (for example, tape, direct access device) which caused the error.
typ	Device type, that is, whether direct access or tape
ddname	The name of the DD statement describing the device
op	The operation that was being performed at the time the error occurred
err	The error that occurred (i.e., wrong length record)
block	The block of records being processed
accm	The access method (QSAM)

(Condition code - 16)

Programmer Response: Call IBM for FE service.

IFE429I PERMANENT I/O ERROR, jobname, stepname, add, typ, SYSPRINT, op, err, block, accm

Explanation: A hardware error on the device described in the SYSPRINT DD statement caused the operating system to terminate compiler processing. This message prints on the same unit as system messages. In the message text, the variable information is the following:

Name	Meaning
jobname	The name of the job
stepname	The name of the job step
add	The address of the physical device (for example, printer, direct access device) which caused the error.
typ	Device type, that is, whether direct access or tape
op	The operation that was being performed at the time the error occurred
err	The error that occurred (i.e., wrong length record)
block	The block of records being processed
accm	The access method (QSAM)

(Condition code - 16)

Programmer Response: Call IBM for FE service.

IFE430I AN INVALID OPTION HAS BEEN SPECIFIED IN THE PARM FIELD. IT HAS BEEN IGNORED.

Explanation: The compiler could not accept or recognize an option.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct or delete the invalid option. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.

IFE431I AN INVALID AUTODBL OPTION HAS BEEN SPECIFIED IN THE PARM FIELD. DBLPAD HAS BEEN ASSUMED.

Explanation: The AUTODBL option specified is invalid or incorrectly specified.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Correct the option specification. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE432I INVALID OPTION ON *PROCESS CARD. OPTION IGNORED.

Explanation: Either an invalid option was specified on a * PROCESS card or a valid option was specified incompletely. In the latter case, an option which requires a compiler data set was specified by the *PROCESS card (either explicitly, or implicitly through installation defaults), but was not specified for the first compilation in the batch (either explicitly or implicitly). Requests to compile with options that require a compiler data set are honored only when the data set was opened for the first compilation.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Specify all options that may require system data sets on the EXEC statement for the first compilation.

IFE433I MISSING DD CARD FOR SYSTERM, NOTERMINAL OPTION USED.

Explanation: No SYSTERM DD statement is specified. Error messages and compiler statistics are not grouped together at the end of the compilation output listing.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Include the missing DD statement, or specify the NOTERMINAL option. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE434I SYSTERM I/O ERROR, NOTERMINAL OPTION USED.

Explanation: Because a SYSTERM I/O error occurred, the NOTERMINAL EXEC option was used for the compilation(s).

(Condition code - 4)

System Action: Job processing continues, but the duplicate listing of error messages and source statements in error is not produded.

Programmer Response: Ensure that both the SYSTERM and SYSUT1 DD cards are specified and check for any system messages. Rerun the job. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE435I MISSING DD CARD FOR SYSUT1, NOTERMINAL OPTION USED.

Explanation: Because the DD card for SYSUT1, which holds source statements for use with the TERM and FORMAT compiler EXEC options, was missing, the NOTERMINAL option was used for the compilation(s). This message is issued whenever the SYSUT1 DD card is missing and errors have been encountered, regardless of whether the TERMINAL option was specified.

(Condition code - 4)

System Action: Job processing continues, but the duplicate listing of error messages and source statements in error is not produced.

Programmer Response: Probable user error. Ensure that both SYSUT1 and SYSTERM DD cards are specified and rerun the job. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE436I OPTIMIZATION MAY NOT BE INITIALLY REQUESTED ON * PROCESS CARD in CMS, NOOPTIMIZE OPTION USED.

Explanation: Under CMS, optimization must be requested on the execute (EXEC) card and be in effect for the entire batch compilation.

(Condition code - 4)

System Action: Job processing continues, but optimization is not performed.

Programmer Response: Specify optimization on the execute card and delete it from the *PROCESS card. Run programs for which optimization is desired separately or in a separate batch from those for which optimization is not desired. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE490I SOURCE PROGRAM TOO LARGE. TOO MANY BRANCH ADDRESSES REQUIRED.

Explanation: Source program contains too many branches, causing a compiler table to overflow.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Subdivide the program. If OPTIMIZE(0) was specified, resubmit the program specifying OPTIMIZE(1) or (2). If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE491I SOURCE PROGRAM TOO LARGE. CONSTANT GENERATED BY FLOAT, FIX, OR AMOD FAMILY CANNOT BE ADDRESSED.

Explanation: Too many formats, or too many computed GO TO statements or too many uses of members of the FLOAT, FIX, or AMOD family occurred, causing program addresses to be generated beyond the range of register 13.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Remove formats and computed GO TO statements; reduce the use of members of the FLOAT, FIX, and AMOD families. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE500I A REFERENCE TO A FORTRAN-SUPPLIED FUNCTION HAS INCORRECT NUMBER OF ARGUMENTS, OR HAS ARGUMENTS OF THE WRONG DATA TYPE.

Explanation: The arguments of a FORTRAN-supplied function are not of the correct type or mode, or an incorrect number has been specified, or a user-supplied function may not have been defined in an EXTERNAL statement, causing a FORTRAN-supplied function to be substituted.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. If the function is user supplied, then make sure the function name appears in an EXTERNAL statement. Make sure that an argument mode is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE501I THE EXPRESSION HAS A COMPLEX EXPONENT.

Explanation: The exponent has been expressed as a complex number.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the exponent so that it is integral with a complex base, and otherwise integral or real. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE502I THE EXPRESSION HAS A BASE WHICH IS COMPLEX BUT THE EXPONENT IS NON-INTEGER.

Explanation: A non-integer exponent has been specified.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the expression so that a complex base has an integer exponent. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE503I A GENERIC FUNCTION REFERENCE HAS INCORRECT NUMBER OF ARGUMENTS, OR HAS ARGUMENTS OF THE WRONG DATA TYPE.

Explanation: The arguments are not of the correct type or an incorrect number has been specified.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid reference. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE504I THE EXPONENTIAL EXPRESSION USES A LOGICAL VARIABLE.

Explanation: A logical variable has been specified as either a base or an exponent.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Make sure that base and/or exponent are only of type real, integer, or complex. Check placement of parentheses. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE505I THE INPUT/OUTPUT STATEMENT REFERS TO THE STATEMENT NUMBER OF A NON-FORMAT STATEMENT.

Explanation: A non-FORMAT statement number is specified in an input/output statement as the statement number describing record format.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid statement number. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE506I THERE IS A MISSING OPERAND PRECEDING A RIGHT PARENTHESIS.

Explanation: The compiler expected to encounter an operand but detected a right parenthesis first.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the statement and resubmit the job. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE507I AN ARRAY NAME OR FUNCTION NAME APPEARS IMPROPERLY AS AN ARGUMENT TO A FORTRAN-SUPPLIED FUNCTION.

Explanation: An array item not having a subscript, or a function name improperly used appears as an argument.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid item. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE508I AN ARRAY NAME OR FUNCTION NAME APPEARS IMPROPERLY AS AN ARGUMENT IN A GENERIC FUNCTION REFERENCE.

Explanation: An array item not having a subscript, or a function name improperly used appears as an argument.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete items in the argument list. If the problems recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE509I THE PROGRAM DOES NOT END WITH ONE OF THE ALLOWABLE LAST EXECUTABLE STATEMENTS.

Explanation: A terminal statement is missing.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Insert the necessary terminal statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE510I THE EXPRESSION HAS A LOGICAL OPERATOR WITH A NON-LOGICAL OPERAND.

Explanation: A non-logical operand appears with a logical operator.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the operand. Make sure that a logical primary or logical expressions have correct form. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE511I AN ASYNCHRONOUS I/O STATEMENT HAS AN INVALID NON-INTEGER ID PARAMETER.

Explanation: A non-integer value has been specified in the ID parameter.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Change the ID parameter to an integer variable, constant, or expression. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listings available.

IFE512I THE LOGICAL IF DOES NOT CONTAIN A LOGICAL EXPRESSION.

Explanation: A logical expression does not appear in a logical IF statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE513I A SYMBOLIC NAME IS USED TO REFER TO BOTH A FUNCTION AND A SUBROUTINE.

Explanation: A name has been specified twice.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Specify unique names for the function and subroutine. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE514I PUSHDOWN OR ASF ARGUMENT TABLE EXCEEDED, OR DEPTH OF FUNCTION NESTING TOO GREAT.

Explanation: Too many entries to a compiler table caused the table to overflow, i.e., exceed its limits.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Change the program structure. Restructure deeply nested function statements or eliminate some ASF (arithmetic statement function) arguments if possible. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE515I THE EXPRESSION HAS A RELATIONAL OPERATOR WITH A COMPLEX OPERAND.

Explanation: A complex operand occurs in an expression with a relational operator.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the complex operand. Equivalence a real array of 2 elements to a complex variable to permit use of the relational operator, if necessary. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE516I THE ARITHMETIC IF CONTAINS A COMPLEX EXPRESSION.

Explanation: A complex expression occurs in an arithmetic IF statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the expression. Convert the expression, if possible, to a permissible type. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE517I A NON-SUBSCRIPTED ARRAY ITEM APPEARS IMPROPERLY IN AN ARITHMETIC STATEMENT, A FUNCTION REFERENCE, OR A SUBROUTINE CALL.

Explanation: An array item not having a subscript is specified improperly.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Indicate which array item is to be processed. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE520I THERE IS A COMMA IN AN INVALID POSITION.

Explanation: The compiler expected to encounter a character other than a comma.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or reposition the comma. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE521I THE EXPRESSION HAS AT LEAST ONE EXTRA RIGHT PARENTHESIS.

Explanation: Unbalanced parentheses occurred.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Delete or correct extraneous and invalid parentheses. Make sure parentheses are balanced. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE522I THE EXPRESSION HAS AT LEAST ONE TOO FEW RIGHT PARENTHESES.

Explanation: Unbalanced parentheses occurred.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or add needed parentheses. Make sure parentheses are balanced. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE523I THE EQUAL SIGN IS IMPROPERLY USED.

Explanation: The compiler expected to encounter a character other than an equal sign.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the invalid equal sign. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE524I THE EXPRESSION HAS AN OPERATOR MISSING AFTER A RIGHT PARENTHESIS.

Explanation: The compiler expected to encounter an operator in an expression.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Insert the missing operator or delete the erroneous parenthesis. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE525I THE EXPRESSION USES A LOGICAL OR RELATIONAL OPERATOR INCORRECTLY.

Explanation: A relational operator is used to compare expressions not integer or real, or a logical operator is used to compare expressions not logical, or an operator is misplaced.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Either correct the logical or relational operator, or change invalid operand expressions. Make sure operators are preceded and followed by a period. Verify that expressions precede operators where required. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE526I THE EXPRESSION USES A LOGICAL VARIABLE WITH A RELATIONAL OPERATOR

Explanation: A relational operator is used to join logical quantities.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Substitute integer and/or real quantities for the logical quantities. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE529I A FUNCTION NAME APPEARING AS AN ARGUMENT HAS NOT BEEN DECLARED EXTERNAL.

Explanation: The name of a subprogram or function passed as an argument to another subprogram has not been defined in an EXTERNAL statement in the calling program.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer response: Probable user error. Insert the required EXTERNAL statement, or delete or correct the invalid function reference. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE530I THE EXPRESSION HAS A VARIABLE WITH AN IMPROPER NUMBER OF SUBSCRIPTS.

Explanation: The number of subscripts specified differs from the number declared in the associated specification statement.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Check for all necessary delimiters. Make sure that there are as many subscripts as are declared in the associated specification statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE531I THE EXPRESSION HAS A STATEMENT FUNCTION REFERENCE WITH AN IMPROPER NUMBER OF ARGUMENTS.

Explanation: The number of arguments in the function reference does not agree with the number of arguments specified in the function definition.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid function reference. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE532I THE CONSTANT SUBSCRIPT IS OUT-OF-RANGE.

Explanation: A subscript constant used in an array reference in an arithmetic assignment statement or an I/O list is negative or zero or it exceeds its respective dimension.

(Condition code - 12)

System Action: Job processing terminates at the end of compilation.

Programmer, Response: Probable user error. Correct the subscript in the array reference so it is within the range of the array dimension. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE541I A REFERENCE TO AN EXTENDED LANGUAGE BUILT-IN FUNCTION HAS AN INCORRECT NUMBER OR TYPE OF ARGUMENTS.

Explanation: The type or number of arguments in the function reference does not agree with the type or number of arguments in the function definition.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct the invalid argument. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE542I A LOGICAL EXPRESSION APPEARS IN INVALID CONTEXT.

Explanation: An invalid logical expression was specified.

(Condition code - 12)

System Action: Job processing is terminated at the end of compilation.

Programmer Response: Probable user error. Correct or delete the source statement. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IFE550I PUSHDOWN, ADCON, OR ASF ARGUMENT TABLE EXCEEDED.

Explanation: The number of entries to be stored exceeded the storage spare allocated for the pushdown, adcon, or ASF argument table.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Change the program structure. If there are many subroutine references in a program unit, subdivide the program unit. Restructure deeply nested expressions or eliminate some ASF arguments where many occur, if possible. Where parentheses are deeply nested, restructure the source statement, if possible, to eliminate some of the nesting. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE552I SOURCE PROGRAM TOO LARGE. TOO MANY CALL ARGUMENT ADDRESSES.

Explanation: Too many adcons for CALL arguments are required by the object program.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Subdivide the program, reduce the number of CALL arguments by using COMMON where possible to pass arguments, and recompile. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE570I TABLE EXCEEDED, OPTIMIZATION DOWNGRADED.

Explanation: The program is too large to permit optimization. The compiler performs OPTIMIZE(1) register allocation only; no other optimization is performed.

(Condition code - 0)

System Action: Job processing continues.

Programmer Response: Either the program should be segmented or the size of the compiler table should be increased. A compiler table may be increased at program installation time. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE580I COMPILER ERROR.

Explanation: One of the following conditions occurred: an invalid adjective code was detected; an illegal element length was detected; no equivalence group was found; an unusual primary adjective code was detected; a NAMELIST list was not found.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure that the source program is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE600I INTERNAL COMPILER ERROR. LOGICALLY IMPOSSIBLE BRANCH TAKEN IN A COMPILER SUBROUTINE.

Explanation: An error in a compiler routine caused the compiler to take an impossible branch. The label of a statement in the compiler routine appears before the message text.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure that source code is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'TRACE(4114)' as an option in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE610I THE STATEMENT NUMBER OR GENERATED LABEL IS UNREACHABLE.

Explanation: Incorrect branch target statements are specified, or a statement following an arithmetic IF is unlabeled. The unlabeled statement is ignored.

(Condition code - 4)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that control statements indicate correct branch targets. Verify that an unlabeled STOP, RETURN, or GO TO does not immediately follow any one of these same three source statements. Make sure that the statement following an arithmetic IF is labeled. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE620I THE STATEMENT NUMBER OR GENERATED LABEL IS A MEMBER OF AN UNREACHABLE LOOP.

Explanation: Control statements do not indicate correct branch targets.

(Condition code - 8)

System Action: Job processing continues.

Programmer Response: Probable user error. Make sure that control statements indicate correct branch targets. Correct statement numbers so that the loop may be the target of a branch. Delete invalid terminal source statements. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE630I INTERNAL STRUCTURAL-ANALYSIS TABLES EXCEEDED.

Explanation: The source program contains more than 1000 statement numbers and generated labels.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Split the source program so that it contains fewer statement numbers. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE640I SOURCE PROGRAM TOO LARGE, TOO MANY INTERMEDIATE COMPUTATIONS.

Explanation: The source program contains too many items resulting in excessive storage requirements for compiler generated constants and intermediate values.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Subdivide the program and recompile. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE650I INTERNAL ADCON TABLE EXCEEDED.

Explanation: No space remains in the internal adcon table for compiler generated constants and intermediate values.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Subdivide the program and recompile. If SIZE was specified, indicate a larger SIZE, if possible. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IFE660I INTERNAL COMPILER ERROR. TEMPORARY FETCHED BUT NEVER STORED.

Explanation: The compiler references an undefined one-byte temporary (intermediate value).

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure that source is correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'TRACE(4114)' as an option in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE661I INTERNAL COMPILER ERROR. UNABLE TO FREE A REGISTER.

Explanation: The compiler requires the use of a register and none is available.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Segment large spans of unlabeled source code into smaller extents by inserting statement numbers. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'TRACE=(4354)' as an option in the PARM field of the EXEC statement and include a SYSABEND DD statement.

IFE662I INTERNAL COMPILER ERROR. AN INTER-BLOCK TEMPORARY HAS NOT BEEN PROMOTED.

Explanation: A one-byte temporary (intermediate value) moved to a target by a text optimization technique was not promoted.

(Condition code - 16)

System Action: Job processing is terminated immediately.

Programmer Response: Make sure that source statements are correct. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Specify 'TRACE(4114)' as an option in the PARM field of the EXEC statement and include a SYSABEND DD statement.

The FORTRAN IV library (Mod II) generates three types of messages:

- Program interrupt messages
- Execution error messages
- Operator messages

All library messages are numbered. Program interrupt messages are written when an exception to a system restriction occurs, such as when an invalid storage address or a violation of protected storage is requested. Execution error messages are written when a FORTRAN library function or subroutine is misused or an input/output error occurs. Operator messages are written when a STOP n or PAUSE statement is executed.

The descriptions of library messages often mention the use of the extended error handling facility. A full discussion of this facility appears in the publication IBM System/360 Operating System: FORTRAN IV (H Extended) Compiler Programmer's Guide, Order No. SC28-6852.

PROGRAM INTERRUPT MESSAGES

Program interrupt messages are written with the old Program Status Word (PSW) which aids the programmer in determining the nature of the error.

Program interrupt messages consist of messages IHO207, IHO208, IHO209, and IHO210. Messages IHO207-IHO209 are produced only when the extended error handling facility is specified; message IHO210 is produced any time an exception occurs.

IHO207I IBCOM - PROGRAM INTERRUPT-OVERFLOW OLD PSW IS xxxxxxxCxxxxxxxx REGISTER CONTAINED YYY

Explanation: This message is produced only when the extended error message facility is specified. The message indicates that an exponent-overflow exception, identified by the character C in the eighth position of the PSW, has occurred. This exception occurs when the result of a floating-point arithmetic operation is greater than or equal to 16^{63} (approximately $7.2x10^{75}$).

Supplemental Data: The floating point number before alteration.

Standard Corrective Action: Execution continues at the point of the interrupt with the result register set to the largest possible floating-point number that can be represented in short precision ($16^{63}*(1-16^{-6})$), in long precision ($16^{63}*(1-16^{-14})$, or in extended precision ($16^{63}*(1-16^{-28})$).

Programmer Response: Probable user error. Make sure that a variable or variable expression does not exceed the allowable magnitude. Verify that all variables have been initialized correctly in previous source statements and have not been inadvertently modified in intermediate source. If the problem recurs, do the following before calling IBM for programming support.

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and the necessary DD statement is included.
- Have source and associated listing available.

IHO208I IBCOM - PROGRAM INTERRUPT-UNDERFLOW OLD PSW IS xxxxxxxDxxxxxxxx REGISTER CONTAINED YYY

Explanation: This message is produced only when the extended error message facility is specified. The message indicates that an exponent-underflow exception, identified by a D in the eighth position of the PSW, has occurred. This exception occurs when the result of a floating-point arithmetic operation is less than 16^{-65} (approximately $5.4x10^{-79}$).

Supplemental Data: The floating point number before alteration.

Standard Corrective Action: Execution continues at the point of the interrupt with the result register set to a true zero of correct precision.

Programmer Response: Probable user error. Make sure that a variable or variable expression is not smaller than the allowable magnitude. Verify that all variables have been initialized correctly in previous source statements and have not been inadvertently modified in intermediate source. If the problem recurs, do the following before calling IBM for programming support.

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF option is specified in the PARM field of the EXEC statement and that the nessary DD statement is included.
- Have source and associated listing available.

IHO209I IBCOM - PROGRAM INTERRUPT-DIVIDE CHECK OLD PSW IS

$\text{xxxxxxx}\begin{Bmatrix}9\\ F\end{Bmatrix}\text{xxxxxxxx}$ REGISTER CONTAINED YYY

Explanation: This message is produced only when the extended error message facility is specified. This message indicates that an attempt to divide by zero has occurred. A fixed-point-divide exception is identified by a 9 in the eighth position of the PSW; a floating-point-divide exception by an F.

Supplemental Data: Floating-point registers before alteration, for a floating-point interrupt.

Standard Corrective Action: For floating-point-divide, execution continues at the point of the interrupt with the result registers set to:

a. True zero of correct precision for case of n/0 where n=0.

b. Largest possible floating-point number of correct precision for case of n/0 where n≠0. For fixed-point-divide, leave registers unmodified and continue execution.

Programmer Response: Probable user error. Either correct the source where division by zero is occurring, or modify previous source statements to test for the possibilities, or bypass the invalid division. If the problem recurs, do the following before calling IBM for programming support.

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the neccessary DD statement is included.
- Have source and associated listing available.

IHO210I PROGRAM INTERRUPT ($\begin{Bmatrix}\text{A}\\ \text{P}\\ \text{O}\end{Bmatrix}$) OLD PSW IS XXXXXXX$\begin{pmatrix}4\\5\\6\\7\\9\\ \text{C}\\ \text{D}\\ \text{F}\end{pmatrix}$XXXXXXXX

Explanation: The operating system has detected a condition that causes a program interruption.

Note: Codes 4, 5, 6, and 7 are associated with the execution-time adjustment of boundary alignment errors and appear only when BOUNDRY=ALIGN is specified in the FORTLIB macro instruction during program installation. Codes 5 and 6 are also associated with execution time floating-point-extended-precision simulation.

If the letter A appears in the message, boundary adjustment has taken place. The letter P in the message indicates that the interruption was precise. This will always be the case for non-specification interrupt messages in FORTRAN except when using machines with special hardware on which imprecise interruptions may occur. The letter O in the message indicates that extended precision floating point simulation has taken place and a secondary interrupt occurred. The eighth character in the PSW (i.e., 4, 5, 6, 7, C, D, or F) represents the code number (in hexadecimal) associated with the type of interruption. The following text describes these interruptions.

PROTECTION EXCEPTION: The protection exception (code 4) is recognized when the key of an operand in storage does not match the protection key in the PSW. A message is issued only if a specification exception (code 6) has already been recognized in the same instruction. Otherwise, the job terminates abnormally.

If the extended error handling facility is specified, the following information is provided:

```
IBCOM - PROGRAM INTERRUPT - ALIGNMENT OLD PSW IS
   xxxxxxx4xxxxxxxx
```

Supplemental Data: None.

Standard Corrective Action: Execution continues at point of interrupt.

Programmer Response: Probable user error. If the job has been terminated with a completion code of SYSTEM=0C6 (Specification Interrupt), correct the source causing boundary misalignment. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, DUMP, and LIST have been specified as parameters on the EXEC statement, and that the SYSUDUMP or SYSABEND DD statement is included.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

ADDRESSING EXCEPTION: The addressing exception (code 5) is recognized when the address of the data is outside of the available storage for the particular installation. A message is issued only if exception codes 5 or 6 have already been recognized in the same instruction. Otherwise, the job terminates abnormally.

If the extended error handling facility is specified, the following information is provided:

```
                            (ALIGNMENT)
IBCOM - PROGRAM INTERRUPT - {         } OLD PSW IS
                            (OPERATION)

   xxxxxxx5xxxxxxxx
```

Supplemental Data: None.

Standard Corrective Action: Execution continues at point of interrupt.

Programmer Response: Probable user error. If the job has been terminated with a completion code of SYSTEM=0C6 (Specification Interrupt), correct the source causing boundary misalignment. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, DUMP, and LIST have been specified as parameters on the EXEC statement, and that the SYSUDUMP or SYSABEND DD statement is included.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

SPECIFICATION EXCEPTION: The specification exception (code 6) is recognized when a data address does not specify an integral boundary for that unit of information. A specification error would occur during execution of the following instructions:

```
DOUBLE-PRECISION D, E
COMMON A, B, C
EQUIVALENCE (B, D)
D = 3.0D02.
```

Note: If an instruction contains a boundary violation, a specification interrupt will occur and the message will be issued with code 6. The boundary-adjustment routine will then be invoked, if the BOUNDRY=ALIGN option was specified in the FORTLIB macro instruction during program installation. If an instruction which has been processed for boundary misalignment also contains a protection, addressing, or data error, the interrupt message will be reissued with the appropriate code (4, 5, or 7). The job will then terminate because both a specification error and a protection, addressing, or data error have been detected. The completion code in the dump will specify that the job terminated because of the specification error.

If the extended error handling facility is specified, the following information is provided:

$$\text{IBCOM - PROGRAM INTERRUPT - } \left\{\begin{array}{c}\text{OPERATION}\\ \text{ALIGNMENT}\end{array}\right\} \text{ OLD PSW IS}$$

```
xxxxxxx6xxxxxxxx
```

Supplemental Data: None.

Standard Corrective Action: Execution continues at the point of the interrupt.

Programmer Response: Probable user error. Make sure that proper alignment of variables is guaranteed. Arrange variables in fixed descending order according to length, or force proper alignment with dummy variables. Construct COMMON blocks so that the displacement of each variable can be evenly divided by the reference number associated with the variable. Use the MAP option for information on the relative address of each variable in the block. Make sure that EQUIVALENCE statements do not cause misalignment. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, DUMP, and LIST have been specified as parameters on the EXEC statement, and that the SYSUDUMP or SYSABEND DD statement is included.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

DATA EXCEPTION: The data exception (code 7) is recognized when the sign or digit codes for a CONVERT TO BINARY instruction are incorrect. A message is issued only if a specification exception (code 6) has already been recognized in the same instruction. Otherwise, the job terminates abnormally.

If the extended error handling facility is specified, the following information is provided:

```
IBCOM - PROGRAM INTERRUPT - ALIGNMENT OLD PSW IS
   xxxxxxx7xxxxxxxx
```

Supplemental Data: None.

Standard Corrective Action: Execution continues at point of interrupt.

Programmer Response: Probable user error. If the job has been terminated with a completion code of SYSTEM=0C6 (Specification Interrupt), correct the source causing boundary misalignment. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, DUMP, and LIST have been specified as parameters on the EXEC statement, and that the SYSUDUMP or SYSABEND DD statement is included.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

FIXED-POINT-DIVIDE EXCEPTION: The fixed-point-divide exception (code 9) is recognized when division of a fixed-point number by zero is attempted. For example, a divide exception would occur during execution of the following statement:

```
K=I/J
```

where:
    J=0 and I=7

Supplemental Data: None.

Standard Corrective Action: Execution continues at the point of the interrupt.

Programmer Response: Probable user error. Either correct the source where division by zero is occurring, or modify previous source statements to test for the possibility of or bypass the invalid division. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

EXPONENT-OVERFLOW EXCEPTION: The exponent-overflow exception (code C) is recognized when the result of a floating-point addition, subtraction, multiplication, or division is greater than or equal to 16^{63} (approximately $7.2x10^{75}$). For example, an exponent-overflow would occur during execution of the statement:

```
A=1.0E+75+7.2E+75
```

When the interrupt occurs, the result register contains a floating-point number whose fraction is normalized and whose sign is correct. However, the number is not usable for further computation since its characteristic field no longer reflects the true exponent. The content of the result register as it existed when the interrupt occurred is printed following the program interrupt message with the format:

```
REGISTER CONTAINED hhhhhhhhhhhhhhhh
```

where: hhhhhhhhhhhhhhhh is the floating-point number in hexadecimal notation. (An additional 16 hexadecimal characters are printed for extended-precision numbers.)

If the improved floating-point engineering change is not in effect, the register content cannot be used to calculate the true value.

If the improved floating-point engineering change is in effect, exponent overflow causes "exponent wraparound" - i.e., the characteristic field represents an exponent that is 128 smaller than the correct one. Treating bits 1 to 7 (the exponent characteristic field) of the floating-point number as a binary integer, the true exponent may be computed as follows:

```
TE = (Bits 1 to 7) + 128 - 64
```

Standard Corrective Action: The result register is set to the largest possible floating-point number that can be represented in short precision ($16^{63}*(1-16^{-6})$) in long precision ($16^{63}*(1-16^{-14}$), or in extended precision ($16^{63}*(1-16^{-28})$), but the sign of the result is not changed. The condition code is not altered.

Programmer Response: Probable user error. Make sure that a variable expression does not exceed the allowable magnitude. Verify that all variables have been initialized correctly in previous source statements and have not been inadvertently modified in intermediate source. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

EXPONENT-UNDERFLOW EXCEPTION: The exponent-underflow exception (code D) is recognized when the result of a floating-point addition, subtraction, multiplication, or division is less than 16^{-65} (approximately $5.4x10^{-79}$). For example, an exponent-underflow exbeption would occur during execution of the statement:

```
A=-1.0E-50*1.0E-50
```

Although exponent underflows are maskable, FORTRAN jobs are executed without the mask so that the library will handle such interrupts.

When the interrupt occurs, the result register contains a floating-point number whose fraction is normalized and whose sign is correct. However, the number is not usable for further computation since its characteristic field no longer reflects the true exponent. The content of the result register as it existed when the interrupt occurred is printed following the program interrupt message with the format:

```
REGISTER CONTAINED hhhhhhhhhhhhhhhh
```

where: hhhhhhhhhhhhhhhh is the floating-point number in hexadecimal notation. (An additional 16 hexadecimal characters are printed for extended-precision numbers.)

If the improved floating-point engineering change is not in effect, the exponent underflow always leaves a zero in the result register.

If the improved floating-point engineering change is in effect, exponent underflow causes "exponent wraparound" - i.e., the characteristic field represents an exponent that is 128 larger than the correct one. Treating bits 1 to 7 (the exponent characteristic field) of the floating-point number as a binary integer, the true exponent may be computed as follows:

TE = (Bits 1 to 7) - 128 - 64

Standard Corrective Action: The result register is set to a true zero of correct precision. If the interrupt resulted from a floating-point addition or subtraction operation, the condition code is set to zero to reflect the setting of the result register.

Programmer Response: Probable user error. Make sure that a variable or variable expression is not smaller than the allowable magnitude. Verify that all variables have been initialized correctly in previous source statements and have not been inadvertently modified in intermediate source. To take advantage of the "exponent wraparound" feature and override the FORTRAN interruption routine, a programmer may handle the interrupt in his own program, but must call an assembly language subroutine to issue a SPIE macro instruction. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

FLOATING-POINT-DIVIDE EXCEPTION: The floating-point-divide exception (code number F) is recognized when division of a floating-point number by zero is attempted. For example, a floating-point divide exception would occur during execution of the statement:

```
C=A/B
```

where:
B=0.0 and A=1.0 or B0.0 and A0.0

If the extended error message facility is specified, the following information is provided:

```
IBCOM - PROGRAM INTERRUPT - ALIGNMENT OLD PSW IS X
```

Supplemental Data: None.

Standard Corrective Action: None.

or

IBCOM - PROGRAM INTERRUPT - $\left\{\begin{matrix}\text{ALIGNMENT}\\ \text{OPERATION}\end{matrix}\right\}$ OLD PSW IS

(contents of old PSW)

Supplemental Data: Registers before alteration.

Standard Corrective Action: Execution continues at point of interrupt with result registers set to:

a. True zero correct precision for case of n/0 where n=0.

b. Largest possible floating point number of correct precision for case of n/0 where n≠0.

Programmer Response: Probable user error. Either correct the source where division by zero is occurring, or modify previous source statements to test for the possibility of or bypass the invalid division. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

EXECUTION ERROR MESSAGES

Execution error messages have the form:

IHOxxxI [message text]

TRACEBACK FOLLOWS-ROUTINE ISN REG. 14,REG. 15,REG. 0,REG. 1

The description of each diagnostic message contains the error code, the abbreviated name for the origin of the error, and an explanation describing the type of error. In addition, if the extended error handling facility was specified (by indicating OPTERR=INCLUDE in the FORTLIB macro instruction at program installation time), supplemental data is provided and standard corrective action to be taken to correct the error is described.

Variable information in the message is shown as X, and, in the corrective action descriptions, • denotes the largest possible number that can be represented for a floating-point value.

If the extended error handling facility is not specified, all execution error messages generate a condition code of 16 with resultant program termination, except the following:

- IHO210I (described above under the heading "Program Interrupt Messages")
- IHO217I, provided that an END=n parameter is specified in the appropriate READ statement, and
- IHO218I, provided that an ERR=n parameter is specified in the appropriate READ statement.

The abbreviated name for the origin of the error is:

ASYNC -- IHOASYNC routine (performs asynchronous input/output processing).

DIOCS -- IHODIOSE routine (performs direct access input/output operations for FORTRAN load module execution).

FCVTH -- IHOFCVTH routine (performs conversion).

FIOCS -- IHOFIOSH routine (performs input/output operations for FORTRAN load module execution).

IBC -- IHOFCOMH routine (performs interruption and error procedures).

IBERR -- IHOIBERH routine (performs the processing of errors detected during execution of the load modules).

LDFIO -- IHOLDFIO routine (performs list-directed input/output processing).

LIB -- SYS1.FORTLIB. In the explanation of the messages, the module name is given followed by the entry point name(s) enclosed in parentheses.

NAMEL -- IHONAMEL routine (performs namelist processing).

IHO203I IBCOM - INVALID COMBINATION OF INITIAL, TEST, AND INCREMENT VALUE FOR READ/WRITE IMPLIED DO.

Explanation: IBC -- A READ/WRITE statement with an implied DO had an invalid combination of initial, test, and increment values (I1, I2, and I3, respectively) for one of its nested levels:

1. I3=0, or

2. I2<I1 and I3≤I2-I1, or

3. I1<I2 and I3<0.

Standard Corrective Action: The job is terminated.

Programmer Response: Probable user error. Check the statements which set the initial, test, and increment variables for an error. If the problem recurs, do the following before calling IBM for programming support:

- Include the following statement as the first executable statement in the program:

```
CALL ERRSET (203,2,2,2,1)
```

- Recompile the program specifying LIST and MAP.
- Rerun the GO step including a SYSUDUMP DD statement.

IHO204I Explanation: LDFIO -- For a non-complex number, the number is longer than the buffer. For a complex number, half the length of the number plus a comma is longer than the buffer.

If the extended error message facility is specified, the following information is provided:

LDFIO -- ITEM SIZE EXCEEDS BUFFER LENGTH.

Standard Corrective Action: Remainder of I/O list ignored.

Programmer Response: Probable user error. Make sure that the DCB subparameter LRECL specifies a number large enough to contain the longest item in the I/O list. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO205I ASYNC - I/O SUBTASK ABENDED

Explanation: ASYNC - An asynchronous I/O subtask resulted in an abnormal termination.

Standard Corrective Action: None; the job step is terminated.

Programmer Response: Probable user error. Verify that all DD statements are coded correctly and point to the appropriate data sets. Check all READ and WRITE statements and any END FILE, REWIND, and BACKSPACE statements. Check the system completion code for assistance in the type of error that caused abnormal termination. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Make sure that LIST, XREF, and MAP have been specified as parameters on the EXEC statement.
- Provide a GO.SYSUDUMP or GO.SYSABEND DD statement.

IHO206I CONVERT INTEGER NUMBER OUT OF RANGE X

Explanation: FCVTH - An integer number was too large to be processed by the load module. (The largest number that can be processed is $16^{63}-1$.)

Supplemental Data: Integer specified.

Standard Corrective Action: Replace number with as much of the lower order part of the given number as will fit for the integer size (I*2 or I*4) specified.

Programmer Response: Probable user error. Make sure that all integer input used is within the required range for the integer size. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST are specified as compiler options.
- Have source and associated listing available.

(For explanations of messages IHC0207-210, see "Program Interrupt Messages.")

IHO211I Explanation: IBC -- An invalid character has been detected in a FORMAT statement.

If the extended error message facility is specified, the following information is provided:

IBCOM - ILLEGAL COMPILED FORMAT CHARACTER SPECIFIED
or
IBCOM - ILLEGAL VARIABLE FORMAT CHARACTER SPECIFIED X

Supplemental Data: Character in error.

Standard Corrective Action: Format field treated as an end of format.

Programmer Response: Probable user error. Make sure that all format specifications read in at object time are valid. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO212I Explanation: IBC -- An attempt has been made to read or write a record, under FORMAT control, that exceeds the buffer length.

If the extended error message facility is specified, the following information is provided:

IBCOM - FORMATTED I/O, END OF RECORD ON UNIT X

Supplemental Data: Unit number.

Standard Corrective Action: For a read, the remainder of the I/O list is ignored; for a write, a new record is started with no control character.

Programmer Response: Probable user error. If the error occurs on input, verify that a FORMAT statement does not define a FORTRAN record longer than the record referred to in the data set. If reading in data, either keep a counter to avoid exceeding the end of record or file, or insert an END= parameter on the READ statement for appropriate transfer of control on the end of data set. No record to be punched should be specified as longer than 80 characters. For printed output make sure that no specification is longer than the printer's line length. Check all DD statements. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Make sure that LIST has been specified as a parameter on the EXEC statement.

IH0213I This message may be produced by IBC, ASYNC, or LDFIO.

FOR IBC AND ASYNC:

Explanation: The input list in an input/output statement without a FORMAT specification is larger than the logical record.

If the extended error message facility is specified, the following information is provided:

$\left\{\begin{matrix}\text{ASYNC}\\ \text{IBCOM}\end{matrix}\right\}$ - UNFORMATTED READ, END OF RECORD ON UNIT X

Supplemental Data: Unit number.

Standard Corrective Action: The remainder of the I/O list is ignored.

Programmer Response: Probable user error. Either keep a counter to avoid exceeding end of record or file, or insert an END= parameter on the READ statement for appropriate transfer of control on end of data set. Check all DD statements. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Make sure that LIST and XREF have been specified as compiler options.

FOR LDFIO:

Explanation: A FORTRAN list-directed READ statement attempted to read more items from a variable spanned logical record than were present in the record. (This message can be issued only when RECFM=VS is specified).

If the extended error handling facility is specified, the following information is provided:

LDFIO -- LIST-DIRECTED READ, END OF DATA SET ON UNIT X.

Supplemental Data: Unit number.

Standard Corrective Action: The remainder of the I/O list is ignored.

Programmer Response: Make sure that the records and the input data agree in number. Either delete extra variable names or supply additional logical records. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.

IHO214I This message may be produced by either FIOCS or ASYNC.

FOR FIOCS:

Explanation: For unformatted records read or written in EBCDIC sequentially organized data sets, the record format (RECFM) specification must include the characters VS (variable spanned); any of the optional characters (B, A, M, or T) may be specified with the characters VS. This message appears if the programmer has coded RECFM=V, RECFM=U, or RECFM=F, or if an ASCII tape was specified.

If the extended error message facility is specified, the following information is provided:

FIOCS - UNFORMATTED I/O, RECORD FORMAT SPECIFIED AS F, U, OR V OR UNFORMATTED I/O REQUEST ON ASCII TAPE; UNIT = X

Supplemental Data: Unit number.

Standard Corrective Action: For non-ASCII data sets, the read request is ignored; for a write request, the record form is changed to VS.

Programmer Response: Probable user error. For ASCII data sets, include Format Specifications for non-ASCII data sets. Correct the RECFM subparameter. Change a V (variable) or U (undefined) or F (fixed) specification to VS. If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing and the associated job stream available.
- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.

FOR ASYNC:

ASYNC-UNFORMATTED I/O, RECORD FORMAT IS NOT VS OR VST ON UNIT X

Explanation: For unformatted records in an asynchronous input/output operation, the record format specification (RECFM) did not include the characters VS or VST.

Supplemental Data: Unit number.

Standard Corrective Action: For an input operation, the read request is ignored; for an output operation, VS is assumed.

Programmer Response: Probable user error. Include the record format specification as VS (or VST). If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO215I CONVERT - ILLEGAL DECIMAL CHARACTER X

Explanation: FCVTH -- An invalid character exists for the decimal input corresponding to an I, E, F, or D format code.

Supplemental Data: Display the record in which character appeared.

Standard Corrective Action: 0 replaces the character encountered.

Note: If the standard or corrective user action results in a null format, no output will result. If the FORMAT statement is terminated in such a way that no conversion type is called for, an alphanumeric literal may be repeated for each list item.

Programmer Response: Probable user error. If an IHO214I message has occurred previously, correct the source causing the error. Otherwise, make sure that all decimal input is valid. Correct any FORMAT statements specifying decimal input where character should be indicated. If the problem recurs, do the before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO216I SLITE-SLITET X IS AN ILLEGAL VALUE

Explanation: LIB -- An invalid sense light number was detected in the argument list in a call to the SLITE or SLITET subroutine.

Supplemental Data: The sense light value supplied.

Standard Corrective Action: For SLITE, no action is taken; for SLITET, the OFF indication is returned, i.e., J=2.

Programmer Response: Probable user error. If CALL SLITE(i) is specified, make sure that i is an integer expression with a value of 0-4, inclusive. If CALL SLITET(i,j) is specified, make sure that i is an integer expression with a value of 0-4, inclusive, and j is an integer variable. If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO217I *Explanation*: IBC -- An end of data set was sensed during a READ operation; that is, a program attempted to read beyond the data.

If the extended error message facility is specified, the following information is provided:

$\left\{\begin{array}{l}\text{ASYNC}\\ \\ \text{FIOCS}\end{array}\right\}$ END OF DATA SET ON UNIT X

Supplemental Data: Unit number.

Standard Corrective Action: Next file is read, i.e., sequence number incremented by 1.

Programmer Response: Probable user error. Make sure that a FORMAT statement does not define a FORTRAN record longer than the record referred to in the data set. Either keep a counter to avoid exceeding the end of record or file, or insert an END=n parameter on the READ statement for appropriate transfer of control on end of data set. Check all DD statements. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Make sure that LIST has been specified as a parameter on the EXEC statement.

IHO218I I/O ERROR xxx...xxx (see also IHO218I BACKSPACE ERROR)

Explanation: ASYNC, FIOCS or DIOCS -- One of the following occurred:

- A permanent input/output error has been encountered.
- For sequential I/O, the length of a physical record is inconsistent with the default block size or the blocksize specified on the DD card.
- An attempt has been made to read or write with magnetic tape a record that is less than 18 bytes long.

xxx...xxx is the character string formatted by the SYNADAF macro instruction. For an interpretation of this information, see the publication *IBM System/360 Operating System: Supervisor and Data Management Macro Instructions*, Order No. GC28-6647. After the traceback is completed, control is returned to the call routine statement designated in the ERR parameter of a FORTRAN READ statement if that parameter was specified.

Note: If a permanent input/output error has been detected while writing in the object error unit data set, the error message is written in the SYSOUT data set, and job execution is terminated.

If the extended error message facility is specified, the following information is provided:

FIOCS - I/O ERROR (text provided by data management)

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues and the input/output request is ignored.

Note: ERR=parameter is honored.

Programmer Response: Probable user error. Make sure that, for sequential input/output, the length of the physical record is consistent with the default or specified blocksize. Check all DD statements. Make sure that no attempt has been made to read or write with magnetic tape from a record that is less than 18 bytes in length. If extended error handling has not been specified, to prevent job termination include an ERR=n parameter in the READ statement to transfer control to another statement when an error occurs. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Make sure that LIST has been specified as a parameter on the EXEC statement.

IHO218I BACKSPACE ERROR (See also IHC218I I/O ERROR)

Explanation: An error was encountered in a backspace operation. This message is written to WTO and the program terminates.

Standard Corrective Action: None; the job is terminated.

Programmer Response: Rerun the program.

IHC219I This message may be generated for either EBCDIC or ASCII data sets.

FOR EBCDIC DATA SETS:

Explanation: FIOCS or ASYNC -- Either a data set is referred to in the load module, but no DD statement is supplied for it, or a DD statement has an erroneous ddname.

If the extended error message facility is specified, the following information is provided:

ASYNC - MISSING DD CARD FOR (DDNAME)

FIOCS - MISSING DD CARD OR DCB ERROR FOR ASCII TAPE FOR(DDNAME)

or

DIOCS - MISSING DD CARD FOR UNIT X

Supplemental Data: DDname or unit number.

Standard Corrective Action: Execution continues and the input/output request is ignored.

Note: If no DD statement has been supplied for the object error unit data set, the message is written in the SYSOUT data set, and the job is terminated.

Programmer Response: Probable user error. Either provide the missing DD statement, or correct any erroneous ddname.

Example: If Unit 6 is the installation data set reference number for the printer and an attempt is made to write on Unit 3, then the following DD statement should be included:

```
//GO.FT03F001 DD SYSOUT=A
```

If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Make sure that LIST has been specified as a parameter on the EXEC statement.
- Have output from program installation available.

FOR ASCII DATA SETS:

Explanation: A data set may have been referred to in the load module but had no corresponding DD statement, or the DD statement may have had an erroneous ddname.

If the extended error message facility is specified, the following information is provided:

FIOCS - MISSING DD CARD OR DCB ERROR FOR ASCII TAPE FOR X

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues and the input/output request is ignored.

Programmer Response: Probable user error. Either provide the missing DD statement or correct any erroneous ddname. Also be sure that the LABEL parameter on the DD statement specifies AL (or NL provided that the DCB subparameter OPTCD=Q is also specified). Also be sure that the operating system permits the use of ASCII data sets (that is, that the option ASCII=INCRES or ASCII=INCTRAN was specified at program installation time.) If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Have output from program installation available.

IHO220I _Explanation_: FIOCS or ASYNC -- A data set reference number exceeds the limit specified for data set reference numbers when the library was installed.

If the extended error message facility is specified, the following information is provided:

$\left\{\begin{array}{l}\text{ASYNC}\\ \text{DIOCS}\\ \text{FIOCS}\end{array}\right\}$ - UNIT NUMBER OUT OF RANGE. UNIT=X

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues and the input/output request is ignored.

Programmer Response: Probable user error. Correct the invalid data set reference number. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.
- Have output from program installation available.

IHO221I NAMEL-NAME LARGER THAN EIGHT CHARACTERS. NAME=X

Explanation: NAMEL -- An input variable name exceeds eight characters.

Supplemental Data: Name specified (first eight characters).

Standard Corrective Action: The remainder of the namelist request is ignored.

Programmer Response: Correct the invalid NAMELIST input variable, or provide any missing delimiters. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IHO222I NAMEL-NAME NOT IN NAMELIST DICTIONARY NAME=X

Explanation: NAMEL -- An input variable name is not in the NAMELIST dictionary, or an array is specified with an insufficient amount of data.

Supplemental Data: Name specified.

Standard Corrective Action: The remainder of the namelist request is ignored.

Programmer Response: Probable user error. Make sure that a correct NAMELIST statement is included in the source module for all variable and/or array names read in using NAMELIST. If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO223I NAMEL-END OF RECORD ENCOUNTERED BEFORE EQUAL SIGN. NAME=X

Explanation: NAMEL -- An input variable name or a subscript has no delimiter.

Supplemental Data: Name of item.

Standard Corrective Action: The remainder of the namelist request is ignored.

Programmer Response: Probable user error. Make sure that all NAMELIST input data is correctly specified and all delimiters are correctly positioned. Check all delimiters. Make sure that sequence numbers are not present in columns 73-80. If the problem recurs, do the following before calling IBM for programming support:
- Have source and associated listing available.

IHO224I NAMEL-SUBSCRIPT FOR NON-DIMEMSIONED VARIABLE OR SUBSCRIPT OUT OF RANGE. NAME=X

Explanation: NAMEL -- A subscript is encountered after an undimensioned input name, or the subscript is too big.

Supplemental Data: Name of item.

Standard Corrective Action: The remainder of the namelist request is ignored.

Programmer Response: Probable user error. Insert any missing DIMENSION statements, or correct the invalid array reference. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO225I CONVERT - ILLEGAL HEXADECIMAL CHARACTER X

Explanation: FCVTH -- An invalid character is encountered on input for the Z format code.

Supplemental Data: Display the record in which the character appeared.

Standard Corrective Action: 0 replaces the encountered character.

Programmer Response: Probable user error. Either correct the invalid character, or correct or delete the Z format code. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO226I CONVERT -- REAL NUMBER OUT OF RANGE X

Explanation: FCVTH -- A real number was too large to be processed by the load module. (The largest number that can be processed is 16^{63}-1.)

Supplemental Data: The field of input characters.

Standard Corrective Action: If the number was too large, the number is set to 16^{63}-1. If the number was too small, the number is set to zero.

Programmer Response: Make sure that all real input is within the required range for the number specified. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.

IHO227I LDFIO -- ERROR IN REPEAT COUNT

Explanation: A repeat count (k*---) was not followed by a blank, comma, or integer.

Supplemental Data: Contents of the buffer.

Standard Corrective Action: The remainder of the I/O list is ignored.

Programmer Response: Make sure that all repeat counts are followed by a valid character: a blank, a comma, or an integer. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.

IHO228I ASYNC -- LAST ELEMENT IN THE I/O LIST HAS A LOWER ADDRESS THAN THE FIRST ELEMENT. UNIT = X

Explanation: An I/O list contained an element having a lower storage address than the first element in the list.

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues and the input/output request is ignored.

Programmer Response: Probable user error. Make sure that all elements in the I/O list are specified in the correct order. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that LIST has been specified as a parameter on the EXEC statement.

| IHO229I LDFIO - INVALID CHARACTER '/' IN INPUT STREAM

Explanation: An end-of-data indicator ('/') was encountered before any data in the input record.

Supplemental Data: Contents of the buffer.

Standard Corrective Action: The remainder of the I/O list is ignored.

Programmer Response: Remove slash from the beginning of the data list. A slash should appear only when no further data is to be read in for a given READ statement. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.

IHO230I SOURCE ERROR AT ISN xxxx - EXECUTION FAILED [AT SUBROUTINE-name]

Explanation: IBERR -- During load module execution, a source statement error is encountered. The internal statement number for the source statement is xxxx; the routine that contains the statement is specified by "name".

Supplemental Data: None.

Standard Corrective Action: Execution is terminated.

Programmer Response: Make sure that all source module code is correct. Ensure that source statements that the compiler diagnosed have been corrected. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO231I Explanation: DIOCS -- Direct access input/output statements are used for a sequential data set, or input/output statements for a sequential data set are used for a direct access data set.

If the extended error message facility is specified, the following information is provided:

```
IBCOM - DIRECT ACCESS STATEMENT USED WITHOUT DEFINE FILE ON
        UNIT X
or

DIOCS - DIRECT ACCESS STATEMENT USED FOR SEQUENTIAL DATA SET X

cr

FIOCS - SEQUENTIAL I/O STATEMENTS USED FOR DIRECT ACCESS DATA
        SET X
```

Supplemental Data: Unit number.

Standard Corrective Action: The input/output request is ignored.

Programmer Response: Probable user error. Either include the necessary DEFINE FILE statement for direct access or delete the DEFINE FILE for a sequential data set. Make sure that all DD statements are correct. Verify that all data sets are referenced with valid FORTRAN statements for the data set type. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO232I Explanation: DIOCS -- Relative position of a record is not a positive integer, or the relative position exceeds the number of records in the data set.

If the extended error message facility is specified, the following information is provided:

```
DIOCS - RECORD NUMBER X OUT OF RANGE ON UNIT X
```

Supplemental Data: Unit number and record number.

Standard Corrective Action: The input/output request is ignored.

Programmer Response: Probable user error. Make sure that the relative position on the data set has been specified correctly. Check all DD statements. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO233I Explanation: DIOCS -- The record length specified in the DEFINE FILE statement exceeds the physical limitation of the volume assigned to the data set in the DD statement.

If the extended error message facility is specified, the following information is provided:

```
DIOCS - RECORD LENGTH GREATER THAN 32K-1 SPECIFIED FOR UNIT X
```

Supplemental Data: Unit number specified.

Standard Corrective Action: Record length is set to 32,000.

Programmer Response: Probable user error. Make sure that parameters of the DD statement conform to specifications in the DEFINE FILE statement; the record length in both must be equivalent and within the physical limitations of the assigned volume. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO234I Explanation: DIOCS -- The data set assigned to print execution error messages cannot be a direct access data set.

If the extended error message facility is specified, the following information is provided:

DIOCS - ATTEMPT TO DEFINE THE OBJECT ERROR UNIT AS DIRECT ACCESS DATA SET. UNIT=X

Supplemental Data: Unit number.

Standard Corrective Action: The DEFINE FILE entry is ignored.

Programmer Response: Probable user error. Make sure that the object error unit specified is not direct access. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO235I Explanation: DIOCS -- A data set reference number assigned to a direct access data set is used for a sequential data set.

If the extended error message facility is specified, the following information is provided:

DIOCS - DEFINE A DATA SET WHICH HAS BEEN USED SEQUENTIALLY AS A DIRECT ACCESS DATA SET. UNIT=X

Supplemental Data: Unit number.

Standard Corrective Action: The DEFINE FILE entry is ignored.

Programmer Response: Probable user error. Make sure that use of and/or reference to sequential data sets do not conflict with FORTRAN defined direct access data sets. Verify that device classes assigned by the installation do not conflict with the specification on the UNIT parameter of the DD statement. Make sure that the DEFINE FILE statement defines a direct access data set. Check all DD statements. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO236I Explanation: IBC -- A READ is executed for a direct access data set that has not been created.

If the extended error message facility is specified, the following information is provided:

DIOCS - READ REQUEST FOR AN UNCREATED DATA SET ON UNIT X

Supplemental Data: Unit number.

Standard Corrective Action: The input/output request is ignored.

Programmer Response: Probable user error. Make sure that either a data set utility program has been used, or appropriate parameters have been specified on the associated DD statement. Verify that, if a DD statement is used, DSNAME, UNIT, VOLUME, SPACE, LABEL, DISP, SYSOUT, and DCB are specified correctly or omitted where appropriate. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO237I Explanation: DIOCS -- length of record did not correspond to length of record specified in DEFINE FILE statement.

If the extended error message facility is specified, the following information is provided:

DIOCS - INCORRECT RECORD LENGTH SPECIFIED FOR UNIT X

Supplemental data: Unit for which error occurred.

Standard Corrective Action: The input/output request is ignored.

Programmer Response: Make sure that length of records supplied matches the length specified in the DEFINE FILE statement. If necessary, change the DEFINE FILE statement to specify the correct record length. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO238I LDFIO -- INCORRECT DELIMITER IN COMPLEX OR LITERAL INPUT

Explanation: A literal string in the input record(s) was not closed with a quotation mark (or was longer than 256 characters); alternatively, a complex number in the input record(s) contained embedded blanks, no internal comma, or no closing right parenthesis.

Supplemental Data: Contents of the buffer.

Standard Corrective Action: The remainder of the I/O list is ignored.

Programmer Response: Supply the missing quotation mark or amend the literal data to keep within the 256 character limit if the error was in the literal input. Check complex input numbers to see that they contain no embedded blanks, and that they contain an internal comma and a closing right parenthesis. If the problem recurs, do the following before calling IBM for programming support:

- Have source and associated listing available.

IHO240I STAE -- ABEND CODE IS: SYSTEM XXXX, USER YYYY.

$\text{IO} - \left\{\begin{array}{l}\text{NOT RESTORED}\\ \text{RESTORED}\\ \text{NONE}\end{array}\right\}$. SCB = xxxxxx. PSW IS xxxxxxxxxxxxxxxx.

Explanation: An abnormal termination occurred. In some instances, pointers to subroutine entry points may have been destroyed, causing the traceback map to be incomplete. If an incomplete traceback map is printed, the following additional text appears between message IHO240I and the traceback map:

TRACEBACK MAY NOT BEGIN WITH ABENDING ROUTINE.

Supplemental Data: XXXX is the completion code if a system code caused termination; YYYY is the completion code if a program code caused termination. If YYYY is the value 203, a READ/WRITE Implied DO statement had an invalid combination of initial, test, and increment values. The error handling option table indicated that more than one such error should be allowed before job termination; however, IBCOM# terminated the job with a user 203 completion code after the first occurrence of the invalid parameters.

For specific explanations of the completion codes, see the following publications:

For OS:
IBM System/360 Operating System Messages and Codes, Order No. GC28-6631

For VS:
OS/VS Message Library: VS1 System Codes, Order No. GC38-1003, or
OS/VS Message Library: VS2 System Codes, Order No. GC38-1008

For VM/370:
IBM Virtual Machine Facility/370: System Messages, Order No. GC20-1808.

The SCB field gives the address of the STAE Control Block, which contains the old PSW and the contents of general registers at the time of abnormal termination. For a description of the contents of the STAE Control Block, see the publication IBM System/360 Operating System: Supervisor and Data Management Services, Order No. GC28-6646. The PSW field gives the contents of the last FORTRAN program status word when abnormal termination occurred.

Input/output operations associated with the error are defined as NOT RESTORED, RESTORED, or NONE, as follows:

NOT RESTORED--Input/output has been halted and cannot be restored.

RESTORED--Input/output has been halted. FORTRAN will attempt to restart input/output and then close data sets.

NONE--No active input/output operations were present at abnormal termination time. FORTRAN will close data sets.

Standard Corrective Action: None.

Programmer Response: Probable user error. Use the abend code and the contents of the SCB and PSW to determine the nature of the error. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Make sure that the LIST, XREF, and MAP options were specified in the PARM field of the EXEC statement.
- Have source and associated listing available.

IHO241I FIXPI INTEGER BASE=0, INTEGER EXPONENT=X, LE 0

Explanation: LIB -- For an exponentiation operation (I**J) in the subprogram IHOFIXPI (FIXPI#) where I and J represent integer variables or integer constants, I is equal to zero and J is less than or equal to zero.

IHO245I FDXPD REAL*8 BASE=0.0, REAL*8 EXPONENT=X.X, LE 0

Explanation: LIB -- For an exponentiation operation (D**P) in the subprogram IHOFDXPD(FDXPD#), where D and P represent REAL*8 variables or REAL*8 constants, D is equal to zero and P is less than or equal to zero.

Supplemental Data: Exponent specified.

Standard Corrective Action: Result=0.

Programmer Response: Probable user error. Make sure that both the real variable or constant base and exponent for an exponentiation operation are within the allowable range. If the base and exponent may or will fall outside that range during program execution, then either modify the operand(s), or insert source code to test for the situation and make compensation appropriate to the program unit. Bypass the exponentiation operation if necessary. (See similar example for IHO241I.) If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

| IHO246I FCXPI COMPLEX*8 BASE=0.0+0.0I, INTEGER EXPONENT=X, LE 0

Explanation: LIB -- For an exponentiation operation (Z**J) in the subprogram IHOFCXPI(FCXPI#), where Z represents a COMPLEX*8 variable or COMPLEX*8 constant and J represents an integer variable or integer constant, Z is equal to zero and J is less than or equal to zero.

Supplemental Data: Exponent specified.

Standard Corrective Action:

If BASE=0,EXP<0,RESULT=•;
If BASE=0.0,EXP=0,RESULT=1

Programmer Response: Probable user error. Make sure that both the complex variable or constant base and the integer variable or constant exponent for an exponentiation operation are within the allowable range. If the base and exponent may or will fall outside that range during program execution, then either modify the operand(s), or insert source code to test for the situation and make the compensation appropriate to the program unit. Bypass the exponentiation operation if necessary. (See similar example for IHO241I). If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO247I FCDXI COMPLEX*16 BASE=0.0+0.0I, INTEGER EXPONENT=X, LE 0

Explanation: LIB -- For an exponentiation operation (Z**J) in the subprogram IHOFCDXI(FCDXI#), where Z represents a COMPLEX*16 variable or COMPLEX*16 constant and J represents an integer variable or integer constant, Z is equal to zero and J is less than or equal to zero.

Supplemental Data: Exponent specified.

Standard Corrective Action:

If BASE=0,EXP<0,RESULT=•;
If BASE=0.0,EXP=0,RESULT=1

Programmer Response: Probable user error. Make sure that both the complex variable or constant base and the integer variable or constant exponent for an exponentiation operation are within the allowable range. If the base and exponent may or will fall outside that range during program execution, then either modify the operand(s), or insert source code to test for the situation and make the compensation appropriate to the program unit. Bypass the exponentiation operation if necessary. (See similar example for IHO241I). If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO248I FQXPI# REAL*16 BASE=0.0, INTEGER EXPONENT=X, LE 0

Explanation: LIB -- For an exponentiation operation (D**J) in the subprogram IHOFQXPI(FQXPI#), where D represents a REAL*16 variable or constant and J represents an integer variable or constant, D is equal to zero and J is less than or equal to zero.

Supplemental Data: Exponent specified.

Standard Corrective Action:

If BASE=0,EXP<0,RESULT=•;
If BASE=0.0,EXP=0,RESULT=1

Programmer Response: Probable user error. Make sure that both the real variable or constant base and the integer variable or constant exponent for an exponentiation operation are within the allowable range. If the base and exponent may or will fall outside that range during execution, then either modify the operand(s),• or insert source code to test for the situation and make the compensation appropriate to the program unit. Bypass the exponentiation operation if necessary. (See similar example for IHO241I.) If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO249I FQXPQ# REAL*16 BASE=X.X, REAL*16 EXP=X.X, BASE=0.0, AND EXP LE 0 OR BASE LT 0 AND EXP NE 0

Explanation: LIB -- For an exponentiation operation (X**Y) in the subprogram IHOFQXPQ(FQXPQ#), where X and Y represent REAL*16 variables or constants, if X equals 0, Y must be greater than 0; if X is less than 0, Y must equal 0. One of these conditions is violated.

Supplemental Data: Exponent specified.

Standard Corrective Action:

If BASE=0 and EXP<0,RESULT=•;
If BASE=0.0 and EXP=0,RESULT=1
If BASE=<0.0 and EXP≠0,RESULT=|X|**y

Programmer Response: Probable user error. Make sure that both the real variable or constant base and exponent for an exponentiation operation are within the allowable range. If the base and exponent may or will fall outside that range during program execution, then either modify the operand(s), or insert source code to test for the situation and make compensation appropriate to the program unit. Bypass the exponentiation operation if necessary. (See similar example for IHO241I.) If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO250I FQXPQ# REAL*16 BASE=X.X, REAL*16 EXPONENT=Y.Y, ARGUMENT COMBINATION EXP.*LOG2(BASE) GE 252

Explanation: LIB -- For an exponentiation operation in the subprogram IHOFQXPQ, the argument combination of y*log2(x) generates a number equal to or greater than 252.

Supplemental Data: The arguments specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the base and exponent are within the allowable range. If necessary, restructure arithmetic operations. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, LIST, and XREF have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO251I SQRT NEGATIVE ARGUMENT=X.X

Explanation: LIB -- In the subprogram IHOSSQRT(SQRT), the argument is less than 0.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=$|X|^{1/2}$.

Programmer Response: Probable user error. Make sure that the argument is within the allowable range. Either modify the argument, or insert source code to test for a negative argument and make the necessary compensation. Bypass the function reference if necessary.

> Example: Assume ARG (REAL*4) is to be the input argument to SQRT. Then a simple test might appear:

```
      IF (ARG) 10,20,20
   10 ARG = ABS (ARG)
   20 ANS = SQRT (ARG)
```

If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement:
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO252I EXP ARG=X.X, GT 174.673

Explanation: LIB -- In the subprogram IHOSEXP(EXP), the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the exponential function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO253I ALOG-ALOG10 ARG=X.X, LE ZERO

Explanation: LIB -- In the subprogram IHOSLOG(ALOG and ALOG10), the argument is less than or equal to zero. Because this subprogram is called by an exponential subprogram, this message also indicates that an attempt has been made to raise a negative base to a real power.

Supplemental Data: Argument specified.

Standard Corrective Action:

If X=0, RESULT=-•;
If X<0, RESULT=log|X| or $\log_{10}$ |X|.

Programmer Response: Probable user error. Make sure that the argument to the logarithmic function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO254I SIN-COS/ARG/=/X.X(HEX=X)/, GE PI*2**18

Explanation: LIB -- In the subprogram IHOSSCN(SIN and COS), the absolute value of an argument is greater than or equal to $2^{18}*\pi$ ($2^{18}*\pi$ =.82354966406249996D+06).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=$\sqrt{2}/2$.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO255I ATAN2 ARGUMENTS=0.0

Explanation: LIB -- In the subprogram IHOSATN2, when entry name ATAN2 is used, both arguments are equal to zero.

Supplemental Data: Arguments specified.

Standard Corrective Action: Result=0.

Programmer Response: Probable user error. Make sure that both arguments do not become 0 during program execution, or are not inadvertently initialized or modified to 0. Provide code to test for the situation and, if necessary, modify the arguments or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO256I SINH-COSH/ARG/=/X.X/, GE 175.366

Explanation: LIB -- In the subprogram IHOSSCNH(SINH or COSH), the argument is greater than or equal to 175.366.

Supplemental Data: Argument specified.

Standard Corrective Action: SINH(X)=(the sign of x)•; COSH(X)=•

Programmer Response: Probable user error. Make sure that the argument to the hyperbolic sine or cosine function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO257I ARSIN-ARCOS/ARG/=/X.X/ GT 1

Explanation: LIB -- In the subprogram IHOSASCN (ARSIN or ARCOS), the absolute value of the argument is greater than 1.

Supplemental Data: Argument specified.

Standard Corrective Action:

If x>1.0,ARCOS(x)=0; if x<1.0,ARCOS(x)= π ;
If x>1.0,ARSIN= $\frac{\pi}{2}$; if x<1.0,ARSIN=$-\frac{\pi}{2}$

Programmer Response: Probable User Error. Make sure that the argument to the arcsine or arccosine function is between -1 and +1, inclusive. If the argument may or will fall outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO258I TAN-COTAN/ARG/=/X.X(HEX=X)/, GE PI*2**18

Explanation: LIB -- In the subprogram IHOSTNCT (TAN or COTAN), the absolute value of the argument is greater than or equal to $2^{18}*\pi$ ($2^{18}*\pi$ =.82354966406249996D+06).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=1.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian≈57.2957795131°) to the trigonometric tangent or cotangent function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO259I TAN-COTAN/ARG/=/X.X(HEX=X)/, APPROACHES SINGULARITY

Explanation: LIB -- In the subprogram IHOSTNCT (TAN or COTAN), the argument value is too close to one of the singularities ($\pm\frac{\pi}{2}$, $\pm\frac{3\pi}{2}$, ... for the tangent or $\pm\pi$, $\pm 2\pi$, ... for the cotangent).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian≈ 57.2957795131°) to the trigonometric tangent or cotangent function is within the allowable range. If the argument may or will approach the corresponding singularites for the function during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO260I FQXP2# REAL*16 EXPONENT=X.X, GE 252

Explanation: LIB -- In the subprogram IHOFQXPR(FQXP2#), the exponent is beyond the range of 2^{252}.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the exponent is within the allowable range. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, LIST, and XREF have been specified as parameters on the EXEC statement.
- Have source and associated listing available.

IHO261I DSQRT NEGATIVE ARGUMENT=X.X

Explanation: LIB -- In the subprogram IHOLSQRT(DSQRT), the argument is less than 0.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=$|X|^{1/2}$.

Programmer Response: Probable user error. Make sure that the argument is within the allowable range. Either modify the argument, or insert source code to test for a negative argument and make the necessary compensation. Bypass the function reference if necessary.

Example: Assume DARG (REAL*8) is to be the input argument to DSQRT. Then a simple test might appear:

```
    IF (DARG) 10,20,20
 10 DARG = DABS (DARG)
 20 ANS = DSQRT (DARG)
```

If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO262I DEXP ARG=X.X, GT 174.673

Explanation: LIB -- In the subprogram IHOLEXP(DEXP), the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the exponential function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO263I DLOG-DLOG10 ARG=X.X, LE 0

Explanation: LIB -- In the subprogram IHOLLOG(DLOG and DLOG10), the argument is less than or equal to zero. Because the subprogram is called by an exponential subprogram, this message also indicates that an attempt has been made to raise a negative base to a real power.

Supplemental Data: Argument specified.

Standard Corrective Action: If X=0, result=-•; if X<0, result=$\log|X|$ or $\log_{10}|X|$.

Programmer Response: Probable user error. Make sure that the argument to the logarithmic function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO264I DSIN-DCOS/ARG/=/X.X(HEX=X)/, GE PI*2**50

Explanation: LIB -- In the subprogram IHOLSCN(DSIN and DCOS), the absolute value of the argument is greater than or equal to $2^{50}*\pi$ ($2^{50}*\pi$ =.35371188737802239D+16).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=$\sqrt{2}/2$.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO265I DATAN2 ARGUMENTS=0.0

Explanation: LIB -- In subprogram IHOLATN2, when entry name DATAN2 is used, both arguments are equal to zero.

Supplemental Data: Arguments specified.

Standard Corrective Action: Result=0.

Programmer Response: Probable user error. Make sure that both arguments do not become zero during program execution, or are not inadvertently initialized or modified to zero. Provide code to test for the situation and, if necessary, modify the arguments or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO266I DSINH-DCOSH/ARG/=/X.X/, GE 175.366

Explanation: LIB -- In the subprogram IHOLSCNH (DSINH or DCOSH), the absolute value of the argument is greater than or equal to 175.366.

Supplemental Data: Argument specified.

Standard Corrective Action: DSINH(X)=(the sign of x)•; DCOSH(X)=•

Programmer Response: Probable user error. Make sure that the argument to the hyperbolic sine or cosine function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO267I DARSIN-DARCOS/ARG/=/X.X/, GT 1

Explanation: LIB -- In the subprogram IHOLASCN (DARSIN or DARCOS), the absolute value of the argument is greater than 1.

Supplemental Data: Argument specified.

Standard Corrective Action:

If x>1.0 DARCOS(x) = 0;
if x<1.0 DARCOS(x) = π;
if x>1.0 DARSIN = $\frac{\pi}{2}$;
if x<1.0 DARSIN = $-\frac{\pi}{2}$

Programmer Response: Probable user error. Make sure that the argument to the arcsine or arccosine function is between -1 and +1, inclusive. If the argument may or will fall outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO268I DTAN-DCOTAN/ARG/=/X.X(HEX=X)/, GE PI*2**50

Explanation: LIB -- In the subprogram IHOLTNCT (DTAN or DCOTAN), the absolute value of the argument is greater than or equal to $2^{50}*\pi$ ($2^{50}*\pi$ =.3537118876014220lD+16).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=1.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric tangent or cotangent function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO269I DTAN-DCOTAN/ARG/=/X.X(HEX=X)/, APPROACHES SINGULARITY

Explanation: LIB -- In the subprogram IHOLTNCT (DTAN or DCOTAN), the argument value is too close to one of the singularities ($\pm \frac{\pi}{2}$, $\pm \frac{3\pi}{2}$, ... for the tangent; $\pm \pi$, $\pm 2\pi$, ... for the cotangent).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric tangent or cotangent function is within the allowable range. If the argument may or will approach the corresponding singularities for the function during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO270I FCQXI# COMPLEX*32 BASE=0.0+0.0I, INTEGER EXPONENT=X, LE 0

Explanation: LIB -- In the subprogram IHOFCQXI(FCQXI#), a base 0 number has been raised to a power less than or equal to zero.

Supplemental Data: Argument specified.

Standard Corrective Action:

If X_1+X_2i=0.0+0.0i and J=0, Y=1+0.i
If X_1+X_2i=0+0.i and J<0, Y=•+0.i
(where J=exponent and Y=answer)

Programmer Response: Probable user error. Make the base a non-zero number or raise the exponent to a non-zero value. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP, LIST, and XREF have been specified as parameters in the EXEC statement.
- Have source and associated listing available.

IHO271I CEXP REAL ARG=X.X(HEX=X), GT 174.673

Explanation: LIB -- In the subprogram IHOCSEXP (CEXP), the value of the real part of the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•(COS X + iSIN X) where X is the imaginary portion of the argument.

Programmer Response: Probable user error. Make sure that the argument to the exponential function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO272I CEXP IMAG ARG=X(HEX=X), ABS VALUE GE PI*2**18

Explanation: LIB -- In the subprogram IHOCSEXP (CEXP), the absolute value of the imaginary part of the argument is greater than or equal to $2^{18*\pi}$ ($2^{18*\pi}$=.82354966406249996D+06).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=e^{x_1}+0*i.

Programmer Response: Probable user error. Make sure that the argument to the exponential function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation, and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO273I CLOG ARGUMENT=0.0+0.0I

Explanation: LIB -- In the subprogram IHOCSLOG (CLOG), the real and imaginary parts of the argument are equal to zero.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=-•+0i.

Programmer Response: Probable user error. Make sure that both the real and imaginary parts of the argument do not become zero during program execution, or are not inadvertently initialized or modified to zero. Provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO274I CSIN-CCOS/REAL ARG/=/X.X(HEX=X)/, GE PI*2**18

Explanation: LIB -- In the subprogram IHOCSSCN (CSIN or CCOS), the absolute value of the real part of the argument is greater than or equal to $2^{18}*\pi$ ($2^{18}*\pi$ =.8235496640624996D+06).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=COSH(x_2)+0*i; CSIN=0+SINH(x_2)*i.

Programmer Response: Probable user error. Make sure that the real part of the argument (in radians where 1 radian≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the real part of the argument may or will exceed the range during program execution, then provide code to test for the situation and, if necessary, modify the real part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO275I CSIN-CCOS/IMAG ARG=/X.X(HEX=X)/, GT 174.673

Explanation: LIB -- In the subprogram IHOCSSCN (CSIN or CCOS), the absolute value of the imaginary part of the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: If imaginary part >0, (X is real portion of argument):
1. For sine, result=•/2(SIN X + iCOS X).
2. For cosine, result=•/2(COS X - iSIN X).

If imaginary part <0, (X is real portion of argument):
1. For sine, result=•/2(SIN X - iCOS X).
2. For cosine, result=•/2(COS X + iSIN X).

Programmer Response: Probable user error. Make sure that the imaginary part of the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the imaginary part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the imaginary part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO276I CQEXP REAL ARG = X.X, GT 174.673

Explanation: LIB -- In the subprogram IHOCQEXP(CQEXP), the value of the real part of the argument is greather than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•(COS X + iSIN X) where X is the imaginary portion of the argument.

Programmer Response: Probable user error. Make sure that the real part of the argument to the exponential function is within the allowable range. If the real part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the real part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

| IHO277I CQEXP IMAG ARG = X.X, ABS VALUE GT PI*2**100

Explanation: LIB -- In the subprogram IHOCQEXP(CQEXP), the absolute value of the imaginary part of the argument is greater
| than $2^{100}*\pi$ ($2^{100}*\pi$ =.39824418129956973D+31)

Supplemental Data: Argument specified.

Standard Corrective Action: Result=e^{x_1}+0*i.

Programmer Response: Probable user error. Make sure that the imaginary part of the argument to the exponential function is within the allowable range. If the imaginary part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the imaginary part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO278I CQLOG ARGUMENT = 0.0+0.0I

Explanation: LIB -- In the subprogram IHOCQLOG(CQLOG), the real and imaginary parts of the argument are equal to zero.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=-•+0i.

Programmer Response: Probable user error. Make sure that both the real and imaginary parts of the argument do not become zero during program execution, or are not inadvertently initialized or modified to zero. Provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO279I CQSIN-CQCOS /REAL ARG/ = /X.X/, GE 2**100

Explanation: LIB -- In the subprogram IHOCQSCN(CQSIN or CQCOS), the absolute value of the real part of the argument is greater than (or equal to) 2^{100}.

Supplemental Data: Argument specified.

Standard Corrective Action: For CQSIN, Y=0+DSINH(X_2)*i; for CQCOS, Y=DCOSH(X_2)+0*i.

Programmer Response: Probable user error. Make sure that the real part of the argument (in radians where 1 radian≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the part of the argument may or will exceed the range during program execution, then provide code to test for the situation and, if necessary, modify the real part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO280I CQSIN-CQCOS /IMAG ARG/ = /X.X/, GT 174.673

Explanation: LIB -- In the subprogram IHOCQSCN(CQSIN or CQCOS), the absolute value of the imaginary part of the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: If imaginary part >0, (X is real portion of argument):
1. For sine, result=•/2(SIN X + iCOS X).
2. For cosine, result=•/2(COS X - iSIN X).

If imaginary part <0, (X is real portion of argument):
1. For sine, result=•/2(SIN X - iCOS X).
2. For cosine, result=•/2(COS X + iSIN X).

Programmer Response: Probable user error. Make sure that the imaginary part of the argument (in radians where 1 radian≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the imaginary part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the imaginary part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO281I CDEXP REAL ARG=X.X(HEX=X), GT 174.673

Explanation: LIB -- In the subprogram IHOCLEXP (CDEXP), the value of the real part of the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=*(COS X + iSIN X) where X is the imaginary portion of the argument.

Programmer Response: Probable user error. Make sure that the real part of the argument to the exponential function is within the allowable range. If the real part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the real part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO282I CDEXP IMAG ARG=X.X(HEX=X),ABS VALUE GE PI*2**50

Explanation: LIB -- In the subprogram IHOCLEXP (CDEXP), the absolute value of the imaginary part of the argument is greater than or equal to $2^{50} * \pi$
($2^{50} * \pi$ = .35371188760142201D+16).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=e^{x1}+0*i.

Programmer Response: Probable user error. Make sure that the imaginary part of the argument to the exponential function is within the allowable range. If the imaginary part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the imaginary part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO283I CDLOG ARGUMENT =0.D0+0.D0I

Explanation: LIB -- In the subprogram IHOCLLOG (CDLOG), the real and imaginary parts of the argument are equal to zero.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=-*+0i.

Programmer Response: Probable user error. Make sure that both the real and imaginary parts of the argument do not become zero during program execution, or are not inadvertently initialized or modified to zero. Provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO284I CDSIN-CDCOS/REAL ARG/ =/X.X(HEX=X)/, GE PI*2**50

Explanation: LIB -- In the subprogram IHOCLSCN (CDSIN or CDCOS), the absolute value of the real part of the argument is greater than or equal to $2^{50}*\pi$
(2^{50}* =.35371188760142201D+16).

Supplemental Data: Argument specified.

Standard Corrective Action: For CDSIN, Y=0+DSINH(X_2)+i; for CDCOS, Y=DCOSH(X_2)+0*i.

Programmer Response: Probable user error. Make sure that the real part of the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the part of the argument may or will exceed the range during program execution, then provide code to test for the situation and, if necessary, modify the real part of the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO285I CDSIN-CDCOS/IMAG ARG/=/X.X(HEX=X)/, GT 174.673

Explanation: LIB -- In the subprogram IHOCLSCN (CDSIN or CDCOS), the absolute value of the imaginary part of the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: If imaginary part >0, (X is real portion of argument):
1. For sine, result=•/2(SIN X + iCOS X).
2. For cosine, result=•/2(COS X - iSIN X).

If imaginary part <0, (X is real portion of argument):
1. For sine, result=•/2(SIN X - iCOS X).
2. For cosine, result=•/2(COS X + iSIN X).

Programmer Response: Probable user error. Make sure that the imaginary part of the argument (in radians where 1 radian≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the imaginary part of the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the imaginary part of the argument or bypass the source referencing the function subprogram. If the program recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO286I Explanation: ASYNC or FIOCS -- a data set that has been using one mode of input/output operations (that is, either synchronous or asynchronous) must be rewound before changing modes. An attempt was made to change the mode without rewinding the data set.

If the extended error message facility is specified, the following information is provided:

$\left\{\begin{array}{l}\text{ASYNC}\\\text{FIOCS}\end{array}\right\}$ - ATTEMPT TO ISSUE SYNCHRONOUS AND ASYNCHRONOUS I/O

REQUESTS WITHOUT INTERVENING REWIND. UNIT=X.

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues and the I/O request is ignored.

Programmer Response: Probable user error. Insert a REWIND statement at an appropriate point in the program. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST are specified in the PARM field of the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO287I ASYNC - A WAIT ISSUED WITH NO OUTSTANDING READ/WRITE REQUEST ON UNIT X

Explanation: ASYNC--A WAIT statement was issued with no corresponding READ or WRITE request.

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues and the WAIT statement is ignored.

Programmer Response: Probable user error. Remove the WAIT statement or include a corresponding READ or WRITE statement. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST are specified in the PARM field of the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO288I ASYNC - NO WAIT ISSUED FOR AN OUTSTANDING READ/WRITE REQUEST ON UNIT X

Explanation: ASYNC--No WAIT statement was issued for an outstanding READ or WRITE request.

Supplemental Data: Unit number.

Standard Corrective Action: Execution continues with an implied WAIT.

Programmer Response: Probable user error. Include the WAIT statement or remove the READ or WRITE statement. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST are specified in the PARM field of the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO289I QSQRT NEGATIVE ARGUMENT=X.X

Explanation: LIB -- In the subprogram IHOQSQRT, the argument is less than zero.

Supplemental Data: Argument specified.

Standard Corrective Action: Result = $|x|^{1/2}$.

Programmer Response: Probable user error. Make sure that the argument is within the allowable range. Either modify the argument, or insert source code to test for a negative argument and make the necessary compensation. Bypass the function reference if necessary.

Example: Assume QARG(REAL*16) is to be the input argument to QSQRT. Then a simple test might appear:

```
      IF (QARG) 10,20,20
   10 QARG = QABS(QARG)
   20 ANS = QSQRT(QARG)
```

If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO290I GAMMA ARG=X.X(HEX=X),LE 2**-252 OR GE 57.5744

Explanation: LIB -- In the subprogram IHOSGAMA (GAMMA), the value of the argument is outside the valid range ($2^{-252}<x<57.5744$).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the gamma function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO291I ALGAMA ARG=X.X(HEX=X),LE 0. OR GE 4.2937*10**73

Explanation: LIB -- In the subprogram IHOSGAMA (ALGAMA), the value of the argument is outside the valid range ($0<x<4.2937x10^{73}$).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the algama function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the necessary DD statement included.
- Have source and associated listing available.

IHO292I QEXP ARG = X.X, GT 174.673

Explanation: LIB -- in the subprogram IHOFQXPR (QEXP), the argument is greater than 174.673.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the exponential function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO293I QLOG-QLOG10 ARG = X.X, LE 0

Explanation: LIB -- In the subprogram IHOQLOG (QLOG and QLOG10), the argument is less than or equal to zero. Because the subprogram is called by an exponential subprogram, this message also indicates that an attempt has been made to raise a negative base to a real power.

Supplemental Data: Argument specified.

Standard Corrective Action: If X=0, result=-•; if X<0, result=log|X| or $\log_{10}$ |X|.

Programmer Response: Probable user error. Make sure that the argument to the logarithmic function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO294I QSIN-QCOS /ARG/=/X.X/, GE 2**100

Explanation: LIB -- In the subprogram IHOQSCN (QSIN and QCOS), the absolute value of the argument is greater than or equal to 2^{100}.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=$\sqrt{2}/2$

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian ≈ 57.2957795131°) to the trigonometric sine or cosine function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO295I QATAN2 ARGUMENTS = 0.0

Explanation: LIB -- In subprogram IHOQATN2, when entry name QATAN2 is used, both arguments are equal to zero.

Supplemental Data: Arguments specified.

Standard Corrective Action: Result=0.

Programmer Response: Probable user error error. Make sure that both arguments do not become zero during program execution, or are not inadvertently initialized or modified to zero. Provide code to test for the situation and, if necessary, modify the arguments or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO296I QSINH-SCOSH /ARG/=/X.X/, GE 175.366

Explanation: LIB -- In the subprogram IHOQSCNH (QSINH or QCOSH), the absolute value of the argument is greater than (or equal to) 175.366.

Supplemental Data: Argument specified.

Standard Corrective Action: QSINH(X)=(the sign of x)•;
QCOSH(X)=•

Programmer Response: Probable user error. Make sure that the argument to the hyperbolic sine or cosine function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO297I QARSIN-QARCOS /ARG/=/X.X/, GT 1

Explanation: LIB -- In the subprogram IHOQASCN (QARSIN or QARCOS), the absolute value of the argument is greater than 1.

Supplemental Data: Argument specified.

Standard Corrective Action:
If X>1.0 QARCOS(X)=0;
if X<1.0 QARCOS(X) $=\pi$;
if X>1.0 QARSIN $=\frac{\pi}{2}$;
if X<1.0 DARSIN $=-\frac{\pi}{2}$

Programmer Response: Probable user error. Make sure that the argument to the arcsine or arccosine function is between -1 and +1, inclusive. If the argument may or will fall outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO298I QTAN-QCOTAN /ARG/=/X.X/, GE 2**100

Explanation: LIB -- In the subprogram IHOQTNCT (QTAN or QCOTAN), the absolute value of the argument is greater than or equal to 2^{100}.

Supplemental Data: Argument specified.

Standard Corrective Action: Result=1.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian≈ 57.2957795131°) to the trigonometric tangent or cotangent function is within the allowable range. If the argument may or will exceed that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO299I QTAN-QCOTAN /ARG/=/X.X/, APPROACHES SINGULARITY

Explanation: LIB -- In the subprogram IHOQTNCT (QTAN or QCOTAN), the argument value is too close to one of the singularities ($\pm\frac{\pi}{2}$, $\pm\frac{3\pi}{2}$,... for the tangent; $\pm\pi$, $\pm 2\pi$,... for the cotangent).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument (in radians where 1 radian≈57.2957795131°) to the trigonometric tangent or cotangent function is within the allowable range. If the argument may or will approach the corresponding singularities for the function during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO300I DGAMMA ARG=X.X(HEX=X),LE 2**-252 OR GE 57.5744

Explanation: LIB -- In the subprogram IHOLGAMA (DGAMMA), the value of the argument is outside the valid range (2^{-252}<x<57.5744).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the DGAMMA function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO301I DLGAMA ARG=X.X(HEX=X), LE 0. OR GE 4.2937*10**73

Explanation: LIB -- In the subprogram IHOLGAMA (DLGAMA), the value of the argument is outside the valid range ($0<x<4.2937\text{x}10^{73}$).

Supplemental Data: Argument specified.

Standard Corrective Action: Result=•.

Programmer Response: Probable user error. Make sure that the argument to the DLGAMA function is within the allowable range. If the argument may or will be outside that range during program execution, then provide code to test for the situation and, if necessary, modify the argument or bypass the source referencing the function subprogram. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO900I EXECUTION TERMINATING DUE TO ERROR COUNT FOR ERROR NUMBER X

Explanation: This error has occurred frequently enough to reach the count specified as the number at which execution should be terminated.

(Condition Code - 16)

System Action: The job is terminated.

Programmer Response: Probable user error. Make sure that occurrences of the error number indicated are eliminated. For alternative action, see the Extended Error Handling Facility section in the publication IBM System/360 Operating System: FORTRAN IV (H Extended) Compiler Programmer's Guide, Order No. SC28-6852. If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Make sure that MSGLEVEL=(1,1) was specified on the JOB statement.
- Have source and associated listing available.

IHO901I EXECUTION TERMINATING DUE TO SECONDARY ENTRY TO ERROR MONITOR FOR ERROR X WHILE PROCESSING ERROR X

Explanation: In a user's corrective action routine, an error has occurred that has called the error monitor before it has returned from processing a diagnosed error.

(Condition Code - 16)

System Action: The job is terminated.

Note: If Traceback follows this message, it may be unreliable.

Programmer Response: Probable user error. Make sure that the error monitor is not called prior to processing the diagnosed error.

> Example: A statement such as R=A**B cannot be used in the exit routine for error 252, because FRXPR# uses EXP, which detects error 252.

If the problem recurs, do the following before calling IBM for programming support:

- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.

IHO902I ERROR NUMBER X OUT OF RANGE OF ERROR TABLE

Explanation: A request has been made to reference a non-existent Option Table entry.

System Action: The request is ignored and execution continues. IRETCD is set to 0.

Programmer Response: Probable user error. Make sure that the value assigned to an error condition is within the range of entries in the option table. If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.
- Have information from system generation time on the extended error handling facility available.

IHO903I ATTEMPT TO CHANGE UNMODIFIABLE TABLE ENTRY. NUMBER=X

Explanation: The Option Table specifies that no changes may be made in this entry, but a change request has been made by use of CALL ERRSET or CALL ERRSTR.

System Action: The request is ignored and execution continues.

Programmer Response: Probable user error. Make sure that no attempt has been made to alter dynamically an unmodifiable entry in the Option Table. If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.
- Have information from system generation time on the extended error handling facility available.

IHO904I ATTEMPT TO DO I/O DURING FIXUP ROUTINE FOR AN I/O TYPE ERROR

Explanation: When attempting to correct an input/output error, the user may not issue a READ, WRITE, BACKSPACE, ENDFILE, REWIND, PDUMP, DEBUG, or ERRTRA.

(Condition Code - 16)

System Action: The job is terminated.

Programmer Response: Probable user error. Make sure that, if an input/output error is detected, the user exit routine does not attempt to execute any FORTRAN input/output statement. If the problem recurs, do the following before calling IBM for programming support:
- Make sure that MAP and LIST have been specified as parameters on the EXEC statement.
- Make sure that the 'XREF' option is specified in the PARM field of the EXEC statement and that the necessary DD statement is included.
- Have source and associated listing available.
- Have information from system generation time on the extended error handling facility available.

OPERATOR MESSAGES

Operator messages for PAUSE and STOP statements may be generated during load module execution as follows:

yy IHO001A PAUSE x

Explanation: A FORTRAN PAUSE statement has been executed. The yy is an identification number assigned to the message. The x can be:

- An unsigned 1 to 5 digit integer constant specified in the PAUSE statement.
- A literal constant specified in the PAUSE statement.
- A zero to indicate that the PAUSE statement contained no constant.

System Action: The program enters the wait state.

Operator Response: Follow the instructions given by the programmer when he submitted the program for execution; his instructions should indicate the action to be taken for any constant printed in the message text or for a PAUSE statement without a constant.

To resume execution, press the REQUEST key. When the PROCEED light comes on, enter REPLY yy,'z' where yy is the message identification number and z is any letter or number. Then press the alternate coding key and enter a numeric 5.

IHO002A STOP x

Explanation: A FORTRAN STOP statement has been executed. The x can be an unsigned 1 to 5 digit integer constant, not a zero, specified in the STOP statement.

System Action: The STOP statement caused the program to terminate.

Operator Response: None.

INDEX

IBM Technical Newsletter

This Newsletter No. SN20-9192
Date January 14, 1977

Base Publication No. SC28-6865-1

Previous Newsletters None

IBM OS FORTRAN IV (H EXTENDED) COMPILER AND LIBRARY (MOD II) MESSAGES

© IBM Corp. 1971, 1974

This technical newsletter, a part of Release 2.1 of FORTRAN IV (H Extended) Compiler and FORTRAN IV Library (Mod II), program number, 5734-FO3 and 5734-LM3, provides replacement pages for the subject publication. These replacement pages remain in effect for subsequent releases unless specifically altered. Pages to be inserted and removed are:

Title page, Edition notice	61 - 64
Summary of Amendments, Contents	69, 70
7, 8	77, 78
37, 38	87 - 88.1 (88.1 added)
47 - 48.1 (48.1 added)	107 - 110

Each technical change is marked by a vertical bar to the left of the change.

Summary of Amendments

Changes included in this newsletter are summarized under "Summary of Amendments."

Note: Please file this cover letter at the back of the publication to provide a record of changes.

IBM Corporation, P.O. Box 50020, Programming Publishing, San Jose, California 95150

Printed in U.S.A.

READER'S COMMENTS

TITLE:	IBM OS FORTRAN IV (H Extended) Compiler and Library (Mod II) Messages	**FORM:**	SC28-6865-1

Your comments assist us in improving the usefulness of our publications; they are an important part of the input used in preparing updates to the publications. All comments and suggestions become the property of IBM.

Please do not use this form for technical questions about the system or for requests for additional publications; this only delays the response. Instead, direct your inquiries or requests to your IBM representative or to the IBM Branch Office serving your locality.

Corrections or clarifications needed:

Page *Comment*

Please include your name and address in the space below if you wish a reply.

Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

cut along this line

fold

fold

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

IBM CORPORATION
1271 Avenue of the Americas
New York, New York 10020

Attention: PUBLICATIONS

fold

fold

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
[U.S.A. only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]

SC28-6865-1

IBM OS FORTRAN (H Ext) Compiler & Mod II Lib Msgs Printed in U.S.A. SC28-6865-1

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)