Program Product

MVS/370
Linkage Editor and Loader
Messages

Data Facility Product 5665-295

Release 1.0

IBM

PREFACE

This publication lists the messages produced by the Linkage Editor and Loader programs. The causes of the messages are explained, the accompanying actions by the system are described, and the appropriate responses are suggested. The messages are presented in alphameric order for the Linkage Editor and for the Loader.

This manual is intended for use by application and system programmers.

Those publications that introduce concepts and terminology relevant to the linkage editor and loader messages are:

- OS/VS2 System Programming Library: Debugging Handbook, Volumes 1 through 3, GC28-1047 through GC28-1049

- OS/VS2 MVS JCL, GC28-0692

- MVS/370 Linkage Editor and Loader, GC26-4061

- MVS/370 Linkage Editor Logic, LY26-3921

- MVS/370 Loader Logic, LY26-3922

- Operator's Library: OS/VS2 Reference (JES2), GC38-0210

CONTENTS

The messages in this publication are produced by the linkage editor and loader programs. These messages are intended for application and system programmers, and are produced as diagnostic output listings in the SYSPRINT data set.

Message Format: IEWnnns xxx—text

where:

nnn
>Message serial number, which is coded to indicate the program:
>
>   0 nn   Linkage editor program
>
>   1 nn   Loader program

s
>Severity code (linkage editor program):
>
>0   Informational message: appears when control statement is printed as result of LIST option.
>
>1   Warning message: condition may cause error during execution of output module; output module is marked executable.
>
>2   Error message: condition may make execution of output module impossible; output module is marked not executable, unless LET option was specified; module processing is continued.
>
>3   Error message: condition will make execution of output module impossible; output module is marked not executable; module processing is continued.
>
>4   Error message: no recovery from error condition is possible; output module is not produced; module processing is terminated; only output is diagnostic messages.
>
>Severity code (loader program):
>
>0   Informational message: condition that will not cause an error during execution of the loaded module.
>
>1   Warning message: condition that may cause an error during execution of the loaded module.
>
>2   Error message: error that could make execution of the loaded module impossible. Processing continues.
>
>3   Error message: error that will make execution of the loaded module impossible. Processing continues.
>
>4   Error message: error condition from which no recovery is possible. Processing continues.
>
>I   Information: no operator action is required.

xxx
>WARNING or ERROR to indicate nature of message.

text
>Message text.

Associated Publication: Linkage Editor and Loader

Problem Determination: This is the activity required to identify a failing hardware unit or program, and to determine who is responsible for maintenance.

Problem determination is accomplished by using procedures specified by IBM. In some cases, these procedures may be initiated by a message or code that requires operator or programmer response. The response may include the requirement for additional problem-related data to be collected and will attempt, where possible, to indicate "probable" failure responsibility.

Problem determination information is included for applicable messages and codes under the heading "Problem Determination." Standard problem determination actions are identified as items of Tables I and II, which follow. Unique actions follow the list of standard actions to be taken. In any case, it is intended that the specified actions be taken before calling IBM for support.

PROBLEM DETERMINATION TABLE I

Note: If the problem recurs, follow the problem determination aids specified by the associated message or code before calling IBM for support.

1. Make sure that MSGLEVEL=(1,1) was specified in the JOB statement.

2. Save the console sheet from the primary console. In systems with remote consoles, save the remote console sheet. In systems with multiple console support (MCS), save a copy of the hardcopy log.

3. Save the job stream associated with the job.

4. Save the system output (SYSOUT) associated with the job.

5. Make sure that the failing job step includes a:

   a. SYSABEND DD statement

   b. SYSUDUMP DD statement

   c. PL1DUMP DD statement

   d. SYSMDUMP DD statement

6. Make sure that the PARM parameter of the EXEC statement specifies the following:

   a. MAP

   b. LIST

   c. DIAG

   d. MSG=AP

   e. CORE, if applicable

   f. XREF

   g. DUMP

7. If SMP is used to make all changes to the system, execute the LIST CDS and, if SMP3, LIST PTFBY, if SMP4, LIST CDS SYSMOD and LIST PTS SYS, functions of SMP to obtain a list of the current maintenance from the SMP control data set (CDS). If SMP4 and LIST PTS SYS indicate the LKEDPRINT parameter is used, make certain that the substitute name for SYSPRINT ddname statement is included.

If any changes are made to the system without using SMP, execute the LISTIDR function of the service aid program, AMBLIST, to obtain a list of all members with a program temporary fix (PTF) or local fix, and save the output. Execute the program against the:

a. SYS1.LINKLIB data set

b. SYS1.SVCLIB data set

c. Library containing the program that issued the message

8. Execute the service aid program, IMCOSJQD, to obtain a formatted copy of the contents of the SYS1.SYSJOBQE or SYS1.SYSWADS data sets, SWADS, or the resident job list.

9. Execute the service aid program, AMBLIST, to obtain:

a. An object module listing, specifying the LISTOBJ function

b. A load module map and cross-reference listing, specifying the OUTPUT=BOTH option of the LISTLOAD function

10. Have a copy of the message control program (MCP) available.

11. Execute the service aid program, AMDSADMP, to dump the contents of real storage and page data sets on magnetic tape.

After restarting the system, execute the GO function of the service aid program, AMBPRDMP, to print the real storage portion of the dump tape produced by AMDSADMP.

Save the tape from AMDSADMP (should further information from the tape be required) and the listing from AMBPRDMP.

12. Execute the SEREP program, and save the resulting output.

13. Save all the associated output.

14. The normal response to this message requests the programmer/operator to execute a specific program. Save all the output from that program.

15. Save the program listing associated with the job.

16. Save the dump.

17. Have the system generation (SYSGEN) output available from:

a. Stage I

b. Stage II

18. Execute the IFCEREP1 service aid program, specifying PARM=(N), to dump the SYS1.LOGREC data set. Save the resulting output.

For MSS, execute the following program to dump the SYS1.LOGREC data set:

a. Service aid IFCISDAO

b. Program ISDASDAO with the DETAIL(ALL) parameter

19. Save the assembly listing associated with the job.

20. Save the control cards associated with the job.

21. Save the compiler output associated with the job.

22. Save the source input associated with the job.

23. Save the source program listing associated with the job.

24. Run OLTEP diagnostics for the problem device and save the output.

25. Execute the IEHLIST system utility program to obtain a list of the:

    a. Volume table of contents of the associated volume, specifying the FORMAT option

    b. Volume table of contents of the associated volume, specifying the DUMP option

    c. Directory of the associated data set

    d. System catalog

26. Execute the IEBPTPCH data set utility to print the:

    a. Directory of the applicable data set

    b. Applicable data set

    c. Applicable member

    d. Applicable procedure

27. Have the linkage editor/loader map available.

28. Save the associated volume.

29. Contact IBM for programming support.

PROBLEM DETERMINATION TABLE II

GTF for Problem Determination

Format 1: Tracing without prompting for event keywords.

Before reproducing the problem, have the system operator issue a START GTF command specifying tape output, MODE=EXT and TIME=YES. In response to message AHL100A, the operator should type TRACE=opt, where opt is the trace option keyword indicated for the particular message or code, within the text of the reply.

When data for the problem has been recorded, run the service aid program, AMDPRDMP, using the EDIT statement to format the trace output, specifying DDNAME=(ddname of the trace data set).

Format 2: Tracing with prompting for event keywords.

Before reproducing the problem, have the system operator issue a START GTF command specifying tape output, MODE=EXT and TIME=YES. In response to the message AHL100A, the system operator should specify the trace option keywords indicated for the associated message or code within the text of the reply. Then, in response to the message AHL101A, the operator should specify the event keywords also indicated with the associated message or code.

When data for the problem has been recorded, run the service aid program, AMDPRDMP, using the EDIT statement to format the trace output, specifying DDNAME=(ddname of the trace data set).

Format 3: Specialized tracing action.

Before reproducing the problem, have the system operator issue a
START GTF command specifying tape output, MODE=EXT and TIME=YES.
In response to message AHL100A, the operator should type
'TRACE=SYS,USR'. The DD statement for a data set in error
should specify DCB=DIAGNS=TRACE.

When data for the problem has been recorded, execute the EDIT
function of AMDPRDMP, specifying the options SYS and USR=DMA.

IEW0000 (control statement)

Explanation: The control statement is printed as a result of the LIST option.

IEW0012 ERROR—INPUT CONTAINS INVALID TWO-BYTE RELOCATABLE ADDRESS CONSTANT, CONSTANT HAS NOT BEEN RELOCATED.

Explanation: A relocatable A-type or V-type address constant of less than 3 bytes has been found in the input.

System Action: The constant is not relocated.

Programmer Response: Probable user error. Check assembler language input for V-type address constants, which cannot be relocated. Delete or correct the invalid address constant.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0022 ERROR—INPUT CONTAINS INVALID V-TYPE ADDRESS CONSTANT, CONSTANT HAS NOT BEEN RELOCATED.

Explanation: A V-type address constant of less than 4 bytes has been found in the overlay structure.

System Action: The constant is not relocated.

Programmer Response: Probable user error. Either (1) specify a length of 4 bytes for all V-type address constants; or (2) if a 3-byte V-type address constant refers to a symbol within its overlay segment, you can assemble it as an A-type address constant with an EXTRN statement. One method of isolating an invalid address constant is (1) link-edit with OVLY and XREF options specified; (2) link-edit again without the OVLY option; and (3) compare the external reference lists. Any reference appearing in the second run and not in the first is invalid in an overlay structure.

Problem Determination: Table I, items 6b, 6f, 22, 23, 29. Have the output used to isolate the address constant available.

IEW0033 ERROR—INVALID ENTRY POINT FROM END CARD, NO ENTRY POINT ASSIGNED.

Explanation: The entry point for the program was specified as a relative address in an END card. The entry point

that was specified appeared to be valid when the END card was processed; however, the entry point was found to be invalid when the entry point of the load module was being determined.

System Action: No entry point is assigned.

Programmer Response: Check object module input for completeness. Then, either specify an entry point name on the ENTRY control statement; or, if entry points were specified at compilation or assembly, make sure the object module containing the desired entry point precedes all other object modules with assembled or compiled entry points.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0043 ERROR—INPUT CONTAINS INVALID EXTERNAL SYMBOL ID.

Explanation: END card is probably mispunched.

System Action: The invalid item is ignored.

Programmer Response: Probable user error. Check the input object modules for completeness and proper sequence. If necessary, either (1) re-create any module that has been in card form, or (2) isolate the incorrect module by executing the linkage editor with the NCAL option specified, using the NAME control statement for each input module. Diagnostic IEW0043 should recur and isolate the incorrect module. Re-create the module, and rerun the step.

Problem Determination: Table I, items 6b, 6f, 9a, 9b, 29. Have available the output used to isolate the module as described above.

IEW0053 ERROR—ENTRY STATEMENT SYMBOL PRINTED IS INVALID (NOT AN EXTERNAL NAME), NO ENTRY POINT ASSIGNED.

Explanation: The symbolic entry point specified in an ENTRY statement is not a control section or an entry name.

System Action: No entry point is assigned.

Programmer Response: Probable user error. Correct the ENTRY control statement, or make sure that the control

section containing the entry point is included in the input and has not been accidentally deleted or redefined by a REPLACE or CHANGE control statement.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0063   ERROR—END CARD SYMBOL PRINTED IS INVALID (NOT AN EXTERNAL NAME), NO ENTRY POINT ASSIGNED.

Explanation: The symbolic entry point specified in an END statement is not a control section or an entry name.

System Action: No entry point is assigned.

Programmer Response: Check that the entry point control section or entry name has not been accidentally deleted or redefined by a REPLACE or CHANGE control statement. Check the module containing the entry point for completeness.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0073   ERROR—ENTRY STATEMENT SYMBOL PRINTED IS NOT IN ROOT SEGMENT OF OVERLAY STRUCTURE, NO ENTRY POINT ASSIGNED.

Explanation: The entry point specified is in a segment other than the root segment. Either (1) the module containing the entry point was placed in a segment other than the root segment by means of the INSERT statement, or (2) the entry point is incorrectly specified on the ENTRY statement.

System Action: No entry point is assigned.

Programmer Response: Probable user error. Either correct the ENTRY control statement, or move the module containing the entry point to the root segment.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0083   ERROR—END CARD SYMBOL PRINTED IS NOT IN ROOT SEGMENT OF OVERLAY STRUCTURE, NO ENTRY POINT ASSIGNED.

Explanation: The entry point is in a segment other than the root segment. Either (1) the INSERT statement was used to place the control section containing the entry point in another segment, or (2) the symbol specified on the END statement is incorrect.

System Action: No entry point is assigned.

Programmer Response: Probable user error. Move the object module containing the entry point to the root segment, or specify an entry point in the root segment using the ENTRY control statement.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0093   ERROR—END CARD ENTRY POINT ADDRESS PRINTED IS NOT IN ROOT SEGMENT OF OVERLAY STRUCTURE, NO ENTRY POINT ASSIGNED.

Explanation: The entry point is in a segment other than the root segment. Either (1) the INSERT statement was used to place the control section containing the entry point in another segment, or (2) the address specified on the END statement is incorrect.

System Action: No entry point is assigned.

Programmer Response: Probable user error. Move the object module containing the entry point to the root segment, or specify an entry point in the root segment using the ENTRY control statement.

Problem Determination: Table I, items 6b, 6f, 9a, 13, 29. Save the object module input.

IEW0102   ERROR—INVALID ENTRY POINT ON END CARD, ENTRY, POINT IGNORED.

Explanation: A possible entry point for the program was specified as a relative address in an END card. When the END card was processed, the control section identification of the specified entry point was found to be invalid.

System Action: The entry point is ignored. The first valid entry point encountered is used; if there is none, no entry point is assigned.

Programmer Response: Probable user error. Check the input object modules for completeness and proper sequence. If necessary, either (1) re-create any module which has been in card form, or (2) isolate the incorrect module by executing the linkage editor with the NCAL option specified, using the NAME control statement for each input object module. Diagnostic IEW0102 should recur and isolate the incorrect module. Re-create the module, and rerun the step.

Problem Determination: Table I, items 3, 6b, 6f, 9a, 13, 29. Have available the output used to isolate the module.

IEW0113 ERROR—OUTPUT MODULE CONTAINS NO CONTROL SECTIONS IN ROOT SEGMENT OF OVERLAY STRUCTURE, NO ENTRY POINT ASSIGNED.

Explanation: There are no control sections in the root segment. Either (1) all control sections originally in the root segment have been deleted, or (2) there were no control sections originally in the root segment, or (3) an OVERLAY statement preceded the input.

System Action: No entry point is assigned.

Programmer Response: Probable user error. Place at least one control section in the root segment.

Problem Determination: Table I, items 6b, 6f, 9b, 29. Save the root segment module and its associated listings.

IEW0123 ERROR—NO ESD ENTRIES, EXECUTION IMPOSSIBLE.

Explanation: There are no external symbol dictionary entries. There are no control sections in the output.

System Action: Processing is terminated.

Programmer Response: Probable user error. Check other messages issued for cause of error (that is, invalid input from object module). Ensure that at least one control section appears in the input and is not deleted by the REPLACE control statement.

Problem Determination: Table I, items 1, 3, 6b, 6f, 9a, 13, 22, 23, 29.

IEW0132 ERROR—SYMBOL PRINTED IS AN UNRESOLVED EXTERNAL REFERENCE.

Explanation: An external reference is unresolved at the end of input processing. None of the following is specified: restricted no-call, never-call, or NCAL.

System Action: The module is marked not executable unless LET is specified.

Programmer Response: Probable user error. Check that the reference is valid and not the result of a keypunch or programming error. If the reference is valid, add the needed module or alias to one of the input data sets. Make sure the SYSLIB data set DD statement has been specified, if needed. If resolution is not desired, specify NCAL, never-call, or restricted no-call. If the reference was found in a control section replaced by another control section not containing that same reference, delete the reference, or specify NCAL, never-call, or restricted no-call.

Problem Determination: Table I, items 1, 6b, 6f, 9b, 13, 22, 23, 25c, 29.

IEW0143 ERROR—NO TEXT.

Explanation: No text remains in the output module. Either (1) all the control sections originally in the input are deleted, or (2) there are no control sections that originally contained text.

System Action: Processing is terminated.

Programmer Response: Probable user error. Check other messages issued for cause of error (that is, invalid input from object module). Ensure that at least one control section contains text and is not deleted by the REPLACE control statement or by automatic replacement.

Problem Determination: Table I, items 1, 6b, 6f, 9a, 9b, 13, 29. Save a module containing text.

IEW0152 ERROR—INVALID OVERLAY STRUCTURE, NO CALLS OR BRANCHES MADE FROM ROOT SEGMENT.

Explanation: There are no calls or branches from the root segment to a segment lower in the tree structure. Other segments cannot be loaded.

System Action: The module is marked not executable unless LET is specified.

Programmer Response: Probable user error. Make sure the root segment contains a control section that refers to at least one other segment in the overlay structure by means of a V-type address constant.

Problem Determination: Table I, items 1, 6b, 6f, 9b, 13, 29. Have a root segment module that calls another segment available with its associated listing.

IEW0161 WARNING—EXCLUSIVE CALL FROM SEGMENT NUMBER PRINTED TO SYMBOL PRINTED—XCAL WAS SPECIFIED.

Explanation: There is a valid exclusive branch-type reference; The XCAL option is specified for this job step.

System Action: Processing continues.

Programmer Response: Normally, no response is necessary. You can check that the printed branch-type references between exclusive segments are correct according to your overlay structure.

Problem Determination: If you suspect that the message fails to appear when it should, or appears incorrectly, see

Table I, items 6b, 6f, 9b, 29. Have
modules that contain the calls and
symbol available with associated source
listings.

IEW0172   ERROR—EXCLUSIVE CALL FROM
              SEGMENT NUMBER PRINTED TO
              SYMBOL PRINTED.

Explanation: A valid branch-type
reference is made from a segment to an
exclusive segment: The XCAL option is
not specified.

System Action: The module is marked not
executable unless the LET option is
specified.

Programmer Response: Probable user
error. Either (1) rearrange the overlay
structure to place both segments in the
same path, or (2) specify the XCAL
option.

Problem Determination: Table I, items
6b, 6f, 9b, 13, 29. Have the modules
containing the symbol and the calls to
it available with associated listings.

IEW0182   ERROR—INVALID EXCLUSIVE CALL
              FROM SEGMENT NUMBER PRINTED TO
              SYMBOL PRINTED.

Explanation: There is an invalid
exclusive branch-type reference from a
segment to a symbol in an exclusive
segment.

System Action: The module is marked not
executable unless the LET option is
specified.

Programmer Response: Probable user
error. Either (1) place the segments in
the same path, or (2) place a V-type
address constant in a common segment.

Problem Determination: Table I, items
6b, 6f, 9b, 13, 29. Have the modules
containing the symbol and the calls to
it available with associated listings.

IEW0201   WARNING—OVERLAY STRUCTURE
              CONTAINS ONLY ONE
              SEGMENT—OVERLAY OPTION
              CANCELLED.

Explanation: There are no OVERLAY
statements in the input.

System Action: The overlay option is
canceled.

Programmer Response: Probable user
error. Either place OVERLAY statements
in the input, or remove the OVLY option
from the EXEC statement.

Problem Determination: Table I, items 1,
3, 6b, 6f, 9b, 13, 29.

IEW0212   ERROR—EXPECTED CONTINUATION
              CARD NOT FOUND.

Explanation: A linkage editor control
statement specifying a continuation
(nonblank in column 72) is not followed
by a continuation card.

System Action: The card is not processed
as a continuation, but as normal input.

Programmer Response: Probable user
error. Either remove the nonblank
character in column 72 or insert the
necessary continuation record.

Problem Determination: Table I, items
6b, 6f, 13, 29.

IEW0222   ERROR—CARD PRINTED CONTAINS
              INVALID INPUT FROM OBJECT
              MODULE.

Explanation: One of the following
occurred during the processing of an
object module:

*   A record of invalid type was
    encountered.

*   A text (TXT) record was encountered
    in which the data length (columns
    11-12) is invalid or mispunched.

*   An invalid, probably mispunched
    relocation dictionary (RLD) record
    was encountered in an object module.

System Action: The record in error is
ignored, and processing continues.

Programmer Response: (1) Remove all
extraneous records from the input to the
linkage editor. (2) Ensure that linkage
editor control statements are placed
either before or after object modules.
(3) Ensure that all records in the
object module have a 12-2-9 punch in
column 1. (4) Ensure that all records
in the object module contain one of the
following in columns 2-4: ESD, SYM, TXT,
RLD, or END. (5) Locate the TXT or RLD
record having the invalid or mispunched
data; regenerate the object module, or
investigate the punching device or
generating processor for malfunctions.

Problem Determination: Table I, items
6b, 6f, 22, 29.

IEW0234   ERROR—INPUT FROM LOAD MODULE
              IS INVALID.

Explanation: In the order of
probability, either (1) a text record
was encountered in an input load module
that is larger than the load module
buffer; (2) the member being read does
not contain a valid load module; or (3)
the region size specified was too small
to contain the largest load module.

System Action: The erroneous record is
ignored and processing continues. The
output load module is marked not
executable. In case (3), the operation
is terminated.

Programmer Response: In case (1),
specify value2 of the SIZE parameter as
a number equal to or greater than the
size of the largest text record in any
input load module and rerun the job
step. Value2 must be equal to or
greater than one-half the value2
specified in the link-edit of any input
load module. In case (2), check that
all input data sets are specified
correctly on the DD statements. If load
module input occurs in the primary
input, rerun the job step with the NCAL
option specified. If error message
IEW0234 recurs, the incorrect load
module is in primary input; otherwise,
it is in SYSLIB input. Isolate the
incorrect load module by executing the
linkage editor with INCLUDE and NAME
statements for each suspected load
module. When the incorrect load module
is isolated, re-create it and rerun the
job step. In case (3), correct the
region size, and resubmit the job.

If the problem recurs, do the following
before calling IBM for programming
support:

• If an incorrect load module was
  created, execute the service aid
  program, AMBLIST, using the
  OUTPUT=MODLIST option of the
  LISTLOAD function, and save the
  resultant listing.

• Make sure that the XREF and LIST
  options were specified for the
  failing job step.

Problem Determination: Table I, items
6b, 6f, 13, 29.

**IEW0241   WARNING—EXTERNAL SYMBOL
            PRINTED IS DOUBLY DEFINED—ESD
            TYPE DEFINITIONS CONFLICT.**

Explanation: Two identical external
names have been found in the input. (1)
The invalid match involves a label
reference (LR) or label definition (LD)
matching an existing section definition
(SD), common (CM), or label reference
(LR). The section definition for the
input LR or LD must be marked delete in
order for this not to be an error. (2)
It is always invalid for a CM to match
an existing LR.

System Action: References to the name
are resolved with respect to the first
occurrence of the name.

Programmer Response: Probable user
error. Correct the existing symbol
conflict. To isolate the problem,
execute the following functions of the
service aid program AMBLIST. Load
module symbols can be printed using the
LISTLOAD function, specifying the
OUTPUT=XREF option. Object module
symbols can be printed using the LISTOBJ
function of the service aid program.

Problem Determination: Table I, items
6b, 6f, 14, 22, 29.

**IEW0254   ERROR—TABLE OVERFLOW—TOO
            MANY EXTERNAL SYMBOLS IN ESD.**

Explanation: There are too many external
symbols or control statement operands in
the problem program.

System Action: Processing is terminated.

Programmer Response: Probable user
error. Check that no unnecessary
modules or control statements are
included in the input. Then, either (1)
increase the linkage editor's table
space by increasing value1 (or
decreasing value2) of the SIZE
parameter, making sure the region or
partition size is also increased, if
necessary; or (2) reduce the number of
external symbols in the input (control
sections, entry points, and named common
areas).

Problem Determination: Table I, items
6b, 6f, 29.

**IEW0264   ERROR—TABLE OVERFLOW—INPUT
            LOAD MODULE CONTAINS TOO MANY
            EXTERNAL SYMBOLS IN ESD.**

Explanation: Either (1) an input module
contains too many external symbols in
the ESD, or (2) an ESD card is
mispunched.

System Action: Processing is terminated.

Programmer Response: Probable user
error. Check that input object modules
are complete and not mispunched. Then
either (1) break down any large input
module into a number of smaller modules,
or (2) increase the linkage editor's
table space by increasing value1 (or
decreasing value2) of the SIZE parameter
making sure the region or partition size
is also increased, if necessary.

Problem Determination: Table I, items
6b, 6f, 9a, 29.

**IEW0272   ERROR—LOAD MODULE FROM
            LIBRARY SPECIFIED
            UNACCEPTABLE.**

Explanation: When the load module was
created, it was marked not editable.

System Action: The load module was not
accepted as input.

Programmer Response: Probable user
error. If the module is unacceptable
because it is marked not editable, it
must be re-created before it can be
input to the linkage editor. Execute
the IEHLIST utility program, using the
LISTPDS function with the FORMAT option,
to print out the module's directory
entry and show the not editable
indicator.

Problem Determination: Table I, items 6b, 6f, 29.

IEW0284   ERROR—DDNAME PRINTED CANNOT BE OPENED.

Explanation: The specified data set cannot be opened. The DD statement defining the data set is missing.

System Action: Processing is terminated.

Programmer Response: Probable user error. Either (1) supply the missing DD statement, or (2) correct erroneous information on the DD statement. If the linkage editor was invoked by a macro instruction such as LINK rather than through the EXEC statement, make sure the ddname list, if passed, was correct.

Problem Determination: Table I, items 1, 3, 13, 25b, 29.   Table II, format 3.

IEW0294   ERROR—DDNAME PRINTED HAD SYNCHRONOUS ERROR.

Explanation: Either (1) a physical uncorrectable I/O error occurred, or (2) an object module is missing an END card as the last card, or (3) if the data definition name that was printed is for a DD statement that defines a blocked input data set of fixed format, an input record larger than the specified block size or logical record length was found, or (4) the data set may be too full for STOW to write an EOF mark.

System Action: Processing is terminated. The data definition name in the name field of the DD statement for the input data set was printed after the message code. If an input/output error occurred, the information provided by the SYNADAF macro instruction was printed after the message code in the following format: SYNAD EXIT, jobname, stepname, unit address, device type, ddname, operation attempted, error description, block count or BBCCHHR, and access method.

Programmer Response: For any fixed format, specify the correct block size. If the block size was correct and the data set was an input data set, re-create or restore the data set. For condition (4), compress the data set, and rerun the job.

Problem Determination: Table I, items 1, 3, 13, 25b, 29.

IEW0302   ERROR—INVALID STATEMENT—SCAN TERMINATED.

Explanation: Either (1) there is an error on a linkage editor control statement, or (2) an OVERLAY control statement was encountered and the OVLY attribute was not specified on the EXEC statement.

System Action: A statement in error is accepted as input up to the point of the error; the OVERLAY statements are ignored, and the module is not in overlay format.

Programmer Response: Probable user error. Either (1) correct the error, if necessary, or (2) specify OVLY on the EXEC statement.

Problem Determination: Table I, items 1, 6b, 6f, 3, 13, 29.

IEW0314   ERROR—MAXIMUM NUMBER OF REGIONS (4) EXCEEDED.

Explanation: There are 5 or more regions specified in this overlay structure.

System Action: Processing is terminated.

Programmer Response: Probable user error. Reduce the number of regions in the overlay structure to 4.

Problem Determination: Table I, items 6b, 6f, 29.

IEW0324   ERROR—MAXIMUM NUMBER OF SEGMENTS EXCEEDED.

Explanation: The number of segments exceeded 255.

System Action: Processing is terminated.

Programmer Response: Reduce the number of segments in the overlay structure to 255, or less.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0332   ERROR—MAXIMUM NUMBER OF ALIASES (16) EXCEEDED, EXCESS IGNORED.

Explanation: More than 16 aliases were specified for the output load module.

System Action: The excess aliases are ignored.

Programmer Response: Probable user error. Either (1) reduce the number of aliases, or (2) create a second copy of the module under a different name with the additional aliases specified.

Problem Determination: Table I, items 6b, 25b, 29.

IEW0342   ERROR—LIBRARY SPECIFIED DOES NOT CONTAIN MODULE.

Explanation: STOW returned a nonzero return code for one of the following reasons.

1.   The library specified on an INCLUDE or LIBRARY statement does not contain the module.

2. Permanent I/O error encountered while searching the directory.

3. Insufficient virtual storage available.

System Action: Any references to the module are not resolved. The output load module is marked not executable unless the LET option has been specified.

Programmer Response: Probable user error. Correct the library or module name on the DD, INCLUDE, or LIBRARY control statement.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 25b, 29.

IEW0354 ERROR—TABLE OVERFLOW—TOO MANY CALLS BETWEEN CONTROL SECTIONS.

Explanation: There are too many V-type address constants referring to external symbols in a program that is structured in overlay. The table recording these V-type address constants has overflowed.

System Action: Processing is terminated.

Programmer Response: Probable user error. Either (1) increase the linkage editor's table space by increasing value1 (or decreasing value2) of the SIZE parameter, making sure the region or partition size is also increased, if necessary; or (2) reduce the number of V-type address constants by combining control sections; or (3) change V-type address constants that do not refer across segments to A-type address constants with EXTRN statements.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0364 ERROR—TABLE OVERFLOW—INPUT TEXT EXCEEDED MAXIMUM OR TOO MANY CHANGES OF ORIGIN IN INPUT.

Explanation: Either (1) the text of a single CSECT exceeds 512 times the block size for the SYSUT1 or SYSLMOD data set, or (2) the internal tables used by the linkage editor to account for the text of a load module have overflowed because of discontinuities in the input text or the division of the text into pieces equal to the block size for the SYSLMOD data set.

System Action: Processing is terminated.

Programmer Response: Probable user error.

1. In case (1), ensure that the record length for the output load module is greater than the length of the longest input CSECT divided by 512.

2. In case (2), perform one of the following:

   a. Increase the linkage editor's table space by increasing value1 (or decreasing value2) of the SIZE parameter, making sure the region or partition size is also increased if necessary.

   b. Increase the linkage editor's buffer space by increasing both value1 and value2 of the SIZE parameter, making sure the region or partition size is increased proportionally.

   c. Reduce the number of ORG statements specified in assembler language routines.

   d. Segment the step into a number of link-edits, performing only part of the necessary linkage function in each successive step.

   e. Verify that all input object modules have ESD records.

   f. Place output and SYSUT1 data on a device with the largest available track size, and specify value1 of the SIZE parameter as large as feasible.

   g. Sort object module text in ascending address sequence

   h. A combination of b, c, f, and g.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0374 ERROR—TABLE OVERFLOW—INPUT CONTAINS TOO MANY RELOCATABLE ADDRESS CONSTANTS OR TOO MANY CONTROL SECTIONS CONTAINING SUCH CONSTANTS.

Explanation: Either (1) there are too many control sections with relocation dictionaries, or (2) there are too many relocatable address constants.

System Action: Processing is terminated.

Programmer Response: Probable user error. Either (1) increase the linkage editor's table space by increasing value1 (or decreasing value2) of the SIZE parameter making sure the region or partition size is also increased, if necessary; or (2) reduce the number of relocatable address constants in the input. (One method is to assemble the coding of two or more control sections into one control section.)

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0382 ERROR—TEXT RECORD ID IS INVALID, CARD IGNORED.

Explanation: The ID of the text record refers to an invalid external symbol dictionary entry; that is, it does not refer to a section definition entry or a private code entry. The input deck may be out of sequence or incomplete.

System Action: The record is ignored. Processing continues.

Programmer Response: Probable user error. Check the input object modules for completeness and proper sequence. If necessary, either (1) re-create any module that has been in card form, or (2) isolate the incorrect module by executing the linkage editor with the NCAL option specified, using the NAME control statement for each input module. Diagnostic IEW0382 should recur and isolate the incorrect module. Re-create the module, and rerun the step.

Problem Determination: Table I, items 6b, 6f, 9a, 9b, 13, 29. Have available the output used to isolate the module as indicated above.

IEW0394   ERROR—MEMBER NOT STORED IN LIBRARY—PERMANENT DEVICE ERROR.

Explanation: This is either an input/output error or no space was allocated for the library directory.

System Action: Processing terminates.

Programmer Response: Check the SYSLMOD data set to make sure it is a partitioned data set with space allocated for a directory. If necessary, restore the library to a different volume, and rerun the job. Execute the IEHLIST utility program, using the LISTVTOC function to print out the data set control block for the SYSLMOD data set.

Problem Determination: Table I, items 1, 3, 13, 29. Have the output from a run with the library on a different volume available.

IEW0404   ERROR—MEMBER NOT STORED IN LIBRARY—NO SPACE LEFT IN DIRECTORY.

Explanation: All the directory blocks allocated when the output data set was created have been used.

Programmer Response: Probable user error. Either (1) reprocess, placing the output module in a new library; when the original library is used as input, concatenate the new one with it; or (2) use a utility program to copy the library, allowing for more directory entries. Edit the member into the new library. Execute the IEHLIST utility program, using the LISTVTOC and LISTPDS

statements to print out the data set control block and directory entries for the SYSLMOD data set.

Problem Determination: Table I, items 1, 3, 6b, 13, 29.

IEW0412   ERROR—ALIAS NOT STORED IN LIBRARY—NO SPACE LEFT IN DIRECTORY.

Explanation: All directory blocks allocated when the output data set was created have been used.

System Action: The ALIAS is not stored in the specified library; however, the member can be referred to by the member name.

Programmer Response: Probable user error. Either (1) reprocess, placing the output module in a new library; when the original library is used as input, concatenate the new one with it, or (2) use a utility program to copy the entire library (except the member whose alias was not stored), and allow for more directory entries. Edit the member into the new library. Execute the IEHLIST utility program, using the LISTVTOC and LISTPDS statements to print out the data set control block and directory entries for the SYSLMOD data set.

Problem Determination: Table I, items 1, 3, 6b, 13, 29.

IEW0421   WARNING—MEMBER NOT STORED IN LIBRARY—IDENTICAL NAME IN DIRECTORY, WILL TRY TO STORE UNDER 'TEMPNAME'.

Explanation: The output module name has been used previously in the library. The replace function is not specified.

System Action: An attempt is made to store the output module into the library under the name TEMPNAME.

Programmer Response: Probable user error. Either (1) reprocess, using a different name in the SYSLMOD DD statement or NAME statement; or (2) reprocess, and specify the replacement function for the name originally specified in the SYSLMOD DD statement or the NAME statement. Execute the IEHLIST utility program, using the LISTPDS statement to print out the directory entries for the SYSLMOD data set.

Problem Determination: Table I, items 1, 3, 6b, 13, 29.

IEW0432   ERROR—LIBRARY NAME PRINTED CANNOT BE OPENED, DD CARD MAY BE MISSING.

Explanation: The DD statement that defines the library is probably missing. This message also results when a

sequential data set (encountered in the processing of an INCLUDE statement) cannot be opened.

System Action: Processing continues without input from the specified library.

Programmer Response: Probable user error. Either supply the missing DD statement, or correct erroneous information on the DD statement. Execute the IEHLIST utility program using the LISTVTOC statement to print out the data set control block for the data set that cannot be opened.

Problem Determination: Table I, items 1, 3, 6b, 13, 29. Table II, format 3.

IEW0444   ERROR—TABLE OVERFLOW—TOO MANY DOWNWARD CALLS.

Explanation: There are too many V-type address constants that refer to segments lower in the tree structure.

System Action: Processing is terminated.

Programmer Response: Probable user error. Either (1) increase the linkage editor's table space by increasing value1 (or decreasing value2) of the SIZE parameter, making sure the region or partition size is also increased if necessary; or (2) use an overlay structure with fewer segments.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0454   ERROR—TABLE OVERFLOW—SEGMENT CONTAINS TOO MANY DOWNWARD CALLS.

Explanation: One segment in the overlay structure contains too many V-type address constants that refer to segments lower in the tree structure. The maximum is determined by the size of an output load module record.

System Action: Processing is terminated.

Programmer Response: Probable user error. Either (1) increase the size of an output load module record by specifying SYSLMOD as a library with a larger block size, or (2) incorporate some of the called control sections in the requesting segment, or (3) divide the requesting segment into two or more segments.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0461   WARNING—SYMBOL PRINTED IS AN UNRESOLVED EXTERNAL REFERENCE; NCAL WAS SPECIFIED, OR THE REFERENCE WAS MARKED FOR RESTRICTED NO-CALL OR NEVERCALL.

Explanation: The NCAL option, restricted no-call, or never-call function was specified for the external reference.

System Action: The automatic library call mechanism does not attempt to resolve the external reference.

Programmer Response: Normally, no response is necessary. Check that the reference is valid and not the result of a keypunch or programming error. If you wish the reference resolved, either (1) add the needed module to the primary or included input data sets; (2) remove the NCAL option, if specified; (3) remove the LIBRARY statement specifying restricted no-call or never-call; or (4) if an input load module contained a never-call reference, re-create the load module without specifying never-call.

Problem Determination: Table I, items 1, 3, 6b, 6f, 9a, 9b, 13, 29. Have available each object module that contains a call to the reference, and the associated source listing.

IEW0472   ERROR—INVALID ALIAS ENTRY POINT IN OVERLAY STRUCTURE.

Explanation: The specified alias entry point is not in the root segment.

System Action: The entry point for the member name is used.

Programmer Response: Probable user error. Respecify the alias, entry point, or overlay structure.

Problem Determination: Table I, items 6b, 6f, 9, 29. Have the module containing the alias entry point and its associated listing available.

IEW0484   ERROR—TABLE OVERFLOW—TOO MANY EXTERNAL SYMBOLS AFFECTED BY RELOCATION.

Explanation: There are too many symbols being relocated.

System Action: Processing is terminated.

Programmer Response: Probable user error. Either (1) increase the linkage editor's table space by increasing value1 (or decreasing value2) of the SIZE parameter making sure the region or partition size is also increased if necessary; or (2) segment the step into a number of link-edits, performing only part of the necessary editing function in each successive step.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

IEW0492   ERROR—NAME CARD FOUND IN LIBRARY, CARD IGNORED.

Explanation: A NAME statement has been encountered in an included data set or an automatic call library. NAME statements may be placed only in the primary input.

System Action: The record is ignored. Processing continues.

Programmer Response: Remove the NAME statement from the library or sequential data set. Reprocess if the load module is incorrect. Execute the IEBPTPCH utility program or print out all included and automatic call library modules.

Problem Determination: Table I, items 1, 3, 6b, 6f, 13, 29.

**IEW0502   ERROR—ALIAS NOT STORED IN
             LIBRARY—PERMANENT DEVICE
             ERROR.**

Explanation: Because of an input/output error, the alias could not be stored in the library directory.

System Action: The load module has already been stored.

Programmer Response: Execution of the module is possible using the member name or aliases already stored. The module can be link-edited again with the new alias specified. If message IEW0502 appears again, restore the library to a different volume and rerun the job. Execute the IEHLIST utility program, using the LISTVTOC and LISTPDS statements to print out the data set control block and directory entries for the SYSLMOD data set.

Problem Determination: Table I, items 1, 3, 6b, 13, 29. Have the output from a run with the library on a different volume available.

**IEW0512   ERROR—INCLUDE STATEMENT
             SYNTAX CONFLICTS WITH RECORD
             FORMAT OF SPECIFIED DATA
             SET—DD NAME PRINTED.**

Explanation: The INCLUDE statement syntax conflicts with the characteristics of the data set specified on the DD statement.

System Action: The specified module is ignored.

Programmer Response: Probable user error. Either (1) specify a member name on the INCLUDE or DD statement if the data set is partitioned; or (2) remove all member names from the INCLUDE statement if the data set is not partitioned. Execute the IEHLIST utility program, using the LISTVTOC statement to print out the data set control block for the specified data set.

Problem Determination: Table I, items 1, 3, 6b, 13, 29.

**IEW0522   ERROR—SPECIFIED DATA SET HAS
             UNACCEPTABLE RECORD
             FORMAT—DDNAME PRINTED.**

Explanation: The record format of the specified data set is not type U or F and cannot be processed by the linkage editor.

System Action: The data set is not processed.

Programmer Response: Probable user error. Correct the data set specification. Execute the IEHLIST utility program, using the LISTVTOC statement to print out the data set control block for the rejected data set.

Problem Determination: Table I, items 1, 3, 13, 29.

**IEW0532   ERROR—BLOCKSIZE OF LIBRARY
             DATA SET EXCEEDED
             MAXIMUM—DDNAME PRINTED.**

Explanation: The block size of the specified library data set cannot be handled by the linkage editor.

System Action: The data set is not processed.

Programmer Response: Probable user error. Either (1) decrease the block size of the data set, (2) rerun in a larger region or partition, (3) increase value2 of the SIZE parameter to allow for larger buffers, and, if necessary, increase value1 and region size accordingly. Execute the IEHLIST utility program, using the LISTVTOC statement to print out the data set control block for the specified data set. For information on block size, region size, and SIZE parameter values, see Linkage Editor and Loader.

Problem Determination: Table I, items 1, 3, 13, 29.

**IEW0543   ERROR—IDENTICAL NAME IN
             DIRECTORY.**

Explanation: The member name already exists in the directory. In the case of a member, an attempt was made to store under TEMPNAME; however, TEMPNAME was also found in the directory.

System Action: The output module is not stored under this member name.

Programmer Response: Probable user error. Either (1) specify a unique member name for the module on the NAME control statement or the SYSLMOD DD statement, or (2) specify the replace function on the NAME statement. Execute the IEHLIST utility program, using the

LISTPDS statement to print out the directory entries for the SYSLMOD data set.

Problem Determination: Table I, items 1, 3, 6b, 13, 29.

IEW0552   ERROR—COMMON PRINTED EXCEEDED SIZE OF CONTROL SECTION·WITH IDENTICAL NAME.

Explanation: A named COMMON area has been encountered that is larger than a control section with the same name.

System Action: The linkage editor uses the length specified for the control section. Processing continues.

Programmer Response: Ensure that no named COMMON area is larger than the control section initializing it. FORTRAN programmers should make sure that any named COMMON in a BLOCK DATA subprogram is at least as large as any named COMMON with the same name in any other FORTRAN program or subprogram with which the BLOCK DATA subprogram is to be link-edited. To isolate the problem, you can run the step with the NCAL option specified. If the error recurs, the long COMMON occurs in the primary data set or in an included data set. Otherwise, it occurs in a module from the automatic call library. In either case, execute the following functions of the service aid program AMBLIST. Execute the LISTOBJ function to list all object module symbols, and execute the LISTLOAD function with the OUTPUT=XREF option to list all load module symbols in the appropriate input data sets. Check the listings for all modules that contain the named COMMON in question, and correct the lengths.

Problem Determination: Table I, items 6b, 6f, 14, 29.

IEW0564   ERROR—INVALID TEXT ORIGIN—LINKAGE EDITOR PROCESSING TERMINATED.

Explanation: Text has been found that has an origin address outside the limits of the control section to which it belongs.

System Action: Processing is terminated.

Programmer Response: A text record in an object module input to the link-edit has an invalid text origin address, probably mispunched. List the object module using the LISTOBJ function of the AMBLIST service aid program. Examine the ADDR= field of the TXT records to locate the invalid address. Re-create the object module, then reattempt the link-edit.

Problem Determination: Table I, items 3, 4, 9a, 27, 29.

IEW0572   ERROR—COMMON PRINTED AND SUBROUTINE HAVE IDENTICAL NAME.

Explanation: This message appears only when the linkage editor is processing an object program originally written in FORTRAN. It is issued when a COMMON defined in the program has the same name as a subprogram.

System Action: Processing continues. The output module is marked not executable unless the LET option is specified.

User Response: Change the name of either the COMMON or the subprogram so that the names are no longer the same. Compile and link-edit the program again.

Problem Determination: Table I, items 3, 6b, 6f, 13, 29. Save the linkage editor output listing.

IEW0581   WARNING—INVALID MEMBER NAME—WILL TRY TO STORE UNDER "TEMPNAME."

Explanation: The member name to be assigned to the output load module was taken from the SYSLMOD DD statement, but the name was found to be invalid.

System Action: An attempt is made to store the output module into the library under the name TEMPNAME.

Programmer Response: Correct the member name on the SYSLMOD DD statement to conform to the rules for a name on the NAME control statement.

Problem Determination: Table I, items 1, 3, 4, 6b, 27, 29.

IEW0594   ERROR—INPUT DATA SET BLOCKSIZE IS INVALID.

Explanation: The block size for the primary input data (SYSLIN) is not an even multiple of the logical record length, or exceeds the allowable maximum.

System Action: Linkage editor processing terminates.

Programmer Response: Probable user error. Determine whether the values specified in the SIZE parameter are sufficient to accommodate the blocking factor of the primary input data set (SYSLIN). Blocking factors are discussed under "SIZE Option" in Linkage Editor and Loader. If the SIZE values are not large enough, increase them and execute the linkage editor step again. The region for the job step must be large enough to allow the size values specified, as described under "EXEC Statement—REGION Parameter," in Linkage Editor and Loader. If the region is not

large enough, increase the REGION parameter before executing the linkage editor step again.

If the blocking factor is greater than 40 to 1, or is not a multiple of the logical record length, correct the BLKSIZE field, or re-create the data set, or both. Execute the linkage editor step again. If possible, execute the IEHLIST utility program, using the LISTVTOC statement to print out the data set control block for the specified data set.

Problem Determination: Table I, items 1, 3, 13, 29.

IEW0602   ERROR—INPUT FROM OBJECT MODULE IS INVALID—END CARD MISSING.

Explanation: The END card of an object module being processed by the linkage editor is missing.

System Action: Linkage editor processing continues. The load module produced is marked not-executable unless the LET option has been specified.

Programmer Response: If input to the linkage editor was in the form of an object deck, verify that the last card is an END card (END in columns 2, 3, and 4). If the card is not an END card, recompile or reassemble the source program. If input to the linkage editor was not in the form of an object deck, recompile or reassemble the source program with the DECK option specified.

In either case, verify that the last card is an END card. Rerun the linkage editor step using the object deck. Execute the service aid program, AMBLIST, using the LISTOBJ function and save the resultant listing of the questionable object module.

Problem Determination: Table I, items 1, 3, 13, 22, 23, 29.

IEW0614   ERROR—LENGTH NOT SPECIFIED FOR EXTERNAL SYMBOL PRINTED.

Explanation: An object module contained a control section that had a length field containing zero in its external symbol dictionary (ESD) entry, and either (1) the control section was not last in the object module, or (2) the length was not specified on the END card.

System Action: The module was not processed, and the linkage editor terminated processing.

Programmer Response: Probable user error. Check the input object modules for completeness and proper sequence. If necessary, either (1) re-create any module that has been in card form, or

(2) isolate the incorrect module by executing the linkage editor with the NCAL option specified, using the NAME control statement for each input object module. Diagnostic IEW0614 should recur and isolate the incorrect module. Re-create the module, and rerun the step.

Problem Determination: Table I, items 3, 6b, 6f, 9a, 9b, 13, 22, 23, 29.

IEW0622   ERROR—ADDRESS CONSTANT REFERENCES NULL UNNAMED CONTROL SECTION.

Explanation: An address constant has been found that references a symbol defined in an unnamed control section having a length of zero.

System Action: The processing of the input RLD record is terminated at the invalid item. Processing resumes with the next record.

Programmer Response: Either re-create the input (referencing) module, by eliminating or changing the reference, or re-create the referenced module, by eliminating or redefining the symbol being referenced.

Problem Determination: Table I, items 1, 3, 4, 9a, 9b, 23, 27, 29.

IEW0630   ERROR—DDNAME PRINTED HAD SYNCHRONOUS ERROR—XREF ABORTED.

Explanation: A permanent input/output error occurred while attempting to produce a cross-reference table. The output module was successfully edited.

System Action: The information provided by the SYNADAF macro instruction was printed after the message code in the following format: SYNAD EXIT, jobname, stepname, unit address, device type, ddname, operation attempted, error description, block count or BBCCHHR, access method.

Programmer Response: Rerun the linkage editor step. If possible, execute the IEHLIST utility program, using the LISTVTOC function to print out the data set control block for the data set specified in the SYNAD output.

Problem Determination: Table I, items 1, 3, 13, 29. Save the output from the SYNADAF macro instruction.

IEW0642   ERROR—SYMBOL PRINTED APPEARED ON CONTROL STATEMENT BUT WAS NOT MATCHED.

Explanation: Either (1) a control section name or common name appearing on an ORDER or PAGE control statement was not found in the primary or additional input sources; or (2) alignment or

sequencing of a label reference (such as a FORTRAN ENTRY statement) was specified.

System Action: The name is ignored. Processing continues.

Programmer Response: Probable user error. Either (1) include the specified control section or common area in the input, or delete the name from the control statement; or (2) verify that only control section or common area names are specified on the control statement.

Have available the job stream and associated output listings.

Problem Determination: Table I, items 3, 6b, 6f, 13, 29.

IEW0652   ERROR—CONFLICT IN ORDER SPECIFIED FOR SYMBOL PRINTED.

Explanation: A control section or common area was named more than once on one or in a series of ORDER statements. After a name appears once, any subsequent use of the name is invalid unless the name appears as the last operand on one ORDER statement and as the first operand on the next.

System Action: The first use of the name determines the order of the control section or common area in the output load module. Any subsequent use of the name is ignored, as is the balance of the control statement it appears on. Any control sections or common areas named on the balance of the statement are included in the output load module but are not sequenced. Linkage editor processing continues.

Programmer Response: Probable user error. Correct the ORDER statement so the name appears only once or appears as the last operand on one statement and the first operand on the next.

Problem Determination: Table I, items 3, 6b, 6f, 13, 29.

IEW0670   THE SPECIFIED IDENTIFY DATA HAS BEEN ADDED TO THE IDR FOR THE CONTROL SECTION NAME PRINTED.

Explanation: The linkage editor has added the data specified on the IDENTIFY control statement to the IDR record for the control section indicated.

System Action: Processing continues.

Programmer Response: None. This message is for information only.

IEW0682   ERROR—CONTROL SECTION NAME ON AN IDENTIFY CONTROL STATEMENT

IS INCORRECT OR THE STATEMENT IS MISPLACED—IDENTIFY DATA IGNORED.

Explanation: The control section named on the IDENTIFY control statement either (1) does not exist in the load module, or (2) had not been read in by the linkage editor by the time it encountered the IDENTIFY statement.

System Action: The data specified on the IDENTIFY statement is ignored. Linkage editor processing continues.

Programmer Response: Probable user error. Check the IDENTIFY statement to verify that the control section name has been specified correctly and that the IDENTIFY statement has been placed correctly in the input. Verify that the required control section has been included in the input to the linkage editor step. Correct the input, and rerun the linkage editor step.

Problem Determination: Table I, items 1, 3, 6a, 6b, 6f, 13, 29.

IEW0694   ERROR—TABLE OVERFLOW—SIZE VALUE SPECIFIED NOT LARGE ENOUGH FOR CSECT IDR INPUT—LINKAGE EDITOR PROCESSING TERMINATED.

Explanation: The space available for CSECT identification records was insufficient for the actual input.

System Action: Linkage editor processing terminates.

Programmer Response: Rerun the link-edit, increasing the space available to the linkage editor by increasing value1 (or decreasing value2, or both) of the SIZE option or the region or partition size. If this fails, divide the link-edit into two or more smaller link-edits.

Problem Determination: Table I, items 1, 6b, 6f, 29.

IEW0704   UNRECOVERABLE ERROR DETECTED IN CSECT IDR INPUT—LINKAGE EDITOR PROCESSING TERMINATED.

Explanation: An unrecoverable error was detected while processing an input module containing CSECT identification (IDR) records. The cause of the error was a load module IDR record that contained an invalid code in its subtype field (the third byte of the record).

System Action: Linkage editor processing terminates.

Programmer Response: Probable user error. Examine all data sets containing input load modules. Check all secondary input sources (either defined by the SYSLIB DD statement or specified on an

INCLUDE statement). If any user
modifications were made to any record
other than text in any of these modules,
re-create any affected modules from the
source or object level and execute the
linkage editor step again. Execute the
LISTLOAD function of the service aid
program, AMBLIST, specifying the
OUTPUT=BOTH option to list all load
modules in the input to the linkage
editor. Execute the service aid
program, AMBLIST, with the LISTIDR
function to list CSECT IDR records for
all members of the SYS1.LINKLIB data set
that was cataloged on the system at the
time of the error.

Problem Determination: Table I, items 1,
3, 6b, 13, 22, 29.

IEW0714   ERROR—MEMBER NOT STORED IN
          LIBRARY—STOW WORKSPACE
          UNAVAILABLE.

Explanation: The conditional GETMAIN
macro instruction issued by the STOW
routine to obtain work space in virtual
storage was unsuccessful (that is, not
enough contiguous virtual storage was
available).

System Action: The member is not stored
in the specified library; linkage editor
processing is terminated.

Programmer Response: Rerun the linkage
editor job step. The error may be a
temporary one caused by fragmentation of
virtual storage. If the problem
persists, check for user-written
programs or user-written SVC (supervisor
call) routines that may be executing
concurrently with the linkage editor and
causing virtual storage fragmentation,
as would occur when a GETMAIN macro is
issued without a FREEMAIN in an
uncontrolled loop.

Problem Determination: Table I, items 1,
2, 3, 13, 29. Table II, format 1, using
TRACE=SVC.

IEW0722   ERROR—INVALID ALIAS NAME.

Explanation: An ALIAS name has been
specified that either does not begin
with an alphabetic character, $, #, a,
or 12-0 punch, or contains a character
that is not alphameric, $, #, a, or 12-0
punch.

System Action: The ALIAS name is
ignored.

Programmer Response: Correct the invalid
character(s) in the ALIAS name according
to the rules above, and rerun the
link-edit job step.

Problem Determination: Table I, items 3,
4, 6b, 20, 27, 29.

IEW0731   WARNING—ALIAS MATCHES MEMBER
          NAME—ALIAS IGNORED.

Explanation: An ALIAS name has been
specified that duplicates the member
name of the output load module.

System Action: The ALIAS name is
ignored.

Programmer Response: Either (1) delete
the ALIAS name, or (2) make the ALIAS
name unique.

IEW0740   THE INDICATED ACTION WAS TAKEN
          FOR AN EXPAND REQUEST.

Explanation: The linkage editor has
increased the size of a control section
or named common section by the number of
bytes specified in an EXPAND control
statement. Details of the expansion are
provided in the message text that
appears immediately following the EXPAND
control statement.

System Action: Processing continues.

Programmer Response: None. This message
is for information only; no error has
occurred and no response is required.

IEW0751   WARNING—INVALID AMODE/RMODE
          COMBINATION FOUND IN MODE
          CONTROL STATEMENT—IGNORED.

Explanation: An invalid conbination of
AMODE and RMODE parameters was specified
on the MODE control statement.

System Action: Processing continues but
the MODE control statement is ignored as
a source of AMODE/RMODE data applicable
to the output load module.

Programmer Response: Either (1) remove
the MODE control statement, or (2)
correct the MODE control statement so
that the combination of AMODE/RMODE
specifications is valid.

Problem Determination: Table I, items 1,
3, 4, 6a, 6b, 13, 15, 20, 27, 29.

IEW0761   WARNING—INVALID AMODE/RMODE
          COMBINATION FOUND IN PARM
          FIELD—IGNORED.

Explanation: An invalid combination of
AMODE and RMODE parameters was specified
in the PARM field of the EXEC statement.

System Action: Processing continues, but
the PARM field of the EXEC statement is
ignored as a source of AMODE/RMODE data
applicable to the output load module.

Programmer Response: Either (1) remove
the AMODE and RMODE specification(s)
from the PARM field, or (2) correct the
PARM field so that the combination of
AMODE/RMODE specifications is valid.

Problem Determination: Table I, items 1,
3, 4, 6a, 13, 15, 27, 29.

IEW0771 **WARNING—AMODE/RMODE DATA IN MODE CONTROL STATEMENT INCOMPATIBLE WITH OVLY OPTION—IGNORED.**

Explanation: The AMODE and/or RMODE parameters specified on the MODE control statement are incompatible with the overlay option.

System Action: Processing continues, but the MODE control statement is ignored. The overlay load module is assigned an AMODE of 24 and an RMODE of 24.

Programmer Response: Either (1) remove the mode control statement, or (2) remove the OVLY option from the PARM field of the EXEC statement.

Problem Determination: Table I, items 1, 3, 4, 6a, 6b, 13, 15, 20, 27, 29.

IEW0781 **WARNING—AMODE/RMODE DATA IN PARM FIELD INCOMPATIBLE WITH OVLY OPTION—IGNORED.**

Explanation: The AMODE and/or RMODE parameters specified in the PARM field of the EXEC statement are incompatible with the overlay option.

System Action: Processing continues, but the AMODE and/or RMODE data in the PARM field of the EXEC statement is ignored. The overlay load module is assigned an AMODE of 24 and an RMODE of 24.

Programmer Response: Either (1) remove the AMODE and/or RMODE options from the PARM field of the EXEC statement, or (2) remove the OVLY option from the PARM field of the EXEC statement.

Problem Determination: Table I, items 1, 3, 4, 6a, 13, 15, 27, 29.

IEW0791 **WARNING—INVALID AMODE/RMODE COMBINATION IN ESD DATA FOR THE NAMED CSECT—IGNORED.**

Explanation: An invalid AMODE/RMODE combination, 24/ANY, was found in the ESD data.

System Action: Processing continues, but the control section is processed as having an AMODE of 24 and an RMODE of 24.

Programmer Response: Either (1) correct the ESD data to indicate a valid AMODE/RMODE combination, or (2)

recompile/reassemble the source program to obtain an object module without the erroneous AMODE/RMODE indicators, or (3) correct the processor generating the object module to provide only valid AMODE/RMODE indicators.

Problem Determination: Table I, items 1, 3, 4, 6a, 6b, 9a, 13, 15, 27, 29.

IEW0984 **ERROR—SYSPRINT BLOCKSIZE EXCEEDS MAXIMUM—LINKEDIT PROCESSING TERMINATED.**

Explanation: The block size specified for the SYSPRINT data set cannot be handled by the linkage editor.

System Action: The data set is not opened. Linkage editor processing terminates.

Programmer Response: Probable user error. Either (1) decrease the block size of the data set, or (2) decrease value2 of the SIZE option to allow for a larger SYSPRINT buffer, and if necessary, increase value1 accordingly. Increase the region or partition size correspondingly, if necessary. Rerun the linkage editor step. Execute the IEHLIST utility program, using the LISTVTOC statement to print out the data set control block for the SYSPRINT data set.

Problem Determination: Table I, items 1, 3, 13, 29.

IEW0994 **ERROR—SYSPRINT DD CARD MISSING—LINKAGE EDITOR PROCESSING TERMINATED.**

Explanation: The SYSPRINT data set cannot be opened.

System Action: Linkage editor processing terminates.

Programmer Response: Probable user error. The SYSPRINT DD statement is probably missing. Supply the missing SYSPRINT DD statement, and execute the job step again.

Problem Determination: Table I, items 1, 3, 7, 7a (SYS1.LINKLIB), 13, 29. Table II, format 3.

IEW1001   WARNING—UNRESOLVED EXTERNAL REFERENCE (NOCALL SPECIFIED).

Explanation: The NCAL, NOCALL, or NORES option or never-call function was specified for the external reference.

System Action: The SYSLIB data set is not searched if the NCAL or NOCALL option has been specified. The link pack area queue is not searched if the NORES option has been specified. Neither the SYSLIB data set nor the link pack area queue is searched if the ER is marked 'never-call' from a previous linkage editor run.

Programmer Response: Normally, no response is necessary. If you wish the reference resolved, either (1) add the needed module to the SYSLIN input data set; (2) remove the NOCALL, NCAL, or NORES option, if specified; or (3) if an input load module contained a never-call reference, re-create the load module without specifying never-call.

Problem Determination: Run the failing step, using the linkage editor instead of loader, and save the resulting output. Table I, items 6a, 9b, 13, 23, 29. Have available each object module that contains a call to the reference.

IEW1012   ERROR—UNRESOLVED EXTERNAL REFERENCE.

Explanation: The external reference was not found on the SYSLIB-defined data set or in the link pack area.

System Action: No attempt is made to execute the module unless the LET option is specified.

Programmer Response: Probable user error. Make sure that the reference is valid and not the result of a keypunch or programming error. If the reference is valid, add the needed module or alias to either (1) the SYSLIB data set, (2) the link pack area, or (3) the SYSLIN input data set. Make sure the SYSLIB DD statement has been specified if needed. If the needed module is in a SYSLIB or SYSLIN partitioned data set, execute the IEHLIST utility program using the LISTPDS statement to print out the data set directory.

Problem Determination: Execute the failing job step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 1, 3, 6a, 9b, 13, 22, 23, 29.

IEW1024   ERROR—DDNAME CANNOT BE OPENED.

Explanation: The SYSLIN data set cannot be opened. The DD statement defining the data set is missing or incorrect.

System Action: Processing terminates. The loader returns to the caller with a condition code of 16.

Programmer Response: Probable user error. Either (1) supply a missing SYSLIN DD statement, (2) correct erroneous information on the SYSLIN DD statement, or (3) make sure the correct DDNAME has been specified for the SYSLIN data set. If the loader was invoked by a macro instruction such as LINK, rather than through the EXEC statement, make sure that the SYSLIN ddname, if passed, is correct.

Problem Determination: Table I, items 1, 29. Either have the output of the SYSGEN of the loader available, or execute the AMASPZAP service aid program with the DUMPT IEWLOADR IEWLDDEF statement, and save the resulting dump of the loader default ddnames.

IEW1034   ERROR—DDNAME HAD SYNCHRONOUS ERROR.

Explanation: A physical uncorrectable input/output error occurred. If it occurred on a blocked data set, the block size may have been specified incorrectly.

System Action: The message supplied by the SYNADAF macro was printed. Processing was terminated.

Programmer Response: For any fixed format, specify the correct block size. If the block size was correct and the data set was an input data set, re-create or restore the data set.

Problem Determination: Execute the failing step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 1, 29.

IEW1044   ERROR—UNACCEPTABLE RECORD FORMAT (VARIABLE ON INPUT).

Explanation: Only object module (FIXED record format) and load module (UNDEFINED record format) data sets are accepted by the loader.

System Action: Processing was terminated. The loader returns to caller with a condition code of 16.

Programmer Response: Probable user error. (1) Make sure that the record format specification is correct. The

record format may have been mispunched.
(2) Make sure that the correct data set
has been specified. Execute the IEHLIST
utility program using the LISTVTOC
statement to print out the data set
control block for the input data sets,
and save the resulting output.

Problem Determination: Execute the
failing step, using the linkage editor
instead of the loader, and save the
resulting output. Table I, items 1, 6a,
29.

**IEW1053   ERROR—I/O ERROR WHILE
              SEARCHING LIBRARY DIRECTORY.**

Explanation: A permanent I/O error
occurred while attempting a BLDL.

System Action: Automatic library call
processing is terminated.

Programmer Response: Ensure that the
SYSLIB defined data set is partitioned.
If it is, re-create or restore the data
set and rerun the job step.

Problem Determination: Execute the
failing step, using the linkage editor
instead of the loader, and save the
resulting output. Table I, items 1, 6a,
25c, 29.

IEW1072   ERROR—BLKSIZE IS INVALID.

Explanation: In the specified data set,
the BLKSIZE was not an integral multiple
of LRECL.

System Action: BLKSIZE was rounded up to
the next higher multiple of LRECL and
processing continued.

Programmer Response: Probable user
error. Change BLKSIZE to be an integral
multiple of LRECL. If the data set was
an input data set, execute the IEHLIST
utility program, using the LISTVTOC
statement to print out the data set
control block, and save the resulting
output.

Problem Determination: Table I, items 1,
29.

**IEW1082   ERROR—INVALID LENGTH
              SPECIFIED.**

Explanation: The length of a control
section in an object module was not
specified in either its FSD entry or on
the END record, and text was received
for the control section.

System Action: The total length of the
text received was used.

Programmer Response: Check if an END
record in any object module is missing
or has been replaced. If so, re-create
the object module and rerun the job.

Problem Determination: Execute the
failing step, using the linkage editor
instead of the loader, and save the
resulting output. Table I, items 6a,
13, 22, 23, 29.

IEW1093   ERROR—NO TEXT RECEIVED.

Explanation: No valid text has been
received for the loaded module.

System Action: The loader returns to the
caller with a condition code of 12.

Programmer Response: Probable user
error. (1) Make sure that the SYSLIN
data was specified correctly. (2) Check
other error messages issued for cause of
error (for example, invalid record).
Correct the error, and rerun the job
step. Execute the service aid program,
AMBLIST, using the LISTOBJ function and
save the resultant listing of the
questionable input module. Have all
SYSLIN input available.

Problem Determination: Execute the
failing step, using the linkage editor
instead of the loader, and save the
resulting output. Table I, items 1, 6a,
29.

IEW1102   ERROR—DOUBLY DEFINED ESD.

Explanation: Two identical external
names have been found in the input. (1)
The invalid match involves a label
reference (LR) or label definition (LD)
matching an existing section definition
(SD), common (CM), or label reference
(LR). The section definition for the
input LR or LD must be marked delete in
order for this not to be an error. (2)
It is always invalid for a CM to match
an existing LR.

System Action: References to the name
are resolved with respect to the first
occurrence of the name.

Programmer Response: Probable user
error. Correct the existing symbol
conflict. To isolate the problem,
execute the following functions of the
service aid program AMBLIST. Execute
the LISTOBJ function to list all object
module symbols, and execute the LISTLOAD
function with the OUTPUT=XREF option to
list all load module symbols. Object
module symbols can be printed using the
IEBPTPCH utility program with the PRINT
statement.

Problem Determination: Execute the
failing step, using the linkage editor
instead of the loader, and save the
resulting output. Table I, items 6a,
29. Have all object and load module
input available.

IEW1112   ERROR—INVALID 2-BYTE ADCON.

Explanation: A relocatable A-type or V-type address constant of less than 3 bytes has been found in the input.

System Action: The constant is not relocated.

Programmer Response: Probable user error. Check assembler language input for Y-type address constants, which cannot be relocated. Delete or correct the invalid address constant.

Problem Determination: Rerun the step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 6a, 9a, 29. Have object module input and associated listings available.

IEW1123   ERROR—INVALID RECORD FROM LOAD MODULE.

Explanation: An unrecognizable type record was found while reading a load module.

System Action: The record is ignored and processing continues.

Programmer Response: (1) Check that all input data sets are specified correctly on DD statements. (2) If load module input occurs in the SYSLIN data set, rerun the step with the NOCALL option specified. If error message IEW1123 recurs, the incorrect load module is in SYSLIN input. Otherwise, it is in SYSLIB input. (3) Isolate the incorrect load module by executing the linkage editor with the NCAL option specified, using the INCLUDE and NAME statements for each suspect load module. When the incorrect load module is isolated, re-create it and rerun the job step.

Problem Determination: Execute the failing step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 9b, 29.

IEW1132   ERROR—INVALID ID RECEIVED.

Explanation: Input contains an invalid external symbol ID.

This error is the result of the following conditions:

1.   The SD for an ID does not appear in the input module.

2.   Text is received before the external symbol dictionary (ESD) defining it is received.

3.   An RLD is received before the ESDs to which it pertains.

4.   The ID defining the entry point on the END card is not a defined SD, PC, or LR ESD type.

System Action: The invalid item is ignored.

Programmer Response: (1) Check that input object modules are complete and that assembly or compilation errors did not occur when object modules were generated. (2) Rerun the step with the NOCALL option specified. If error message IEW1132 recurs, the incorrect module is in SYSLIN input. Otherwise, it is in SYSLIB input. (3) Isolate the incorrect module by executing the linkage editor with the NCAL option specified, using the INCLUDE and NAME statements for each suspect module. When the incorrect module is isolated, re-create it and rerun the step.

Problem Determination: Run the failing step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 9b, 29. If an incorrect object module was created, have the module and its associated listing available.

IEW1141   WARNING—CARD RECEIVED NOT AN OBJECT RECORD.

Explanation: The card read has a blank in column one.

System Action: The card is ignored.

Programmer Response: Probable user error. Check input for a blank card or linkage editor control card. If other errors occur, re-create all object modules that have been in card form.

Problem Determination: Rerun the step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 1, 3, 6a, 13, 29.

IEW1152   ERROR—INVALID RECORD FROM OBJECT MODULE.

Explanation: An unrecognizable record type was received while reading an object module.

System Action: The card is ignored.

Programmer Response: Probable user error. Check object module input for invalid records. Column 1 should contain a 12-2-9 punch. Columns 2 through 4 should contain a TXT, RLD, ESD, END, or SYM identifier. Remove incorrect records or re-create the module, and rerun.

Problem Determination: Rerun the step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 9a, 29. Have object module input available.

IEW1161   WARNING—NO ENTRY POINT RECEIVED.

Explanation: No entry point was specified in the parameter field or END card. The END card entry point specification could be incorrect (that is, invalid ID, bad column alignment, etc.). The parameter field specification could also be incorrect.

System Action: The first assigned address is used as the entry point.

Programmer Response: Probable user error. (1) Specify the entry point name in the loader parameter list, EP=. If the entry point occurs in load module input, this parameter must be specified. (2) If you cannot use the EP= parameter and the entry point occurs in an object module, make sure that the module is included in the SYSLIN or SYSLIB input, and that an entry point was specified during compilation or assembly.

Problem Determination: Rerun the step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 6a, 29. Have the module containing the entry point and its associated listing available.

IEW1173   ERROR—ENTRY POINT RECEIVED BUT NOT MATCHED.

Explanation: The entry point name specified in the parameter field or on an END card was not matched to an incoming LR, SD, or PC.

System Action: The first assigned address is used as the entry point address.

Programmer Response: Probable user error. (1) Check to see if the EP= parameter was specified correctly. (2) Check to see if the module containing the entry point is included in either the SYSLIN or SYSLIB input. (3) Check other messages issued for the cause of error (that is, invalid record).

Problem Determination: Rerun the step, using the linkage editor instead of the loader, and save the resulting output. Table I, items 6a, 9a, 9b, 29.

IEW1182   ERROR—NO END CARD RECEIVED.

Explanation: An END card is missing for an input object module.

System Action: Processing continues.

Programmer Response: Probable user error. Check input object modules. The last record of each should have a 12-2-9 punch in column 1 and the END identifier in columns 2 through 4. If an END record is missing, re-create the module and rerun.

Problem Determination: Rerun the step, using the linkage editor instead of the loader, and save the resulting output.

Table I, items 6a, 9a, 29. Have object module input available.

IEW1194   ERROR—AVAILABLE STORAGE EXCEEDED.

Explanation: The amount of virtual storage available to the loader is insufficient to allow construction of the required tables and loaded program.

System Action: The loader returns to the caller with a completion code of 16.

Programmer Response: Probable user error. (1) Increase the SIZE parameter, or (2) make sure the REGION specification is sufficient, or (3) make sure that sufficient virtual storage is available to satisfy the SIZE specification. Either have the output of the SYSGEN of the loader available or execute the AMASPZAP service aid program with the DUMPT IEWLOADR IEWLDDEF statement, and save the resulting dump of the loader's default SIZE value.

Problem Determination: Table I, items 1, 6a, 13, 29.

IEW1204   ERROR—TOO MANY EXTERNAL NAMES IN INPUT MODULE.

Explanation: The external symbol ID is too large to fit in the translation table.

System Action: Processing is terminated. The loader returns to the caller with a completion code of 16.

Programmer Response: If the program is large and/or complex, either (1) run the step using the linkage editor, or (2) break down the large program module into a number of smaller routines. If the program is not particularly large or complex, check other messages issued for the cause of error. Object module input may be incomplete or mispunched. Re-create the object modules, and rerun the job.

Problem Determination: Table I, items 6a, 9a, 9b, 13, 29.

IEW1214   ERROR—IDENTIFICATION FAILED—DUPLICATE PROGRAM NAME FOUND.

Explanation: When trying to identify the loaded program to the system, the IDENTIFY routine found a duplicate program name in the user's region or partition or in the link pack area.

System Action: Processing is terminated. The loader returns to the caller with a condition code of 16.

Programmer Response: Probable user error. Specify a unique program name, using the NAME option, or let the loader

default the name to **GO. Rerun the job.

Problem Determination: Table I, items 3, 13, 29. Execute the IEBPTPCH utility program to obtain a listing of the SYS1.PARMLIB data set. Save the output.

IEW1224   ERROR—IDENTIFICATION FAILED.

Explanation: The IDENTIFY routine located an error in the parameter list passed to it by the loader. The appropriate IDENTIFY macro instruction support may not be included in the operating system.

System Action: Processing is terminated. The loader returns to the caller with a condition code of 16.

Programmer Response: Verify that the appropriate IDENTIFY macro instruction support is included in the system. The release level of the IDENTIFY macro instruction should be the same as the release level of the loader.

Problem Determination: Table I, items 3, 13, 29.

IEW1232   ERROR—COMMON EXCEEDS SIZE OF CSECT WITH SAME NAME.

Explanation: A named COMMON area has been encountered that is larger than the control section with same name.

System Action: The loader uses the length of the control section. Processing continues.

Programmer Response: Ensure that no named COMMON area is larger than the control section initializing it. FORTRAN programmers should make sure that any named COMMON in a BLOCK DATA subprogram is at least as large as any named COMMON with the same name in any other FORTRAN program or subprogram with which the BLOCK DATA subprogram is to be link-edited. To isolate the problem, you can run the step with the NCAL option specified. If the error recurs, the long COMMON occurs in the primary data set. Otherwise, it occurs in a module from the automatic call library. In either case, execute the following functions of the service aid program AMBLIST. Execute the LISTOBJ function to list all object module symbols, and execute the LISTLOAD function with the OUTPUT=XRFF option to list all load module symbols in the appropriate input data sets. Check the listings for all modules that contain the named COMMON in question, and correct the lengths.

Problem Determination: Table I, item 9.

IEW1241   WARNING—INVALID AMODE/RMODE COMBINATION FOUND IN PARM FIELD—IGNORED.

Explanation: An invalid combination of AMODE and RMODE parameters was specified in the PARM field of the EXEC statement.

System Action: Processing continues, but the PARM field of the EXEC statement is ignored as a source of AMODE/RMODE data for the loaded module.

Programmer Response: Either (1) remove the AMODE and RMODE specification(s) from the PARM field, or (2) correct the PARM field so that the combination of AMODE/RMODE specifications is valid.

Problem Determination: Table I, items 1, 3, 4, 6a, 13, 15, 27, 29.

IEW1251   WARNING—INVALID AMODE/RMODE COMBINATION IN ESD DATA FOR THE NAMED CSECT—IGNORED.

Explanation: An invalid AMODE/RMODE combination, 24/ANY, was found in the ESD data.

System Action: Processing continues, but the control section is processed as having an AMODE of 24 and an RMODE of 24.

Programmer Response: Either (1) correct the ESD data to indicate a valid AMODE/RMODE combination, or (2) recompile/reassemble the source program to obtain an object module without the erroneous AMODE/RMODE indicators, or (3) correct the processor generating the object module to provide only valid AMODE/RMODE indicators.

Problem Determination: Table I, items 1, 3, 4, 6a, 9a, 13, 15, 27, 29.

IEW1262   ERROR—INVALID 3-BYTE ADCON.

Explanation: A 3-byte address constant cannot be relocated, because it requires more than the 24 bits available for relocation.

System Action: Processing continues but the 3-byte address constant is not relocated.

Programmer Response: Either (1) redefine the address constant, as a 4-byte address constant, or (2) specify an RMODE of 24 bits for the loaded module.

Problem Determination: Table I, items 1, 3, 4, 6a, 9a, 9b, 13, 15, 19, 21, 22, 23, 27, 29.

IEW1271   WARNING—INCONSISTENT RMODE DATA—RMODE = 24 FORCED.

Explanation: The loading of the module was initiated above the 16-megabyte virtual storage line because the external symbol dictionary (ESD) data for the first control section encountered indicated an RMODE of ANY.

However, a control section has been encountered that indicates an RMODE of 24 in the ESD data.

System Action: The loading of the module above the 16-megabyte virtual storage line is stopped, and loading is restarted below the 16-megabyte virtual storage line.

Programmer Response: Either (1) specify an RMODE and an AMODE (if necessary) in the PARM field of the EXEC statement for the loaded module; (2) cause the control section that indicates an RMODE of 24 in the ESD data to be the first control section encountered; or (3) recode and/or recompile/reassemble the source program for the control section that has an RMODE of 24, making the RMODE = ANY.

Problem Determination: Table I, items 1, 3, 4, 6a, 9, 13, 15, 27, 29.

IEW1991   ERROR—USER PROGRAM HAS ABNORMALLY TERMINATED.

Explanation: This message is issued by the loader when it determines that the loaded program has terminated abnormally.

System Action: Loaded program execution is terminated abnormally, and control is returned to the loader. (Unless the user has included a SYSUDUMP DD statement for the loaded program, this message is the only indication that the program has terminated abnormally.)

Operator Response: None.

Programmer Response: To obtain a dump to aid in determining the cause of the abnormal termination, include a SYSUDUMP DD statement for the loaded program and rerun the job.

Problem Determination: Table I, items 3, 13, 29.

GC26-4067-0

IBM®

MVS/370
Linkage Editor and Loader
Messages
GC26-4067-0

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

List TNLs here:

If you have applied any technical newsletters (TNLs) to this book, please list them here:

Last TNL _____
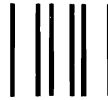
Previous TNL _____

Previous TNL _____

Fold on two lines, tape, and mail. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.) Thank you for your cooperation.

GC26-4067-0

Reader's Comment Form

IBM

MVS/370 Linkage Editor and Loader Messages (File No. S370-40) Printed in U.S.A. GC26-4067-0

GC26-4067-0