



# Computer Science and Technology

NBS Special Publication 500-70/2

## NBS Minimal BASIC Test Programs—Version 2, User's Manual

Volume 2—Source Listings  
and Sample Output

## NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards<sup>1</sup> was established by an act of Congress on March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, and the Institute for Computer Sciences and Technology.

**THE NATIONAL MEASUREMENT LABORATORY** provides the national system of physical and chemical and materials measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; conducts materials research leading to improved methods of measurement, standards, and data on the properties of materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government agencies; develops, produces, and distributes Standard Reference Materials; and provides calibration services. The Laboratory consists of the following centers:

Absolute Physical Quantities<sup>2</sup> — Radiation Research — Thermodynamics and Molecular Science — Analytical Chemistry — Materials Science.

**THE NATIONAL ENGINEERING LABORATORY** provides technology and technical services to the public and private sectors to address national needs and to solve national problems; conducts research in engineering and applied science in support of these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the ultimate user. The Laboratory consists of the following centers:

Applied Mathematics — Electronics and Electrical Engineering<sup>2</sup> — Mechanical Engineering and Process Technology<sup>2</sup> — Building Technology — Fire Research — Consumer Product Technology — Field Methods.

**THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY** conducts research and provides scientific and technical services to aid Federal agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Institute consists of the following centers:

Programming Science and Technology — Computer Systems Engineering.

<sup>1</sup>Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted; mailing address Washington, DC 20234.

<sup>2</sup>Some divisions within the center are located at Boulder, CO 80303.

# **Computer Science and Technology**

---

National Bureau of Standards  
Library, E-01 Admin. Bldg.  
DEC 4 1980

NBS<sup>t</sup> Special Publication 500-70/2

## **NBS Minimal BASIC Test Programs—Version 2, User's Manual**

### **Volume 2—Source Listings and Sample Output**

John V. Cugini  
Joan S. Bowden  
Mark W. Skall

Center for Programming Science and Technology  
Institute for Computer Sciences and Technology  
National Bureau of Standards  
Washington, DC 20234



**U.S. DEPARTMENT OF COMMERCE  
Philip M. Klutznick, Secretary**

Luther H. Hodges, Jr., Deputy Secretary  
Jordan J. Baruch, Assistant Secretary for Productivity,  
Technology and Innovation

**National Bureau of Standards**  
Ernest Ambler, Director

Issued November 1980

## **Reports on Computer Science and Technology**

The National Bureau of Standards has a special responsibility within the Federal Government for computer science and technology activities. The programs of the NBS Institute for Computer Sciences and Technology are designed to provide ADP standards, guidelines, and technical advisory services to improve the effectiveness of computer utilization in the Federal sector, and to perform appropriate research and development efforts as foundation for such activities and programs. This publication series will report these NBS efforts to the Federal computer community as well as to interested specialists in the academic and private sectors. Those wishing to receive notices of publications in this series should complete and return the form at the end of this publication.

National Bureau of Standards Special Publication 500-70/2

Nat. Bur. Stand. (U.S.), Spec. Publ. 500-70/2, 487 pages (Nov. 1980)

CODEN: XNBSAV

Library of Congress Catalog Card Number: 80-600163

**U.S. GOVERNMENT PRINTING OFFICE  
WASHINGTON: 1980**

---

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402  
Price \$9.50  
(Add 25 percent for other than U.S. mailing)

NBS Minimal BASIC Test Programs - Version 2  
User's Manual  
Volume 2 - Source Listings and Sample Output

John V. Cugini  
Joan S. Bowden  
Mark W. Skall

Abstract: This publication describes the set of programs developed by NBS for the purpose of testing conformance of implementations of the computer language BASIC to the American National Standard for Minimal BASIC, ANSI X3.60-1978. The Department of Commerce has adopted this ANSI standard as Federal Information Processing Standard 68. By submitting the programs to a candidate implementation, the user can test the various features which an implementation must support in order to conform to the standard. While some programs can determine whether or not a given feature is correctly implemented, others produce output which the user must then interpret to some degree. This manual describes how the programs should be used so as to interpret correctly the results of the tests. Such interpretation depends strongly on a solid understanding of the conformance rules laid down in the standard, and there is a brief discussion of these rules and how they relate to the test programs and to the various ways in which the language may be implemented.

Key words: BASIC; language processor testing; Minimal BASIC; programming language standards; software standards; software testing

Acknowledgments: Version 2 owes its existence to the efforts and example of many people. Dr. David Gilsinn and Mr. Charles Sheppard, the authors of version 1\*, deserve credit for construction of that first system, of which version 2 is a refinement. In addition, they were generous in their advice on many of the pitfalls to avoid on the second iteration. Mr. Landon Dyer assisted with the testing and document preparation. It is also important to thank the many people who sent in comments and suggestions on Version 1. We hope that all the users of the resulting Version 2 will help us improve it further.

\* issued as an NBS Internal Report; no longer available.



## Introduction

This volume is the companion to Volume 1 which explains the rationale underlying the design of the Version 2 Minimal BASIC Test Programs and also contains detailed instructions for operating and interpreting the tests. This volume complements the information in Volume 1, by allowing you, first to examine the source code in its original form, and second to get an idea of what the output from a standard-conforming implementation should look like. This second purpose is especially important since the implementation under test may not allow some of the programs to fail gracefully. If you have any doubt about whether the results of a given test signify passing or failing, be sure to check the sample output for that program in this volume.

Because the standard allows certain features to be implementation defined, not all standard processors need produce exactly the same output. You must compare the output of your processor with this sample output carefully, making sure that any variation between the two is within the scope allowed by the standard. Appendix C of the ANSI standard lists all the features for which implementations may differ.

This volume is ordered according to the sequential numbering system of the programs (see section 6.2 in Volume 1). The source code appears first, preceded by three lines of asterisks. The output for that program follows immediately, separated from the program itself by a line of hyphens. The output is that which would be produced by a truly minimal standard conforming processor, that is, one which provided for no syntactic enhancements of the Minimal BASIC standard.

## Table of Contents

Group	Program	Page
1 Simple PRINTing of string constants .....	1	5
2 END and STOP .....	2-5	8
2.1 END .....	2-4	8
2.2 STOP .....	5	11
3 PRINTing and simple assignment (LET) .....	6-14	11
3.1 String variables and TAB .....	6-8	11
3.2 Numeric constants and variables .....	9-14	22
4 Control Statements and REM .....	15-21	55
4.1 REM and GOTO .....	15-16	55
4.2 GOSUB and RETURN .....	17	59
4.3 IF-THEN .....	18-21	60
5 Variables .....	22-23	71
6 Numeric Constants, Variables, and Operations .....	24-43	74
6.1 Standard Capabilities .....	24-27	74
6.2 Exceptions .....	28-35	99
6.3 Errors .....	36-38	119
6.4 Accuracy tests - Informative .....	39-43	121
7 FOR-NEXT .....	44-55	137
7.1 Standard Capabilities .....	44-49	137
7.2 Errors .....	50-55	166
8 Arrays .....	56-84	171
8.1 Standard Capabilities .....	56-62	171
8.2 Exceptions .....	63-72	189
8.3 Errors .....	73-84	204

Group	Program	Page
9 Control Statements .....	85-91	217
9.1 GOSUB and RETURN .....	85-87	217
9.2 ON-GOTO .....	88-91	223
10 READ, DATA, and RESTORE .....	92-106	228
10.1 Standard Capabilities .....	92-95	228
10.2 Exceptions .....	96-101	243
10.3 Errors .....	102-106	251
11 INPUT .....	107-113	256
11.1 Standard Capabilities .....	107-110	256
11.2 Exceptions .....	111-112	287
11.3 Errors .....	113	300
12 Implementation-supplied Functions .....	114-150	300
12.1 Precise functions: ABS, INT, SGN .....	114-116	300
12.2 Approximated functions: SQR, ATN, COS, EXP, LOG, SIN, TAN .....	117-129	307
12.3 RND and RANDOMIZE .....	130-142	352
12.3.1 Standard Capabilities .....	130-134	352
12.3.2 Informative tests .....	135-142	367
12.4 Errors .....	143-150	389
13 User-defined Functions .....	151-163	395
13.1 Standard Capabilities .....	151-152	395
13.2 Errors .....	153-163	402

Group	Program	Page
14 Numeric Expressions .....	164-184	412
14.1 Standard Capabilities in context of LET-statement .....	164	412
14.2 Expressions in other contexts: PRINT, IF, ON-GOTO, FOR .....	165-166	416
14.3 Exceptions in subscripts and arguments .....	167-171	422
14.4 Exceptions in other contexts: PRINT, IF, ON-GOTO, FOR .....	172-184	431
15 Miscellaneous Checks .....	185-208	452
15.1 Missing keyword .....	185	452
15.2 Spaces .....	186-191	453
15.3 Quotes .....	192-195	459
15.4 Line numbers .....	196-201	462
15.5 Line longer than 72 characters .....	202	467
15.6 Effect of zones and margin on PRINT .....	203	468
15.7 Lowercase characters .....	204-205	476
15.8 Ordering relations between strings .....	206	477
15.9 Mismatch of types in assignment .....	207-208	482

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 1: NULL PRINT AND PRINTING QUOTED STRINGS."  
20 PRINT " ANSI STANDARD 3.2, 3.4, 5.2, 5.4, 12.2, 12.4"  
30 PRINT  
40 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF THE END STATEMENT"  
50 PRINT " WHICH HAS NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"  
60 PRINT " THIS FEATURE TO BE INCORRECTLY IMPLEMENTED, THEN THE "  
70 PRINT " VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."  
80 PRINT  
90 PRINT  
100 PRINT " * * * * * * * * * * * * * * * *"  
110 PRINT " *"  
120 PRINT " * MINIMAL BASIC TEST PROGRAMS *"  
130 PRINT " * VERSION 2 *"  
135 PRINT " *"  
140 PRINT " *"  
150 PRINT " * PRODUCT OF *"  
150 PRINT " * NATIONAL BUREAU OF STANDARDS *"  
160 PRINT " *"  
170 PRINT " * * * * * * * * * * * * * * * *"  
180 PRINT  
190 PRINT "SECTION 1.1: NULL PRINT. "  
200 PRINT  
210 PRINT " BEGIN TEST."  
220 PRINT " THIS IS LINE 1."  
230 PRINT " THIS IS LINE 2."  
240 PRINT  
250 PRINT " THIS IS LINE 4, 3 SHOULD HAVE BEEN SKIPPED."  
260 PRINT " THIS IS LINE 5."  
270 PRINT  
280 PRINT  
290 PRINT " THIS IS LINE 8, 6 AND 7 SHOULD HAVE BEEN SKIPPED."  
300 PRINT  
310 PRINT "**** TEST PASSED IF LINES 2 AND 4 HAVE A BLANK LINE BETWEEN"  
320 PRINT " THEM, AND LINES 5 AND 8 TWO BLANK LINES ***"  
330 PRINT  
340 PRINT " END TEST."  
350 PRINT  
360 PRINT "SECTION 1.2: LEGAL CHARACTERS IN PRINTING QUOTED STRINGS."  
390 PRINT  
400 PRINT " BEGIN TEST."  
410 PRINT  
420 PRINT " LINE BELOW MUST PRINT OUT UPPER CASE LETTERS A THROUGH Z"  
430 PRINT " ABCDEFGHIJKLMNOPQRSTUVWXYZ"  
440 PRINT  
450 PRINT " LINE BELOW MUST PRINT OUT DIGITS ZERO THROUGH NINE."  
460 PRINT " 0123456789"  
470 PRINT
```

```
480 PRINT "LINE BELOW MUST PRINT OUT, IN THE FOLLOWING ORDER, LEFT"
490 PRINT "PARENTHESIS, RIGHT PARENTHESIS, AMPERSAND, EXCLAMATION"
500 PRINT "POINT, NUMBER SIGN, PERCENT, APOSTROPHE, QUESTION MARK,"
510 PRINT "ASTERISK, AND DOLLAR SIGN."
520 PRINT "                               ()&!#%!?*$"
530 PRINT
540 PRINT "LINE BELOW MUST PRINT OUT, IN THE FOLLOWING ORDER, PLUS,"
550 PRINT "COMMA, MINUS, SLANT, LESS-Than, EQUALS, GREATER-Than,"
560 PRINT "CIRCUMFLEX, AND PERIOD."
570 PRINT "                               +,-/<=>^."
580 PRINT
590 PRINT "LINE BELOW MUST PRINT OUT, IN THE FOLLOWING ORDER, COLON,"
600 PRINT "SPACE, SEMI-COLON, AND UNDERLINE."
610 PRINT "                               : ;_"
620 PRINT
630 PRINT "*** TEST PASSED IF THE ABOVE SPECIFICATIONS WERE MET ***"
640 PRINT
650 PRINT "                               END TEST."
660 PRINT
670 PRINT "SECTION 1.3: SPACES IN QUOTED STRINGS."
690 PRINT
700 PRINT "                               BEGIN TEST."
710 PRINT
720 PRINT "                               ABCDEFGHIJKLMNOPQRSTUVWXYZ"
730 PRINT "                               A CD GH LM RS YZ"
740 PRINT
750 PRINT "*** TEST PASSED IF THE SECOND LINE OF ALPHA CHARACTERS IN "
760 PRINT "      THE ABOVE TEST HAS BLANKS WHERE THE LETTERS B, E, F,"
770 PRINT "      I, J, K, N, O, P, Q, T, U, V, W, AND X APPEAR IN THE "
775 PRINT "      FIRST LINE. ***"
780 PRINT
790 PRINT "                               END TEST."
800 PRINT
810 PRINT "SECTION 1.4: PRINTING THE NULL STRING."
830 PRINT
840 PRINT "                               BEGIN TEST."
850 PRINT
870 PRINT "THE NEXT STATEMENT PRINTS A NULL QUOTED STRING."
890 PRINT ""
900 PRINT "THE LINE ABOVE MUST BE BLANK AS A RESULT OF PRINTING"
910 PRINT "A NULL QUOTED STRING."
920 PRINT
930 PRINT "*** TEST PASSED IF THE ABOVE SPECIFICATIONS WERE MET ***"
940 PRINT
950 PRINT "                               END TEST."
960 PRINT
970 PRINT "END PROGRAM 1"
980 END
```

\*\*\* NOTE: THIS PROGRAM MAKES USE OF THE END STATEMENT WHICH HAS NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW THIS FEATURE TO BE INCORRECTLY IMPLEMENTED, THEN THE VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

```
* * * * * * * * * * * * * * * * *  
* *  
* MINIMAL BASIC TEST PROGRAMS *  
* VERSION 2 *  
* *  
* PRODUCT OF *  
* NATIONAL BUREAU OF STANDARDS *  
* * * * * * * * * * * * * * * * *
```

#### SECTION 1.1: NULL PRINT.

BEGIN TEST.

THIS IS LINE 1.  
THIS IS LINE 2.

THIS IS LINE 4, 3 SHOULD HAVE BEEN SKIPPED.  
THIS IS LINE 5.

THIS IS LINE 8, 6 AND 7 SHOULD HAVE BEEN SKIPPED.

\*\*\* TEST PASSED IF LINES 2 AND 4 HAVE A BLANK LINE BETWEEN THEM, AND LINES 5 AND 8 TWO BLANK LINES \*\*\*

END TEST.

#### SECTION 1.2: LEGAL CHARACTERS IN PRINTING QUOTED STRINGS.

BEGIN TEST.

LINE BELOW MUST PRINT OUT UPPER CASE LETTERS A THROUGH Z  
ABCDEFGHIJKLMNPQRSTUVWXYZ

LINE BELOW MUST PRINT OUT DIGITS ZERO THROUGH NINE.  
0123456789

LINE BELOW MUST PRINT OUT, IN THE FOLLOWING ORDER, LEFT PARENTHESIS, RIGHT PARENTHESIS, AMPERSAND, EXCLAMATION POINT, NUMBER SIGN, PERCENT, APOSTROPHE, QUESTION MARK, ASTERISK, AND DOLLAR SIGN.

( )&!#%'?\*\$

LINE BELOW MUST PRINT OUT, IN THE FOLLOWING ORDER, PLUS, COMMA, MINUS, SLANT, LESS-THAN, EQUALS, GREATER-THAN, CIRCUMFLEX, AND PERIOD.

+,-/<=>^.

LINE BELOW MUST PRINT OUT, IN THE FOLLOWING ORDER, COLON,  
SPACE, SEMI-COLON, AND UNDERLINE.

: ;\_

\*\*\* TEST PASSED IF THE ABOVE SPECIFICATIONS WERE MET \*\*\*

END TEST.

SECTION 1.3: SPACES IN QUOTED STRINGS.

BEGIN TEST.

ABCDEFGHIJKLMNPQRSTUVWXYZ  
A CD GH LM RS YZ

\*\*\* TEST PASSED IF THE SECOND LINE OF ALPHA CHARACTERS IN  
THE ABOVE TEST HAS BLANKS WHERE THE LETTERS B, E, F,  
I, J, K, N, O, P, Q, T, U, V, W, AND X APPEAR IN THE  
FIRST LINE. \*\*\*

END TEST.

SECTION 1.4: PRINTING THE NULL STRING.

BEGIN TEST.

THE NEXT STATEMENT PRINTS A NULL QUOTED STRING.

THE LINE ABOVE MUST BE BLANK AS A RESULT OF PRINTING  
A NULL QUOTED STRING.

\*\*\* TEST PASSED IF THE ABOVE SPECIFICATIONS WERE MET \*\*\*

END TEST.

END PROGRAM 1

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 2: THE END-STATEMENT."  
20 PRINT "      ANSI STANDARD 4.2, 4.4"  
30 PRINT  
40 PRINT "SECTION 2.1: THE END-STATEMENT."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR TERMINATES"  
70 PRINT "THE BASIC PROGRAM WHEN IT ENCOUNTERS AN END STATEMENT."  
80 PRINT  
90 PRINT "                                BEGIN TEST."  
100 PRINT  
110 PRINT "**** TEST PASSES IF 'END PROGRAM 2' IS THE LAST OUTPUT"  
120 PRINT "      GENERATED BY THIS PROGRAM, FOLLOWED BY NORMAL "  
125 PRINT "      TERMINATION OF EXECUTION. ***"  
130 PRINT
```

```
140 PRINT "END TEST."
150 PRINT
160 PRINT "END PROGRAM 2"
170 END
```

---

## PROGRAM FILE 2: THE END-STATEMENT.

ANSI STANDARD 4.2, 4.4

## SECTION 2.1: THE END-STATEMENT.

THIS PROGRAM TESTS TO SEE IF THE PROCESSOR TERMINATES  
THE BASIC PROGRAM WHEN IT ENCOUNTERS AN END STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSES IF 'END PROGRAM 2' IS THE LAST OUTPUT  
GENERATED BY THIS PROGRAM, FOLLOWED BY NORMAL  
TERMINATION OF EXECUTION. \*\*\*

END TEST.

END PROGRAM 2

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 3: ERROR - MISPLACED END-STATEMENT."
20 PRINT "    ANSI STANDARD 4.2, 4.4"
30 PRINT
40 PRINT "SECTION 3.1: ERROR - MISPLACED END-STATEMENT."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"
70 PRINT "ACCEPTS PROGRAMS CONTAINING AN END STATEMENT WHICH IS"
80 PRINT "NOT PHYSICALLY LAST."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "      BY THE PROCESSOR, OR"
160 PRINT
170 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "BEGIN TEST."
230 PRINT
```

```
240 PRINT "IF THIS SENTENCE IS THE LAST SENTENCE PRINTED, THE PROCESSOR"
250 PRINT "DID NOT EXECUTE ANY MORE STATEMENTS AFTER ENCOUNTERING AN"
260 PRINT "END-STATEMENT IN THE MIDDLE OF THE PROGRAM."
270 END
280 PRINT
290 PRINT "IF THIS SENTENCE IS PRINTED OUT, THE PROCESSOR CONTINUED TO"
300 PRINT "EXECUTE STATEMENTS AFTER ENCOUNTERING AN END-STATEMENT"
310 PRINT "IN THE MIDDLE OF THE PROGRAM."
320 PRINT
330 PRINT "                                END TEST."
340 PRINT
350 PRINT "END PROGRAM 3"
360 END
```

---

? END IS NOT LAST IN LINE 270

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 4: ERROR - MISSING END-STATEMENT."
20 PRINT "      ANSI STANDARD 4.2, 4.4"
30 PRINT
40 PRINT "SECTION 4.1: ERROR - MISSING END-STATEMENT."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"
70 PRINT "ACCEPTS PROGRAMS WHICH DO NOT CONTAIN AN END STATEMENT."
80 PRINT
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
110 PRINT
120 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
130 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
140 PRINT "      BY THE PROCESSOR, OR"
150 PRINT
160 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
170 PRINT
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
190 PRINT "FOR DETAILED CRITERIA."
200 PRINT
210 PRINT "                                BEGIN TEST."
220 PRINT
230 PRINT "THE PROCESSOR IS EXECUTING A PROGRAM WHICH DOES NOT CONTAIN"
240 PRINT "AN END STATEMENT."
250 PRINT
260 PRINT "                                END TEST."
270 PRINT
280 PRINT "END PROGRAM 4"
```

---

? NO END INSTRUCTION

```
*****
*****  

10 PRINT "PROGRAM FILE 5: THE STOP-STATEMENT."  

20 PRINT "    ANSI STANDARD 10.2, 10.4"  

30 PRINT  

40 PRINT "SECTION 5.1: THE STOP-STATEMENT."  

50 PRINT  

60 PRINT "                                BEGIN TEST."  

70 PRINT  

80 PRINT "IF PROGRAM EXECUTION TERMINATES AFTER THE NEXT LINE,"  

90 PRINT " *** TEST PASSED ***"  

100 STOP  

110 PRINT "PROGRAM EXECUTION WAS NOT TERMINATED:"  

120 PRINT " *** TEST FAILED ***"  

130 PRINT  

140 PRINT "                                END TEST."  

150 PRINT  

160 PRINT "END PROGRAM 5"  

170 END
```

---

PROGRAM FILE 5: THE STOP-STATEMENT.  
ANSI STANDARD 10.2, 10.4

#### SECTION 5.1: THE STOP-STATEMENT.

BEGIN TEST.

IF PROGRAM EXECUTION TERMINATES AFTER THE NEXT LINE,  
\*\*\* TEST PASSED \*\*\*

```
*****
*****  

*****  

10 PRINT "PROGRAM FILE 6: PRINT-SEPARATORS, TABS,"  

15 PRINT "        AND STRING VARIABLES."  

20 PRINT "    ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 12.2, 12.4"  

30 PRINT  

40 PRINT "SECTION 6.1: SEMICOLON SEPARATOR WITH QUOTED STRINGS."  

60 PRINT  

70 PRINT "                                BEGIN TEST."  

80 PRINT  

90 PRINT "                                1. 123"  

100 PRINT "                                2. 1";"2";"3"  

110 PRINT "                                3. 1";  

120 PRINT "23"  

130 PRINT "                                4. 12";  

140 PRINT "3"  

150 PRINT "                                5. 1";
```

```
160 PRINT "2";
170 PRINT "3"
180 PRINT
190 PRINT "**** TEST PASSED IF THERE ARE EXACTLY FIVE NUMBERED LINES"
200 PRINT "    ABOVE, EACH CONTAINING THE ADJACENT DIGITS '123', AND"
205 PRINT "    IF THEY ARE LINED UP IN A COLUMN ***"
210 PRINT
220 PRINT "                                END TEST."
380 PRINT
390 PRINT "SECTION 6.2: THE COMMA SEPARATOR WITH QUOTED STRINGS."
410 PRINT
420 PRINT "                                BEGIN TEST."
430 PRINT
440 PRINT "THE PRINT LINE SHOULD BE DIVIDED INTO EQUAL ZONES WHICH "
450 PRINT "ENABLE THE COMMA SEPARATOR TO ACT AS A TAB WHEN PRINTING"
460 PRINT "A SERIES OF COMMA-SEPARATED ITEMS."
470 PRINT
480 PRINT "000000000111111112222222233333333444444445";
490 PRINT "5555555556"
500 PRINT "12345678901234567890123456789012345678901234567890";
510 PRINT "1234567890"
520 PRINT "XYZ", "XYZ", "XYZ"
530 PRINT
540 PRINT "**** TEST PASSED IF 3 SETS OF THE CHARACTERS 'XYZ' ARE EVENLY"
550 PRINT "    SPACED ON A SINGLE LINE IN THE ABOVE TEST AND WIDTH"
555 PRINT "    OF PRINT ZONE IS AS DOCUMENTED ***"
560 PRINT
570 PRINT "                                END TEST."
580 PRINT
590 PRINT "SECTION 6.3: TABBING QUOTED STRINGS."
610 PRINT
620 PRINT "                                BEGIN TEST."
630 PRINT
640 PRINT "000000000111111112222222233333333444444445";
650 PRINT "5555555556"
660 PRINT "12345678901234567890123456789012345678901234567890";
670 PRINT "1234567890"
680 PRINT TAB(24); "1"
690 PRINT TAB(48); "2"
700 PRINT TAB(59); "3"
710 PRINT
720 PRINT "**** TEST PASSED IF A '1' IS PRINTED BELOW COLUMN HEADING 24,"
730 PRINT "    A '2' BELOW 48 AND A '3' BELOW 59 IN THE ABOVE TEST ***"
740 PRINT
750 PRINT "                                END TEST."
760 PRINT
770 PRINT "SECTION 6.4: THE ASSIGNMENT OF QUOTED STRINGS."
790 PRINT
800 PRINT "                                BEGIN TEST."
810 PRINT
820 PRINT "THIS TEST ASSIGNS THE SAME QUOTED STRING TO ALL LEGAL STRING"
830 PRINT "VARIABLES: A$ THROUGH Z$, USING A SERIES OF LET-STATEMENTS."
840 PRINT "A$ IS ASSIGNED THE QUOTED STRING, THEN B$ IS ASSIGNED A$ ..."
850 PRINT "THROUGH Z$ ASSIGNED Y$."
860 PRINT
```

```
870 LET A$="18 CHARACTERS LONG"
880 LET B$=A$
890 LET C$=B$
900 LET D$=C$
910 LET E$=D$
920 LET F$=E$
930 LET G$=F$
940 LET H$=G$
950 LET I$=H$
960 LET J$=I$
970 LET K$=J$
980 LET L$=K$
990 LET M$=L$
1000 LET N$=M$
1010 LET O$=N$
1020 LET P$=O$
1030 LET Q$=P$
1040 LET R$=Q$
1050 LET S$=R$
1060 LET T$=S$
1070 LET U$=T$
1080 LET V$=U$
1090 LET W$=V$
1100 LET X$=W$
1110 LET Y$=X$
1120 LET Z$=Y$
1130 PRINT "           Z$ = ";Z$
1140 PRINT
1150 PRINT "*** TEST PASSED IF 'Z$ = 18 CHARACTERS LONG' IS PRINTED"
1160 PRINT "      IN THE ABOVE TEST ***"
1170 PRINT
1180 PRINT "           END TEST."
1190 PRINT
1200 PRINT "SECTION 6.5: SEMICOLON SEPARATOR WITH ASSIGNED STRINGS"
1210 PRINT "           AND CONSTANTS."
1220 PRINT
1230 PRINT "           BEGIN TEST."
1240 PRINT
1250 LET A$="123"
1260 LET B$="1"
1270 LET C$="2"
1280 LET D$="3"
1290 LET E$="12"
1300 LET F$="23"
1310 PRINT "           1.";A$
1320 PRINT "           2.";B$;C$;D$
1330 PRINT "           3.";B$;
1340 PRINT F$
1350 PRINT "
1360 PRINT D$
1370 PRINT "
1380 PRINT "2";
1390 PRINT D$
1400 PRINT
```



```

1930 PRINT "0000000001111111122222222333333344444444445";
1940 PRINT "5555555556"
1950 PRINT "12345678901234567890123456789012345678901234567890";
1960 PRINT "1234567890"
1965 PRINT "PRINT-ZONE IDENTIFIERS:"
1970 PRINT "1","2","3","4"
1980 PRINT , , , "A"
1990 PRINT
2000 PRINT "*** TEST PASSED IF AN 'A' IS IN THE BEGINNING COLUMN OF THE"
2010 PRINT "    FOURTH PRINT ZONE FOR THIS SYSTEM ***"
2020 PRINT
2030 PRINT "                                END TEST."
2040 PRINT
2050 PRINT "END PROGRAM 6"
2060 END

```

---

PROGRAM FILE 6: PRINT-SEPARATORS, TABS,  
AND STRING VARIABLES.

ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 12.2, 12.4

#### SECTION 6.1: SEMICOLON SEPARATOR WITH QUOTED STRINGS.

BEGIN TEST.

1. 123
2. 123
3. 123
4. 123
5. 123

\*\*\* TEST PASSED IF THERE ARE EXACTLY FIVE NUMBERED LINES  
ABOVE, EACH CONTAINING THE ADJACENT DIGITS '123', AND  
IF THEY ARE LINED UP IN A COLUMN \*\*\*

END TEST.

#### SECTION 6.2: THE COMMA SEPARATOR WITH QUOTED STRINGS.

BEGIN TEST.

THE PRINT LINE SHOULD BE DIVIDED INTO EQUAL ZONES WHICH  
ENABLE THE COMMA SEPARATOR TO ACT AS A TAB WHEN PRINTING  
A SERIES OF COMMA-SEPARATED ITEMS.

```

0000000001111111122222222333333334444444455555555556
123456789012345678901234567890123456789012345678901234567890
XYZ      XYZ      XYZ

```

\*\*\* TEST PASSED IF 3 SETS OF THE CHARACTERS 'XYZ' ARE EVENLY  
SPACED ON A SINGLE LINE IN THE ABOVE TEST AND WIDTH  
OF PRINT ZONE IS AS DOCUMENTED \*\*\*

END TEST.

## SECTION 6.3: TABMING QUOTED STRINGS.

BEGIN TEST.

000000000111111112222222233333333444444445555555555  
12345678901234567890123456789012345678901234567890

1

2

3

\*\*\* TEST PASSED IF A '1' IS PRINTED BELOW COLUMN HEADING 24,  
A '2' BELOW 48 AND A '3' BELOW 59 IN THE ABOVE TEST \*\*\*

END TEST.

## SECTION 6.4: THE ASSIGNMENT OF QUOTED STRINGS.

BEGIN TEST.

THIS TEST ASSIGNS THE SAME QUOTED STRING TO ALL LEGAL STRING  
VARIABLES: A\$ THROUGH Z\$, USING A SERIES OF LET-STATEMENTS.  
A\$ IS ASSIGNED THE QUOTED STRING, THEN B\$ IS ASSIGNED A\$ ...  
THROUGH Z\$ ASSIGNED Y\$.

Z\$ = 18 CHARACTERS LONG

\*\*\* TEST PASSED IF 'Z\$ = 18 CHARACTERS LONG' IS PRINTED  
IN THE ABOVE TEST \*\*\*

END TEST.

SECTION 6.5: SEMICOLON SEPARATOR WITH ASSIGNED STRINGS  
AND CONSTANTS.

BEGIN TEST.

1.123  
2.123  
3.123  
4.123  
5.123

\*\*\* TEST PASSED IF THERE ARE EXACTLY FIVE NUMBERED LINES  
ABOVE, EACH CONTAINING THE ADJACENT DIGITS '123', AND  
IF THEY ARE LINED UP IN A COLUMN \*\*\*

END TEST.

SECTION 6.6: COMMA SEPARATOR - ASSIGNED STRINGS  
AND CONSTANTS.

BEGIN TEST.

00000000011111111122222222333333334444444455555555556  
 12345678901234567890123456789012345678901234567890  
 XYZ XYZ XYZ

\*\*\* TEST PASSED IF 3 SETS OF THE CHARACTERS 'XYZ' ARE  
 EVENLY SPACED ON A SINGLE LINE IN THE ABOVE TEST AND  
 WIDTH OF PRINT ZONE IS AS DOCUMENTED \*\*\*

END TEST.

#### SECTION 6.7: TABMING ASSIGNED STRINGS AND CONSTANTS.

BEGIN TEST.

00000000011111111122222222333333334444444455555555556  
 12345678901234567890123456789012345678901234567890  
 1  
 2  
 3

\*\*\* TEST PASSED IF A '1' PRINTS BELOW COLUMN HEADING 24,  
 A '2' BELOW 48, AND A '3' BELOW 59 \*\*\*

END TEST.

#### SECTION 6.8: NULL PRINT ITEMS.

THIS TEST DETERMINES WHETHER THE PROCESSOR RECOGNIZES  
 CONSECUTIVE COMMAS IN A PRINT-STATEMENT AS TAB POSITIONS.

BEGIN TEST.

00000000011111111122222222333333334444444455555555556  
 12345678901234567890123456789012345678901234567890  
 PRINT-ZONE IDENTIFIERS:

1	2	3	4
			A

\*\*\* TEST PASSED IF AN 'A' IS IN THE BEGINNING COLUMN OF THE  
 FOURTH PRINT ZONE FOR THIS SYSTEM \*\*\*

END TEST.

END PROGRAM 6

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

10 PRINT "PROGRAM FILE 7: EXCEPTION - STRING OVERFLOW USING"  
 15 PRINT " THE LET-STATEMENT."  
 20 PRINT " ANSI STANDARD 9.5, 12.4"  
 30 PRINT

```
40 PRINT "SECTION 7.1: EXCEPTION - STRING OVERFLOW USING"
50 PRINT "                      THE LET-STATEMENT."
60 PRINT
70 PRINT "STRING OVERFLOW OCCURS WHEN THE VALUE ASSIGNED TO A STRING"
80 PRINT "VARIABLE CONTAINS MORE CHARACTERS THAN CAN BE RETAINED."
82 PRINT
85 PRINT "TO PASS THIS TEST:"
90 PRINT
95 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
100 PRINT "          DISPLAYED AND EXECUTION MUST TERMINATE, OR"
105 PRINT
110 PRINT "      2) STRING OVERFLOW MUST NOT OCCUR"
120 PRINT
130 PRINT "                                BEGIN TEST."
140 PRINT
145 PRINT "ABOUT TO ASSIGN STRING OF 19 CHARACTERS - "
150 LET A$=?*****19*****!
155 PRINT "ABOUT TO ASSIGN STRING OF 20 CHARACTERS - "
160 LET B$=?*****20*****!
165 PRINT "ABOUT TO ASSIGN STRING OF 30 CHARACTERS - "
170 LET C$=?*****30*****!
175 PRINT "ABOUT TO ASSIGN STRING OF 40 CHARACTERS - "
180 LET D$=?*****40*****!
185 PRINT "ABOUT TO ASSIGN STRING OF 50 CHARACTERS - "
190 LET E$=?*****50*****!
195 PRINT "ABOUT TO ASSIGN STRING OF 58 CHARACTERS - "
200 LET F$=?*****58*****!
202 PRINT "ALL ASSIGNMENTS COMPLETED."
205 PRINT
210 PRINT "?*****19*****!"
220 PRINT A$
230 PRINT
240 PRINT "?*****20*****!"
250 PRINT B$
260 PRINT
270 PRINT "?*****30*****!"
280 PRINT C$
290 PRINT
300 PRINT "?*****40*****!"
310 PRINT D$
320 PRINT
330 PRINT "?*****50*****!"
340 PRINT E$
350 PRINT
360 PRINT "?*****58*****!"
370 PRINT F$
380 PRINT
390 PRINT "IF THE PAIRED LINES IN THE OUTPUT ARE IDENTICAL AND EACH"
400 PRINT "    LINE STARTS WITH A QUESTION MARK AND ENDS WITH AN"
410 PRINT "    EXCLAMATION POINT, THEN"
420 PRINT "*** TEST PASSED ***"
440 PRINT
450 PRINT "                                END TEST."
```

```
460 PRINT
470 PRINT "END PROGRAM 7"
480 END
```

---

PROGRAM FILE 7: EXCEPTION - STRING OVERFLOW USING  
THE LET-STATEMENT.

ANSI STANDARD 9.5, 12.4

SECTION 7.1: EXCEPTION - STRING OVERFLOW USING  
THE LET-STATEMENT.

STRING OVERFLOW OCCURS WHEN THE VALUE ASSIGNED TO A STRING  
VARIABLE CONTAINS MORE CHARACTERS THAN CAN BE RETAINED.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED AND EXECUTION MUST TERMINATE, OR
- 2) STRING OVERFLOW MUST NOT OCCUR

BEGIN TEST.

ABOUT TO ASSIGN STRING OF 19 CHARACTERS -  
ABOUT TO ASSIGN STRING OF 20 CHARACTERS -  
ABOUT TO ASSIGN STRING OF 30 CHARACTERS -  
% STRING OVERFLOW IN LINE 170

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 8: EXCEPTION - TAB ARGUMENT"
15 PRINT "      LESS THAN ONE."
20 PRINT "      ANSI STANDARD 12.5"
50 PRINT
60 PRINT "TO PASS THIS TEST:"
65 PRINT
70 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTIONS MUST BE"
75 PRINT "          DISPLAYED, AND"
80 PRINT
85 PRINT "      2) THE ARGUMENT TO TAB MUST BE REPLACED WITH A"
90 PRINT "          VALUE OF ONE AND EXECUTION CONTINUES."
100 PRINT
110 PRINT "SECTION 8.1: TAB ARGUMENT IS 0."
120 PRINT
130 PRINT "                  BEGIN TEST."
140 PRINT
```

```
145 LET A=0
150 PRINT "000000000111111112222222233333334444444445";
160 PRINT "5555555556"
170 PRINT "12345678901234567890123456789012345678901234567890";
180 PRINT "1234567890"
190 PRINT TAB(A); "X"
200 PRINT
210 PRINT "*** TEST PASSED IF AN X IS PRINTED IN COLUMN 1, AFTER A "
220 PRINT "      SYSTEM MESSAGE IDENTIFYING THE EXCEPTION ***"
230 PRINT
240 PRINT "                                END TEST."
250 PRINT
260 PRINT "SECTION 8.2: TAB ARGUMENT IS NEGATIVE."
270 PRINT
280 PRINT "                                BEGIN TEST."
290 PRINT
295 LET A=-10
300 PRINT "000000000111111112222222233333334444444445";
310 PRINT "5555555556"
320 PRINT "12345678901234567890123456789012345678901234567890";
330 PRINT "1234567890"
340 PRINT TAB(A); "X"
350 PRINT
360 PRINT "*** TEST PASSED IF AN X IS PRINTED IN COLUMN 1, AFTER A"
370 PRINT "      SYSTEM MESSAGE IDENTIFYING THE EXCEPTION ***"
380 PRINT
390 PRINT "                                END TEST."
400 PRINT
430 PRINT
440 PRINT "SECTION 8.3: ROUNDED TAB ARGUMENT EQUAL TO ONE."
450 PRINT
460 PRINT "                                BEGIN TEST."
470 PRINT
475 LET A=.6
480 PRINT "000000000111111112222222233333334444444445";
490 PRINT "5555555556"
500 PRINT "12345678901234567890123456789012345678901234567890";
510 PRINT "1234567890"
530 PRINT TAB(A); "X"
540 PRINT
550 PRINT "*** TEST PASSED IF AN X IS PRINTED IN COLUMN 1 "
555 PRINT "      AND NO EXCEPTION REPORTED ***"
560 PRINT
570 PRINT "                                END TEST."
580 PRINT
590 PRINT "SECTION 8.4: FRACTIONAL TAB ARGUMENT LESS THAN ONE AFTER"
600 PRINT "      ROUNDING."
610 PRINT
620 PRINT "                                BEGIN TEST."
630 PRINT
```

```

640 PRINT "00000000011111111222222223333333344444444445";
650 PRINT "5555555556";
660 PRINT "12345678901234567890123456789012345678901234567890";
670 PRINT "1234567890";
680 LET A=.4;
690 PRINT TAB(A);"X";
700 PRINT;
710 PRINT "*** TEST PASSED IF AN X IS PRINTED IN COLUMN 1, AFTER A";
720 PRINT "    SYSTEM MESSAGE IDENTIFYING THE EXCEPTION ***";
730 PRINT;
740 PRINT "                                END TEST.";
750 PRINT;
760 PRINT "END PROGRAM 8";
770 END

```

---

PROGRAM FILE 8: EXCEPTION - TAB ARGUMENT  
LESS THAN ONE.  
ANSI STANDARD 12.5

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTIONS MUST BE DISPLAYED, AND
- 2) THE ARGUMENT TO TAB MUST BE REPLACED WITH A VALUE OF ONE AND EXECUTION CONTINUES.

SECTION 8.1: TAB ARGUMENT IS 0.

BEGIN TEST.

```

0000000001111111122222222333333334444444455555555556
12345678901234567890123456789012345678901234567890
% TAB OUT OF RANGE IN LINE 190
X

```

\*\*\* TEST PASSED IF AN X IS PRINTED IN COLUMN 1, AFTER A  
SYSTEM MESSAGE IDENTIFYING THE EXCEPTION \*\*\*

END TEST.

SECTION 8.2: TAB ARGUMENT IS NEGATIVE.

BEGIN TEST.

```

0000000001111111122222222333333334444444455555555556
12345678901234567890123456789012345678901234567890
% TAB OUT OF RANGE IN LINE 340
X

```

\*\*\* TEST PASSED IF AN X IS PRINTED IN COLUMN 1, AFTER A  
SYSTEM MESSAGE IDENTIFYING THE EXCEPTION \*\*\*

END TEST.

### SECTION 8.3: ROUNDED TAB ARGUMENT EQUAL TO ONE.

BEGIN TEST.

000000000111111112222222233333333444444445555555556  
12345678901234567890123456789012345678901234567890  
X

\*\*\* TEST PASSED IF AN X IS PRINTED IN COLUMN 1  
AND NO EXCEPTION REPORTED \*\*\*

END TEST.

## SECTION 8.4: FRACTIONAL TAB ARGUMENT LESS THAN ONE AFTER ROUNDING.

BEGIN TEST.

00000000111111112222222233333333444444445555555556  
12345678901234567890123456789012345678901234567890  
% TAB OUT OF RANGE IN LINE 690  
X

\*\*\* TEST PASSED IF AN X IS PRINTED IN COLUMN 1, AFTER A  
SYSTEM MESSAGE IDENTIFYING THE EXCEPTION \*\*\*

END TEST

END PROGRAM 8

```
10 PRINT "PROGRAM FILE 9: PRINTING NR1 AND NR2 NUMERIC CONSTANTS."
20 PRINT "      ANSI STANDARD 5.2, 5.4, 12.4"
30 PRINT
40 PRINT "SECTION 9.1: CONSTANTS IN NR1 FORM (INTEGERS)."
50 PRINT
60 PRINT "COLUMNS 1 AND 3 SHOW THE OUTPUT AS IT SHOULD BE."
70 PRINT
80 PRINT "COLUMNS 2 AND 4 SHOW THE ACTUAL OUTPUT."
90 PRINT
100 PRINT "BEGIN TEST."
110 PRINT
120 PRINT
130 PRINT "000000000111111112222222333333334444444445";
140 PRINT "55555555666666666777"
150 PRINT "1234567890123456789012345678901234567890";
160 PRINT "1234567890123456789012"
170 PRINT "SHOULD BE","ACTUAL","SHOULD BE","ACTUAL"
180 PRINT "0",+0,"0",-0
190 PRINT "1",+1,"-1",-1
200 PRINT "12",12,"-12",-12
```

```
230 PRINT " 123",+123,"-123",-123
240 PRINT " 1234",1234,"-1234",-1234
250 PRINT " 12345",+12345,"-12345",-12345
260 PRINT " 123456",123456,"-123456",-123456
270 PRINT " 999999",+999999,"-999999",-999999
280 PRINT
290 PRINT "*** TEST PASSED IF COLUMNS 1 & 2, AND 3 & 4 CONTAIN THE SAME"
300 PRINT "      NUMERIC REPRESENTATION FOR EACH PAIR OF INTEGERS ***"
310 PRINT
320 PRINT "                                END TEST."
330 PRINT
340 PRINT "SECTION 9.2: NR1 CONSTANTS SEPARATED BY COMMAS TO PRODUCE"
350 PRINT "      TABULAR OUTPUT."
360 PRINT
370 PRINT "                                BEGIN TEST."
380 PRINT
390 PRINT "000000000111111112222222233333334444444445";
400 PRINT "555555555666666666777"
410 PRINT "12345678901234567890123456789012345678901234567890";
420 PRINT "1234567890123456789012"
430 PRINT 1,-12,123
440 PRINT -1234,12345,-123456
450 PRINT 123456,-1234,123
460 PRINT
470 PRINT "*** TEST PASSED IF THE FIRST THREE PRINT ZONES EACH CONTAIN"
480 PRINT "      A COLUMN OF INTEGERS ***"
490 PRINT
500 PRINT "                                END TEST."
510 PRINT
520 PRINT "SECTION 9.3: SPACE ALLOTTED FOR PRINTED INTEGER CONSTANTS."
530 PRINT
540 PRINT "PRINTED INTEGER CONSTANTS ARE PRECEDED BY A SPACE IF"
550 PRINT "POSITIVE, A MINUS IF NEGATIVE AND FOLLOWED BY A SPACE."
560 PRINT "THE SPACE FOR THE CONSTANT IS DELIMITED BY ASTERISKS."
570 PRINT
580 PRINT "                                BEGIN TEST."
590 PRINT
600 PRINT "000000000111111112222222233333334444444445";
610 PRINT "555555555666666666777"
620 PRINT "12345678901234567890123456789012345678901234567890";
630 PRINT "1234567890123456789012"
640 PRINT "SHOULD BE:", "* 1 **, **-23 **, ** 456 **"
650 PRINT "    ACTUAL:", "**;+1;**;**;-23;**;**;456;**"
660 PRINT
670 PRINT "SHOULD BE:", "* 7890 **, **-12345 **, ** 678901 **"
680 PRINT "    ACTUAL:", "**;7890;**,**;-12345;**,**;+678901;**"
690 PRINT
720 PRINT "*** TEST PASSED IF 'ACTUAL' LINES MATCH CORRESPONDING"
730 PRINT "    'SHOULD BE' LINES ***"
740 PRINT
750 PRINT "                                END TEST."
```

```
760 PRINT
770 PRINT "SECTION 9.4: NR1 CONSTANTS SEPARATED BY SEMICOLONS."
780 PRINT
790 PRINT "PRINTED INTEGER CONSTANTS ARE PRECEDED BY A SPACE IF"
800 PRINT "POSITIVE OR A MINUS IF NEGATIVE AND ARE FOLLOWED BY A SPACE."
810 PRINT "THE SEMICOLON SEPARATOR PRODUCES A NULL STRING."
820 PRINT
830 PRINT "                                BEGIN TEST."
840 PRINT
850 PRINT "0000000001111111122222222333333334444444445";
860 PRINT "55555555566666666777"
870 PRINT "12345678901234567890123456789012345678901234567890";
880 PRINT "1234567890123456789012"
885 PRINT "SHOULD BE: 0 0 1 -12 123 -1234 12345 -123456"
890 PRINT "    ACTUAL:";0;-0;1;-12;123;-1234;12345;-123456
900 PRINT
910 PRINT "*** TEST PASSED IF 'ACTUAL' LINE MATCHES CORRESPONDING"
920 PRINT "    'SHOULD BE' LINE ***"
930 PRINT
940 PRINT "                                END TEST."
950 PRINT
960 PRINT "SECTION 9.5: CONSTANTS IN NR2 FORM."
970 PRINT
980 PRINT "COLUMNS 1 AND 3 SHOW THE OUTPUT AS IT SHOULD BE."
995 PRINT "COLUMNS 2 AND 4 SHOW THE ACTUAL OUTPUT."
1001 PRINT
1002 PRINT "EXPECTED OUTPUT IS DISPLAYED WITH TRAILING ZEROS"
1003 PRINT "SUPPRESSED. IMPLEMENTATIONS MAY ADD TRAILING ZEROS UP TO"
1004 PRINT "THEIR SIGNIFICAND-WIDTH, D. THUS .12 MAY BE DISPLAYED AS"
1005 PRINT "'1.200' OR '.1200000' IF, FOR INSTANCE, D = 7."
1010 PRINT
1020 PRINT "                                BEGIN TEST."
1030 PRINT
1040 PRINT "0000000001111111122222222333333334444444445";
1050 PRINT "55555555566666666777"
1060 PRINT "12345678901234567890123456789012345678901234567890";
1070 PRINT "1234567890123456789012"
1080 PRINT "SHOULD BE","ACTUAL","SHOULD BE","ACTUAL"
1100 PRINT ".1 ",.1,"-.1 ",-.1
1110 PRINT ".12 ",+.12,"-.12 ",-.12
1120 PRINT ".123 ",.123,"-.123 ",-.123
1130 PRINT ".1234 ",+.1234,"-.1234 ",-.1234
1140 PRINT ".12345 ",.12345,"-.12345 ",-.12345
1150 PRINT ".123456 ",+.123456,"-.123456 ",-.123456
1160 PRINT ".234567 ",.234567,"-.234567 ",-.234567
1170 PRINT ".345678 ",+.345678,"-.345678 ",-.345678
1180 PRINT ".456789 ",.456789,"-.456789 ",-.456789
1190 PRINT ".56789 ",+.567890,"-.56789 ",-.567890
1200 PRINT ".6789 ",.678900,"-.6789 ",-.678900
1210 PRINT ".789 ",+.789000,"-.789 ",-.789000
1220 PRINT ".89 ",.890000,"-.89 ",-.890000
1230 PRINT ".9 ",+.900000,"-.9 ",-.900000
1240 PRINT "0 ",.0,0," 0 ",-.0,0
1250 PRINT "1.23456 ",1.23456,"-1.23456",-1.23456
1260 PRINT "9.876 ",+9.87600,"-9.876 ",-9.87600
```

```

1270 PRINT " 12.3456 ",12.3456,"-12.3456 ",-12.3456
1280 PRINT " 123.456 ",+123.456,"-123.456 ",-123.456
1290 PRINT " 1234.56 ",1234.56,"-1234.56 ",-1234.56
1300 PRINT " 12345.6 ",+12345.6,"-12345.6 ",-12345.6
1310 PRINT " 23456.7 ",23456.7,"-23456.7 ",-23456.7
1320 PRINT " 34567.8 ",+34567.8,"-34567.8 ",-34567.8
1330 PRINT " 45678.9 ",45678.9,"-45678.9 ",-45678.9
1340 PRINT " 56789.1 ",+56789.1,"-56789.1 ",-56789.1
1350 PRINT " 67891.2 ",67891.2,"-67891.2 ",-67891.2
1360 PRINT " 78912.3 ",+78912.3,"-78912.3 ",-78912.3
1370 PRINT " 89123.4 ",89123.4,"-89123.4 ",-89123.4
1380 PRINT " 91234.5 ",+91234.5,"-91234.5 ",-91234.5
1390 PRINT " 99999.9 ",99999.9,"-99999.9 ",-99999.9
1400 PRINT
1410 PRINT "*** TEST PASSED IF COLUMNS 1 & 2, AND 3 & 4 CONTAIN THE "
1420 PRINT " SAME REPRESENTATION FOR EACH PAIR OF NUMBERS "
1425 PRINT " (ALLOWING FOR OPTIONAL TRAILING ZEROS) ***"
1430 PRINT
1433 PRINT " END TEST."
1436 PRINT
1440 PRINT "SECTION 9.6: USING NR2 INSTEAD OF NR3 FORMAT FOR"
1450 PRINT " SIX-DIGIT CONSTANTS."
1460 PRINT
1490 PRINT "NUMBERS THAT CAN BE REPRESENTED IN THE UNSCALED"
1500 PRINT "REPRESENTATION NO LESS ACCURATELY THAN AS IN"
1510 PRINT "SCALED FORMAT, SHALL BE OUTPUT USING THE UNSCALED FORMAT."
1520 PRINT
1523 PRINT " BEGIN TEST."
1526 PRINT
1540 PRINT "SHOULD BE","ACTUAL","SHOULD BE","ACTUAL"
1550 PRINT
1560 PRINT " .022222 ",+.022222,"-.022222 ",-.022222
1570 PRINT " .004444 ",.004444,"-.004444 ",-.004444
1580 PRINT " .000888 ",+.000888,"-.000888 ",-.000888
1590 PRINT " .000044 ",.000044,"-.000044 ",-.000044
1600 PRINT " .000002 ",+.000002,"-.000002 ",-.000002
1610 PRINT
1620 PRINT "*** TEST PASSED IF COLUMNS 1 & 2, AND 3 & 4 CONTAIN THE "
1630 PRINT " SAME REPRESENTATION FOR EACH PAIR OF NUMBERS "
1635 PRINT " (ALLOWING FOR OPTIONAL TRAILING ZEROS) ***"
1640 PRINT
1650 PRINT " END TEST."
1660 PRINT
1670 PRINT "SECTION 9.7: NR2 CONSTANTS SEPARATED BY COMMAS TO PRODUCE"
1680 PRINT " TABULAR OUTPUT."
1690 PRINT
1700 PRINT " BEGIN TEST."
1710 PRINT
1720 PRINT "00000000011111111122222222333333334444444445";
1730 PRINT "5555555566666666777"
1740 PRINT "12345678901234567890123456789012345678901234567890";
1750 PRINT "1234567890123456789012"
1760 PRINT .123456,-99999.9,91234.5

```

```

1770 PRINT -1.23456,89123.4,-2.34567
1780 PRINT +295.65,.023,-67.8954
1790 PRINT
1800 PRINT "**** TEST PASSED IF THE FIRST THREE PRINT ZONES EACH CONTAIN"
1810 PRINT "      A COLUMN OF REAL NUMBERS USING EXPLICIT POINT,"
1820 PRINT "      UNSCALED REPRESENTATION ***"
1825 PRINT
1830 PRINT "                                END TEST."
1840 PRINT
1850 PRINT "SECTION 9.8: SPACE ALLOTMENT FOR PRINTED NR2 NUMBERS."
1860 PRINT
1870 PRINT "PRINTED NR2 CONSTANTS ARE PRECEDED BY A SPACE IF POSITIVE"
1880 PRINT "OR BY A MINUS IF NEGATIVE, AND ARE FOLLOWED BY A SPACE."
1890 PRINT "THE FIELDS FOR THE CONSTANTS ARE DELIMITED BY ASTERISKS."
1900 PRINT
1910 PRINT "                                BEGIN TEST."
1920 PRINT
1930 PRINT "0000000001111111122222222333333334444444445";
1940 PRINT "555555555666666666777"
1950 PRINT "1234567890123456789012345678901234567890";
1960 PRINT "1234567890123456789012"
1970 PRINT "***;90000.1;***;*.000123;***;*.000009;***"
1980 PRINT "***;+900.001;***;*;+0.000123;***;*;+0.000009;***"
1990 PRINT "***;-0.900001;***;*;-0.000123;***;*;-0.000009;***"
2000 PRINT
2010 PRINT "***;.234567;***;*;1.23456;***;*;91.2345;***"
2020 PRINT "***;+0.234567;***;*;+1.23456;***;*;91.2345;***"
2030 PRINT "***;-0.234567;***;*;-1.23456;***;*;-91.2345;***"
2040 PRINT
2050 PRINT "***;865.789;***;*;1234.56;***;*;99999.9;***"
2060 PRINT "***;+865.789;***;*;+1234.56;***;*;+99999.9;***"
2070 PRINT "***;-865.789;***;*;-1234.56;***;*;-99999.9;***"
2080 PRINT
2090 PRINT "**** TEST PASSED IF EACH NUMBER IS PRECEDED BY A SPACE OR"
2100 PRINT "      A MINUS SIGN, AND FOLLOWED BY A SPACE AND THE NUMBERS"
2110 PRINT "      USE EXPLICIT POINT, UNSCALED REPRESENTATION ***"
2120 PRINT
2130 PRINT "                                END TEST."
2335 PRINT
2340 PRINT "END PROGRAM 9"
2350 END

```

---

PROGRAM FILE 9: PRINTING NR1 AND NR2 NUMERIC CONSTANTS.

ANSI STANDARD 5.2, 5.4, 12.4

SECTION 9.1: CONSTANTS IN NR1 FORM (INTEGERS).

COLUMNS 1 AND 3 SHOW THE OUTPUT AS IT SHOULD BE.  
COLUMNS 2 AND 4 SHOW THE ACTUAL OUTPUT.

BEGIN TEST.

0000000001111111122222222233333333344444444455555555566666666777  
 12345678901234567890123456789012345678901234567890123456789012  
 SHOULD BE    ACTUAL            SHOULD BE    ACTUAL  
 0            0                0            0  
 1            1                -1          -1  
 12          12              -12         -12  
 123        123             -123       -123  
 1234       1234           -1234       -1234  
 12345       12345        -12345      -12345  
 123456       123456      -123456     -123456  
 999999       999999      -999999     -999999

\*\*\* TEST PASSED IF COLUMNS 1 & 2, AND 3 & 4 CONTAIN THE SAME  
 NUMERIC REPRESENTATION FOR EACH PAIR OF INTEGERS \*\*\*

END TEST.

SECTION 9.2: NR1 CONSTANTS SEPARATED BY COMMAS TO PRODUCE  
 TABULAR OUTPUT.

BEGIN TEST.

000000000111111112222222223333333344444444455555555566666666777  
 12345678901234567890123456789012345678901234567890123456789012  
 1            -12            123  
 -1234       12345       -123456  
 123456       -1234       123

\*\*\* TEST PASSED IF THE FIRST THREE PRINT ZONES EACH CONTAIN  
 A COLUMN OF INTEGERS \*\*\*

END TEST.

SECTION 9.3: SPACE ALLOTTED FOR PRINTED INTEGER CONSTANTS.

PRINTED INTEGER CONSTANTS ARE PRECEDED BY A SPACE IF  
 POSITIVE, A MINUS IF NEGATIVE AND FOLLOWED BY A SPACE.  
 THE SPACE FOR THE CONSTANT IS DELIMITED BY ASTERISKS.

BEGIN TEST.

0000000001111111122222222233333333344444444455555555566666666777  
 12345678901234567890123456789012345678901234567890123456789012

SHOULD BE:    \* 1 \*            \*-23 \*            \* 456 \*  
 ACTUAL:       \* 1 \*            \*-23 \*            \* 456 \*

SHOULD BE:    \* 7890 \*        \*-12345 \*       \* 678901 \*  
 ACTUAL:       \* 7890 \*        \*-12345 \*       \* 678901 \*

\*\*\* TEST PASSED IF 'ACTUAL' LINES MATCH CORRESPONDING  
 'SHOULD BE' LINES \*\*\*

END TEST.

## SECTION 9.4: NR1 CONSTANTS SEPARATED BY SEMICOLONS.

PRINTED INTEGER CONSTANTS ARE PRECEDED BY A SPACE IF  
 POSITIVE OR A MINUS IF NEGATIVE AND ARE FOLLOWED BY A SPACE.  
 THE SEMICOLON SEPARATOR PRODUCES A NULL STRING.

BEGIN TEST.

000000000111111112222222223333333344444444455555555666666666777  
 12345678901234567890123456789012345678901234567890123456789012  
 SHOULD BE: 0 0 1 -12 123 -1234 12345 -123456  
 ACTUAL: 0 0 1 -12 123 -1234 12345 -123456

\*\*\* TEST PASSED IF 'ACTUAL' LINE MATCHES CORRESPONDING  
 'SHOULD BE' LINE \*\*\*

END TEST.

## SECTION 9.5: CONSTANTS IN NR2 FORM.

COLUMNS 1 AND 3 SHOW THE OUTPUT AS IT SHOULD BE.  
 COLUMNS 2 AND 4 SHOW THE ACTUAL OUTPUT.

EXPECTED OUTPUT IS DISPLAYED WITH TRAILING ZEROS  
 SUPPRESSED. IMPLEMENTATIONS MAY ADD TRAILING ZEROS UP TO  
 THEIR SIGNIFICAND-WIDTH, D. THUS .12 MAY BE DISPLAYED AS  
 '.1200' OR '.120000' IF, FOR INSTANCE, D = 7.

BEGIN TEST.

000000000111111112222222233333333444444445555555566666666777	12345678901234567890123456789012345678901234567890123456789012	SHOULD BE	ACTUAL	SHOULD BE	ACTUAL
.1	.1	-.1	-.1		
.12	.12	-.12	-.12		
.123	.123	-.123	-.123		
.1234	.1234	-.1234	-.1234		
.12345	.12345	-.12345	-.12345		
.123456	.123456	-.123456	-.123456		
.234567	.234567	-.234567	-.234567		
.345678	.345678	-.345678	-.345678		
.456789	.456789	-.456789	-.456789		
.56789	.56789	-.56789	-.56789		
.6789	.6789	-.6789	-.6789		
.789	.789	-.789	-.789		
.89	.89	-.89	-.89		
.9	.9	-.9	-.9		
0	0	0	0		
1.23456	1.23456	-1.23456	-1.23456		
9.876	9.876	-9.876	-9.876		
12.3456	12.3456	-12.3456	-12.3456		
123.456	123.456	-123.456	-123.456		
1234.56	1234.56	-1234.56	-1234.56		
12345.6	12345.6	-12345.6	-12345.6		
23456.7	23456.7	-23456.7	-23456.7		

34567.8	34567.8	-34567.8	-34567.8
45678.9	45678.9	-45678.9	-45678.9
56789.1	56789.1	-56789.1	-56789.1
67891.2	67891.2	-67891.2	-67891.2
78912.3	78912.3	-78912.3	-78912.3
89123.4	89123.4	-89123.4	-89123.4
91234.5	91234.5	-91234.5	-91234.5
99999.9	99999.9	-99999.9	-99999.9

\*\*\* TEST PASSED IF COLUMNS 1 & 2, AND 3 & 4 CONTAIN THE  
SAME REPRESENTATION FOR EACH PAIR OF NUMBERS  
(ALLOWING FOR OPTIONAL TRAILING ZEROS) \*\*\*

END TEST.

SECTION 9.6: USING NR2 INSTEAD OF NR3 FORMAT FOR  
SIX-DIGIT CONSTANTS.

NUMBERS THAT CAN BE REPRESENTED IN THE UNSCALED  
REPRESENTATION NO LESS ACCURATELY THAN AS IN  
SCALED FORMAT, SHALL BE OUTPUT USING THE UNSCALED FORMAT.

BEGIN TEST.

SHOULD BE	ACTUAL	SHOULD BE	ACTUAL
.022222	.022222	-.022222	-.022222
.004444	.004444	-.004444	-.004444
.000888	.000888	-.000888	-.000888
.000044	.000044	-.000044	-.000044
.000002	.000002	-.000002	-.000002

\*\*\* TEST PASSED IF COLUMNS 1 & 2, AND 3 & 4 CONTAIN THE  
SAME REPRESENTATION FOR EACH PAIR OF NUMBERS  
(ALLOWING FOR OPTIONAL TRAILING ZEROS) \*\*\*

END TEST.

SECTION 9.7: NR2 CONSTANTS SEPARATED BY COMMAS TO PRODUCE  
TABULAR OUTPUT.

BEGIN TEST.

00000000011111111222222223333333444444445555555566666666777
12345678901234567890123456789012345678901234567890123456789012
.123456 -99999.9 91234.5
-1.23456 89123.4 -2.34567
295.65 .023 -67.8954

\*\*\* TEST PASSED IF THE FIRST THREE PRINT ZONES EACH CONTAIN  
A COLUMN OF REAL NUMBERS USING EXPLICIT POINT,  
UNSCALED REPRESENTATION \*\*\*

END TEST.

SECTION 9.8: SPACE ALLOTMENT FOR PRINTED NR2 NUMBERS.

PRINTED NR2 CONSTANTS ARE PRECEDED BY A SPACE IF POSITIVE OR BY A MINUS IF NEGATIVE, AND ARE FOLLOWED BY A SPACE. THE FIELDS FOR THE CONSTANTS ARE DELIMITED BY ASTERisks.

BEGIN TEST.

```

00000000011111111122222222233333333344444444555555555666666666777
123456789012345678901234567890123456789012345678901234567890123456789012
* 90000.1 *   * .000123 *   * .000009 *
* 900.001 *   * .000123 *   * .000009 *
*-.900001 *   *-.000123 *   *-.000009 *

* .234567 *   * 1.23456 *   * 91.2345 *
* .234567 *   * 1.23456 *   * 91.2345 *
*-.234567 *   *-1.23456 *   *-91.2345 *

* 865.789 *   * 1234.56 *   * 99999.9 *
* 865.789 *   * 1234.56 *   * 99999.9 *
*-.865.789 *   *-1234.56 *   *-99999.9 *

```

\*\*\* TEST PASSED IF EACH NUMBER IS PRECEDED BY A SPACE OR  
A MINUS SIGN, AND FOLLOWED BY A SPACE AND THE NUMBERS  
USE EXPLICIT POINT, UNSCALED REPRESENTATION \*\*\*

END TEST.

END PROGRAM 9

A large grid of black asterisks on a white background, arranged in approximately 20 rows and 30 columns.

```
10 PRINT "PROGRAM FILE 10: PRINTING NR3 NUMERIC CONSTANTS."
20 PRINT "      ANSI STANDARD 5.2, 5.4, 12.4"
30 PRINT
40 PRINT "THESE TESTS ASSUME THAT A MINIMUM OF SIX SIGNIFICANT DIGITS"
50 PRINT "WILL BE PRINTED. PROCESSORS WHICH PRINT MORE SIGNIFICANT"
60 PRINT "DIGITS SHOULD HAVE COMPARABLE ACCURACY."
70 PRINT
80 PRINT
90 PRINT
100 PRINT "NOTE THAT THE DISPLAYED STANDARD OUTPUT IS ORIENTED TO"
110 PRINT "THE MINIMAL REQUIREMENTS OF SIGNIFICAND-WIDTH, D, "
120 PRINT "EQUAL TO SIX, AND EXRAD-WIDTH, E, EQUAL TO TWO. FOR D > 6"
130 PRINT "AND E > 2, THERE MAY BE EXTRA TRAILING AND LEADING ZEROS,"
140 PRINT "RESPECTIVELY."
150 PRINT
160 PRINT "SECTION 10.1: POSITIVE NUMBER AND SIGNED SIGNIFICAND."
170 PRINT
180 PRINT "BEGIN TEST."
190 PRINT
200 PRINT "SOURCE FORM: SIGNED SIGNIFICAND - BOTH COLUMNS"
210 PRINT "              UNSIGNED EXRAD      - COLUMN 1"
220 PRINT "              SIGNED EXRAD      - COLUMN 2"
230 PRINT
```

```
200 PRINT +123456E27,+123456E+27
210 PRINT +123456.E27,+123456.E+27
220 PRINT +12345.6E28,+12345.6E+28
230 PRINT +1234.56E29,+1234.56E+29
240 PRINT +123.456E30,+123.456E+30
250 PRINT +12.3456E31,+12.3456E+31
260 PRINT +1.23456E32,+1.23456E+32
270 PRINT +.123456E33,.123456E+33
280 PRINT +1.2345600000000E32,+1.234560000000E+32
290 PRINT +.00000123456E38,.00000123456E+38
300 PRINT +.00001234560000E37,.00001234560000E+37
310 PRINT
320 PRINT "*** TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT PRINT"
330 PRINT " AS '1.23456E+32' ***"
340 PRINT
350 PRINT " END TEST."
360 PRINT
363 PRINT "SECTION 10.2: POSITIVE NUMBER AND UNSIGNED SIGNIFICAND."
366 PRINT
367 PRINT " BEGIN TEST."
368 PRINT
370 PRINT "SOURCE FORM: UNSIGNED SIGNIFICAND - BOTH COLUMNS"
380 PRINT " UNSIGNED EXRAD - COLUMN 1"
390 PRINT " SIGNED EXRAD - COLUMN 2"
400 PRINT
430 PRINT 123456E27,123456E+27
440 PRINT 123456.E27,123456.E+27
450 PRINT 12345.6E28,12345.6E+28
460 PRINT 1234.56E29,1234.56E+29
470 PRINT 123.456E30,123.456E+30
480 PRINT 12.3456E31,12.3456E+31
490 PRINT 1.23456E32,1.23456E+32
500 PRINT .123456E33,.123456E+33
510 PRINT 1.2345600000000E32,1.234560000000E+32
520 PRINT .00000123456E38,.00000123456E+38
530 PRINT .00001234560000E37,.00001234560000E+37
540 PRINT
550 PRINT "*** TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT PRINT"
560 PRINT " AS '1.23456E+32' ***"
570 PRINT
580 PRINT " END TEST."
590 PRINT
593 PRINT "SECTION 10.3: NEGATIVE NUMBER AND SIGNED SIGNIFICAND."
595 PRINT
597 PRINT " BEGIN TEST."
598 PRINT
600 PRINT "SOURCE FORM: SIGNED SIGNIFICAND - BOTH COLUMNS"
610 PRINT " UNSIGNED EXRAD - COLUMN 1"
620 PRINT " SIGNED EXRAD - COLUMN 2"
630 PRINT
```

```
660 PRINT -123456E27,-123456E+27
670 PRINT -123456.E27,-123456.E+27
680 PRINT -12345.6E28,-12345.6E+28
690 PRINT -1234.56E29,-1234.56E+29
700 PRINT -123.456E30,-123.456E+30
710 PRINT -12.3456E31,-12.3456E+31
720 PRINT -1.23456E32,-1.23456E+32
730 PRINT -.123456E33,-.123456E+33
740 PRINT -1.2345600000000E32,-1.2345600000000E+32
750 PRINT -.00000123456E38,-.00000123456E+38
760 PRINT -.00001234560000E37,-.00001234560000E+37
770 PRINT
780 PRINT "*** TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT PRINT"
790 PRINT "      AS '-1.23456E+32' ***"
800 PRINT
810 PRINT "                                END TEST."
820 PRINT
822 PRINT "SECTION 10.4: POSITIVE NUMBER AND NEGATIVE EXRAD."
824 PRINT
826 PRINT "                                BEGIN TEST."
828 PRINT
830 PRINT "SOURCE FORM: SIGNED EXRAD          - BOTH COLUMNS"
840 PRINT "              SIGNED SIGNIFICAND - COLUMN 1"
850 PRINT "              UNSIGNED SIGNIFICAND - COLUMN 2"
860 PRINT
880 PRINT +123456E-29,123456E-29
890 PRINT +123456.E-29,123456.E-29
900 PRINT +12345.6E-28,12345.6E-28
910 PRINT +1234.56E-27,1234.56E-27
920 PRINT +123.456E-26,123.456E-26
930 PRINT +12.3456E-25,12.3456E-25
940 PRINT +1.23456E-24,1.23456E-24
950 PRINT +.123456E-23,.123456E-23
960 PRINT +1.2345600000000E-24,1.2345600000000E-24
970 PRINT +.00000123456E-18,.00000123456E-18
980 PRINT +.00001234560000E-19,.00001234560000E-19
990 PRINT
1000 PRINT "*** TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT"
1010 PRINT "      PRINT AS '1.23456E-24' ***"
1020 PRINT
1030 PRINT "                                END TEST."
1040 PRINT
1042 PRINT "SECTION 10.5: NEGATIVE NUMBER AND NEGATIVE EXRAD."
1044 PRINT
1046 PRINT "                                BEGIN TEST."
1048 PRINT
1050 PRINT "SOURCE FORM: SIGNED SIGNIFICAND - COLUMN 1"
1060 PRINT "              SIGNED EXRAD          - COLUMN 1"
1070 PRINT
```



```

1610 PRINT "00000000011111111122222222333333334444444445";
1620 PRINT "55555555566666666777"
1630 PRINT "12345678901234567890123456789012345678901234567890";
1640 PRINT "1234567890123456789012"
1650 PRINT 1E30,-9.87E-37,1.23456E32
1660 PRINT -1.23456E32,+1.7865E36,5E-20
1670 PRINT +99E10,-765.32E32,+0.00082E-9
1700 PRINT
1710 PRINT "*** TEST PASSED IF THE FIRST THREE PRINT ZONES EACH CONTAIN"
1720 PRINT "      A COLUMN OF NR3 NUMBERS ***"
1730 PRINT
1733 PRINT "                                END TEST."
1736 PRINT
1740 PRINT "SECTION 10.8: WIDTH FOR PRINTED E-FORMAT CONSTANTS"
1745 PRINT
1750 PRINT "THE SPACE FOR THE CONSTANT IS DELIMITED BY ASTERISKS."
1760 PRINT
1770 PRINT "                                BEGIN TEST."
1780 PRINT
1790 PRINT "00000000011111111122222222333333334444444445";
1800 PRINT "55555555566666666777"
1810 PRINT "12345678901234567890123456789012345678901234567890";
1820 PRINT "1234567890123456789012"
1830 PRINT "*";1.23456E32; "*"
1840 PRINT "*";-1.23456E32; "*"
1850 PRINT "*";1.23456E-32; "*"
1860 PRINT "*";-1.23456E-32; "*"
1870 PRINT "*";0.123456E32; "*"
1880 PRINT
1890 PRINT "*** TEST PASSED IF THE CONSTANTS ABOVE ARE PRECEDED BY A"
1900 PRINT "      SPACE OR A MINUS AND ARE FOLLOWED BY A SPACE ***"
1910 PRINT
1920 PRINT "                                END TEST."
1930 PRINT
2140 PRINT "END PROGRAM 10"
2150 END

```

---

#### PROGRAM FILE 10: PRINTING NR3 NUMERIC CONSTANTS.

ANSI STANDARD 5.2, 5.4, 12.4

THESE TESTS ASSUME THAT A MINIMUM OF SIX SIGNIFICANT DIGITS WILL BE PRINTED. PROCESSORS WHICH PRINT MORE SIGNIFICANT DIGITS SHOULD HAVE COMPARABLE ACCURACY.

NOTE THAT THE DISPLAYED STANDARD OUTPUT IS ORIENTED TO THE MINIMAL REQUIREMENTS OF SIGNIFICAND-WIDTH, D, EQUAL TO SIX, AND EXRAD-WIDTH, E, EQUAL TO TWO. FOR D > 6 AND E > 2, THERE MAY BE EXTRA TRAILING AND LEADING ZEROS, RESPECTIVELY.

SECTION 10.1: POSITIVE NUMBER AND SIGNED SIGNIFICAND.

BEGIN TEST.

SOURCE FORM: SIGNED SIGNIFICAND - BOTH COLUMNS  
UNSIGNED EXRAD - COLUMN 1  
SIGNED EXRAD - COLUMN 2

\*\*\* TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT PRINT AS '1.23456E+32' \*\*\*

END TEST.

## SECTION 10.2: POSITIVE NUMBER AND UNSIGNED SIGNIFICAND.

BEGIN TEST.

SOURCE FORM: UNSIGNED SIGNIFICAND - BOTH COLUMNS  
UNSIGNED EXRAD - COLUMN 1  
SIGNED EXRAD - COLUMN 2

\*\*\* TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT PRINT AS '1.23456E+32' \*\*\*

END TEST.

SECTION 10.3: NEGATIVE NUMBER AND SIGNED SIGNIFICANT.

BEGIN TEST.

SOURCE FORM: SIGNED SIGNIFICAND - BOTH COLUMNS

UNSIGNED EXRAD - COLUMN 1

SIGNED EXRAD - COLUMN 2

-1.23456E+32 -1.23456E+32  
-1.23456E+32 -1.23456E+32

\*\*\* TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT PRINT  
AS '-1.23456E+32' \*\*\*

END TEST.

SECTION 10.4: POSITIVE NUMBER AND NEGATIVE EXRAD.

BEGIN TEST.

SOURCE FORM: SIGNED EXRAD - BOTH COLUMNS

SIGNED SIGNIFICAND - COLUMN 1

UNSIGNED SIGNIFICAND - COLUMN 2

1.23456E-24 1.23456E-24  
1.23456E-24 1.23456E-24

\*\*\* TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT  
PRINT AS '1.23456E-24' \*\*\*

END TEST.

SECTION 10.5: NEGATIVE NUMBER AND NEGATIVE EXRAD.

BEGIN TEST.

SOURCE FORM: SIGNED SIGNIFICAND - COLUMN 1

SIGNED EXRAD - COLUMN 1

-1.23456E-24  
-1.23456E-24

\*\*\* TEST PASSED IF ALL THE NUMBERS IN THE ABOVE OUTPUT  
PRINT AS '-1.23456E-24' \*\*\*

END TEST.

#### SECTION 10.6: MISCELLANEOUS NR3 CONSTANTS.

ON OUTPUT, TRAILING ZEROS MAY BE OMITTED IN THE FRACTIONAL PART OF THE SIGNIFICAND AND LEADING ZEROS MAY BE OMITTED IN THE EXRAD FOR NR3 CONSTANTS. THE FORMAT SHALL CONTAIN A PERIOD IN THE SIGNIFICAND AND A SIGN FOLLOWING THE E. THE SIGNIFICAND MUST BE  $\geq 1$  AND  $< 10$ .

BEGIN TEST.

(EXPECTED OUTPUT DISPLAYED WITH ZEROS SUPPRESSED.)

SOURCE CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
1.E30	1.E+30	1.00000E+30
10.E30	1.E+31	1.00000E+31
1E09	1.E+9	1.00000E+9
25.01E036	2.501E+37	2.50100E+37
25.01E36	2.501E+37	2.50100E+37
-1.E30	-1.E+30	-1.00000E+30
1E-30	1.E-30	1.00000E-30
-1E-30	-1.E-30	-1.00000E-30
1.000E034	1.E+34	1.00000E+34

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.

#### SECTION 10.7: NR3 CONSTANTS SEPARATED BY COMMAS TO PRODUCE TABULAR OUTPUT.

BEGIN TEST.

```
000000000111111111222222222333333334444444455555555666666666777
12345678901234567890123456789012345678901234567890123456789012
1.00000E+30 -9.87000E-37 1.23456E+32
-1.23456E+32 1.78650E+36 5.00000E-20
9.90000E+11 -7.65320E+34 8.20000E-13
```

\*\*\* TEST PASSED IF THE FIRST THREE PRINT ZONES EACH CONTAIN  
A COLUMN OF NR3 NUMBERS \*\*\*

END TEST.

#### SECTION 10.8: WIDTH FOR PRINTED E-FORMAT CONSTANTS

THE SPACE FOR THE CONSTANT IS DELIMITED BY ASTERISKS.

BEGIN TEST.

```
000000000111111111222222222333333334444444455555555666666666777
12345678901234567890123456789012345678901234567890123456789012
* 1.23456E+32 *
*-1.23456E+32 *
* 1.23456E-32 *
*-1.23456E-32 *
* 1.23456E+31 *
```

\*\*\* TEST PASSED IF THE CONSTANTS ABOVE ARE PRECEDED BY A  
SPACE OR A MINUS AND ARE FOLLOWED BY A SPACE \*\*\*

END TEST.

END PROGRAM 10

```
*****
*****
```

```
10 PRINT "PROGRAM FILE 11: PRINTING NUMERIC VARIABLES ASSIGNED"
15 PRINT "          NR1 AND NR2 CONSTANTS."
20 PRINT "          ANSI STANDARD 5.2, 5.4, 6.2, 6.4, 9.2, 9.4, 12.4"
70 PRINT
80 PRINT "SECTION 11.1: PRINTING VARIABLES WHICH HAVE BEEN ASSIGNED"
85 PRINT "          INTEGER (NR1) CONSTANTS."
90 PRINT
100 PRINT "                                BEGIN TEST."
110 PRINT
310 LET M2=+2
320 LET L2=-3
330 LET N2=-998765
340 LET O2=+6912
342 LET M1=1
344 LET J1=12345
346 LET K1=000
350 PRINT "ASSIGNED","OUTPUT","ACTUAL"
360 PRINT "CONSTANT","SHOULD BE","OUTPUT"
370 PRINT
```

```
372 PRINT " 1 "," 1 ",M1
374 PRINT " 12345 "," 12345 ",J1
376 PRINT " 000 "," 0 ",K1
380 PRINT "+2"," 2 ",M2
390 PRINT "-3","-3",L2
400 PRINT "-998765","-998765",N2
410 PRINT "+6912"," 6912",O2
420 PRINT
430 PRINT "*** TEST PASSED IF THE ACTUAL OUTPUT MATCHES THE OUTPUT"
440 PRINT "    AS IT SHOULD BE ***"
450 PRINT
460 PRINT "                                END TEST."
470 PRINT
480 PRINT "SECTION 11.2: TRANSITIVE ASSIGNMENT OF AN NR1 CONSTANT."
490 PRINT
500 PRINT "                                BEGIN TEST."
510 PRINT
520 LET A=-99999
530 LET B=A
540 LET C=B
550 LET D=C
560 LET E=D
570 LET F=E
580 LET G=F
590 LET H=G
600 LET I=H
610 LET J=I
620 LET K=J
630 LET L=K
640 LET M=L
650 LET N=M
660 LET O=N
670 LET P=O
680 LET Q=P
690 LET R=Q
700 LET S=R
710 LET T=S
720 LET U=T
730 LET V=U
740 LET W=V
750 LET X=W
760 LET Y=X
770 LET Z=Y
780 PRINT "ASSIGNED","OUTPUT","ACTUAL"
790 PRINT "CONSTANT","SHOULD BE","OUTPUT"
800 PRINT
810 PRINT "-99999 ","-99999 ",Z
820 PRINT
830 PRINT "*** TEST PASSES IF THE ACTUAL OUTPUT MATCHES THE OUTPUT"
840 PRINT "    AS IT SHOULD BE ***"
850 PRINT
860 PRINT "                                END TEST."
```

```
870 PRINT
900 PRINT "SECTION 11.3: PRINTING VARIABLES WHICH HAVE BEEN ASSIGNED"
905 PRINT "          NR2 CONSTANTS."
910 PRINT
1000 PRINT "                                BEGIN TEST."
1120 PRINT
1130 LET A4=-2.1
1140 LET B4=+3.1
1150 LET C4=-2714.25
1160 LET D4=+29.3054
1170 LET E4=-0.00
1172 LET A3=1.05
1174 LET B3=358.672
1176 LET C3=0.0
1180 PRINT "ASSIGNED", "OUTPUT", "ACTUAL"
1190 PRINT "CONSTANT", "SHOULD BE", "OUTPUT"
1200 PRINT
1202 PRINT " 1.05 "," 1.05 ",A3
1204 PRINT " 358.672 "," 358.672 ",B3
1206 PRINT " 0.0 "," 0 ",C3
1210 PRINT "-2.1 ","-2.1 ",A4
1220 PRINT "+3.1 "," 3.1 ",B4
1230 PRINT "-2714.25 ","-2714.25 ",C4
1240 PRINT "+29.3054 "," 29.3054 ",D4
1250 PRINT "-0.00 "," 0 ",E4
1260 PRINT
1270 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
1280 PRINT "      SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS) ***"
1290 PRINT
1300 PRINT "                                END TEST."
1310 PRINT
1320 PRINT "SECTION 11.4: TRANSITIVE ASSIGNMENT OF AN NR2 CONSTANT."
1330 PRINT
1340 PRINT "                                BEGIN TEST."
1350 PRINT
1360 LET A0=-9999.99
1370 LET B0=A0
1380 LET C0=B0
1390 LET D0=C0
1400 LET E0=D0
1410 LET F0=E0
1420 LET G0=F0
1430 LET H0=G0
1440 LET I0=H0
1450 LET J0=I0
1460 LET K0=J0
1470 LET L0=K0
1480 LET M0=L0
1490 LET N0=M0
1500 LET O0=N0
1510 LET P0=O0
1520 LET Q0=P0
1530 LET R0=Q0
1540 LET S0=R0
1550 LET T0=S0
```

```
1560 LET U0=T0
1570 LET V0=U0
1580 LET W0=V0
1590 LET X0=W0
1600 LET Y0=X0
1610 LET Z0=Y0
1620 PRINT "ASSIGNED","OUTPUT","ACTUAL"
1630 PRINT "CONSTANT","SHOULD BE","OUTPUT"
1640 PRINT
1650 PRINT "-9999.99 ","-9999.99 ",Z0
1660 LET A5=1.5
1670 LET B5=-2.5
1680 LET C5=+3.5
1690 LET D5=4.5
1700 LET E5=+5.5
1710 LET F5=-6.5
1720 LET A1=A5
1730 LET B1=B5
1740 LET C1=C5
1750 LET D1=D5
1760 LET E1=E5
1770 LET F1=F5
1780 LET G1=A1
1790 LET H1=B1
1800 LET I1=C1
1810 LET J1=D1
1820 LET K1=E1
1830 LET L1=F1
1840 LET M1=G1
1850 LET N1=H1
1860 LET O1=I1
1870 LET P1=J1
1880 LET Q1=K1
1890 LET R1=L1
1900 LET S1=M1
1910 LET T1=N1
1920 LET U1=O1
1930 LET V1=P1
1940 LET W1=Q1
1950 LET X1=R1
1960 PRINT " 1.5 "," 1.5 ",S1
1970 PRINT "-2.5 ","-2.5 ",T1
1980 PRINT "+3.5 "," 3.5 ",U1
1990 PRINT " 4.5 "," 4.5 ",V1
2000 PRINT "+5.5 "," 5.5 ",W1
2010 PRINT "-6.5 ","-6.5 ",X1
2020 LET Y1=9.99999
2030 LET Z1=Y1
2040 PRINT " 9.99999 "," 9.99999 ",Z1
2050 PRINT
2060 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
2070 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS) ***"
2080 PRINT
2090 PRINT " END TEST."
```

```
2100 PRINT
2110 PRINT "END PROGRAM 11"
2120 END
```

---

PROGRAM FILE 11: PRINTING NUMERIC VARIABLES ASSIGNED  
NR1 AND NR2 CONSTANTS.

ANSI STANDARD 5.2, 5.4, 6.2, 6.4, 9.2, 9.4, 12.4

SECTION 11.1: PRINTING VARIABLES WHICH HAVE BEEN ASSIGNED  
INTEGER (NR1) CONSTANTS.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
1	1	1
12345	12345	12345
000	0	0
+2	2	2
-3	-3	-3
-998765	-998765	-998765
+6912	6912	6912

\*\*\* TEST PASSED IF THE ACTUAL OUTPUT MATCHES THE OUTPUT  
AS IT SHOULD BE \*\*\*

END TEST.

SECTION 11.2: TRANSITIVE ASSIGNMENT OF AN NR1 CONSTANT.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
-99999	-99999	-99999

\*\*\* TEST PASSES IF THE ACTUAL OUTPUT MATCHES THE OUTPUT  
AS IT SHOULD BE \*\*\*

END TEST.

SECTION 11.3: PRINTING VARIABLES WHICH HAVE BEEN ASSIGNED  
NR2 CONSTANTS.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
----------------------	---------------------	------------------

1.05	1.05	1.05
358.672	358.672	358.672
0.0	0	0
-2.1	-2.1	-2.1
+3.1	3.1	3.1
-2714.25	-2714.25	-2714.25
+29.3054	29.3054	29.3054
-0.00	0	0

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS) \*\*\*

END TEST.

#### SECTION 11.4: TRANSITIVE ASSIGNMENT OF AN NR2 CONSTANT.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
-9999.99	-9999.99	-9999.99
1.5	1.5	1.5
-2.5	-2.5	-2.5
+3.5	3.5	3.5
4.5	4.5	4.5
+5.5	5.5	5.5
-6.5	-6.5	-6.5
9.99999	9.99999	9.99999

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS) \*\*\*

END TEST.

END PROGRAM 11

TIME: 2.68 SECS.

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 12: PRINTING NUMERIC VARIABLES ASSIGNED"
15 PRINT "          NR3 CONSTANTS."
20 PRINT "          ANSI STANDARD 5.2, 5.4, 6.2, 6.4, 9.2, 9.4, 12.4"
50 PRINT
60 PRINT "SECTION 12.1: ASSIGNING NR3 CONSTANTS WHICH PRINT EITHER"
70 PRINT "          AS NR1 OR NR2 CONSTANTS, SIGNED AND UNSIGNED. "
80 PRINT
90 PRINT "          BEGIN TEST."
100 PRINT

```

```
110 LET A7=1.05E02
120 LET B7=-7.6E1
130 LET C7=+332.4E0
140 LET D7=51.32E-1
150 LET E7=+5.34E-3
160 LET F7=-14.19E-2
170 LET G7=-9.9E+2
180 LET H7=+10.5210E+3
190 LET I7=4.56E+1
200 PRINT "ASSIGNED","OUTPUT","ACTUAL"
210 PRINT "CONSTANT","SHOULD BE","OUTPUT"
220 PRINT
230 PRINT " 1.05E02"," 105 ",A7
240 PRINT "-7.6E1 ","-76 ",B7
250 PRINT "+332.4E0"," 332.4 ",C7
260 PRINT " 51.32E-1"," 5.132 ",D7
270 PRINT "+5.34E-3 "," .00534 ",E7
280 PRINT "-14.19E-2 ","-.1419 ",F7
290 PRINT "-9.9E+2 ","-990 ",G7
300 PRINT "+10.5210E+3 "," 10521 ",H7
310 PRINT " 4.56E+1 "," 45.6 ",I7
320 PRINT
330 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
340 PRINT "      SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS) ***"
350 PRINT
360 PRINT "
370 PRINT
380 PRINT "SECTION 12.2: ASSIGNING NR3 FORM CONSTANTS WHICH PRINT AS"
390 PRINT "      NR3 CONSTANTS, SIGNED AND UNSIGNED."
400 PRINT
410 PRINT "
420 PRINT
430 LET A8=1.E30
440 LET B8=+123.E20
450 LET C8=-11.E30
460 LET D8=144.E-21
470 LET E8=-12.E-22
480 LET F8=+3645.E-23
490 LET G8=1.E+34
500 LET H8=-200.E+21
510 LET I8=+99.E+32
520 LET A9=.234E20
530 LET B9=-.3E22
540 LET C9=+.44E17
550 LET D9=.36E-33
560 LET E9=+.9E-24
570 LET F9=-.10E-25
580 LET G9=.777E+18
590 LET H9=-.29E+31
600 LET I9=+.04E+26
610 LET J1=709876000000000E+22
620 LET J2=+81E36
630 LET J3=-000.00000000192837E-25
640 LET J4=627E+27
650 LET J5=+53E+19
```

```
660 LET J6=-4E+28
670 LET J7=1463E-29
680 LET J8=+2E-37
690 LET J9=-355E-19
700 PRINT "ASSIGNED","OUTPUT","ACTUAL"
710 PRINT "CONSTANT","SHOULD BE","OUTPUT"
720 PRINT
730 PRINT " 1.E30 "," 1.E+30 ",A8
740 PRINT "+123.E20 "," 1.23E+22 ",B8
750 PRINT "-11.E30 ","-1.1E+31 ",C8
760 PRINT " 144.E-21 "," 1.44E-19 ",D8
770 PRINT "-12.E-22 ","-1.2E-21 ",E8
780 PRINT "+3645.E-23 "," 3.645E-20 ",F8
790 PRINT " 1.E+34 "," 1.E+34 ",G8
800 PRINT "-200.E+21 ","-2.E+23 ",H8
810 PRINT "+99.E+32 "," 9.9E+33 ",I8
820 PRINT " .234E20 "," 2.34E+19 ",A9
830 PRINT "-.3E22 ","-3.E+21 ",B9
840 PRINT "+.44E17 "," 4.4E+16 ",C9
850 PRINT " .36E-33 "," 3.6E-34 ",D9
860 PRINT "+.9E-24 "," 9.E-25 ",E9
870 PRINT "-.10E-25 ","-1.E-26 ",F9
880 PRINT " .777E+18 "," 7.77E+17 ",G9
890 PRINT "-.29E+31 ","-2.9E+30 ",H9
900 PRINT "+.04E+26 "," 4.E+24 ",I9
910 PRINT " 709876000000000E+22 "
915 PRINT      " 7.09876E+35 ",J1
920 PRINT "+81E36 "," 8.1E+37 ",J2
930 PRINT "-000.000000000192837E-25"
935 PRINT      " -1.92837E-35 ",J3
940 PRINT " 627E+27 "," 6.27E+29 ",J4
950 PRINT "+53E+19 "," 5.3E+20 ",J5
960 PRINT "-4E+28 ","-4.E+28 ",J6
970 PRINT " 1463E-29 "," 1.463E-26 ",J7
980 PRINT "+2E-37 "," 2.E-37 ",J8
990 PRINT "-355E-19 ","-3.55E-17 ",J9
1000 PRINT
1010 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
1020 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN"
1030 PRINT " THE SIGNIFICAND OR LEADING ZEROS IN THE EXRAD) ***"
1040 PRINT
1050 PRINT "                                              END TEST."
1060 PRINT
1070 PRINT "SECTION 12.3: TRANSITIVE ASSIGNMENT OF AN NR3 CONSTANT."
1080 PRINT
1090 PRINT "                                              BEGIN TEST."
1100 PRINT
1110 LET A2=1.E30
1120 LET B2=A2
1130 LET C2=B2
1140 LET D2=C2
1150 LET C3=D2
1160 LET D3=C3
1170 LET A6=D3
1180 LET B6=A6
```

```
1190 LET C6=B6
1200 LET D6=C6
1210 LET E2=D6
1220 LET E3=E2
1230 LET E4=E3
1240 LET E6=E4
1250 LET F2=E6
1260 LET F3=F2
1270 LET F4=F3
1280 LET F6=F4
1290 LET G2=F6
1300 LET G3=G2
1310 LET G4=G3
1320 LET G5=G4
1330 LET G6=G5
1340 LET H2=G6
1350 LET H3=H2
1360 LET H4=H3
1370 LET H5=H4
1380 LET H6=H5
1390 LET I2=H6
1400 LET I3=I2
1410 LET I4=I3
1420 LET I5=I4
1440 LET I6=I5
1445 PRINT
1450 PRINT "ASSIGNED","OUTPUT","ACTUAL"
1460 PRINT "CONSTANT","SHOULD BE","OUTPUT"
1470 PRINT
1480 PRINT " 1.E30 "," 1.E+30 ",I6
1490 PRINT
1500 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
1510 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN"
1520 PRINT " THE SIGNIFICAND OR LEADING ZEROS IN THE EXRAD) ***"
1530 PRINT
1540 PRINT "
1550 PRINT END TEST."
1560 PRINT "END PROGRAM 12"
1570 END
```

---

PROGRAM FILE 12: PRINTING NUMERIC VARIABLES ASSIGNED  
NR3 CONSTANTS.

ANSI STANDARD 5.2, 5.4, 6.2, 6.4, 9.2, 9.4, 12.4

SECTION 12.1: ASSIGNING NR3 CONSTANTS WHICH PRINT EITHER  
AS NR1 OR NR2 CONSTANTS, SIGNED AND UNSIGNED.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
----------------------	---------------------	------------------

1.05E02	105	105
-7.6E1	-76	-76
+332.4E0	332.4	332.4
51.32E-1	5.132	5.132
+5.34E-3	.00534	.00534
-14.19E-2	-.1419	-.1419
-9.9E+2	-990	-990
+10.5210E+3	10521	10521
4.56E+1	45.6	45.6

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS) \*\*\*

END TEST.

SECTION 12.2: ASSIGNING NR3 FORM CONSTANTS WHICH PRINT AS  
NR3 CONSTANTS, SIGNED AND UNSIGNED.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
1.E30	1.E+30	1.00000E+30
+123.E20	1.23E+22	1.23000E+22
-11.E30	-1.1E+31	-1.10000E+31
144.E-21	1.44E-19	1.44000E-19
-12.E-22	-1.2E-21	-1.20000E-21
+3645.E-23	3.645E-20	3.64500E-20
1.E+34	1.E+34	1.00000E+34
-200.E+21	-2.E+23	-2.00000E+23
+99.E+32	9.9E+33	9.90000E+33
.234E20	2.34E+19	2.34000E+19
-.3E22	-3.E+21	-3.00000E+21
+.44E17	4.4E+16	4.40000E+16
.36E-33	3.6E-34	3.60000E-34
+.9E-24	9.E-25	9.00000E-25
-.10E-25	-1.E-26	-1.00000E-26
.777E+18	7.77E+17	7.77000E+17
-.29E+31	-2.9E+30	-2.90000E+30
+.04E+26	4.E+24	4.00000E+24
709876000000000E+22		
	7.09876E+35	7.09876E+35
+81E36	8.1E+37	8.10000E+37
-000.00000000192837E-25		
	-1.92837E-35	-1.92837E-35
627E+27	6.27E+29	6.27000E+29
+53E+19	5.3E+20	5.30000E+20
-4E+28	-4.E+28	-4.00000E+28
1463E-29	1.463E-26	1.46300E-26
+2E-37	2.E-37	2.00000E-37
-355E-19	-3.55E-17	-3.55000E-17

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN  
THE SIGNIFICAND OR LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.

SECTION 12.3: TRANSITIVE ASSIGNMENT OF AN NR3 CONSTANT.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
----------------------	---------------------	------------------

1.E30	1.E+30	1.00000E+30
-------	--------	-------------

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN  
THE SIGNIFICAND OR LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.

END PROGRAM 12

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 13: FORMAT AND ROUNDING OF PRINTED"
20 PRINT "           NUMERIC CONSTANTS."
30 PRINT "           ANSI STANDARD 12.4, 5.2, 5.4"
40 PRINT
50 PRINT "SECTION 13.1: PRINTED REPRESENTATION OF CONSTANTS."
60 PRINT
70 PRINT "THE FORMAT OF A PRINTED CONSTANT IS DETERMINED BY THE VALUE"
80 PRINT "OF THE CONSTANT RATHER THAN BY ITS ORIGINAL REPRESENTATION."
90 PRINT "THE SUPPRESSION OF LEADING INSIGNIFICANT ZEROS IN THE EXRAD"
100 PRINT "OR TRAILING INSIGNIFICANT ZEROS IN THE SIGNIFICAND"
110 PRINT "IS OPTIONAL. "
120 PRINT
130 PRINT "                               BEGIN TEST."
140 PRINT
150 PRINT "#","CONSTANT","NR1","NR2","NR3"
160 PRINT
170 PRINT "   1"," 76767",76767
180 PRINT "   2"," 76767.0",76767.0
190 PRINT "   3"," 767.670E2",767.670E2
200 PRINT "   4","-.987789",,-.987789
210 PRINT "   5","-.000987789E3",,-.000987789E3
220 PRINT "   6","-9.87789E-1",,-9.87789E-1
230 PRINT "   7"," 1230000000",,,1230000000
240 PRINT "   8"," .0000012345",,,.0000012345
250 PRINT "   9"," 2.3E9",,,2.3E9
260 PRINT

```

```

270 PRINT "*** TEST PASSED IF THE CONSTANTS IN ROWS 1, 2 AND 3 PRINTED"
280 PRINT "    IN NR1 FORMAT; IN ROWS 4, 5 AND 6 IN NR2 FORMAT; AND IN"
290 PRINT "    ROWS 7, 8 AND 9 IN NR3 FORMAT ***"
300 PRINT
310 PRINT "EXCEPT: IF SIGNIFICANCE-WIDTH FOR THIS IMPLEMENTATION IS"
320 PRINT "GREATER THAN 9, ROWS 7, 8, AND 9 MUST BE IN NR1, NR2,"
330 PRINT "AND NR1 FORMAT, RESPECTIVELY."
340 PRINT
350 PRINT "                                END TEST."
360 PRINT
370 PRINT "SECTION 13.2: ROUNDING RESULTS."
380 PRINT
390 PRINT "THE NUMBER OF SIGNIFICANT DIGITS PRINTED IS IMPLEMENTATION"
400 PRINT "DEFINED; HOWEVER, NR2 AND NR3 CONSTANTS WHICH CONTAIN EXCESS"
410 PRINT "DIGITS SHOULD BE PRINTED IN A STANDARD REPRESENTATION."
420 PRINT
430 PRINT "                                BEGIN TEST."
440 PRINT
450 LET T1=30
460 PRINT "SOURCE CONSTANT";TAB(T1);"PROCESSOR OUTPUT"
470 PRINT
480 PRINT "1 1234567886";TAB(T1);1234567886
490 PRINT "2 .000001234567886";TAB(T1);.000001234567886
500 PRINT "3 9.999999999";TAB(T1);9.999999999
510 PRINT "4 923456.7886";TAB(T1);923456.7886
520 PRINT "5 -0.09234567886";TAB(T1);-0.09234567886
530 PRINT "6 .0444444444";TAB(T1);.0444444444
540 PRINT "7 .001200000004";TAB(T1);.001200000004
550 PRINT
560 PRINT "CORRECT REPRESENTATION FOR PROCESSORS PRINTING FROM 6 TO 9"
570 PRINT "SIGNIFICANT DIGITS"
580 PRINT
590 PRINT "D = 6","D = 7","D = 8","D = 9"
600 PRINT
610 PRINT "1 1.23457E+9","1.234568E+9","1.2345679E+9","1.23456789E+9"
620 PRINT "2 1.23457E-6","1.234568E-6","1.2345679E-6","1.23456789E-6"
630 PRINT "3 10","10","10","10"
640 PRINT "(10.)","(10.)","(10.)","(10.)"
650 PRINT "4 923457.","923456.8","923456.79","923456.789"
660 PRINT "(923457)"
670 PRINT "5 -9.23457E-2","-9.234568E-2",
680 PRINT "-9.2345679E-2","-9.23456789E-2"
690 PRINT "6 4.44444E-2","4.444444E-2","4.4444444E-2","4.44444444E-2"
700 PRINT "7 .001200",".0012000",".00120000",".001200000"
710 PRINT
720 PRINT "*** TEST PASSED IF THE PROCESSOR OUTPUT IN THE FIRST SET"
730 PRINT "OF NUMBERED ROWS MATCHES THE COLUMN CORRESPONDING"
740 PRINT "TO THE IMPLEMENTATION-DEFINED SIGNIFICANCE-WIDTH"
750 PRINT "(ALLOWING FOR OPTIONAL TRAILING ZEROS IN THE"
755 PRINT "SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) ***"
760 PRINT
762 PRINT "                                END TEST."

```

```
765 PRINT
770 PRINT "END PROGRAM 13"
780 END
```

---

PROGRAM FILE 13: FORMAT AND ROUNDING OF PRINTED  
NUMERIC CONSTANTS.

ANSI STANDARD 12.4, 5.2, 5.4

SECTION 13.1: PRINTED REPRESENTATION OF CONSTANTS.

THE FORMAT OF A PRINTED CONSTANT IS DETERMINED BY THE VALUE  
OF THE CONSTANT RATHER THAN BY ITS ORIGINAL REPRESENTATION.  
THE SUPPRESSION OF LEADING INSIGNIFICANT ZEROS IN THE EXRAD  
OR TRAILING INSIGNIFICANT ZEROS IN THE SIGNIFICAND  
IS OPTIONAL.

BEGIN TEST.

#	CONSTANT	NR1	NR2	NR3
1	76767	76767		
2	76767.0	76767		
3	767.670E2	76767		
4	-.987789		-0.987789	
5	-.000987789E3		-0.987789	
6	-9.87789E-1		-0.987789	
7	1230000000			1.23000E+9
8	.0000012345			1.23450E-6
9	2.3E9			2.30000E+9

\*\*\* TEST PASSED IF THE CONSTANTS IN ROWS 1, 2 AND 3 PRINTED  
IN NR1 FORMAT; IN ROWS 4, 5 AND 6 IN NR2 FORMAT; AND IN  
ROWS 7, 8 AND 9 IN NR3 FORMAT \*\*\*

EXCEPT: IF SIGNIFICANCE-WIDTH FOR THIS IMPLEMENTATION IS  
GREATER THAN 9, ROWS 7, 8, AND 9 MUST BE IN NR1, NR2,  
AND NR1 FORMAT, RESPECTIVELY.

END TEST.

SECTION 13.2: ROUNDING RESULTS.

THE NUMBER OF SIGNIFICANT DIGITS PRINTED IS IMPLEMENTATION  
DEFINED; HOWEVER, NR2 AND NR3 CONSTANTS WHICH CONTAIN EXCESS  
DIGITS SHOULD BE PRINTED IN A STANDARD REPRESENTATION.

BEGIN TEST.

SOURCE CONSTANT	PROCESSOR OUTPUT
-----------------	------------------

1	1234567886	1.23457E+9
2	.000001234567886	1.23457E-6
3	9.999999999	10.
4	923456.7886	923457.
5	-0.09234567886	-9.23457E-2
6	.044444444444	4.44444E-2
7	.001200000004	.0012

CORRECT REPRESENTATION FOR PROCESSORS PRINTING FROM 6 TO 9  
SIGNIFICANT DIGITS

D = 6            D = 7            D = 8            D = 9

1	1.23457E+9	1.234568E+9	1.2345679E+9	1.23456789E+9
2	1.23457E-6	1.234568E-6	1.2345679E-6	1.23456789E-6
3	10	10	10	10
(10.)	(10.)	(10.)	(10.)	(10.)
4	923457.	923456.8	923456.79	923456.789
(923457)				
5	-9.23457E-2	-9.234568E-2	-9.2345679E-2	-9.23456789E-2
6	4.44444E-2	4.444444E-2	4.4444444E-2	4.44444444E-2
7	.001200	.0012000	.00120000	.001200000

\*\*\* TEST PASSED IF THE PROCESSOR OUTPUT IN THE FIRST SET  
OF NUMBERED ROWS MATCHES THE COLUMN CORRESPONDING  
TO THE IMPLEMENTATION-DEFINED SIGNIFICANCE-WIDTH  
(ALLOWING FOR OPTIONAL TRAILING ZEROS IN THE  
SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.

END PROGRAM 13

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 14: PRINTING AND ASSIGNING NUMERIC VALUES"
15 PRINT "           NEAR TO THE MAXIMUM AND MINIMUM MAGNITUDE."
20 PRINT "           ANSI STANDARD 5.4, 9.4, 12.4"
30 PRINT
40 PRINT "SECTION 14.1: PRINTING NR3 CONSTANTS OF SIX SIGNIFICANT"
50 PRINT "           DIGITS WHICH ARE NEAR THE MAGNITUDE OF:"
60 PRINT "           1E+38 OR 1E-38."
90 PRINT
100 PRINT "           BEGIN TEST."
110 PRINT
120 PRINT "SOURCE", "OUTPUT", "ACTUAL"
130 PRINT "CONSTANT", "SHOULD BE", "OUTPUT"
140 PRINT

```

```
190 PRINT "-9.99999E34 ","-9.99999E+34 ",-9.99999E34
200 PRINT "+9.99999E+35 "," 9.99999E+35 ",+9.99999E+35
210 PRINT "-9.99999E+36 ","-9.99999E+36 ",-9.99999E+36
220 PRINT " 9.99999E37 "," 9.99999E+37 ",9.99999E37
230 PRINT "-1.00001E-36","-1.00001E-36",-1.00001E-36
240 PRINT " 1.00001E-37"," 1.00001E-37",1.00001E-37
250 PRINT "+1.00001E-38"," 1.00001E-38",+1.00001E-38
260 PRINT
270 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
280 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN"
290 PRINT " THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) ***"
300 PRINT
480 PRINT " END TEST."
490 PRINT
500 PRINT "SECTION 14.2: ASSIGNING NR3 CONSTANTS OF SIX SIGNIFICANT"
510 PRINT " DIGITS WHICH ARE NEAR THE MAGNITUDE OF:"
520 PRINT " 1E+38 OR 1E-38."
530 PRINT
560 PRINT " BEGIN TEST."
570 PRINT
580 PRINT "ASSIGNED","OUTPUT","ACTUAL"
590 PRINT "CONSTANT","SHOULD BE","OUTPUT"
600 PRINT
610 LET A=-1.00001E-36
620 LET B=1.00001E-37
630 LET C=+1.00001E-38
650 LET E=-9.99999E34
660 LET F=+9.99999E+35
670 LET G=-9.99999E+36
680 LET H=9.99999E37
690 PRINT "-1.00001E-36","-1.00001E-36",A
700 PRINT " 1.00001E-37"," 1.00001E-37",B
710 PRINT "+1.00001E-38"," 1.00001E-38",C
730 PRINT "-9.99999E34 ","-9.99999E+34",E
740 PRINT "+9.99999E+35"," 9.99999E+35",F
750 PRINT "-9.99999E+36","-9.99999E+36",G
760 PRINT " 9.99999E37"," 9.99999E+37",H
770 PRINT
775 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
780 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN"
790 PRINT " THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) ***"
800 PRINT
810 PRINT " END TEST."
1130 PRINT
1140 PRINT "SECTION 14.3: NR3 NUMERIC CONSTANTS AT EXTREME"
1145 PRINT " MAGNITUDES AS PRINT ITEMS."
1150 PRINT
1160 PRINT " BEGIN TEST."
1170 PRINT
1180 PRINT "SOURCE","OUTPUT","ACTUAL"
1190 PRINT "CONSTANT","SHOULD BE","OUTPUT"
1200 PRINT
```

```

1210 PRINT "+1E38"," 1.E+38",+1E38
1220 PRINT " 1E-38"," 1.E-38",1E-38
1230 PRINT "-1E+38","-1.E+38",-1E+38
1240 PRINT "-1E-38","-1.E-38",-1E-38
1250 PRINT
1260 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
1270 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN"
1280 PRINT " THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) ***"
1290 PRINT
1300 PRINT " END TEST."
1310 PRINT
1320 PRINT "SECTION 14.4: NR3 NUMERIC VARIABLES AT EXTREME"
1325 PRINT " MAGNITUDES AS PRINT ITEMS."
1330 PRINT
1340 PRINT " BEGIN TEST."
1350 PRINT
1360 PRINT "ASSIGNED","OUTPUT","ACTUAL"
1370 PRINT "CONSTANT","SHOULD BE","OUTPUT"
1380 PRINT
1390 LET A=+1E+38
1400 LET B=1E-38
1410 LET C=-1E38
1420 LET D=-1E-38
1430 PRINT "+1E+38"," 1.E+38",A
1440 PRINT " 1E-38"," 1.E-38",B
1450 PRINT "-1E38","-1.E+38",C
1460 PRINT "-1E-38","-1.E-38",D
1470 PRINT
1480 PRINT "*** TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT"
1490 PRINT " SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN"
1500 PRINT " THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) ***"
1510 PRINT
1520 PRINT " END TEST."
1530 PRINT
1540 PRINT "END PROGRAM 14"
1550 END

```

---

PROGRAM FILE 14: PRINTING AND ASSIGNING NUMERIC VALUES  
NEAR TO THE MAXIMUM AND MINIMUM MAGNITUDE.  
ANSI STANDARD 5.4, 9.4, 12.4

SECTION 14.1: PRINTING NR3 CONSTANTS OF SIX SIGNIFICANT  
DIGITS WHICH ARE NEAR THE MAGNITUDE OF:  
1E+38 OR 1E-38.

BEGIN TEST.

SOURCE	OUTPUT	ACTUAL
CONSTANT	SHOULD BE	OUTPUT

-9.99999E34	-9.99999E+34	-9.99999E+34
+9.99999E+35	9.99999E+35	9.99999E+35
-9.99999E+36	-9.99999E+36	-9.99999E+36
9.99999E37	9.99999E+37	9.99999E+37
-1.00001E-36	-1.00001E-36	-1.00001E-36
1.00001E-37	1.00001E-37	1.00001E-37
+1.00001E-38	1.00001E-38	1.00001E-38

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN  
THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.

SECTION 14.2: ASSIGNING NR3 CONSTANTS OF SIX SIGNIFICANT  
DIGITS WHICH ARE NEAR THE MAGNITUDE OF:  
1E+38 OR 1E-38.

BEGIN TEST.

ASSIGNED CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
-1.00001E-36	-1.00001E-36	-1.00001E-36
1.00001E-37	1.00001E-37	1.00001E-37
+1.00001E-38	1.00001E-38	1.00001E-38
-9.99999E34	-9.99999E+34	-9.99999E+34
+9.99999E+35	9.99999E+35	9.99999E+35
-9.99999E+36	-9.99999E+36	-9.99999E+36
9.99999E37	9.99999E+37	9.99999E+37

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN  
THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.

SECTION 14.3: NR3 NUMERIC CONSTANTS AT EXTREME  
MAGNITUDES AS PRINT ITEMS.

BEGIN TEST.

SOURCE CONSTANT	OUTPUT SHOULD BE	ACTUAL OUTPUT
+1E38	1.E+38	1.00000E+38
1E-38	1.E-38	1.00000E-38
-1E+38	-1.E+38	-1.00000E+38
-1E-38	-1.E-38	-1.00000E-38

\*\*\* TEST PASSED IF ACTUAL OUTPUT MATCHES OUTPUT AS IT  
SHOULD BE (ALLOWING FOR OPTIONAL TRAILING ZEROS IN  
THE SIGNIFICAND AND LEADING ZEROS IN THE EXRAD) \*\*\*

END TEST.



```
250 PRINT "IF 1 FOLLOWS THIS LINE, TRANSFER USING 'GOTO' PERFORMED"
260 GOTO 430
270 PRINT "      ERROR: TRANSFER FROM LINE 260 TO 430 NOT PERFORMED"
280 LET M=3
290 PRINT TAB(67);M
300 PRINT "IF 4 FOLLOWS THIS LINE, TRANSFER TO REM-STATEMENT PERFORMED"
310 GO TO 600
320 PRINT "      ERROR: TRANSFER FROM LINE 310 TO 600 NOT PERFORMED"
330 LET M=7
340 PRINT TAB(67);M
350 PRINT "IF 8 FOLLOWS THESE TWO LINES, FORWARD TRANSFER PERFORMED,"
355 PRINT "      WITH LEADING ZERO IN LINE NUMBER OF THE GOTO"
360 GO TO 0480
370 PRINT "      ERROR: TRANSFER FROM LINE 360 TO 480 NOT PERFORMED"
380 LET M=2
390 PRINT TAB(67);M
400 PRINT "IF 3 FOLLOWS THIS LINE, TRANSFER USING 'GO      TO' PERFORMED"
410 GO      TO 280
420 PRINT "      ERROR: TRANSFER FROM LINE 410 TO 280 NOT PERFORMED"
430 LET M=1
440 PRINT TAB(67);M
450 PRINT "IF 2 FOLLOWS THIS LINE, TRANSFER USING 'GO TO' PERFORMED"
460 GO TO 380
470 PRINT "      ERROR: TRANSFER FROM LINE 460 TO 380 NOT PERFORMED"
480 LET M=8
490 PRINT TAB(67);M
495 PRINT
500 PRINT "THE NEXT OUTPUT MUST BE THE '*** TEST PASSED...''"
510 PRINT "MESSAGE FOR TEST TO PASS."
515 PRINT
520 GO TO 710
530 PRINT "      ERROR: TRANSFER FROM LINE 520 TO 710 NOT PERFORMED"
540 LET M=6
550 PRINT TAB(67);M
560 PRINT "IF 7 FOLLOWS THIS LINE, BACKWARD TRANSFER PERFORMED"
570 GO TO 330
580 PRINT "      ERROR: TRANSFER FROM LINE 570 TO 330 NOT PERFORMED"
590 REM
600 REM      TEST GOTO TRANSFERRING CONTROL TO REM STATEMENT.
610 LET M=4
620 PRINT TAB(67);M
630 PRINT "IF 5 FOLLOWS THIS LINE, FORWARD TRANSFER PERFORMED"
640 GO TO 660
650 PRINT "      ERROR: TRANSFER FROM LINE 640 TO 660 NOT PERFORMED"
660 LET M=5
670 PRINT TAB(67);M
680 PRINT "IF 6 FOLLOWS THIS LINE, BACKWARD TRANSFER PERFORMED"
690 GO TO 540
700 PRINT "      ERROR: TRANSFER FROM LINE 690 TO 540 NOT PERFORMED"
710 PRINT "*** TEST PASSED IF THE OUTPUT ABOVE IS ORDERED 1,2...,8   "
720 PRINT "      AND NO ERROR MESSAGES HAVE APPEARED ***"
730 PRINT
740 PRINT "      END TEST."
```

```

760 PRINT
770 PRINT "SECTION 15.3: GOTO-STATEMENT TRANSFERS TO ANOTHER GOTO."
780 PRINT
790 PRINT "                                BEGIN TEST."
800 PRINT
810 GO TO 880
820 PRINT "      ERROR: TRANSFER FROM LINE 810 TO 880 NOT PERFORMED"
830 PRINT "*** TEST PASSED IF THERE ARE NO ERROR MESSAGES ***"
840 GO TO 900
850 PRINT "      ERROR: TRANSFER FROM LINE 840 TO 900 NOT PERFORMED"
860 GO TO 830
870 PRINT "      ERROR: TRANSFER FROM LINE 860 TO 830 NOT PERFORMED"
880 GO TO 860
890 PRINT "      ERROR: TRANSFER FROM LINE 880 TO 860 NOT PERFORMED"
900 PRINT
910 PRINT "                                END TEST."
920 PRINT
930 PRINT "END PROGRAM 15"
940 END

```

---

#### PROGRAM FILE 15: THE REM AND GOTO STATEMENTS.

ANSI STANDARD 18.2, 18.4, 10.2, 10.4

#### SECTION 15.1: THE REM-STATEMENT - FOR PROGRAM COMMENTS.

BEGIN TEST.

\*\*\* REM TEST PASSED IF THESE ARE THE ONLY TWO LINES  
PRINTED BETWEEN 'BEGIN TEST.' AND 'END TEST.' \*\*\*  
END TEST.

#### SECTION 15.2: TEST OPTIONAL SPELLING OF GOTO AND TRANSFER TO REM-STATEMENT.

BEGIN TEST.

IF 1 FOLLOWS THIS LINE, TRANSFER USING 'GOTO' PERFORMED

1

IF 2 FOLLOWS THIS LINE, TRANSFER USING 'GO TO' PERFORMED

2

IF 3 FOLLOWS THIS LINE, TRANSFER USING 'GO TO' PERFORMED

3

IF 4 FOLLOWS THIS LINE, TRANSFER TO REM-STATEMENT PERFORMED

4

IF 5 FOLLOWS THIS LINE, FORWARD TRANSFER PERFORMED

5

IF 6 FOLLOWS THIS LINE, BACKWARD TRANSFER PERFORMED

6

IF 7 FOLLOWS THIS LINE, BACKWARD TRANSFER PERFORMED

7

IF 8 FOLLOWS THESE TWO LINES, FORWARD TRANSFER PERFORMED,  
WITH LEADING ZERO IN LINE NUMBER OF THE GOTO

8

THE NEXT OUTPUT MUST BE THE '\*\*\* TEST PASSED...' MESSAGE FOR TEST TO PASS.

\*\*\* TEST PASSED IF THE OUTPUT ABOVE IS ORDERED 1,2...,8 AND NO ERROR MESSAGES HAVE APPEARED \*\*\*

END TEST.

SECTION 15.3: GOTO-STATEMENT TRANSFERS TO ANOTHER GOTO.

BEGIN TEST.

\*\*\* TEST PASSED IF THERE ARE NO ERROR MESSAGES \*\*\*

END TEST.

END PROGRAM 15

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 16: ERROR - TRANSFER TO A NON-EXISTING LINE"
15 PRINT "      NUMBER USING THE GOTO-STATEMENT."
20 PRINT "      ANSI STANDARD 10.4"
30 PRINT
40 PRINT "SECTION 16.1: ERROR - TRANSFER TO A NON-EXISTING LINE"
50 PRINT "      NUMBER USING THE GOTO-STATEMENT."
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "
200 PRINT
210 LET A$="LINE 280."
240 GOTO 275
250 PRINT "IF THIS MESSAGE APPEARS 'GOTO 275' WAS NOT EXECUTED."
260 GOTO 290
270 LET A$="LINE 270."
280 PRINT "'GOTO 275' JUMPED TO ";A$
290 PRINT
300 PRINT "
```

BEGIN TEST."

END TEST."

```
310 PRINT
320 PRINT "END PROGRAM 16"
330 END
```

---

```
? UNDEFINED LINE NUMBER 275 IN LINE 240
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 17: ELEMENTARY USE OF GOSUB AND RETURN."
20 PRINT "      ANSI STANDARD 10.2, 10.4"
30 PRINT
40 PRINT "SECTION 17.1: ELEMENTARY USE OF GOSUB AND RETURN."
50 PRINT
52 PRINT "THIS PROGRAM TESTS THAT THE SUBROUTINE MECHANISM EXISTS"
54 PRINT "AND THAT A SUBROUTINE CAN BE INVOKED FROM SEVERAL"
56 PRINT "PLACES IN THE MAIN LINE OF CONTROL."
58 PRINT
60 PRINT "                                BEGIN TEST."
70 PRINT
80 PRINT "IF THE NEXT MESSAGE '*** GOSUB TEST PASSED ***' IS"
90 PRINT "SPELLED CORRECTLY, THE TEST PASSED."
100 PRINT
110 GOSUB 1000
120 PRINT "GO";
130 GOSUB 1200
140 PRINT "UB TE";
150 GOSUB 1200
160 PRINT "T PAS";
170 GOSUB 1200
180 PRINT "ED ***"
190 PRINT
200 PRINT "                                END TEST."
210 PRINT
220 PRINT "END PROGRAM 17"
230 STOP
1000 PRINT "*** ";
1010 RETURN
1200 PRINT "S";
1210 RETURN
1220 END
```

---

```
PROGRAM FILE 17: ELEMENTARY USE OF GOSUB AND RETURN.
ANSI STANDARD 10.2, 10.4
```

```
SECTION 17.1: ELEMENTARY USE OF GOSUB AND RETURN.
```

THIS PROGRAM TESTS THAT THE SUBROUTINE MECHANISM EXISTS AND THAT A SUBROUTINE CAN BE INVOKED FROM SEVERAL PLACES IN THE MAIN LINE OF CONTROL.

BEGIN TEST.

IF THE NEXT MESSAGE '\*\*\* GOSUB TEST PASSED \*\*\*' IS SPELLED CORRECTLY, THE TEST PASSED.

\*\*\* GOSUB TEST PASSED \*\*\*

END TEST.

END PROGRAM 17

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 18: THE IF-THEN STATEMENT WITH STRING OPERANDS."
20 PRINT "      ANSI STANDARD 10.2, 10.4"
30 PRINT
40 PRINT "SECTION 18.1: THE IF-THEN STATEMENT WITH STRING OPERANDS."
50 PRINT
60 PRINT "THIS SECTION TESTS THE COMPARISON OF STRING VARIABLES AND"
70 PRINT "CONSTANTS WITH A VARIETY OF VALUES. BOTH RELATIONSHIPS"
80 PRINT "(= AND <>) ARE TESTED FOR EACH PAIR OF VALUES. THE 'V' OR"
90 PRINT "'C' FOLLOWING THE COMPARAND VALUE INDICATES WHETHER IT IS"
100 PRINT "A VARIABLE OR A CONSTANT."
110 PRINT
120 PRINT "'T' OR 'F' IS PRINTED TO SHOW THE OUTCOME OF THE"
130 PRINT "COMPARISON. FOR INCORRECT COMPARES, AN '*' IS PRINTED"
140 PRINT "NEXT TO THE ERRONEOUS RESULT. WHEN DISPLAYING COMPARAND"
150 PRINT "VALUES, THE UNDERLINE CHARACTER, '_', IS USED TO REPRESENT"
160 PRINT "THE SPACE CHARACTER (THUS A DISPLAYED 'ONE BLANK' ")
170 PRINT "INDICATES AN INTERNAL VALUE OF 'ONE BLANK')."
171 PRINT "ALSO, THE NULL STRING (LENGTH OF ZERO) IS DENOTED BY THE"
172 PRINT "LABEL '(NULL STRING)'."
180 PRINT
190 PRINT "CHARACTER STRINGS SHOULD TEST EQUAL IF AND ONLY IF THE"
200 PRINT "STRINGS ARE EQUAL IN LENGTH AND CONTAIN IDENTICAL SEQUENCES"
210 PRINT "OF CHARACTERS."
220 PRINT
230 PRINT "                  BEGIN TEST."
240 PRINT
250 PRINT "LEFT";TAB(22);": RIGHT"
260 PRINT "COMPARAND";TAB(19);"V/C: COMPARAND";TAB(42);"V/C";
270 PRINT TAB(47);"=";TAB(51);"<>";TAB(55);"RESULT"
275 PRINT
```

```
280 LET R$="PASSED"
300 LET T$=" ABC"
310 LET V$=__ABC"
320 GOSUB 2500
330 LET T$="ABC"
340 LET V$="ABC"
350 GOSUB 2000
360 LET T$="ABC "
370 LET V$="ABC_"
380 GOSUB 2500
390 LET T$=" ABC "
400 LET V$=__ABC_
410 GOSUB 2500
420 LET T$=""
430 LET V$="(NULL STRING)"
440 GOSUB 2500
450 LET T$="ABCD"
460 LET V$="ABCD"
470 GOSUB 2500
500 LET T$=" OX9"
510 LET V$=__OX9"
520 GOSUB 3500
530 LET T$="OX9"
540 LET V$="OX9"
550 GOSUB 3500
560 LET T$="OX9 "
570 LET V$="OX9_"
580 GOSUB 3500
590 LET T$=" OX9 "
600 LET V$=__OX9_
610 GOSUB 3000
620 LET T$=" "
630 LET V$=__"
640 LET U$="__"
650 LET W$="____"
660 GOSUB 4500
670 LET T$=" "
680 LET V$=" "
690 LET U$="\""
700 LET W$="(NULL STRING)"
710 GOSUB 4500
720 LET T$="18 CHARACTERS LONG"
730 LET V$="18_CHARACTERS_LONG"
740 LET U$="18_CHARACTERS_LONG"
750 LET W$="18_CHARACTERS_LONG"
760 GOSUB 4000
770 LET U$="18 CHARACTERS LONK"
780 LET W$="18_CHARACTERS_LONK"
790 GOSUB 4500
800 LET T$="!#$%&'()0=+*<>?"
810 LET V$="!#$%&'()0=+*<>?"
820 LET U$="!#$%&'()0=+*<>?"
830 LET W$="!#$%&'()0=+*<>?"
840 GOSUB 4000
850 LET T$=""
```

```
860 LET V$="(NULL STRING)"
870 LET U$=""
880 LET W$="(NULL STRING)"
890 GOSUB 4000
900 LET T$="UVWXYZ "
910 LET V$="UVWXYZ_"
920 LET U$="UVWXYZ_"
930 LET W$="UVWXYZ__"
940 GOSUB 4500
1900 PRINT
1910 PRINT "*** TEST ";R$;" ***"
1915 PRINT
1920 PRINT " END TEST."
1925 PRINT
1930 PRINT "END PROGRAM 18"
1940 STOP
2000 REM SUBROUTINE TO TEST CONSTANT = VARIABLE
2005 LET B$="T*"
2010 LET A$="T"
2020 LET L$=" OK "
2030 IF "ABC" <> T$ THEN 2060
2040 LET B$="F"
2050 GOTO 2080
2060 LET R$="FAILED"
2070 LET L$="FAILED"
2080 IF "ABC"=T$ THEN 2120
2090 LET R$="FAILED"
2100 LET L$="FAILED"
2110 LET A$="F*"
2120 GOSUB 2900
2125 GOSUB 5000
2130 RETURN
2500 REM SUBROUTINE TO TEST CONSTANT <> VARIABLE
2505 LET A$="T*"
2510 LET B$="T"
2520 LET L$=" OK "
2530 IF "ABC"=T$ THEN 2560
2540 LET A$="F"
2550 GOTO 2580
2560 LET R$="FAILED"
2570 LET L$="FAILED"
2580 IF "ABC" <> T$ THEN 2620
2590 LET R$="FAILED"
2600 LET L$="FAILED"
2610 LET B$="F*"
2620 GOSUB 2900
2625 GOSUB 5000
2630 RETURN
2900 REM SUBROUTINE TO PRINT COMPARANDS FOR CONSTANT:VARIABLE
2910 PRINT "ABC";TAB(20);C : ";V$;TAB(43);V";
2920 RETURN
3000 REM SUBROUTINE TO TEST VARIABLE = CONSTANT
3005 LET B$="T*"
3010 LET A$="T"
3020 LET L$=" OK "
```

```
3030 IF T$<>" 0X9 " THEN 3060
3040 LET B$="F"
3050 GOTO 3080
3060 LET R$="FAILED"
3070 LET L$="FAILED"
3080 IF T$=" 0X9 " THEN 3120
3090 LET R$="FAILED"
3100 LET L$="FAILED"
3110 LET A$="F*"
3120 GOSUB 3900
3125 GOSUB 5000
3130 RETURN
3500 REM SUBROUTINE TO TEST VARIABLE <> CONSTANT
3505 LET A$="T*"
3510 LET B$="T"
3520 LET L$=" OK "
3530 IF T$=" 0X9 " THEN 3560
3540 LET A$="F"
3550 GOTO 3580
3560 LET R$="FAILED"
3570 LET L$="FAILED"
3580 IF T$<>" 0X9 " THEN 3620
3590 LET R$="FAILED"
3600 LET L$="FAILED"
3610 LET B$="F*"
3620 GOSUB 3900
3625 GOSUB 5000
3630 RETURN
3900 REM SUBROUTINE TO PRINT COMPARANDS FOR VARIABLE:CONSTANT
3910 PRINT V$;TAB(20);"V : _0X9_";TAB(43);"C";
3920 RETURN
4000 REM SUBROUTINE TO TEST VARIABLE = VARIABLE
4005 LET B$="T*"
4010 LET A$="T"
4020 LET L$=" OK "
4030 IF T$<>U$ THEN 4060
4040 LET B$="F"
4050 GOTO 4080
4060 LET R$="FAILED"
4070 LET L$="FAILED"
4080 IF T$=U$ THEN 4120
4090 LET R$="FAILED"
4100 LET L$="FAILED"
4110 LET A$="F*"
4120 GOSUB 4900
4125 GOSUB 5000
4130 RETURN
4500 REM SUBROUTINE TO TEST VARIABLE <> VARIABLE
4505 LET A$="T*"
4510 LET B$="T"
4520 LET L$=" OK "
4530 IF T$=U$ THEN 4560
4540 LET A$="F"
4550 GOTO 4580
4560 LET R$="FAILED"
```

```

4570 LET L$="FAILED"
4580 IF T$<>U$ THEN 4620
4590 LET R$="FAILED"
4600 LET L$="FAILED"
4610 LET B$="F*"
4620 GOSUB 4900
4625 GOSUB 5000
4630 RETURN
4900 REM SUBROUTINE TO PRINT COMPARANDS FOR VARIABLE:VARIABLE
4910 PRINT V$;TAB(20);"V : ";W$;TAB(43);"V";  

4920 RETURN
5000 REM SUBROUTINE TO PRINT RESULTS OF COMPARE
5010 PRINT TAB(47);A$;TAB(51);B$;TAB(55);L$  

5020 RETURN
9000 END

```

---

PROGRAM FILE 18: THE IF-THEN STATEMENT WITH STRING OPERANDS.  
ANSI STANDARD 10.2, 10.4

SECTION 18.1: THE IF-THEN STATEMENT WITH STRING OPERANDS.

THIS SECTION TESTS THE COMPARISON OF STRING VARIABLES AND CONSTANTS WITH A VARIETY OF VALUES. BOTH RELATIONSHIPS (= AND <>) ARE TESTED FOR EACH PAIR OF VALUES. THE 'V' OR 'C' FOLLOWING THE COMPARAND VALUE INDICATES WHETHER IT IS A VARIABLE OR A CONSTANT.

'T' OR 'F' IS PRINTED TO SHOW THE OUTCOME OF THE COMPARISON. FOR INCORRECT COMPARES, AN '\*' IS PRINTED NEXT TO THE ERRONEOUS RESULT. WHEN DISPLAYING COMPARAND VALUES, THE UNDERLINE CHARACTER, '\_', IS USED TO REPRESENT THE SPACE CHARACTER (THUS A DISPLAYED 'ONE\_BLANK' INDICATES AN INTERNAL VALUE OF 'ONE BLANK'). ALSO, THE NULL STRING (LENGTH OF ZERO) IS DENOTED BY THE LABEL '(NULL STRING)'.

CHARACTER STRINGS SHOULD TEST EQUAL IF AND ONLY IF THE STRINGS ARE EQUAL IN LENGTH AND CONTAIN IDENTICAL SEQUENCES OF CHARACTERS.

BEGIN TEST.

LEFT COMPARAND	: RIGHT V/C: COMPARAND	V/C	=	<>	RESULT
ABC	C : ABC	V	F	T	OK
ABC	C : <u>ABC</u>	V	T	F	OK
ABC	C : ABC <u>_</u>	V	F	T	OK
ABC	C : <u>ABC</u> <u>_</u>	V	F	T	OK
ABC	C : (NULL STRING)	V	F	T	OK
ABC	C : ABCD	V	F	T	OK
OX9	V : OX9	C	F	T	OK
OX9	V : <u>OX9</u>	C	F	T	OK

0X9	V : _0X9	C	F	T	OK
_0X9	V : _0X9	C	T	F	OK
—	V : _____	V	F	T	OK
18_CHARACTERS_LONG	V : (NULL STRING)	V	F	T	OK
18_CHARACTERS_LONG	V : 18_CHARACTERS_LONG	V	T	F	OK
!#\$%&'()0=+*<?>	V : !#\$%&'()0=+*<?>	V	T	F	OK
(NULL STRING)	V : (NULL STRING)	V	T	F	OK
UVWXYZ	V : UVWXYZ	V	F	T	OK

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 18

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 19: THE IF-THEN STATEMENT WITH NUMERIC OPERANDS"
20 PRINT "      ANSI STANDARD 10.2, 10.4"
30 PRINT
40 PRINT "SECTION 19.1: THE IF-THEN STATEMENT WITH NUMERIC OPERANDS."
50 PRINT
60 PRINT "THIS SECTION TESTS THE COMPARISON OF NUMERIC VARIABLES AND"
70 PRINT "CONSTANTS WITH A VARIETY OF NUMERIC VALUES. ALL RELATIONSHIPS"
80 PRINT "(=, <, >, <>, <=, >=) ARE TESTED FOR EACH PAIR OF VALUES."
85 PRINT "THE 'V' OR 'C' FOLLOWING THE COMPARAND VALUE INDICATES"
90 PRINT "WHETHER IT IS A VARIABLE OR A CONSTANT."
95 PRINT
100 PRINT "'T' OR 'F' IS PRINTED TO SHOW THE OUTCOME OF THE"
105 PRINT "COMPARISON. FOR INCORRECT COMPARES, AN '*' IS PRINTED"
110 PRINT "NEXT TO THE ERRONEOUS RESULT."
115 PRINT
120 PRINT          BEGIN TEST."
125 PRINT
130 PRINT "LEFT           RIGHT           ";
135 PRINT "=  <  >  <>  <=  >=  RESULT"
140 PRINT "COMPARAND           COMPARAND   "
145 REM TEST VARIABLE:CONSTANT
150 PRINT
155 LET R$="PASSED"
160 LET A0=3.14159
170 GOSUB 1000
180 GOSUB 4000
190 LET A0=5.4321E-22
200 GOSUB 2000
210 GOSUB 4000
220 LET A0=3.14159E-26
230 GOSUB 3000
240 GOSUB 4000
250 LET A0=-00.0E3
260 GOSUB 3000
270 GOSUB 4000

```

```
280 LET A0=-3.14159E-26
290 GOSUB 3000
300 GOSUB 4000
310 LET A0=-3.14159
320 GOSUB 3000
330 GOSUB 4000
335 REM NOW DO CONSTANT:VARIABLE
340 LET B0=-98765500000
350 GOSUB 1000
360 GOSUB 4200
370 LET B0=-98765400000
380 GOSUB 2000
390 GOSUB 4200
400 LET B0=-3
410 GOSUB 3000
420 GOSUB 4200
430 LET B0=1.23456E-30
440 GOSUB 3000
450 GOSUB 4200
460 LET B0=456.789
470 GOSUB 3000
480 GOSUB 4200
490 REM TEST CONSTANT:CONSTANT
500 GOSUB 1000
510 IF -1.23456E-15 = -.011E-13 THEN 530
520 LET A$=G$
530 IF -1.23456E-15 < -.011E-13 THEN 550
540 LET B$=H$
550 IF -1.23456E-15 > -.011E-13 THEN 570
560 LET C$=I$
570 IF -1.23456E-15 <> -.011E-13 THEN 590
580 LET D$=J$
590 IF -1.23456E-15 <= -.011E-13 THEN 610
600 LET E$=K$
610 IF -1.23456E-15 >= -.011E-13 THEN 630
620 LET F$=L$
630 REM COMPARISONS FINISHED
640 PRINT "-1.23456E-15 C";TAB(18);": -.011E-13 C";
650 GOSUB 5000
660 REM TEST VARIABLE:VARIABLE
670 LET A0=-0
680 LET B0=+0
690 GOSUB 2000
700 GOSUB 4400
710 LET A0=-3E36
720 LET B0=-3.001E36
730 GOSUB 3000
740 GOSUB 4400
750 LET A0=0E22
760 LET B0=1E-38
770 GOSUB 1000
780 GOSUB 4400
900 PRINT
910 PRINT "*** TEST ";R$;" ***"
920 PRINT
```

```
930 PRINT "END TEST."
940 PRINT
950 PRINT "END PROGRAM 19"
960 STOP
1000 REM SUBROUTINE TO INITIALIZE FOR CASE OF <
1010 LET A$="T*"
1020 LET B$="T"
1030 LET C$="T*"
1040 LET D$="T"
1050 LET E$="T"
1060 LET F$="T*"
1070 LET G$="F"
1080 LET H$="F*"
1090 LET I$="F"
1100 LET J$="F*"
1110 LET K$="F*"
1120 LET L$="F"
1140 RETURN
2000 REM SUBROUTINE TO INITIALIZE FOR CASE OF =
2010 LET A$="T"
2020 LET B$="T*"
2030 LET C$="T*"
2040 LET D$="T*"
2050 LET E$="T"
2060 LET F$="T"
2070 LET G$="F*"
2080 LET H$="F"
2090 LET I$="F"
2100 LET J$="F"
2110 LET K$="F*"
2120 LET L$="F*"
2140 RETURN
3000 REM SUBROUTINE TO INITIALIZE FOR CASE OF >
3010 LET A$="T*"
3020 LET B$="T*"
3030 LET C$="T"
3040 LET D$="T"
3050 LET E$="T*"
3060 LET F$="T"
3070 LET G$="F"
3080 LET H$="F"
3090 LET I$="F*"
3100 LET J$="F*"
3110 LET K$="F"
3120 LET L$="F*"
3140 RETURN
4000 REM SUBROUTINE TO TEST CONSTANT:VARIABLE
4010 IF +5.4321E-22 = A0 THEN 4030
4020 LET A$=G$
4030 IF +5.4321E-22 < A0 THEN 4050
4040 LET B$=H$
4050 IF +5.4321E-22 > A0 THEN 4070
4060 LET C$=I$
4070 IF +5.4321E-22 <> A0 THEN 4090
4080 LET D$=J$
```

```
4090 IF +5.4321E-22 <= AO THEN 4110
4100 LET E$=K$
4110 IF +5.4321E-22 >= AO THEN 4130
4120 LET F$=L$
4130 REM COMPARISONS FINISHED
4140 PRINT "+5.4321E-22 C";TAB(18);": ";AO;"V";
4150 GOSUB 5000
4160 RETURN
4200 REM SUBROUTINE TO TEST VARIABLE:CONSTANT
4210 IF BO = -98765400000 THEN 4230
4220 LET A$=G$
4230 IF BO < -98765400000 THEN 4250
4240 LET B$=H$
4250 IF BO > -98765400000 THEN 4270
4260 LET C$=I$
4270 IF BO <> -98765400000 THEN 4290
4280 LET D$=J$
4290 IF BO <= -98765400000 THEN 4310
4300 LET E$=K$
4310 IF BO >= -98765400000 THEN 4330
4320 LET F$=L$
4330 REM COMPARISONS FINISHED
4340 PRINT BO;"V";TAB(18);": -98765400000 C";
4350 GOSUB 5000
4360 RETURN
4400 REM SUBROUTINE TO TEST VARIABLE:VARIABLE
4410 IF AO = BO THEN 4430
4420 LET A$=G$
4430 IF AO < BO THEN 4450
4440 LET B$=H$
4450 IF AO > BO THEN 4470
4460 LET C$=I$
4470 IF AO <> BO THEN 4490
4480 LET D$=J$
4490 IF AO <= BO THEN 4510
4500 LET E$=K$
4510 IF AO >= BO THEN 4530
4520 LET F$=L$
4530 REM COMPARISONS FINISHED
4540 PRINT AO;"V";TAB(18);": ";BO;"V";
4550 GOSUB 5000
4560 RETURN
5000 REM SUBROUTINE TO CHECK OUT AND PRINT RESULTS OF COMPARE
5010 LET N$=" OK "
5020 IF A$="T*" THEN 5200
5025 IF A$="F*" THEN 5200
5030 IF B$="T*" THEN 5200
5035 IF B$="F*" THEN 5200
5040 IF C$="T**" THEN 5200
5045 IF C$="F**" THEN 5200
5050 IF D$="T**" THEN 5200
5055 IF D$="F**" THEN 5200
5060 IF E$="T**" THEN 5200
5065 IF E$="F**" THEN 5200
5070 IF F$="T**" THEN 5200
```

```

5075 IF F$="F*" THEN 5200
5080 GOTO 5210
5200 LET N$="FAILED"
5205 LET R$="FAILED"
5210 REM RESULTS CHECKED
5510 PRINT TAB(38);A$;TAB(42);B$;TAB(46);C$;TAB(50);D$;
5520 PRINT TAB(54);E$;TAB(58);F$;TAB(62);N$
5530 RETURN
9999 END

```

---

PROGRAM FILE 19: THE IF-THEN STATEMENT WITH NUMERIC OPERANDS  
ANSI STANDARD 10.2, 10.4

SECTION 19.1: THE IF-THEN STATEMENT WITH NUMERIC OPERANDS.

THIS SECTION TESTS THE COMPARISON OF NUMERIC VARIABLES AND CONSTANTS WITH A VARIETY OF NUMERIC VALUES. ALL RELATIONSHIPS (=, <, >, <>, <=, >=) ARE TESTED FOR EACH PAIR OF VALUES. THE 'V' OR 'C' FOLLOWING THE COMPARAND VALUE INDICATES WHETHER IT IS A VARIABLE OR A CONSTANT.

'T' OR 'F' IS PRINTED TO SHOW THE OUTCOME OF THE COMPARISON. FOR INCORRECT COMPARES, AN '\*' IS PRINTED NEXT TO THE ERRONEOUS RESULT.

BEGIN TEST.

LEFT COMPARAND	RIGHT COMPARAND	=	<	>	<>	<=	>=	RESULT
+5.4321E-22 C	: 3.14159 V	F	T	F	T	T	F	OK
+5.4321E-22 C	: 5.43210E-22 V	T	F	F	F	T	T	OK
+5.4321E-22 C	: 3.14159E-26 V	F	F	T	T	F	T	OK
+5.4321E-22 C	: 0 V	F	F	T	T	F	T	OK
+5.4321E-22 C	: -3.14159E-26 V	F	F	T	T	F	T	OK
+5.4321E-22 C	: -3.14159 V	F	F	T	T	F	T	OK
-9.87655E+10 V	: -98765400000 C	F	T	F	T	T	F	OK
-9.87654E+10 V	: -98765400000 C	T	F	F	F	T	T	OK
-3 V	: -98765400000 C	F	F	T	T	F	T	OK
1.23456E-30 V	: -98765400000 C	F	F	T	T	F	T	OK
456.789 V	: -98765400000 C	F	F	T	T	F	T	OK
-1.23456E-15 C	: -.011E-13 C	F	T	F	T	T	F	OK
0 V	: 0 V	T	F	F	F	T	T	OK
-3.00000E+36 V	: -3.00100E+36 V	F	F	T	T	F	T	OK
0 V	: 1.00000E-38 V	F	T	F	T	T	F	OK

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 19

```
*****
*****  

10 PRINT "PROGRAM FILE 20: ERROR - IF-THEN STATEMENT WITH A STRING AND"  

20 PRINT "      NUMERIC OPERAND."  

30 PRINT "      ANSI STANDARD 10.2"  

40 PRINT  

50 PRINT "SECTION 20.1: ERROR - IF-THEN STATEMENT WITH A STRING AND"  

60 PRINT "      NUMERIC OPERAND."  

70 PRINT  

80 PRINT "THIS PROGRAM ATTEMPTS TO COMPARE A STRING VARIABLE WITH A"  

90 PRINT "NUMERIC VARIABLE IN AN IF-THEN STATEMENT."  

100 PRINT  

110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  

120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  

130 PRINT  

140 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  

150 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  

160 PRINT "          BY THE PROCESSOR, OR"  

170 PRINT  

180 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  

190 PRINT  

200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  

210 PRINT "FOR DETAILED CRITERIA."  

220 PRINT  

230 PRINT "                  BEGIN TEST."  

240 PRINT  

250 PRINT "SETTING A$='123' AND X=123"  

260 LET A$="123"  

270 LET X=123  

280 PRINT "ABOUT TO EXECUTE: IF A$ = X..."  

290 PRINT  

300 IF A$=X THEN 330  

310 PRINT "IF TESTED FALSE."  

320 GOTO 340  

330 PRINT "IF TESTED TRUE."  

340 PRINT  

350 PRINT "                  END TEST."  

360 PRINT  

370 PRINT "END PROGRAM 20"  

380 END
```

---

? MIXED STRINGS AND NUMBERS IN LINE 300

```
*****
*****  

*****  

*****
```

```
10 PRINT "PROGRAM FILE 21: ERROR - TRANSFER TO NON-EXISTING LINE"  

15 PRINT "      NUMBER USING THE IF-THEN-STATEMENT."  

20 PRINT "      ANSI STANDARD 10.4"  

30 PRINT
```



```
140 LET A=10
150 LET A$="ABC"
155 LET AO=A
160 LET AO=100
170 LET A9=109
180 LET B1=123E22
190 LET B0=-123E-22
200 LET B$="18 CHARACTERS LONG"
210 LET B=33
220 IF A<>10 THEN 320
230 IF A$<>"ABC" THEN 320
240 IF A9<>109 THEN 320
250 IF AO<>100 THEN 320
260 IF B<>33 THEN 320
270 IF B$<>"18 CHARACTERS LONG" THEN 320
280 IF B1<>123E22 THEN 320
290 IF B0<>-123E-22 THEN 320
300 PRINT "*** TEST PASSED ***"
310 GOTO 330
320 PRINT "*** TEST FAILED ***"
330 PRINT
340 PRINT "END TEST."
350 PRINT
360 PRINT "END PROGRAM 22"
370 END
```

---

PROGRAM FILE 22: NUMERIC AND STRING VARIABLE NAMES  
WITH THE SAME INITIAL LETTER.  
ANSI STANDARD 6.2, 6.4

SECTION 22.1: NUMERIC AND STRING VARIABLE NAMES WITH THE  
SAME INITIAL LETTER.

THE STANDARD ESTABLISHES A NAME SPACE OF AT LEAST 286 NUMERIC  
DATA NAMES (A - Z, AO - Z0, A1 - Z1, ..., A9 - Z9) AND 26  
DISTINCT STRING DATA NAMES (A\$ - Z\$).

THIS PROGRAM TESTS THAT DISTINCT VALUES ARE PRESERVED  
IN VARIABLES WITH SIMILAR, BUT NOT IDENTICAL, NAMES.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 22

```
*****
*****  
10 PRINT "PROGRAM FILE 23: INITIALIZATION OF STRING AND NUMERIC"  
20 PRINT "      VARIABLES."  
30 PRINT "      ANSI STANDARD 6.6"  
40 PRINT  
50 PRINT "SECTION 23.1: INITIALIZATION OF STRING AND NUMERIC"  
60 PRINT "      VARIABLES."  
70 PRINT  
80 PRINT "THIS PROGRAM TESTS HOW THE IMPLEMENTATION TREATS"  
90 PRINT "UNINITIALIZED VARIABLES. THE MINIMAL BASIC STANDARD"  
100 PRINT "RECOMMENDS THAT THIS CONDITION BE TREATED AS AN "  
110 PRINT "EXCEPTION. IN ANY CASE, THE DOCUMENTATION MUST"  
120 PRINT "DESCRIBE THE ACTION OF THE PROCESSOR IN ORDER"  
130 PRINT "FOR THE TEST TO PASS."  
140 PRINT  
150 PRINT "                  BEGIN TEST."  
160 PRINT  
170 PRINT "NEITHER A$ OR Y HAS BEEN ASSIGNED A VALUE."  
180 PRINT  
190 PRINT "THE IMPLEMENTATION-DEFINED INITIAL VALUE (SURROUNDED"  
200 PRINT "      BY APOSTROPES) FOR A$=''';A$;'''  
210 PRINT "THE IMPLEMENTATION-DEFINED INITIAL VALUE FOR Y ='';Y  
220 PRINT  
230 PRINT "IF THE DOCUMENTATION FOR THIS IMPLEMENTATION CORRECTLY"  
240 PRINT "      DESCRIBES THE RESULTS ABOVE, THEN"  
250 PRINT "*** TEST PASSED ***"  
260 PRINT  
270 PRINT "                  END TEST."  
280 PRINT  
290 PRINT "END PROGRAM 23"  
300 END
```

---

PROGRAM FILE 23: INITIALIZATION OF STRING AND NUMERIC  
 VARIABLES.  
 ANSI STANDARD 6.6

SECTION 23.1: INITIALIZATION OF STRING AND NUMERIC  
 VARIABLES.

THIS PROGRAM TESTS HOW THE IMPLEMENTATION TREATS  
UNINITIALIZED VARIABLES. THE MINIMAL BASIC STANDARD  
RECOMMENDS THAT THIS CONDITION BE TREATED AS AN  
EXCEPTION. IN ANY CASE, THE DOCUMENTATION MUST  
DESCRIBE THE ACTION OF THE PROCESSOR IN ORDER  
FOR THE TEST TO PASS.

BEGIN TEST.

NEITHER A\$ OR Y HAS BEEN ASSIGNED A VALUE.

THE IMPLEMENTATION-DEFINED INITIAL VALUE (SURROUNDED

BY APOSTROPES) FOR A\$=''

THE IMPLEMENTATION-DEFINED INITIAL VALUE FOR Y = 0

IF THE DOCUMENTATION FOR THIS IMPLEMENTATION CORRECTLY

DESCRIBES THE RESULTS ABOVE, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 23

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 24: PLUS AND MINUS"
20 PRINT "      ANSI STANDARD 7.2, 7.4"
30 PRINT
40 PRINT "THIS PROGRAM TESTS THE GROSS SEMANTIC CHARACTERISTICS OF"
50 PRINT "THE PLUS AND MINUS OPERATORS FOR THE CONSTRUCTION OF"
60 PRINT "NUMERIC EXPRESSIONS."
70 PRINT
80 PRINT "EXPRESSIONS USING ONLY ONE OPERATOR AND CONSTANTS AND"
90 PRINT "SIMPLE VARIABLES ARE EVALUATED, AND THE RESULTS TESTED"
100 PRINT "AGAINST THE CORRECT VALUE. CONSTANTS ARE EXPRESSED IN"
110 PRINT "NR1, NR2, AND NR3 FORM, BUT ARE GENERALLY RESTRICTED TO"
120 PRINT "INTEGER VALUES TO AVOID ACCURACY PROBLEMS (ACCURACY IS"
130 PRINT "ADDRESSED IN LATER TESTS)."
140 PRINT
142 REM S IS A SWITCH TO INDICATE UNARY (S=0) OR BINARY (S=1)
143 REM OPERATIONS
145 LET S=0
150 LET M$="24.1: UNARY MINUS"
160 GOSUB 7000
170 LET N$="CONSTANTS"
180 LET O$="1"
190 LET A=-3
200 IF A>0 THEN 250
210 IF A<>-3 THEN 250
220 IF A<-3.1 THEN 250
230 IF A>-2.9 THEN 250
240 IF A=-3 THEN 260
250 GOSUB 9000
260 LET A=-0
270 LET O$="2"
280 IF A=0 THEN 300
290 GOSUB 9000
300 LET A=-432.0000
310 LET O$="3"
320 IF -432=A THEN 340
330 GOSUB 9000
340 LET A=-2E2
350 LET O$="4"
```

```
360 IF A=-200 THEN 380
370 GOSUB 9000
380 LET A=-200E-1
390 LET O$="5"
400 IF A=-20 THEN 420
410 GOSUB 9000
420 LET N$="VARIABLES"
430 LET B=34
440 LET O$="6"
450 LET A=-B
460 IF A=-34 THEN 480
470 GOSUB 9000
480 LET C=-B
490 LET A=-C
500 LET O$="7"
510 IF A=34 THEN 530
520 GOSUB 9000
530 LET B=-99
540 LET B=-B
550 LET A=-B
560 LET O$="8"
570 IF A=-99 THEN 590
580 GOSUB 9000
590 GOSUB 8000
600 LET M$="24.2: UNARY PLUS"
610 GOSUB 7000
620 LET N$="CONSTANTS"
630 LET O$="1"
640 LET A=+5
650 IF A=5 THEN 670
660 GOSUB 9000
670 LET A=+0
680 LET O$="2"
690 IF A=0 THEN 710
700 GOSUB 9000
710 LET A=+123.00
720 LET O$="3"
730 IF A=123 THEN 750
740 GOSUB 9000
750 LET A=+87E2
760 LET O$="4"
770 IF A=8700 THEN 790
780 GOSUB 9000
790 LET A=+.0054E+5
800 LET O$="5"
810 IF A=540 THEN 830
820 GOSUB 9000
830 LET N$="VARIABLES"
840 LET B=47
850 LET O$="6"
860 LET A=+B
870 IF A=47 THEN 890
880 GOSUB 9000
890 LET B=-88
900 LET O$="7"
```

910 LET C=+B  
920 LET B=+C  
930 LET A=+B  
940 IF A=-88 THEN 960  
950 GOSUB 9000  
960 GOSUB 8000  
965 LET S=1  
970 LET M\$="24.3: ADDITION"  
980 LET N\$="CONSTANTS"  
990 GOSUB 7000  
1000 LET A=3+6  
1010 LET O\$="1"  
1020 LET Y=9  
1030 GOSUB 8500  
1040 LET A=34.00+32  
1050 LET O\$="2"  
1060 LET Y=66  
1070 GOSUB 8500  
1080 LET A=23E2+2  
1090 LET O\$="3"  
1100 LET Y=2302  
1110 GOSUB 8500  
1120 LET A=33.000+880E-1  
1130 LET O\$="4"  
1140 LET Y=121  
1150 GOSUB 8500  
1160 LET N\$="VAR. & CONSTS."  
1170 LET A=-34  
1180 LET B=-98.0  
1190 LET A=A+B  
1200 LET O\$="5"  
1210 LET Y=-132  
1220 GOSUB 8500  
1230 LET B=94  
1240 LET C=-73  
1250 LET A=B+C  
1260 LET O\$="6"  
1270 LET Y=21  
1280 GOSUB 8500  
1290 LET C=434  
1300 LET B=-654  
1310 LET A=B+C  
1320 LET O\$="7"  
1330 LET Y=-220  
1340 GOSUB 8500  
1350 LET B=-98  
1360 LET A=2E2+B  
1370 LET O\$="8"  
1380 LET Y=102  
1390 GOSUB 8500  
1400 LET B=13.0E1  
1410 LET A=B+22  
1420 LET O\$="9"  
1430 LET Y=152  
1440 GOSUB 8500

1450 LET B=22  
1460 LET C=-22  
1470 LET O\$="10"  
1480 LET A=B+C  
1490 LET Y=0  
1500 GOSUB 8500  
1510 LET A=0+C  
1520 LET O\$="11"  
1530 LET Y=-22  
1540 GOSUB 8500  
1550 LET A=C+C  
1560 LET O\$="12"  
1570 LET Y=-44  
1580 GOSUB 8500  
1590 GOSUB 8000  
1600 LET M\$="24.4: SUBTRACTION"  
1610 LET N\$="CONSTANTS"  
1620 GOSUB 7000  
1630 LET O\$="1"  
1640 LET A=48-29  
1650 LET Y=19  
1660 GOSUB 8500  
1670 LET A=47-646  
1680 LET O\$="2"  
1690 LET Y=-599  
1700 GOSUB 8500  
1710 LET A=.00532E5-777.  
1720 LET O\$="3"  
1730 LET Y=-245  
1740 GOSUB 8500  
1750 LET N\$="VARS. & CONSTS."  
1760 LET B=-56  
1770 LET C=-93  
1780 LET A=B-C  
1790 LET O\$="4"  
1800 LET Y=37  
1810 GOSUB 8500  
1820 LET A=C-B  
1830 LET O\$="5"  
1840 LET Y=-37  
1850 GOSUB 8500  
1860 LET B=-345.000  
1870 LET A=2-B  
1880 LET O\$="6"  
1890 LET Y=347  
1900 GOSUB 8500  
1910 LET A=B-.111E3  
1920 LET O\$="7"  
1930 LET Y=-456  
1940 GOSUB 8500  
1950 LET A=B-987  
1960 LET O\$="8"  
1970 LET Y=-1332  
1980 GOSUB 8500  
1990 LET A=888.0-B

```
2000 LET O$="9"
2010 LET Y=1233
2020 GOSUB 8500
2030 LET B=22
2040 LET A=22-B
2050 LET O$="10"
2060 LET Y=0
2070 GOSUB 8500
2080 LET A=0-B
2090 LET O$="11"
2100 LET Y=-22
2110 GOSUB 8500
2120 LET B=-87
2130 LET A=B-B
2140 LET O$="12"
2150 LET Y=0
2160 GOSUB 8500
2170 GOSUB 8000
6000 PRINT
6010 PRINT "END PROGRAM 24"
6020 STOP
7000 REM SUBROUTINE TO BEGIN SECTION
7010 PRINT
7020 PRINT "SECTION ";M$
7030 PRINT
7040 PRINT "           BEGIN TEST"
7050 PRINT
7060 LET R$="PASSED"
7070 IF S=0 THEN 7300
7080 PRINT "CASE #", "SHOULD BE", "ACTUAL", "OUTCOME"
7090 PRINT
7300 RETURN
8000 REM SUBROUTINE TO END SECTION
8010 PRINT
8040 PRINT "*** TEST ";R$;" ***"
8060 PRINT
8070 PRINT "           END TEST"
8080 PRINT
8090 RETURN
8500 REM REPORT RESULTS OF TEST CASE
8505 PRINT O$,Y,A,
8510 IF Y=A THEN 8550
8520 PRINT "TEST FAILS"
8530 LET R$="FAILED"
8540 GOTO 8560
8550 PRINT "TEST PASSES"
8560 RETURN
9000 REM HANDLE ERROR REPORT
9010 PRINT "ERROR IN TEST OF ";M$";", ";N$";", CASE #";O$
9020 LET R$="FAILED"
9030 RETURN
9040 END
```

PROGRAM FILE 24: PLUS AND MINUS  
ANSI STANDARD 7.2, 7.4

THIS PROGRAM TESTS THE GROSS SEMANTIC CHARACTERISTICS OF THE PLUS AND MINUS OPERATORS FOR THE CONSTRUCTION OF NUMERIC EXPRESSIONS.

EXPRESSIONS USING ONLY ONE OPERATOR AND CONSTANTS AND SIMPLE VARIABLES ARE EVALUATED, AND THE RESULTS TESTED AGAINST THE CORRECT VALUE. CONSTANTS ARE EXPRESSED IN NR1, NR2, AND NR3 FORM, BUT ARE GENERALLY RESTRICTED TO INTEGER VALUES TO AVOID ACCURACY PROBLEMS (ACCURACY IS ADDRESSED IN LATER TESTS).

SECTION 24.1: UNARY MINUS

BEGIN TEST

\*\*\* TEST PASSED \*\*\*

END TEST

SECTION 24.2: UNARY PLUS

BEGIN TEST

\*\*\* TEST PASSED \*\*\*

END TEST

SECTION 24.3: ADDITION

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	9	9	TEST PASSES
2	66	66	TEST PASSES
3	2302	2302	TEST PASSES
4	121	121	TEST PASSES
5	-132	-132	TEST PASSES
6	21	21	TEST PASSES
7	-220	-220	TEST PASSES
8	102	102	TEST PASSES
9	152	152	TEST PASSES
10	0	0	TEST PASSES
11	-22	-22	TEST PASSES
12	-44	-44	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

#### SECTION 24.4: SUBTRACTION

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	19	19	TEST PASSES
2	-599	-599	TEST PASSES
3	-245	-245	TEST PASSES
4	37	37	TEST PASSES
5	-37	-37	TEST PASSES
6	347	347	TEST PASSES
7	-456	-456	TEST PASSES
8	-1332	-1332	TEST PASSES
9	1233	1233	TEST PASSES
10	0	0	TEST PASSES
11	-22	-22	TEST PASSES
12	0	0	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 24

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 25: MULTIPLY, DIVIDE, AND INVOLUTE"
20 PRINT "      ANSI STANDARD 7.2, 7.4"
30 PRINT
40 PRINT "THIS PROGRAM TESTS THE GROSS SEMANTIC CHARACTERISTICS OF"
50 PRINT "THE MULTIPLY, DIVIDE, AND INVOLUTION OPERATORS FOR THE"
60 PRINT "CONSTRUCTION OF NUMERIC EXPRESSIONS."
70 PRINT
80 PRINT "EXPRESSIONS USING ONLY ONE OPERATOR AND CONSTANTS AND"
90 PRINT "SIMPLE VARIABLES ARE EVALUATED, AND THE RESULTS TESTED"
100 PRINT "AGAINST THE CORRECT VALUE. CONSTANTS ARE EXPRESSED IN"
110 PRINT "NR1, NR2, AND NR3 FORM. THE VALUES OF THE OPERANDS AND"
120 PRINT "RESULTS ARE RESTRICTED TO INTEGERS AND SIMPLE FRACTIONS"
130 PRINT "TO AVOID ACCURACY PROBLEMS (ACCURACY IS ADDRESSED IN"
135 PRINT "LATER TESTS)."
140 PRINT

```

```
150 LET M$="25.1: MULTIPLY"
160 GOSUB 7000
170 LET N$="CONSTANTS"
180 LET O$="1"
190 LET A=3*7
200 LET Y=21
220 GOSUB 8500
230 LET A=5.0*14
240 LET O$="2"
250 LET Y=70
270 GOSUB 8500
280 LET A=6.000*2.3E+1
290 LET O$="3"
300 LET Y=138
320 GOSUB 8500
330 LET N$="VAR. & CONSTS."
340 LET A=-37
350 LET A=A*3
360 LET O$="4"
370 LET Y=-111
390 GOSUB 8500
400 LET B=-6.0
410 LET C=5
420 LET A=C*B
430 LET O$="5"
440 LET Y=-30
460 GOSUB 8500
470 LET C=-12E-0
480 LET B=-110E-1
490 LET O$="6"
500 LET A=B*C
510 LET Y=132
530 GOSUB 8500
540 LET C=-3
550 LET B=-0
560 LET A=C*B
570 LET O$="7"
580 LET Y=0
600 GOSUB 8500
610 LET A=C*C
620 LET O$="8"
630 LET Y=9
650 GOSUB 8500
660 LET B=4
670 LET A=B*4.0
680 LET O$="9"
690 LET Y=16
710 GOSUB 8500
720 GOSUB 8000
730 LET M$="25.2: DIVIDE"
740 GOSUB 7000
750 LET N$="CONSTANTS"
760 LET O$="1"
770 LET A=12/3
780 LET Y=4
```

790 GOSUB 8500  
800 LET A=60.00/3E1  
810 LET O\$="2"  
820 LET Y=2  
830 GOSUB 8500  
840 LET A=0.0/7  
850 LET O\$="3"  
860 LET Y=0  
870 GOSUB 8500  
880 LET N\$="VARS. & CONSTS."  
882 LET B=63  
884 LET C=9  
886 LET A=B/C  
890 LET O\$="4"  
900 LET Y=7  
910 GOSUB 8500  
915 LET B=-49  
920 LET C=1.00  
925 LET A=B/C  
930 LET O\$="5"  
940 LET Y=-49  
950 GOSUB 8500  
960 LET B=3  
970 LET C=-3  
975 LET A=B/C  
980 LET O\$="6"  
990 LET Y=-1  
1010 GOSUB 8500  
1020 LET B=-111  
1030 LET O\$="7"  
1040 LET C=-3  
1060 LET A=B/C  
1070 LET Y=37  
1080 GOSUB 8500  
1090 LET B=-8  
1100 LET A=0/B  
1110 LET O\$="8"  
1120 LET Y=0  
1130 GOSUB 8500  
1140 LET C=7  
1150 LET A=C/2  
1160 LET O\$="9"  
1170 LET Y=3.5  
1180 GOSUB 8500  
1190 LET B=-26  
1200 LET C=-5  
1210 LET A=B/C  
1220 LET O\$="10"  
1230 LET Y=5.2  
1240 GOSUB 8500  
2000 GOSUB 8000  
2010 LET M\$="25.3: INVOLUTION"  
2020 LET N\$="CONSTANTS"  
2030 GOSUB 7000  
2040 LET A=3^2

```
2050 LET O$="1"
2060 LET Y=9
2070 GOSUB 8500
2080 LET A=4.0^3.E-0
2090 LET O$="2"
2100 LET Y=64
2110 GOSUB 8500
2120 LET A=50^0.0
2130 LET O$="3"
2140 LET Y=1
2150 GOSUB 8500
2155 LET N$="VAR. & CONSTS."
2160 LET B=3
2162 LET C=4
2165 LET A=B^C
2170 LET O$="4"
2180 LET Y=81
2190 GOSUB 8500
2210 LET B=-2
2220 LET C=3
2230 LET A=B^C
2240 LET O$="5"
2250 LET Y=-8
2260 GOSUB 8500
2270 LET B=-3
2290 LET A=B^2.0
2300 LET O$="6"
2310 LET Y=9
2320 GOSUB 8500
2330 LET C=4
2350 LET A=0^C
2360 LET O$="7"
2370 LET Y=0
2380 GOSUB 8500
2390 LET B=4
2400 LET A=B^0
2410 LET O$="8"
2420 LET Y=1
2430 GOSUB 8500
2440 LET B=0
2445 LET C=0.0
2450 LET A=B^C
2460 LET O$="9"
2470 LET Y=1
2480 GOSUB 8500
2490 LET B=40E-1
2500 LET C=1.0
2510 LET A=B^C
2520 LET O$="10"
2530 LET Y=4
2540 GOSUB 8500
2550 LET B=-1
2560 LET A=4^B
2570 LET O$="11"
2580 LET Y=.25
```

```
2590 GOSUB 8500
2600 LET B=-2
2610 LET A=B^B
2620 LET O$="12"
2630 LET Y=.25
2640 GOSUB 8500
2650 LET C=-3
2660 LET A=B^C
2670 LET O$="13"
2680 LET Y=-.125
2690 GOSUB 8500
2700 LET B=-3
2710 LET A=1^B
2720 LET O$="14"
2730 LET Y=1
2740 GOSUB 8500
2750 LET B=-2
2760 LET A=B^0
2770 LET O$="15"
2780 LET Y=1
2790 GOSUB 8500
2800 LET B=-3
2810 LET A=.5^B
2820 LET O$="16"
2830 LET Y=8
2840 GOSUB 8500
2850 LET C=-.5
2860 LET A=.16^C
2870 LET O$="17"
2880 LET Y=2.5
2890 GOSUB 8500
2900 LET A=.36^.5
2910 LET O$="18"
2920 LET Y=.6
2930 GOSUB 8500
2940 LET B=-.5
2950 LET C=-3
2960 LET A=B^C
2970 LET O$="19"
2980 LET Y=-8
2990 GOSUB 8500
3000 LET A=B^3
3010 LET O$="20"
3020 LET Y=-.125
3030 GOSUB 8500
3040 GOSUB 8000
6000 PRINT
6010 PRINT "END PROGRAM 25"
6020 STOP
7000 REM SUBROUTINE TO BEGIN SECTION
7010 PRINT
7020 PRINT "SECTION ";M$
7030 PRINT
7040 PRINT "               BEGIN TEST"
7050 PRINT
```

```
7060 LET E1=0
7080 PRINT "CASE #","SHOULD BE","ACTUAL","OUTCOME"
7090 PRINT
7300 RETURN
8000 REM SUBROUTINE TO END SECTION
8010 PRINT
8020 IF E1=0 THEN 8050
8025 PRINT "*** TEST FAILED IN ";E1;" CASE(S) ***"
8030 GOTO 8060
8050 PRINT "*** TEST PASSED ***"
8060 PRINT
8070 PRINT "           END TEST"
8080 PRINT
8090 RETURN
8500 REM HANDLE TEST CASE REPORT
8510 PRINT O$,Y,A,
8520 LET P$="FAILS"
8530 LET M=0.1
8540 IF Y>1 THEN 8570
8550 IF Y<-1 THEN 8570
8560 LET M=0.01
8570 LET T=A-Y
8580 IF T>M THEN 8630
8590 LET T=Y-A
8600 IF T>M THEN 8630
8610 LET P$="PASSES"
8620 GOTO 8640
8630 LET E1=E1+1
8640 PRINT "TEST ";P$
8650 RETURN
8660 END
```

---

PROGRAM FILE 25: MULTIPLY, DIVIDE, AND INVOLUTE  
ANSI STANDARD 7.2, 7.4

THIS PROGRAM TESTS THE GROSS SEMANTIC CHARACTERISTICS OF  
THE MULTIPLY, DIVIDE, AND INVOLUTION OPERATORS FOR THE  
CONSTRUCTION OF NUMERIC EXPRESSIONS.

EXPRESSIONS USING ONLY ONE OPERATOR AND CONSTANTS AND  
SIMPLE VARIABLES ARE EVALUATED, AND THE RESULTS TESTED  
AGAINST THE CORRECT VALUE. CONSTANTS ARE EXPRESSED IN  
NR1, NR2, AND NR3 FORM. THE VALUES OF THE OPERANDS AND  
RESULTS ARE RESTRICTED TO INTEGERS AND SIMPLE FRACTIONS  
TO AVOID ACCURACY PROBLEMS (ACCURACY IS ADDRESSED IN  
LATER TESTS).

#### SECTION 25.1: MULTIPLY

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	21	21	TEST PASSES
2	70	70	TEST PASSES
3	138	138	TEST PASSES
4	-111	-111	TEST PASSES
5	-30	-30	TEST PASSES
6	132	132	TEST PASSES
7	0	0	TEST PASSES
8	9	9	TEST PASSES
9	16	16	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

### SECTION 25.2: DIVIDE

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	4	4	TEST PASSES
2	2	2	TEST PASSES
3	0	0	TEST PASSES
4	7	7	TEST PASSES
5	-49	-49	TEST PASSES
6	-1	-1	TEST PASSES
7	37	37	TEST PASSES
8	0	0	TEST PASSES
9	3.5	3.5	TEST PASSES
10	5.2	5.2	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

### SECTION 25.3: INVOLUTION

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	9	9	TEST PASSES
2	64	64	TEST PASSES
3	1	1	TEST PASSES
4	81	81	TEST PASSES
5	-8	-8	TEST PASSES
6	9	9	TEST PASSES
7	0	0	TEST PASSES
8	1	1	TEST PASSES
9	1	1	TEST PASSES
10	4	4	TEST PASSES

11	0.25	0.25	TEST PASSES
12	0.25	0.25	TEST PASSES
13	-0.125	-0.125	TEST PASSES
14	1	1	TEST PASSES
15	1	1	TEST PASSES
16	8	8	TEST PASSES
17	2.5	2.5	TEST PASSES
18	0.6	0.6	TEST PASSES
19	-8	-8	TEST PASSES
20	-0.125	-0.125	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 25

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 26: PRECEDENCE RULES FOR NUMERIC EXPRESSIONS."
20 PRINT "    ANSI STANDARD 7.2, 7.4"
30 PRINT
40 PRINT "THIS PROGRAM TESTS WHETHER THE PRECEDENCE RULES ARE"
50 PRINT "CORRECTLY IMPLEMENTED."
60 PRINT
70 PRINT "SECTION 26.1: PRECEDENCE ACCORDING TO OPERATOR AND"
80 PRINT "    POSITION (LEFT TO RIGHT)."
90 PRINT
100 PRINT "THIS SECTION TESTS THE PRECEDENCE RULES USING SIMPLE"
110 PRINT "INTEGER OPERANDS AND ALL ORDERED PAIRS OF OPERATORS"
120 PRINT "FOR WHICH THE ORDER OF EVALUATION MAKES A DIFFERENCE."
130 PRINT "NO PARENTHESES ARE USED."
140 PRINT
150 PRINT "                BEGIN TEST"
160 PRINT
163 PRINT "CASE #", "SHOULD BE", "ACTUAL", "OUTCOME"
166 PRINT
170 LET F=0
180 LET O$="1"
190 LET A=3+4 * 5
200 LET Y=23
210 GOSUB 9000
220 LET A=8+4 / 2
230 LET O$="2"
240 LET Y=10
250 GOSUB 9000
260 LET A=3.+3 ^ 3
270 LET O$="3"
280 LET Y=30
290 GOSUB 9000
300 LET A=3 - 40.00E-1+5
310 LET O$="4"

```

320 LET Y=4  
330 GOSUB 9000  
340 LET A=3.00 - 4-.5E+1  
350 LET O\$="5"  
360 LET Y=-6  
370 GOSUB 9000  
380 LET O\$="6"  
390 LET A=20.E-1-3 \* 4  
400 LET Y=-10  
410 GOSUB 9000  
420 LET A=8-4 / 2  
430 LET O\$="7"  
440 LET Y=6  
450 GOSUB 9000  
460 LET A=3-2 ^ 3  
470 LET O\$="8"  
480 LET Y=-5  
490 GOSUB 9000  
500 LET A=3 \* 4+5  
510 LET O\$="9"  
520 LET Y=17  
530 GOSUB 9000  
540 LET O\$="10"  
550 LET A=3.00E0 \* 4.00E+00-5.00E-00  
560 LET Y=7  
570 GOSUB 9000  
580 LET A=003.0\*2. ^ 3.  
590 LET O\$="11"  
600 LET Y=24  
610 GOSUB 9000  
620 LET A=6 / 2+1  
630 LET O\$="12"  
640 LET Y=4  
650 GOSUB 9000  
660 LET A=6 / 2-1  
670 LET O\$="13"  
680 LET Y=2  
690 GOSUB 9000  
700 LET A=2 / 1\*2  
710 LET O\$="14"  
720 LET Y=4  
730 GOSUB 9000  
740 LET A=8 / 2/2  
750 LET O\$="15"  
760 LET Y=2  
770 GOSUB 9000  
780 LET O\$="16"  
790 LET A=8/2 ^ 2  
800 LET Y=2  
810 GOSUB 9000  
820 LET A=2 ^ 2+2  
830 LET O\$="17"  
840 LET Y=6  
850 GOSUB 9000  
860 LET A=2 ^ 3-2

```
870 LET O$="18"
880 LET Y=6
890 GOSUB 9000
900 LET A=2 ^ 2*2
910 LET O$="19"
920 LET Y=8
930 GOSUB 9000
940 LET A=3 ^ 3/3
950 LET O$="20"
960 LET Y=9
970 GOSUB 9000
980 LET A=2 ^ 3^2
990 LET O$="21"
1000 LET Y=64
1010 GOSUB 9000
1020 LET A=- 2+2
1030 LET O$="22"
1040 LET Y=0
1050 GOSUB 9000
1060 LET A=- 2-2
1070 LET O$="23"
1080 LET Y=-4
1090 GOSUB 9000
1100 LET A=-2 ^ 2
1110 LET O$="24"
1120 LET Y=-4
1130 GOSUB 9000
1140 PRINT
1150 IF F=0 THEN 1180
1160 PRINT "*** TEST FAILED IN ";F;" CASE(S). ****"
1170 GOTO 1190
1180 PRINT "*** TEST PASSED ***"
1190 PRINT
1200 PRINT " END TEST"
1210 PRINT
2000 PRINT
2010 PRINT "SECTION 26.2: MIXED OPERATORS, PARENTHESES, AND VARIABLES"
2020 PRINT
2030 PRINT "THIS SECTION TESTS THE USE OF PARENTHESES TO ALTER THE"
2040 PRINT "ORDER OF EVALUATION, EXPRESSIONS WITH SEVERAL OPERATORS,"
2050 PRINT "AND THE USE OF VARIABLES AND CONSTANTS."
2060 PRINT
2070 PRINT " BEGIN TEST"
2080 PRINT
2083 PRINT "CASE #","SHOULD BE","ACTUAL","OUTCOME"
2086 PRINT
2090 LET F=0
2100 LET O$="1"
2110 LET A=4
2120 LET A=3-(A-5)
2130 LET Y=4
2140 GOSUB 9000
2150 LET O$="2"
2160 LET B=2
2170 LET A=+2-B+2/B*2^B
```

```
2180 LET Y=4
2190 GOSUB 9000
2195 LET O$="3"
2200 LET A=-B-B+B/002.*.2E+01^2
2210 LET Y=0
2220 GOSUB 9000
2225 LET O$="4"
2230 LET A=-3
2240 LET B=-4
2250 LET A=-(A-B)-B-(A-((+B-(B-A))-A))
2260 LET Y=6
2270 GOSUB 9000
2280 LET O$="5"
2290 LET B=4
2300 LET A=-(1-(-3))^(+(-B^1)/(-2))
2310 LET Y=-16
2320 GOSUB 9000
2330 LET O$="6"
2340 LET A=-2
2350 LET B=16
2360 LET C=-30
2370 LET A=(-B+(B^.02E+02-4*A*C)^.5)/(2*A)
2380 LET Y=3
2390 GOSUB 9000
2400 LET O$="7"
2410 LET A=60/(10-(-(9))-7)/2
2420 LET Y=-5
2430 GOSUB 9000
2900 PRINT
2910 IF F=0 THEN 2940
2920 PRINT "*** TEST FAILED IN ";F;" CASE(S). ***"
2930 GOTO 2950
2940 PRINT "*** TEST PASSED ***"
2950 PRINT
2960 PRINT " END TEST"
8970 PRINT
8980 PRINT "END PROGRAM 26"
8990 STOP
9000 REM HANDLE ERROR REPORT
9005 PRINT O$,Y,A,
9007 LET P$="FAILS"
9010 LET M=0.1
9020 IF Y>1 THEN 9050
9030 IF Y<-1 THEN 9050
9040 LET M=0.01
9050 LET T=A-Y
9060 IF T>M THEN 9100
9070 LET T=Y-A
9080 IF T>M THEN 9100
9090 LET P$="PASSES"
9095 GOTO 9110
9100 LET F=F+1
9110 PRINT "TEST ";P$
9130 RETURN
9140 END
```

---

PROGRAM FILE 26: PRECEDENCE RULES FOR NUMERIC EXPRESSIONS.  
ANSI STANDARD 7.2, 7.4

THIS PROGRAM TESTS WHETHER THE PRECEDENCE RULES ARE  
CORRECTLY IMPLEMENTED.

SECTION 26.1: PRECEDENCE ACCORDING TO OPERATOR AND  
POSITION (LEFT TO RIGHT).

THIS SECTION TESTS THE PRECEDENCE RULES USING SIMPLE  
INTEGER OPERANDS AND ALL ORDERED PAIRS OF OPERATORS  
FOR WHICH THE ORDER OF EVALUATION MAKES A DIFFERENCE.  
NO PARENTHESES ARE USED.

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	23	23	TEST PASSES
2	10	10	TEST PASSES
3	30	30	TEST PASSES
4	4	4	TEST PASSES
5	-6	-6	TEST PASSES
6	--10	-10	TEST PASSES
7	6	6	TEST PASSES
8	-5	-5	TEST PASSES
9	17	17	TEST PASSES
10	7	7	TEST PASSES
11	24	24	TEST PASSES
12	4	4	TEST PASSES
13	2	2	TEST PASSES
14	4	4	TEST PASSES
15	2	2	TEST PASSES
16	2	2	TEST PASSES
17	6	6	TEST PASSES
18	6	6	TEST PASSES
19	8	8	TEST PASSES
20	9	9	TEST PASSES
21	64	64	TEST PASSES
22	0	0	TEST PASSES
23	-4	-4	TEST PASSES
24	-4	-4	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

SECTION 26.2: MIXED OPERATORS, PARENTHESES, AND VARIABLES

THIS SECTION TESTS THE USE OF PARENTHESES TO ALTER THE ORDER OF EVALUATION, EXPRESSIONS WITH SEVERAL OPERATORS, AND THE USE OF VARIABLES AND CONSTANTS.

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	4	4	TEST PASSES
2	4	4	TEST PASSES
3	0	0	TEST PASSES
4	6	6	TEST PASSES
5	-16	-16	TEST PASSES
6	3	3	TEST PASSES
7	-5	-5	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 26

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

5 PRINT "PROGRAM FILE 27: ACCURACY OF CONSTANTS AND VARIABLES."
10 PRINT "      ANSI STANDARD 5.2, 5.4, 6.2, 6.4, 10.4"
15 PRINT
20 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF THE ON-GOTO STATEMENT"
25 PRINT "      AND FOR-BLOCKS (THE FOR AND NEXT STATEMENTS) WHICH"
30 PRINT "      HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
35 PRINT "      THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE "
40 PRINT "      VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
45 PRINT
50 PRINT "THIS PROGRAM TESTS THE LEVEL OF ACCURACY OF NUMERIC VALUES,"
55 PRINT "USING THE IF-THEN STATEMENT."
60 LET T1=34
70 LET T2=50
80 LET T3=T1+2
90 LET T4=T2+2
100 PRINT
110 PRINT "SECTION 27.1: ACCURACY IN COMPARISON OF VARIABLES."
120 PRINT
130 PRINT "THIS SECTION COMPARES TWO VARIABLES WHICH HAVE BEEN ASSIGNED"
140 PRINT "NUMERICALLY 'CLOSE' VALUES AND THEN DETERMINES WHETHER THE"
150 PRINT "IMPLEMENTATION CAN DISTINGUISH BETWEEN THEM OR IF THEY ARE"
160 PRINT "TREATED AS EQUAL."
170 PRINT

```

```
180 LET C$=" > ="  
190 GOSUB 8000  
200 LET A = 9.876543212E-7  
210 LET A$="9.876543212E-7 : "  
220 FOR I=6 TO 9  
230 PRINT A$;  
240 LET J=I-5  
250 ON J GOTO 600,700,800,900  
600 LET B= 9.876533E-7  
610 PRINT "9.876533E-7";  
620 GOTO 1000  
700 LET B= 9.8765422E-7  
710 PRINT "9.8765422E-7";  
720 GOTO 1000  
800 LET B= 9.87654311E-7  
810 PRINT "9.87654311E-7";  
820 GOTO 1000  
900 LET B= 9.876543201E-7  
910 PRINT "9.876543201E-7";  
920 GOTO 1000  
1000 LET U$="T"  
1010 LET E$="T"  
1020 IF A > B THEN 1040  
1030 LET U$="F"  
1040 IF A = B THEN 1060  
1050 LET E$="F"  
1060 REM COMPARISONS FINISHED  
1200 GOSUB 8160  
1210 NEXT I  
1220 GOSUB 8280  
3000 PRINT "SECTION 27.2: ACCURACY IN THE COMPARISON OF VARIABLES"  
3010 PRINT " WITH CONSTANTS."  
3020 PRINT  
3030 PRINT "THIS SECTION COMPARES A VARIABLE AND A CONSTANT WHICH HAVE"  
3040 PRINT "NUMERICALLY 'CLOSE' VALUES AND THEN DETERMINES WHETHER THE"  
3050 PRINT "IMPLEMENTATION CAN DISTINGUISH BETWEEN THEM OR IF THEY ARE"  
3060 PRINT "TREATED AS EQUAL."  
3070 PRINT  
3080 LET C$=" > ="  
3090 GOSUB 8000  
3100 LET A$="-9.999988888E36 : "  
3110 FOR I=6 TO 9  
3120 PRINT A$;  
3130 LET J=I-5  
3140 ON J GOTO 3200,3300,3400,3500  
3200 LET B= -9.999999E36  
3210 PRINT "-9.999999E36";  
3220 GOTO 3600  
3300 LET B= -9.9999899E36  
3310 PRINT "-9.9999899E36";  
3320 GOTO 3600  
3400 LET B= -9.99998899E36  
3410 PRINT "-9.99998899E36";  
3420 GOTO 3600  
3500 LET B= -9.999988899E36
```

```
3510 PRINT "-9.999988899E36";
3520 GOTO 3600
3600 LET U$="T"
3610 LET E$="T"
3620 IF -9.999988888E36 > B THEN 3640
3630 LET U$="F"
3640 IF -9.999988888E36 = B THEN 3660
3650 LET E$="F"
3660 REM COMPARISONS FINISHED
3670 GOSUB 8160
3680 NEXT I
3690 GOSUB 8280
5000 PRINT "SECTION 27.3: ACCURACY IN THE COMPARISON OF CONSTANTS."
5010 PRINT
5020 PRINT "THIS SECTION COMPARES TWO CONSTANTS WHICH HAVE"
5030 PRINT "NUMERICALLY 'CLOSE' VALUES AND THEN DETERMINES WHETHER THE"
5040 PRINT "IMPLEMENTATION CAN DISTINGUISH BETWEEN THEM OR IF THEY ARE"
5050 PRINT "TREATED AS EQUAL."
5060 PRINT
5070 LET C$=" < ="
5080 GOSUB 8000
5090 LET A$="9.40395E-37 : "
5100 FOR I=6 TO 9
5110 PRINT A$;
5120 LET U$="T"
5130 LET E$="T"
5140 LET J=I-5
5150 ON J GOTO 5200,5300,5400,5500
5200 IF 9.40395E-37 < 9.40396E-37 THEN 5220
5210 LET U$="F"
5220 IF 9.40395E-37 = 9.40396E-37 THEN 5240
5230 LET E$="F"
5240 PRINT "9.40396E-37";
5250 GOTO 5600
5300 IF 9.40395E-37 < 9.403951E-37 THEN 5320
5310 LET U$="F"
5320 IF 9.40395E-37 = 9.403951E-37 THEN 5340
5330 LET E$="F"
5340 PRINT "9.403951E-37";
5350 GOTO 5600
5400 IF 9.40395E-37 < 9.4039501E-37 THEN 5420
5410 LET U$="F"
5420 IF 9.40395E-37 = 9.4039501E-37 THEN 5440
5430 LET E$="F"
5440 PRINT "9.4039501E-37";
5450 GOTO 5600
5500 IF 9.40395E-37 < 9.40395001E-37 THEN 5520
5510 LET U$="F"
5520 IF 9.40395E-37 = 9.40395001E-37 THEN 5540
5530 LET E$="F"
5540 PRINT "9.40395001E-37";
5550 GOTO 5600
5600 REM COMPARISONS FINISHED
5610 GOSUB 8160
5620 NEXT I
```

```
5630 GOSUB 8280
5800 PRINT "SECTION 27.4: ACCURACY FOR CONSTANTS EXPRESSED WITH"
5810 PRINT "           DIFFERENT EXPONENTS."
5820 PRINT
5830 PRINT "CONSTANTS MUST BE ACCURATE EVEN WHEN NOT EXPRESSED IN"
5840 PRINT "NORMALIZED FORM (1 <= SIGNIFICAND < 10)."
5850 PRINT
5860 PRINT "               BEGIN TEST."
5870 PRINT
5880 IF .0000000999991E-18 >= 99999200000E-36 THEN 5950
5890 IF -.0000000999991E-18 <= -99999200000E-36 THEN 5950
5900 IF .0000000999992E-18 >= 99999300000E-36 THEN 5950
5910 IF -.0000000999992E-18 <= -99999300000E-36 THEN 5950
5920 IF .0000000999993E-18 >= 99999400000E-36 THEN 5950
5930 IF -.0000000999993E-18 <= -99999400000E-36 THEN 5950
5940 PRINT "*** TEST PASSED ***"
5945 GO TO 5955
5950 PRINT "*** TEST FAILED ***"
5955 PRINT
5960 PRINT "               END TEST."
5970 PRINT
6000 PRINT "SECTION 27.5: COMPUTED ACCURACY OF NUMERIC VALUES."
6010 PRINT
6020 PRINT "THIS SECTION USES A COMPUTATIONAL ALGORITHM TO ATTEMPT TO"
6030 PRINT "DISCOVER THIS IMPLEMENTATION'S INTERNAL ACCURACY. BECAUSE"
6040 PRINT "THE ALGORITHM RELIES SOMEWHAT ON ACCURACY OF EXPRESSION"
6050 PRINT "EVALUATION, THE RESULTS ARE NOT ABSOLUTELY TRUSTWORTHY AND"
6060 PRINT "SHOULD BE INTERPRETED IN LIGHT OF GENERAL KNOWLEDGE OF"
6070 PRINT "THE CHARACTERISTICS OF THE SYSTEM."
6080 PRINT
6090 PRINT "*** THIS TEST IS INFORMATIVE ONLY ***"
6100 PRINT
6110 PRINT "               BEGIN TEST."
6120 PRINT
6130 LET B=10
6140 LET B$="DECIMAL"
6150 GOSUB 6230
6160 LET B=2
6170 LET B$="BINARY"
6180 GOSUB 6230
6190 PRINT
6200 PRINT "               END TEST."
6210 PRINT
6220 GOTO 6430
6230 LET S=1
6240 LET E=S
6250 LET D=-1
6260 LET E=E/B
6270 LET T2 = S - E
6280 LET T1=T2
6290 REM THE NEXT FOUR STATEMENTS HAVE NO ALGORITHMIC PURPOSE -
6300 REM THEY ARE AN ATTEMPT TO FORCE THE VARIABLE T1 TO BE STORED
6310 REM RATHER THAN SAVED IN OVERLENGTH REGISTERS, AS IS DONE
6320 REM BY SOME MACHINES, WITH AN OPTIMIZING COMPILER.
6325 LET V=1
```

```
6330 LET U=E+D
6340 LET V=U*D*(-V)
6350 IF V < 0 THEN 6330
6360 LET D=D+1
6370 IF D>500 THEN 6410
6380 IF T1 < S THEN 6260
6390 PRINT "COMPUTED ACCURACY = ";D;B$;" DIGITS."
6400 RETURN
6410 PRINT "UNABLE TO COMPUTE ACCURACY OF ";B$;" DIGITS."
6420 RETURN
6430 REM END SECTION
6440 PRINT "END PROGRAM 27"
6450 STOP
8000 REM SUBROUTINE TO BEGIN SECTION
8010 PRINT "                                BEGIN TEST."
8020 PRINT
8030 PRINT "NUMERIC VALUES";TAB(T1);"DIGITS OF";TAB(T2);
8040 PRINT "RESULT OF"
8050 PRINT "BEING COMPARED";TAB(T1);"DECIMAL";TAB(T2);
8060 PRINT "COMPARISONS"
8070 PRINT TAB(T1);"ACCURACY";TAB(T2);C$
8080 PRINT
8090 REM NO. OF ANOMALOUS RESULTS (BOTH TRUE OR BOTH FALSE)
8100 LET S2=0
8110 REM SIGNIFICANCE SWITCH - 1 IMPLIES LOST SIGNIFICANCE
8120 LET S1=0
8130 REM NO. OF SIGNIFICANT DIGITS
8140 LET S0=5
8150 RETURN
8160 REM SUBROUTINE TO SUMMARIZE LINE
8170 LET W$="
8180 IF E$=U$ THEN 8230
8190 IF U$="F" THEN 8250
8200 IF S1=1 THEN 8260
8210 LET S0=I
8220 GOTO 8260
8230 LET W$="***"
8240 LET S2=S2+1
8250 LET S1=1
8260 PRINT TAB(T3);I;TAB(T4);U$;"    ";E$;"    ";W$
8270 RETURN
8280 REM SUBROUTINE TO END SECTION
8290 LET T$="LEAST"
8300 IF S0=9 THEN 8320
8310 LET T$="MOST"
8320 PRINT
8330 PRINT "*** TEST INDICATES DECIMAL ACCURACY OF AT ";T$;" ";S0;
8340 PRINT "DIGITS. ***"
8350 IF S0 < 6 THEN 8400
8360 PRINT "IF DOCUMENTATION FOR THE IMPLEMENTATION-DEFINED ACCURACY"
8370 PRINT "    FOR NUMERIC VALUES IS COMPATIBLE WITH THIS RESULT, THEN"
8380 PRINT "*** TEST PASSED ***"
8390 GOTO 8410
8400 PRINT "*** TEST FAILED: MINIMUM ACCURACY IS SIX DIGITS. ***"
8410 PRINT
```

```

8420 IF S2=0 THEN 8460
8430 PRINT "*** NOTE ANOMALOUS COMPARISON RESULTS IN THE ";S2;" ROW(S)"
8440 PRINT "    ABOVE FOLLOWED BY ASTERISKS. ***"
8450 PRINT
8460 PRINT "                END TEST."
8470 PRINT
8480 RETURN
8490 END

```

---

## PROGRAM FILE 27: ACCURACY OF CONSTANTS AND VARIABLES.

ANSI STANDARD 5.2, 5.4, 6.2, 6.4, 10.4

\*\*\* NOTE: THIS PROGRAM MAKES USE OF THE ON-GOTO STATEMENT AND FOR-BLOCKS (THE FOR AND NEXT STATEMENTS) WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

THIS PROGRAM TESTS THE LEVEL OF ACCURACY OF NUMERIC VALUES, USING THE IF-THEN STATEMENT.

## SECTION 27.1: ACCURACY IN COMPARISON OF VARIABLES.

THIS SECTION COMPARES TWO VARIABLES WHICH HAVE BEEN ASSIGNED NUMERICALLY 'CLOSE' VALUES AND THEN DETERMINES WHETHER THE IMPLEMENTATION CAN DISTINGUISH BETWEEN THEM OR IF THEY ARE TREATED AS EQUAL.

BEGIN TEST.

NUMERIC VALUES BEING COMPARED	DIGITS OF DECIMAL ACCURACY	RESULT OF COMPARISONS	
		>	=
9.876543212E-7 : 9.876533E-7	6	T	F
9.876543212E-7 : 9.8765422E-7	7	T	F
9.876543212E-7 : 9.87654311E-7	8	T	F
9.876543212E-7 : 9.876543201E-7	9	F	T

\*\*\* TEST INDICATES DECIMAL ACCURACY OF AT MOST 8 DIGITS. \*\*\*  
 IF DOCUMENTATION FOR THE IMPLEMENTATION-DEFINED ACCURACY  
 FOR NUMERIC VALUES IS COMPATIBLE WITH THIS RESULT, THEN  
 \*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 27.2: ACCURACY IN THE COMPARISON OF VARIABLES  
WITH CONSTANTS.

THIS SECTION COMPARES A VARIABLE AND A CONSTANT WHICH HAVE NUMERICALLY 'CLOSE' VALUES AND THEN DETERMINES WHETHER THE IMPLEMENTATION CAN DISTINGUISH BETWEEN THEM OR IF THEY ARE TREATED AS EQUAL.

BEGIN TEST.

NUMERIC VALUES BEING COMPARED	DIGITS OF DECIMAL ACCURACY	RESULT OF COMPARISONS
		>   =
-9.99998888E36 : -9.999999E36	6	T   F
-9.99998888E36 : -9.9999899E36	7	T   F
-9.99998888E36 : -9.99998899E36	8	T   F
-9.99998888E36 : -9.999988899E36	9	F   T

\*\*\* TEST INDICATES DECIMAL ACCURACY OF AT MOST 8 DIGITS. \*\*\*  
 IF DOCUMENTATION FOR THE IMPLEMENTATION-DEFINED ACCURACY  
   FOR NUMERIC VALUES IS COMPATIBLE WITH THIS RESULT, THEN  
 \*\*\* TEST PASSED \*\*\*

END TEST.

### SECTION 27.3: ACCURACY IN THE COMPARISON OF CONSTANTS.

THIS SECTION COMPARES TWO CONSTANTS WHICH HAVE NUMERICALLY 'CLOSE' VALUES AND THEN DETERMINES WHETHER THE IMPLEMENTATION CAN DISTINGUISH BETWEEN THEM OR IF THEY ARE TREATED AS EQUAL.

BEGIN TEST.

NUMERIC VALUES BEING COMPARED	DIGITS OF DECIMAL ACCURACY	RESULT OF COMPARISONS
		<   =
9.40395E-37 : 9.40396E-37	6	T   F
9.40395E-37 : 9.403951E-37	7	T   F
9.40395E-37 : 9.4039501E-37	8	F   T
9.40395E-37 : 9.40395001E-37	9	F   T

\*\*\* TEST INDICATES DECIMAL ACCURACY OF AT MOST 7 DIGITS. \*\*\*  
 IF DOCUMENTATION FOR THE IMPLEMENTATION-DEFINED ACCURACY  
   FOR NUMERIC VALUES IS COMPATIBLE WITH THIS RESULT, THEN  
 \*\*\* TEST PASSED \*\*\*

END TEST.

### SECTION 27.4: ACCURACY FOR CONSTANTS EXPRESSED WITH DIFFERENT EXPONENTS.

CONSTANTS MUST BE ACCURATE EVEN WHEN NOT EXPRESSED IN NORMALIZED FORM ( $1 \leq$  SIGNIFICAND  $< 10$ ).

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 27.5: COMPUTED ACCURACY OF NUMERIC VALUES.

THIS SECTION USES A COMPUTATIONAL ALGORITHM TO ATTEMPT TO DISCOVER THIS IMPLEMENTATION'S INTERNAL ACCURACY. BECAUSE THE ALGORITHM RELIES SOMEWHAT ON ACCURACY OF EXPRESSION EVALUATION, THE RESULTS ARE NOT ABSOLUTELY TRUSTWORTHY AND SHOULD BE INTERPRETED IN LIGHT OF GENERAL KNOWLEDGE OF THE CHARACTERISTICS OF THE SYSTEM.

\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*

BEGIN TEST.

COMPUTED ACCURACY = 8 DECIMAL DIGITS.

COMPUTED ACCURACY = 27 BINARY DIGITS.

END TEST.

END PROGRAM 27

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 28: EXCEPTION - DIVISION BY ZERO."
20 PRINT "      ANSI STANDARD 7.5"
30 PRINT
40 PRINT "THIS PROGRAM TESTS VARIOUS CASES OF DIVISION BY ZERO."
50 PRINT
60 PRINT "SECTION 28.1: POSITIVE NUMBER DIVIDED BY ZERO."
70 PRINT
80 PRINT "THIS SECTION TESTS THE RESULT OF DIVIDING A POSITIVE "
90 PRINT "QUANTITY BY ZERO."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
170 PRINT "          OF THE EXPRESSION AND EXECUTION CONTINUES."
180 PRINT
190 PRINT "                  BEGIN TEST."
200 PRINT
210 LET B=6
220 LET A=5/(B-B)
230 PRINT "VALUE SUPPLIED = ";A
240 PRINT
```

```
250 IF A > .99E38 THEN 290
260 PRINT "**** TEST FAILED: VALUE SUPPLIED LESS THAN MINIMUM"
270 PRINT "    POSITIVE MACHINE INFINITY. ***"
280 GOTO 320
290 PRINT "IF VALUE SUPPLIED = POSITIVE MACHINE INFINITY, AND"
300 PRINT "    EXCEPTION REPORTED, THEN"
310 PRINT "**** TEST PASSED ***"
320 PRINT
330 PRINT "                END TEST."
340 PRINT
1060 PRINT "SECTION 28.2: NEGATIVE NUMBER DIVIDED BY ZERO."
1070 PRINT
1080 PRINT "THIS SECTION TESTS THE RESULT OF DIVIDING A NEGATIVE "
1090 PRINT "QUANTITY BY ZERO."
1100 PRINT
1110 PRINT "TO PASS THIS TEST:"
1120 PRINT
1130 PRINT "    1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
1140 PRINT "        DISPLAYED, AND"
1150 PRINT
1160 PRINT "    2) NEGATIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
1170 PRINT "        OF THE EXPRESSION AND EXECUTION CONTINUES."
1180 PRINT
1190 PRINT "                BEGIN TEST."
1200 PRINT
1210 LET B=6
1220 LET A=-5/(B-B)
1230 PRINT "VALUE SUPPLIED = ";A
1240 PRINT
1250 IF A < -.99E38 THEN 1290
1260 PRINT "**** TEST FAILED: VALUE SUPPLIED GREATER THAN MINIMUM"
1270 PRINT "    NEGATIVE MACHINE INFINITY. ***"
1280 GOTO 1320
1290 PRINT "IF VALUE SUPPLIED = NEGATIVE MACHINE INFINITY, AND"
1300 PRINT "    EXCEPTION REPORTED, THEN"
1310 PRINT "**** TEST PASSED ***"
1320 PRINT
1330 PRINT "                END TEST."
1340 PRINT
2060 PRINT "SECTION 28.3: ZERO DIVIDED BY ZERO."
2070 PRINT
2080 PRINT "THIS SECTION TESTS THE RESULT OF DIVIDING ZERO BY ZERO."
2100 PRINT
2110 PRINT "TO PASS THIS TEST:"
2120 PRINT
2130 PRINT "    1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
2140 PRINT "        DISPLAYED, AND"
2150 PRINT
2160 PRINT "    2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
2170 PRINT "        OF THE EXPRESSION AND EXECUTION CONTINUES."
2180 PRINT
2190 PRINT "                BEGIN TEST."
2200 PRINT
```

```
2210 LET B=6
2220 LET A=0/(B-B)
2230 PRINT "VALUE SUPPLIED = ";A
2240 PRINT
2250 IF A > .99E38 THEN 2290
2260 PRINT "*** TEST FAILED: VALUE SUPPLIED LESS THAN MINIMUM"
2270 PRINT "    POSITIVE MACHINE INFINITY. ***"
2280 GOTO 2320
2290 PRINT "IF VALUE SUPPLIED = POSITIVE MACHINE INFINITY, AND"
2300 PRINT "    EXCEPTION REPORTED, THEN"
2310 PRINT "*** TEST PASSED ***"
2320 PRINT
2330 PRINT "                END TEST."
2340 PRINT
3000 PRINT "END PROGRAM 28"
3010 END
```

---

#### PROGRAM FILE 28: EXCEPTION - DIVISION BY ZERO.

ANSI STANDARD 7.5

THIS PROGRAM TESTS VARIOUS CASES OF DIVISION BY ZERO.

SECTION 28.1: POSITIVE NUMBER DIVIDED BY ZERO.

THIS SECTION TESTS THE RESULT OF DIVIDING A POSITIVE QUANTITY BY ZERO.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE EXPRESSION AND EXECUTION CONTINUES.

BEGIN TEST.

```
% DIVISION BY ZERO IN LINE 220
VALUE SUPPLIED = 1.70141E+38
```

```
IF VALUE SUPPLIED = POSITIVE MACHINE INFINITY, AND
    EXCEPTION REPORTED, THEN
*** TEST PASSED ***
```

END TEST.

SECTION 28.2: NEGATIVE NUMBER DIVIDED BY ZERO.

THIS SECTION TESTS THE RESULT OF DIVIDING A NEGATIVE QUANTITY BY ZERO.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) NEGATIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE EXPRESSION AND EXECUTION CONTINUES.

BEGIN TEST.

```
% DIVISION BY ZERO IN LINE 1220
VALUE SUPPLIED = -1.70141E+38
```

```
IF VALUE SUPPLIED = NEGATIVE MACHINE INFINITY, AND
    EXCEPTION REPORTED, THEN
*** TEST PASSED ***
```

END TEST.

SECTION 28.3: ZERO DIVIDED BY ZERO.

THIS SECTION TESTS THE RESULT OF DIVIDING ZERO BY ZERO.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE EXPRESSION AND EXECUTION CONTINUES.

BEGIN TEST.

```
% DIVISION BY ZERO IN LINE 2220
VALUE SUPPLIED = 1.70141E+38
```

```
IF VALUE SUPPLIED = POSITIVE MACHINE INFINITY, AND
    EXCEPTION REPORTED, THEN
*** TEST PASSED ***
```

END TEST.

END PROGRAM 28

```
*****
*****
```

```
10 PRINT "PROGRAM FILE 29: EXCEPTION - OVERFLOW OF NUMERIC EXPRESSIONS."
20 PRINT "    ANSI STANDARD 7.5"
30 PRINT
40 PRINT "THIS PROGRAM TESTS VARIOUS CASES OF NUMERIC OVERFLOW."
50 PRINT
```

```
60 PRINT "SECTION 29.1: POSITIVE OVERFLOW"
70 PRINT
80 PRINT "THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION"
90 PRINT "EXCEEDING POSITIVE MACHINE INFINITY."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
170 PRINT "          OF THE EXPRESSION AND EXECUTION CONTINUES."
180 PRINT
190 PRINT "          BEGIN TEST"
200 PRINT
210 LET M=10
220 LET F=1
230 LET A=1E36
240 LET B=1
250 PRINT "ABOUT TO COMPUTE ";A;" * ";M
260 LET A=A*M
270 PRINT "RESULT = ";A
280 PRINT
290 PRINT "-----"
300 PRINT
310 IF B>=A THEN 360
320 LET B=A
330 LET F=F*1.1
340 LET M=10^F
350 GOTO 250
360 REM CONVERGE
370 IF A>.99E38 THEN 410
380 PRINT "*** TEST FAILED: VALUE SUPPLIED LESS THAN MINIMUM"
390 PRINT "      POSITIVE MACHINE INFINITY. ***"
400 GOTO 440
410 PRINT "LAST TWO MULTIPLICATIONS SHOULD HAVE BEEN REPORTED"
420 PRINT "AS OVERFLOW, AND POSITIVE INFINITY SUPPLIED; IF SO,"
430 PRINT "*** TEST PASSED *** OTHERWISE *** TEST FAILED ***"
440 PRINT
450 PRINT "          END TEST"
460 PRINT
470 PRINT "SECTION 29.2: NEGATIVE OVERFLOW"
480 PRINT
490 PRINT "THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION"
500 PRINT "EXCEEDING NEGATIVE MACHINE INFINITY."
510 PRINT
520 PRINT "TO PASS THIS TEST:"
530 PRINT
540 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
550 PRINT "          DISPLAYED, AND"
560 PRINT
570 PRINT "      2) NEGATIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
580 PRINT "          OF THE EXPRESSION AND EXECUTION CONTINUES."
590 PRINT
```

```

600 PRINT "                  BEGIN TEST"
610 PRINT
620 LET M=10
630 LET F=1
640 LET A=-1E36
650 LET B=1
660 PRINT "ABOUT TO COMPUTE ";A;" * ";M
670 LET A=A*M
680 PRINT "RESULT = ";A
690 PRINT
700 PRINT "-----"
710 PRINT
720 IF B<=A THEN 770
730 LET B=A
740 LET F=F*1.1
750 LET M=10^F
760 GOTO 660
770 REM CONVERGE
780 IF A<-.99E38 THEN 820
790 PRINT "*** TEST FAILED: VALUE SUPPLIED GREATER THAN MINIMUM"
800 PRINT "      NEGATIVE MACHINE INFINITY. ***"
810 GOTO 850
820 PRINT "LAST TWO MULTIPLICATIONS SHOULD HAVE BEEN REPORTED"
830 PRINT "AS OVERFLOW, AND NEGATIVE INFINITY SUPPLIED; IF SO,"
840 PRINT "*** TEST PASSES *** OTHERWISE *** TEST FAILS *** "
850 PRINT
860 PRINT "                  END TEST"
870 PRINT
880 PRINT "END PROGRAM 29"
890 END

```

---

PROGRAM FILE 29: EXCEPTION - OVERFLOW OF NUMERIC EXPRESSIONS.  
ANSI STANDARD 7.5

THIS PROGRAM TESTS VARIOUS CASES OF NUMERIC OVERFLOW.

#### SECTION 29.1: POSITIVE OVERFLOW

THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION  
EXCEEDING POSITIVE MACHINE INFINITY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE  
OF THE EXPRESSION AND EXECUTION CONTINUES.

BEGIN TEST

ABOUT TO COMPUTE 1.00000E+36 \* 10  
RESULT = 1.00000E+37

---

ABOUT TO COMPUTE 1.00000E+37 \* 12.5893  
RESULT = 1.25893E+38

---

ABOUT TO COMPUTE 1.25893E+38 \* 16.2181

% OVERFLOW IN LINE 260  
RESULT = 1.70141E+38

---

ABOUT TO COMPUTE 1.70141E+38 \* 21.4289

% OVERFLOW IN LINE 260  
RESULT = 1.70141E+38

---

LAST TWO MULTIPLICATIONS SHOULD HAVE BEEN REPORTED AS OVERFLOW, AND POSITIVE INFINITY SUPPLIED; IF SO,  
\*\*\* TEST PASSED \*\*\* OTHERWISE \*\*\* TEST FAILED \*\*\*

END TEST

#### SECTION 29.2: NEGATIVE OVERFLOW

THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION EXCEEDING NEGATIVE MACHINE INFINITY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) NEGATIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE EXPRESSION AND EXECUTION CONTINUES.

BEGIN TEST

ABOUT TO COMPUTE -1.00000E+36 \* 10  
RESULT = -1.00000E+37

---

ABOUT TO COMPUTE -1.00000E+37 \* 12.5893  
RESULT = -1.25893E+38

---

ABOUT TO COMPUTE -1.25893E+38 \* 16.2181

% OVERFLOW IN LINE 670  
RESULT = -1.70141E+38

---

ABOUT TO COMPUTE -1.70141E+38 \* 21.4289

% OVERFLOW IN LINE 670  
RESULT = -1.70141E+38

---

LAST TWO MULTIPLICATIONS SHOULD HAVE BEEN REPORTED AS OVERFLOW, AND NEGATIVE INFINITY SUPPLIED; IF SO,  
\*\*\* TEST PASSES \*\*\* OTHERWISE \*\*\* TEST FAILS \*\*\*

END TEST

END PROGRAM 29

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 30: EXCEPTION - OVERFLOW OF NUMERIC CONSTANTS."
20 PRINT "      ANSI STANDARD 5.4, 5.5"
30 PRINT
40 PRINT "THIS PROGRAM TESTS VARIOUS CASES OF CONSTANT OVERFLOW."
50 PRINT
60 PRINT "SECTION 30.1: POSITIVE CONSTANT OVERFLOW"
70 PRINT
80 PRINT "THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION"
90 PRINT "EXCEEDING POSITIVE MACHINE INFINITY."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
170 PRINT "          OF THE CONSTANT AND EXECUTION CONTINUES."
180 PRINT
190 PRINT "                  BEGIN TEST"
200 PRINT
360 LET A=3E99999
365 PRINT "RESULT OF ASSIGNING 3E99999 = ",A
370 IF A>.99E38 THEN 410
380 PRINT "*** TEST FAILED: VALUE SUPPLIED LESS THAN MINIMUM"
390 PRINT "      POSITIVE MACHINE INFINITY. ***"
400 GOTO 440
410 REM RESULT > MINIMUM

```

```

420 PRINT "IF (RESULT = 3E99999) OR "
425 PRINT " (RESULT = POSITIVE INFINITY AND EXCEPTION REPORTED)"
430 PRINT "*** TEST PASSED *** OTHERWISE *** TEST FAILED ***"
440 PRINT
450 PRINT "           END TEST"
460 PRINT
470 PRINT "SECTION 30.2: NEGATIVE CONSTANT OVERFLOW"
480 PRINT
490 PRINT "THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION"
500 PRINT "EXCEEDING NEGATIVE MACHINE INFINITY."
510 PRINT
520 PRINT "TO PASS THIS TEST:"
530 PRINT
540 PRINT "    1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
550 PRINT "        DISPLAYED, AND"
560 PRINT
570 PRINT "    2) NEGATIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
580 PRINT "        OF THE CONSTANT AND EXECUTION CONTINUES."
590 PRINT
600 PRINT "           BEGIN TEST"
610 PRINT
770 LET A=-3E99999
775 PRINT "RESULT OF ASSIGNING -3E99999 = ";A
780 IF A<-.99E38 THEN 820
790 PRINT "*** TEST FAILED: VALUE SUPPLIED GREATER THAN MINIMUM"
800 PRINT "    NEGATIVE MACHINE INFINITY. ***"
810 GOTO 850
820 REM RESULT < MAXIMUM
830 PRINT "IF (RESULT = -3E99999) OR "
835 PRINT " (RESULT = NEGATIVE INFINITY AND EXCEPTION REPORTED)"
840 PRINT "*** TEST PASSES *** OTHERWISE *** TEST FAILS ***"
850 PRINT
860 PRINT "           END TEST"
870 PRINT
880 PRINT "END PROGRAM 30"
890 END

```

---

PROGRAM FILE 30: EXCEPTION - OVERFLOW OF NUMERIC CONSTANTS.  
ANSI STANDARD 5.4, 5.5

THIS PROGRAM TESTS VARIOUS CASES OF CONSTANT OVERFLOW.

#### SECTION 30.1: POSITIVE CONSTANT OVERFLOW

THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION  
EXCEEDING POSITIVE MACHINE INFINITY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND

2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE CONSTANT AND EXECUTION CONTINUES.

BEGIN TEST

```
% OVERFLOW IN LINE 360
RESULT OF ASSIGNING 3E99999 = 1.70141E+38
IF (RESULT = 3E99999) OR
(RESULT = POSITIVE INFINITY AND EXCEPTION REPORTED)
*** TEST PASSED *** OTHERWISE *** TEST FAILED ***
```

END TEST

## SECTION 30.2: NEGATIVE CONSTANT OVERFLOW

THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION EXCEEDING NEGATIVE MACHINE INFINITY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) NEGATIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE CONSTANT AND EXECUTION CONTINUES.

BEGIN TEST

```
% OVERFLOW IN LINE 770
RESULT OF ASSIGNING -3E99999 = -1.70141E+38
IF (RESULT = -3E99999) OR
(RESULT = NEGATIVE INFINITY AND EXCEPTION REPORTED)
*** TEST PASSES *** OTHERWISE *** TEST FAILS ***
```

END TEST

END PROGRAM 30

```
*****  
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 31: EXCEPTION - ZERO RAISED TO A NEGATIVE POWER."
20 PRINT "      ANSI STANDARD 7.5"
50 PRINT
60 PRINT "SECTION 31.1: EXCEPTION - ZERO RAISED TO A NEGATIVE POWER."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
170 PRINT "          OF THE EXPRESSION AND EXECUTION CONTINUES."
180 PRINT
```

```
190 PRINT "                BEGIN TEST"
200 PRINT
210 LET B=-6
220 LET A=0^B
230 PRINT "VALUE SUPPLIED = ";A
240 PRINT
250 IF A > .99E38 THEN 290
260 PRINT "*** TEST FAILED: VALUE SUPPLIED LESS THAN MINIMUM"
270 PRINT "    POSITIVE MACHINE INFINITY. ***"
280 GOTO 320
290 PRINT "IF VALUE SUPPLIED = POSITIVE MACHINE INFINITY, AND"
300 PRINT "    EXCEPTION REPORTED, THEN"
310 PRINT "*** TEST PASSED ***"
320 PRINT
330 PRINT "                END TEST"
340 PRINT
350 PRINT "END PROGRAM 31"
360 END
```

---

PROGRAM FILE 31: EXCEPTION - ZERO RAISED TO A NEGATIVE POWER.  
ANSI STANDARD 7.5

SECTION 31.1: EXCEPTION - ZERO RAISED TO A NEGATIVE POWER.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE EXPRESSION AND EXECUTION CONTINUES.

BEGIN TEST

% ZERO TO A NEGATIVE POWER IN LINE 220  
VALUE SUPPLIED = 1.70141E+38

IF VALUE SUPPLIED = POSITIVE MACHINE INFINITY, AND  
EXCEPTION REPORTED, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 31

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 32: EXCEPTION - NEGATIVE QUANTITY RAISED TO A"
20 PRINT "      NON-INTEGRAL POWER."
30 PRINT "      ANSI STANDARD 7.5"
40 PRINT
50 PRINT "SECTION 32.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO A"
60 PRINT "      NON-INTEGRAL POWER."
70 PRINT
80 PRINT "TO PASS THIS TEST:"
90 PRINT
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
110 PRINT "          DISPLAYED, AND"
120 PRINT
130 PRINT "      2) EXECUTION MUST TERMINATE"
140 PRINT
150 PRINT "          BEGIN TEST"
160 PRINT
170 PRINT "ABOUT TO ATTEMPT EVALUATION OF (-2) ^ 6.00001:"
200 PRINT
210 LET B=6.00001
220 LET A=-2
230 LET C=A^B
240 PRINT
250 PRINT "VALUE SUPPLIED = ";C
260 PRINT
270 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
280 PRINT
290 PRINT "          END TEST"
300 PRINT
310 PRINT "END PROGRAM 32"
320 END
```

---

PROGRAM FILE 32: EXCEPTION - NEGATIVE QUANTITY RAISED TO A  
NON-INTEGRAL POWER.  
ANSI STANDARD 7.5

SECTION 32.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO A  
NON-INTEGRAL POWER.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE

BEGIN TEST

ABOUT TO ATTEMPT EVALUATION OF (-2) ^ 6.00001:

% NEGATIVE VALUE RAISED TO NON-INTEGRAL POWER IN LINE 230

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 33: EXCEPTION - UNDERFLOW OF"  
15 PRINT "      NUMERIC EXPRESSIONS."  
20 PRINT "      ANSI STANDARD 7.4"  
30 PRINT  
40 PRINT "THIS PROGRAM TESTS VARIOUS CASES OF NUMERIC UNDERFLOW."  
50 PRINT  
60 PRINT "SECTION 33.1: POSITIVE UNDERFLOW"  
70 PRINT  
80 PRINT "THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION"  
90 PRINT "LESS THAN POSITIVE MACHINE INFINITESIMAL."  
100 PRINT  
110 PRINT "TO PASS THIS TEST:"  
120 PRINT  
130 PRINT "      1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE"  
140 PRINT "          DISPLAYED (NOT MANDATORY), AND"  
150 PRINT  
160 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"  
170 PRINT "          OF THE EXPRESSION."  
180 PRINT  
190 PRINT "          BEGIN TEST"  
200 PRINT  
210 REM S IS A SWITCH TO INDICATE FIRST TIME THROUGH LOOP (S=1)  
220 REM OR SUBSEQUENT ITERATIONS (S=0). UNDERFLOW SHOULD NOT OCCUR  
230 REM FIRST TIME.  
240 LET S=1  
250 LET M=10  
260 LET F=1  
270 LET A=.8E-36  
280 LET B=1  
290 PRINT "ABOUT TO COMPUTE ";A;" / ";M  
300 LET A=A/M  
310 PRINT "RESULT = ";A  
320 PRINT  
330 PRINT "-----"  
340 PRINT  
350 IF B<=A THEN 420  
360 IF A=0 THEN 420  
370 LET B=A  
380 LET S=0  
390 LET F=F*1.1  
400 LET M=10^F  
410 GOTO 290  
420 REM CONVERGE  
430 IF S=0 THEN 470  
440 PRINT "*** TEST FAILED: UNDERFLOW OCCURRED ABOVE MAXIMUM"  
450 PRINT "      MACHINE INFINITESIMAL. ***"  
460 GOTO 510  
470 IF A=0 THEN 500  
480 PRINT "*** TEST FAILED: VALUE SUPPLIED NOT EQUAL ZERO. ***"
```

```
490 GOTO 510
500 PRINT "**** TEST PASSED *** "
510 PRINT
520 PRINT "                      END TEST"
530 PRINT
540 PRINT "SECTION 33.2: NEGATIVE UNDERFLOW"
550 PRINT
560 PRINT "THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION"
570 PRINT "GREATER THAN NEGATIVE MACHINE INFINITESIMAL."
580 PRINT
590 PRINT "TO PASS THIS TEST:"
600 PRINT
610 PRINT "      1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE"
620 PRINT "          DISPLAYED (NOT MANDATORY), AND"
630 PRINT
640 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"
650 PRINT "          OF THE EXPRESSION."
660 PRINT
670 PRINT "                      BEGIN TEST"
680 PRINT
690 LET S=1
700 LET M=10
710 LET F=1
720 LET A=-.8E-36
730 LET B=-1
740 PRINT "ABOUT TO COMPUTE ";A;" / ";M
750 LET A=A/M
760 PRINT "RESULT = ";A
770 PRINT
780 PRINT "-----"
790 PRINT
800 IF B>=A THEN 870
810 IF A=0 THEN 870
820 LET B=A
830 LET S=0
840 LET F=F*1.1
850 LET M=10^F
860 GOTO 740
870 REM CONVERGE
880 IF S=0 THEN 920
890 PRINT "**** TEST FAILED: UNDERFLOW OCCURRED BELOW MINIMUM"
900 PRINT "          MACHINE INFINITESIMAL. ***"
910 GOTO 960
920 IF A=0 THEN 950
930 PRINT "**** TEST FAILED: VALUE SUPPLIED NOT EQUAL ZERO. ***"
940 GOTO 960
950 PRINT "**** TEST PASSED *** "
960 PRINT
970 PRINT "                      END TEST"
```

```
980 PRINT
990 PRINT "END PROGRAM 33"
1000 END
```

---

PROGRAM FILE 33: EXCEPTION - UNDERFLOW OF  
NUMERIC EXPRESSIONS.

ANSI STANDARD 7.4

THIS PROGRAM TESTS VARIOUS CASES OF NUMERIC UNDERFLOW.

#### SECTION 33.1: POSITIVE UNDERFLOW

THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION  
LESS THAN POSITIVE MACHINE INFINITESIMAL.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE  
DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE  
OF THE EXPRESSION.

BEGIN TEST

```
ABOUT TO COMPUTE 8.00000E-37 / 10
RESULT = 8.00000E-38
```

---

```
ABOUT TO COMPUTE 8.00000E-38 / 12.5893
RESULT = 6.35463E-39
```

---

```
ABOUT TO COMPUTE 6.35463E-39 / 16.2181
```

```
% UNDERFLOW IN LINE 300
RESULT = 0
```

---

\*\*\* TEST PASSED \*\*\*

END TEST

#### SECTION 33.2: NEGATIVE UNDERFLOW

THIS SECTION TESTS THE RESULT OF EXPRESSION EVALUATION  
GREATER THAN NEGATIVE MACHINE INFINITESIMAL.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE EXPRESSION.

BEGIN TEST

ABOUT TO COMPUTE -8.00000E-37 / 10  
RESULT = -8.00000E-38

---

ABOUT TO COMPUTE -8.00000E-38 / 12.5893  
RESULT = -6.35463E-39

---

ABOUT TO COMPUTE -6.35463E-39 / 16.2181

% UNDERFLOW IN LINE 750  
RESULT = 0

---

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 33

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 34: EXCEPTION - UNDERFLOW OF"  
15 PRINT "      NUMERIC CONSTANTS."  
20 PRINT "      ANSI STANDARD 5.4, 5.6"  
30 PRINT  
40 PRINT "THIS PROGRAM TESTS VARIOUS CASES OF CONSTANT UNDERFLOW."  
50 PRINT  
60 PRINT "SECTION 34.1: POSITIVE CONSTANT UNDERFLOW"  
70 PRINT  
80 PRINT "THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION"  
90 PRINT "LESS THAN POSITIVE MACHINE INFINITESIMAL."  
100 PRINT  
110 PRINT "TO PASS THIS TEST:"  
120 PRINT  
130 PRINT "      1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE"  
140 PRINT "      DISPLAYED (NOT MANDATORY), AND"  
150 PRINT  
160 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"  
170 PRINT "      OF THE CONSTANT AND EXECUTION CONTINUES."  
180 PRINT
```

```

190 PRINT "          BEGIN TEST"
200 PRINT
360 LET A=3E-99999
365 PRINT "RESULT OF ASSIGNING 3E-99999 = ";A
370 IF A=0 THEN 430
380 PRINT "IF RESULT <> 3E-99999, THEN"
390 PRINT "*** TEST FAILED ***"
400 GOTO 440
430 PRINT "*** TEST PASSED ***"
440 PRINT
450 PRINT "          END TEST"
460 PRINT
470 PRINT "SECTION 34.2: NEGATIVE CONSTANT UNDERFLOW"
480 PRINT
490 PRINT "THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION"
500 PRINT "GREATER THAN NEGATIVE MACHINE INFINITESIMAL."
510 PRINT
520 PRINT "TO PASS THIS TEST:"
530 PRINT
540 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE"
550 PRINT "          DISPLAYED (NOT MANDATORY), AND"
560 PRINT
570 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"
580 PRINT "          OF THE CONSTANT AND EXECUTION CONTINUES."
590 PRINT
600 PRINT "          BEGIN TEST"
610 PRINT
770 LET A=-3E-99999
775 PRINT "RESULT OF ASSIGNING -3E-99999 = ";A
780 IF A=0 THEN 840
790 PRINT "IF RESULT <> -3E-99999, THEN"
800 PRINT "*** TEST FAILED ***"
810 GOTO 850
840 PRINT "*** TEST PASSED ***"
850 PRINT
860 PRINT "          END TEST"
870 PRINT
880 PRINT "END PROGRAM 34"
890 END

```

---

PROGRAM FILE 34: EXCEPTION - UNDERFLOW OF  
NUMERIC CONSTANTS.  
ANSI STANDARD 5.4, 5.6

THIS PROGRAM TESTS VARIOUS CASES OF CONSTANT UNDERFLOW.

SECTION 34.1: POSITIVE CONSTANT UNDERFLOW

THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION  
LESS THAN POSITIVE MACHINE INFINITESIMAL.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE CONSTANT AND EXECUTION CONTINUES.

BEGIN TEST

RESULT OF ASSIGNING 3E-99999 = 0  
 \*\*\* TEST PASSED \*\*\*

END TEST

#### SECTION 34.2: NEGATIVE CONSTANT UNDERFLOW

THIS SECTION TESTS THE RESULT OF CONSTANT EVALUATION GREATER THAN NEGATIVE MACHINE INFINITESIMAL.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE CONSTANT AND EXECUTION CONTINUES.

BEGIN TEST

RESULT OF ASSIGNING -3E-99999 = 0  
 \*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 34

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

```

10 PRINT "PROGRAM FILE 35: EXCEPTION - OVERFLOW AND UNDERFLOW WITHIN"
20 PRINT "      SUB-EXPRESSIONS"
30 PRINT "      ANSI STANDARD 7.4, 7.5"
40 PRINT
50 PRINT "THIS PROGRAM TESTS THE EFFECT OF NUMERIC OVERFLOW AND"
60 PRINT "UNDERFLOW WITHIN A SUB-EXPRESSION ON THE CONTAINING"
70 PRINT "EXPRESSION."
80 PRINT
90 PRINT "SECTION 35.1: SUB-EXPRESSION OVERFLOW"
100 PRINT
110 PRINT "THIS SECTION TESTS THE EFFECT ON AN EXPRESSION OF NUMERIC"
120 PRINT "OVERFLOW WITHIN ONE OF ITS SUB-EXPRESSIONS."
130 PRINT
140 PRINT "TO PASS THIS TEST:"
150 PRINT

```

```
160 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
170 PRINT "          DISPLAYED, AND"  
180 PRINT  
190 PRINT "      2) MACHINE INFINITY MUST BE SUPPLIED AS THE VALUE"  
200 PRINT "          OF THE SUB-EXPRESSION."  
210 PRINT  
220 PRINT "              BEGIN TEST"  
230 PRINT  
240 PRINT "ABOUT TO COMPUTE -.01 * (10 ^ 99999)"  
250 LET A=-.01 * (10 ^ 99999)  
260 PRINT "RESULT = ";A  
270 PRINT  
280 IF A <= -.99E36 THEN 310  
290 PRINT "**** TEST FAILED: RESULT > -1E36 ****"  
300 GOTO 340  
310 PRINT "IF RESULT = (-.01 * MACHINE INFINITY)"  
320 PRINT "    AND OVERFLOW EXCEPTION REPORTED, THEN"  
330 PRINT "**** TEST PASSES *** OTHERWISE *** TEST FAILS ***"  
340 PRINT  
350 PRINT "              END TEST"  
360 PRINT  
370 PRINT "SECTION 35.2: SUB-EXPRESSION UNDERFLOW"  
380 PRINT  
390 PRINT "THIS SECTION TESTS THE EFFECT ON AN EXPRESSION OF NUMERIC"  
400 PRINT "UNDERFLOW WITHIN ONE OF ITS OPERATIONS."  
410 PRINT  
420 PRINT "TO PASS THIS TEST:"  
430 PRINT  
440 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE"  
450 PRINT "          DISPLAYED (NOT MANDATORY), AND"  
460 PRINT  
470 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"  
480 PRINT "          OF THE SUB-EXPRESSION."  
490 PRINT  
500 PRINT "              BEGIN TEST"  
510 PRINT  
520 PRINT "ABOUT TO COMPUTE 3 + (10 ^ (-99999))"  
530 LET A=3 + (10 ^ (-99999))  
540 PRINT "RESULT = ";A  
550 PRINT  
560 IF A=3 THEN 590  
570 PRINT "**** TEST FAILED: RESULT NOT = 3 ****"  
580 GOTO 600  
590 PRINT "**** TEST PASSED ***"  
600 PRINT  
610 PRINT "              END TEST"  
620 PRINT  
630 PRINT "END PROGRAM 35"  
640 END
```

## PROGRAM FILE 35: EXCEPTION - OVERFLOW AND UNDERFLOW WITHIN

SUB-EXPRESSIONS

ANSI STANDARD 7.4, 7.5

THIS PROGRAM TESTS THE EFFECT OF NUMERIC OVERFLOW AND UNDERFLOW WITHIN A SUB-EXPRESSION ON THE CONTAINING EXPRESSION.

## SECTION 35.1: SUB-EXPRESSION OVERFLOW

THIS SECTION TESTS THE EFFECT ON AN EXPRESSION OF NUMERIC OVERFLOW WITHIN ONE OF ITS SUB-EXPRESSIONS.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) MACHINE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE SUB-EXPRESSION.

BEGIN TEST

ABOUT TO COMPUTE  $-.01 * (10 ^ 99999)$

% OVERFLOW IN LINE 250  
RESULT =  $-1.70141 \times 10^{36}$

IF RESULT =  $(-.01 * \text{MACHINE INFINITY})$   
AND OVERFLOW EXCEPTION REPORTED, THEN  
\*\*\* TEST PASSES \*\*\* OTHERWISE \*\*\* TEST FAILS \*\*\*

END TEST

## SECTION 35.2: SUB-EXPRESSION UNDERFLOW

THIS SECTION TESTS THE EFFECT ON AN EXPRESSION OF NUMERIC UNDERFLOW WITHIN ONE OF ITS OPERATIONS.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE SUB-EXPRESSION.

BEGIN TEST

ABOUT TO COMPUTE  $3 + (10 ^ (-99999))$

% UNDERFLOW IN LINE 530  
RESULT = 3

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 35

```
*****
*****  

10 PRINT "PROGRAM FILE 36: ERROR - UNMATCHED PARENTHESES IN"  

15 PRINT "      NUMERIC EXPRESSION."  

20 PRINT "      ANSI STANDARD 7.2"  

30 PRINT  

40 PRINT "SECTION 36.1: ERROR - UNMATCHED PARENTHESES IN"  

45 PRINT "      NUMERIC EXPRESSION."  

50 PRINT  

60 PRINT "THIS SECTION TESTS WHETHER THE PROCESSOR DETECTS AND REPORTS"  

70 PRINT "A NUMERIC EXPRESSION WHICH IS IMPROPERLY FORMED BECAUSE THE"  

80 PRINT "PARENTHESES ARE NOT MATCHED."  

100 PRINT  

110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  

120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  

130 PRINT  

140 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  

150 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  

160 PRINT "      BY THE PROCESSOR, OR"  

170 PRINT  

180 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  

190 PRINT  

200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  

210 PRINT "FOR DETAILED CRITERIA."  

220 PRINT  

230 PRINT "      BEGIN TEST"  

240 PRINT  

245 PRINT "ABOUT TO COMPUTE: 8+(7-(9-88)/3+(7-9)-3"  

250 LET A=8+(7-(9-88)/3+(7-9)-3  

260 PRINT "VALUE ASSIGNED FROM ILLEGAL EXPRESSION = ";A  

270 PRINT  

280 PRINT "      END TEST"  

290 PRINT  

300 PRINT "END PROGRAM 36"  

310 END
```

---

? ILLEGAL FORMULA IN LINE 250

```
*****
*****  

10 PRINT "PROGRAM FILE 37: ERROR - USE OF '***' AS OPERATOR."  

20 PRINT "      ANSI STANDARD 7.2"  

30 PRINT
```

```

40 PRINT "SECTION 37.1: ERROR - USE OF '***' AS OPERATOR."
50 PRINT
60 PRINT "THIS SECTION TESTS WHETHER THE PROCESSOR DETECTS AND REPORTS"
70 PRINT "A NUMERIC EXPRESSION WHICH IS IMPROPERLY FORMED BECAUSE A"
80 PRINT "DOUBLE ASTERISK (**) IS USED AS AN OPERATOR, PRESUMABLY"
90 PRINT "FOR INVOLUTION."
100 PRINT
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
130 PRINT
140 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
150 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
160 PRINT "        BY THE PROCESSOR, OR"
170 PRINT
180 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
190 PRINT
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
210 PRINT "FOR DETAILED CRITERIA."
220 PRINT
230 PRINT "                BEGIN TEST"
240 PRINT
250 LET A=5**2
260 PRINT "VALUE ASSIGNED FOR 5**2 = ";A
270 PRINT
280 PRINT "                END TEST"
290 PRINT
300 PRINT "END PROGRAM 37"
310 END

```

---

? ILLEGAL OPERATOR IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 38: ERROR - USE OF ADJACENT OPERATORS."
20 PRINT "    ANSI STANDARD 7.2"
30 PRINT
40 PRINT "SECTION 38.1: ERROR - USE OF ADJACENT OPERATORS."
50 PRINT
60 PRINT "THIS SECTION TESTS WHETHER THE PROCESSOR DETECTS AND REPORTS"
70 PRINT "A NUMERIC EXPRESSION WHICH IS IMPROPERLY FORMED BECAUSE TWO"
80 PRINT "OPERATORS ARE USED IN SUCCESSION."
100 PRINT
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
130 PRINT
140 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
150 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
160 PRINT "        BY THE PROCESSOR, OR"
170 PRINT
180 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
190 PRINT

```

```

200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
210 PRINT "FOR DETAILED CRITERIA."
220 PRINT
230 PRINT "           BEGIN TEST"
240 PRINT
250 LET A=4 ^ -2
260 PRINT "VALUE ASSIGNED FOR 4 ^ -2 = ";A
270 PRINT
280 PRINT "           END TEST"
290 PRINT
300 PRINT "END PROGRAM 38"
310 END

```

---

? ILLEGAL FORMULA IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 39: ACCURACY OF ADDITION"
20 PRINT "      ANSI STANDARD 7.2, 7.4, 7.6"
30 PRINT
40 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'"
50 PRINT "      WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
60 PRINT "      THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE"
70 PRINT "      VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
80 PRINT
100 PRINT "SECTION 39.1 ACCURACY OF ADDITION."
110 PRINT
120 PRINT "THIS SECTION TESTS THE ACCURACY OF ADDITION. THE COMPUTED"
130 PRINT "RESULT IS COMPARED WITH A RANGE ESTABLISHED BY PERTURBING"
140 PRINT "EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT PASSES IF"
150 PRINT "IT FALLS WITHIN THE EXTREME VALUES GENERATED BY THIS"
160 PRINT "PERTURBATION (ACCURATE TO SIX DIGITS)."
220 PRINT
230 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
240 PRINT "DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS."
250 PRINT
260 PRINT "           BEGIN TEST"
270 PRINT
280 PRINT "FIRST","SECOND","TRUE","COMPUTED","TEST"
290 PRINT "OPERAND","OPERAND","VALUE","VALUE","RESULT"
300 PRINT
310 LET F=0
320 READ A1,A2,T,L,H
330 IF A1=999 THEN 480
335 LET M$="PASS"
340 LET C=A1+A2
350 IF C<L THEN 380
360 IF C>H THEN 380
370 GOTO 400

```

```
380 LET M$="FAIL"
390 LET F=F+1
400 PRINT A1,A2,T,C,M$
410 GOTO 320
480 PRINT
490 IF F=0 THEN 520
500 PRINT "**** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ****"
510 GOTO 530
520 PRINT "**** INFORMATIVE TEST PASSED ****"
530 PRINT
540 PRINT " END TEST"
550 PRINT
560 PRINT "END PROGRAM 39"
570 DATA 0,0,0,0,0
2000 DATA 0,765.432,765.432,765.431,765.433
3000 DATA 0.876543000E+03, 0.123453000E+03, 0.999996000E+03
3010 DATA 0.999993499E+03, 0.999998500E+03
3020 DATA 0.811111000E-01, 0.111111000E-01, 0.922222000E-01
3030 DATA 0.922219499E-01, 0.922224500E-01
3040 DATA -0.579999000E+02,-0.111111000E+00,-0.581110110E+02
3050 DATA -0.581111621E+02,-0.581108600E+02
3060 DATA 0.111111100E+07, 0.111111100E+07, 0.222222200E+07
3070 DATA 0.222219700E+07, 0.222224700E+07
3080 DATA 0.654321000E+01, 0.123456000E+01, 0.777777000E+01
3090 DATA 0.777774499E+01, 0.777779500E+01
3100 DATA 0.654321000E-09, 0.123456000E-09, 0.777777000E-09
3110 DATA 0.777774499E-09, 0.777779500E-09
3120 DATA -0.654321000E-19,-0.123456000E-19,-0.777777000E-19
3130 DATA -0.777779500E-19,-0.777774499E-19
3140 DATA 0.654321000E-29, 0.123456000E-29, 0.777777000E-29
3150 DATA 0.777774500E-29, 0.777779500E-29
3160 DATA 0.654321000E-37, 0.123456000E-37, 0.777777000E-37
3170 DATA 0.777774500E-37, 0.777779500E-37
3180 DATA 0.123456000E+11, 0.654321000E+11, 0.777777000E+11
3190 DATA 0.777774500E+11, 0.777779500E+11
3200 DATA -0.123456000E+21,-0.654321000E+21,-0.777777000E+21
3210 DATA -0.777779500E+21,-0.777774500E+21
3220 DATA 0.123456000E+31, 0.654321000E+31, 0.777777000E+31
3230 DATA 0.777774499E+31, 0.777779500E+31
3240 DATA 0.123456000E+38, 0.654321000E+38, 0.777777000E+38
3250 DATA 0.777774499E+38, 0.777779500E+38
3260 DATA 0.499997000E-01, 0.499996000E-01, 0.999993000E-01
3270 DATA 0.999990499E-01, 0.999995500E-01
3280 DATA 0.999993000E-01, 0.400000000E-06, 0.999997000E-01
3290 DATA 0.999995499E-01, 0.999998501E-01
3300 DATA 0.999993000E-01, 0.400000000E-07, 0.999993400E-01
3310 DATA 0.999991899E-01, 0.999994901E-01
3320 DATA 0.999993000E-01, 0.400000000E-08, 0.999993040E-01
3330 DATA 0.999991539E-01, 0.999994541E-01
3340 DATA 0.999993000E-01, 0.400000000E-09, 0.999993004E-01
3350 DATA 0.999991503E-01, 0.999994505E-01
3360 DATA 0.999993000E-01, 0.400000000E-10, 0.999993000E-01
3370 DATA 0.999991500E-01, 0.999994501E-01
3380 DATA -0.102030400E+07, 0.400000000E+01,-0.102030000E+07
3390 DATA -0.102031501E+07,-0.102028499E+07
```

```

3400 DATA 0.102030400E+07,-0.304000000E+03, 0.102000000E+07
3410 DATA 0.101998499E+07, 0.102001501E+07
3420 DATA -0.102030400E+07, 0.203040000E+05,-0.100000000E+07
3430 DATA -0.100001510E+07,-0.999989400E+06
3440 DATA -0.102030400E+07, 0.102030400E+07, 0.000000000E+00
3450 DATA -0.200000500E+02, 0.200000500E+02
3460 DATA -0.102030400E+00, 0.400000000E-06,-0.102030000E+00
3470 DATA -0.102031501E+00,-0.102028499E+00
3480 DATA 0.102030400E+00,-0.304000000E-04, 0.102000000E+00
3490 DATA 0.101998499E+00, 0.102001501E+00
3500 DATA -0.102030400E+00, 0.203040000E-02,-0.100000000E+00
3510 DATA -0.100001510E+00,-0.999989400E-01
3520 DATA -0.102030400E+00, 0.102030400E+00, 0.000000000E+00
3530 DATA -0.200000500E-05, 0.200000500E-05
3540 DATA -0.102030400E+00, 0.102030300E+00,-0.100000000E-06
3550 DATA -0.210000501E-05, 0.190000500E-05
3560 DATA -0.102030400E+00, 0.102030500E+00, 0.100000000E-06
3570 DATA -0.190000500E-05, 0.210000501E-05
5000 DATA 999,0,0,0,0
5010 END

```

---

**PROGRAM FILE 39: ACCURACY OF ADDITION**  
**ANSI STANDARD 7.2, 7.4, 7.6**

\*\*\* NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'  
 WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW  
 THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE  
 VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

**SECTION 39.1 ACCURACY OF ADDITION.**

THIS SECTION TESTS THE ACCURACY OF ADDITION. THE COMPUTED  
 RESULT IS COMPARED WITH A RANGE ESTABLISHED BY PERTURBING  
 EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT PASSES IF  
 IT FALLS WITHIN THE EXTREME VALUES GENERATED BY THIS  
 PERTURBATION (ACCURATE TO SIX DIGITS).

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD  
 DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS.

**BEGIN TEST**

FIRST OPERAND	SECOND OPERAND	TRUE VALUE	COMPUTED VALUE	TEST RESULT
0	0	0	0	PASS
0	765.432	765.432	765.432	PASS
876.543	123.453	999.996	999.996	PASS
8.11111E-2	1.11111E-2	9.22222E-2	9.22222E-2	PASS
-57.9999	-0.111111	-58.111	-58.111	PASS
1111111	1111111	2222222	2222222	PASS
6.54321	1.23456	7.77777	7.77777	PASS
6.54321E-10	1.23456E-10	7.77777E-10	7.77777E-10	PASS

-6.54321E-20	-1.23456E-20	-7.77777E-20	-7.77777E-20	PASS
6.54321E-30	1.23456E-30	7.77777E-30	7.77777E-30	PASS
6.54321E-38	1.23456E-38	7.77777E-38	7.77777E-38	PASS
1.23456E+10	6.54321E+10	7.77777E+10	7.77777E+10	PASS
-1.23456E+20	-6.54321E+20	-7.77777E+20	-7.77777E+20	PASS
1.23456E+30	6.54321E+30	7.77777E+30	7.77777E+30	PASS
1.23456E+37	6.54321E+37	7.77777E+37	7.77777E+37	PASS
4.99997E-2	4.99996E-2	9.99993E-2	9.99993E-2	PASS
9.99993E-2	4.00000E-7	9.99997E-2	9.99997E-2	PASS
9.99993E-2	4.00000E-8	9.99993E-2	9.99993E-2	PASS
9.99993E-2	4.00000E-9	9.99993E-2	9.99993E-2	PASS
9.99993E-2	4.00000E-10	9.99993E-2	9.99993E-2	PASS
9.99993E-2	4.00000E-11	9.99993E-2	9.99993E-2	PASS
-1020304	4	-1020300	-1020300	PASS
1020304	-304	1020000	1020000	PASS
-1020304	20304	-1000000	-1000000	PASS
-1020304	1020304	0	0	PASS
-0.10203	4.00000E-7	-0.10203	-0.10203	PASS
0.10203	-3.04000E-5	0.102	0.102	PASS
-0.10203	2.03040E-3	-0.1	-0.1	PASS
-0.10203	0.10203	0	0	PASS
-0.10203	0.10203	-1.00000E-7	-1.00583E-7	PASS
-0.10203	0.10203	1.00000E-7	9.87202E-8	PASS

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 39

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 40: ACCURACY OF SUBTRACTION"
20 PRINT "      ANSI STANDARD 7.2, 7.4, 7.6"
30 PRINT
40 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'!"
50 PRINT "      WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
60 PRINT "      THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE"
70 PRINT "      VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
80 PRINT
100 PRINT "SECTION 40.1 ACCURACY OF SUBTRACTION."
110 PRINT
120 PRINT "THIS SECTION TESTS THE ACCURACY OF SUBTRACTION. THE COMPUTED"
130 PRINT "RESULT IS COMPARED WITH A RANGE ESTABLISHED BY PERTURBING"
140 PRINT "EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT PASSES IF"
150 PRINT "IT FALLS WITHIN THE EXTREME VALUES GENERATED BY THIS"
160 PRINT "PERTURBATION (ACCURATE TO SIX DIGITS)."
220 PRINT
230 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
240 PRINT "DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS."
250 PRINT
260 PRINT "          BEGIN TEST"
270 PRINT

```

```
280 PRINT "FIRST","SECOND","TRUE","COMPUTED","TEST"
290 PRINT "OPERAND","OPERAND","VALUE","VALUE","RESULT"
300 PRINT
310 LET F=0
320 READ A1,A2,T,L,H
330 IF A1=999 THEN 480
335 LET M$="PASS"
340 LET C=A1-A2
350 IF C<L THEN 380
360 IF C>H THEN 380
370 GOTO 400
380 LET M$="FAIL"
390 LET F=F+1
400 PRINT A1,A2,T,C,M$
410 GOTO 320
480 PRINT
490 IF F=0 THEN 520
500 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
510 GOTO 530
520 PRINT "*** INFORMATIVE TEST PASSED ***"
530 PRINT
540 PRINT " END TEST"
550 PRINT
560 PRINT "END PROGRAM 40"
570 DATA 0,0,0,0,0
2000 DATA 0,-765.432,765.432,765.431,765.433
3000 DATA 0.876543000E+03,-0.123453000E+03, 0.999996000E+03
3010 DATA 0.999993499E+03, 0.999998500E+03
3020 DATA 0.811111000E-01,-0.111111000E-01, 0.922222000E-01
3030 DATA 0.922219499E-01, 0.922224500E-01
3040 DATA -0.579999000E+02, 0.111111000E+00,-0.581110110E+02
3050 DATA -0.581111621E+02,-0.581108600E+02
3060 DATA 0.111111100E+07,-0.111111100E+07, 0.222222200E+07
3070 DATA 0.222219700E+07, 0.222224700E+07
3080 DATA 0.654321000E+01,-0.123456000E+01, 0.777777000E+01
3090 DATA 0.777774499E+01, 0.777779500E+01
3100 DATA 0.654321000E-09,-0.123456000E-09, 0.777777000E-09
3110 DATA 0.777774499E-09, 0.777779500E-09
3120 DATA -0.654321000E-19, 0.123456000E-19,-0.777777000E-19
3130 DATA -0.777779500E-19,-0.777774499E-19
3140 DATA 0.654321000E-29,-0.123456000E-29, 0.777777000E-29
3150 DATA 0.777774500E-29, 0.777779500E-29
3160 DATA 0.654321000E-37,-0.123456000E-37, 0.777777000E-37
3170 DATA 0.777774500E-37, 0.777779500E-37
3180 DATA 0.123456000E+11,-0.654321000E+11, 0.777777000E+11
3190 DATA 0.777774500E+11, 0.777779500E+11
3200 DATA -0.123456000E+21, 0.654321000E+21,-0.777777000E+21
3210 DATA -0.777779500E+21,-0.777774500E+21
3220 DATA 0.123456000E+31,-0.654321000E+31, 0.777777000E+31
3230 DATA 0.777774499E+31, 0.777779500E+31
3240 DATA 0.123456000E+38,-0.654321000E+38, 0.777777000E+38
3250 DATA 0.777774499E+38, 0.777779500E+38
3260 DATA 0.499997000E-01,-0.499996000E-01, 0.999993000E-01
3270 DATA 0.999990499E-01, 0.999995500E-01
3280 DATA 0.999993000E-01,-0.400000000E-06, 0.999997000E-01
```

```

3290 DATA 0.999995499E-01, 0.999998501E-01
3300 DATA 0.999993000E-01,-0.400000000E-07, 0.999993400E-01
3310 DATA 0.999991899E-01, 0.999994901E-01
3320 DATA 0.999993000E-01,-0.400000000E-08, 0.999993040E-01
3330 DATA 0.999991539E-01, 0.999994541E-01
3340 DATA 0.999993000E-01,-0.400000000E-09, 0.999993004E-01
3350 DATA 0.999991503E-01, 0.999994505E-01
3360 DATA 0.999993000E-01,-0.400000000E-10, 0.999993000E-01
3370 DATA 0.999991500E-01, 0.999994501E-01
3380 DATA -0.102030400E+07,-0.400000000E+01,-0.102030000E+07
3390 DATA -0.102031501E+07,-0.102028499E+07
3400 DATA 0.102030400E+07, 0.304000000E+03, 0.102000000E+07
3410 DATA 0.101998499E+07, 0.102001501E+07
3420 DATA -0.102030400E+07,-0.203040000E+05,-0.100000000E+07
3430 DATA -0.100001510E+07,-0.999989400E+06
3440 DATA -0.102030400E+07,-0.102030400E+07, 0.000000000E+00
3450 DATA -0.200000500E+02, 0.200000500E+02
3460 DATA -0.102030400E+00,-0.400000000E-06,-0.102030000E+00
3470 DATA -0.102031501E+00,-0.102028499E+00
3480 DATA 0.102030400E+00, 0.304000000E-04, 0.102000000E+00
3490 DATA 0.101998499E+00, 0.102001501E+00
3500 DATA -0.102030400E+00,-0.203040000E-02,-0.100000000E+00
3510 DATA -0.100001510E+00,-0.999989400E-01
3520 DATA -0.102030400E+00,-0.102030400E+00, 0.000000000E+00
3530 DATA -0.200000500E-05, 0.200000500E-05
3540 DATA -0.102030400E+00,-0.102030300E+00,-0.100000000E-06
3550 DATA -0.210000501E-05, 0.190000500E-05
3560 DATA -0.102030400E+00,-0.102030500E+00, 0.100000000E-06
3570 DATA -0.190000500E-05, 0.210000501E-05
5000 DATA 999,0,0,0,0
5010 END

```

---

PROGRAM FILE 40: ACCURACY OF SUBTRACTION  
ANSI STANDARD 7.2, 7.4, 7.6

\*\*\* NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'  
WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW  
THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE  
VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

SECTION 40.1 ACCURACY OF SUBTRACTION.

THIS SECTION TESTS THE ACCURACY OF SUBTRACTION. THE COMPUTED  
RESULT IS COMPARED WITH A RANGE ESTABLISHED BY PERTURBING  
EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT PASSES IF  
IT FALLS WITHIN THE EXTREME VALUES GENERATED BY THIS  
PERTURBATION (ACCURATE TO SIX DIGITS).

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD  
DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS.

BEGIN TEST

FIRST OPERAND	SECOND OPERAND	TRUE VALUE	COMPUTED VALUE	TEST RESULT
0	0	0	0	PASS
0	-765.432	765.432	765.432	PASS
876.543	-123.453	999.996	999.996	PASS
8.11111E-2	-1.11111E-2	9.22222E-2	9.22222E-2	PASS
-57.9999	0.111111	-58.111	-58.111	PASS
1111111	-1111111	2222222	2222222	PASS
6.54321	-1.23456	7.77777	7.77777	PASS
6.54321E-10	-1.23456E-10	7.77777E-10	7.77777E-10	PASS
-6.54321E-20	1.23456E-20	-7.77777E-20	-7.77777E-20	PASS
6.54321E-30	-1.23456E-30	7.77777E-30	7.77777E-30	PASS
6.54321E-38	-1.23456E-38	7.77777E-38	7.77777E-38	PASS
1.23456E+10	-6.54321E+10	7.77777E+10	7.77777E+10	PASS
-1.23456E+20	6.54321E+20	-7.77777E+20	-7.77777E+20	PASS
1.23456E+30	-6.54321E+30	7.77777E+30	7.77777E+30	PASS
1.23456E+37	-6.54321E+37	7.77777E+37	7.77777E+37	PASS
4.99997E-2	-4.99996E-2	9.99993E-2	9.99993E-2	PASS
9.99993E-2	-4.00000E-7	9.99997E-2	9.99997E-2	PASS
9.99993E-2	-4.00000E-8	9.99993E-2	9.99993E-2	PASS
9.99993E-2	-4.00000E-9	9.99993E-2	9.99993E-2	PASS
9.99993E-2	-4.00000E-10	9.99993E-2	9.99993E-2	PASS
9.99993E-2	-4.00000E-11	9.99993E-2	9.99993E-2	PASS
-1020304	-4	-1020300	-1020300	PASS
1020304	304	1020000	1020000	PASS
-1020304	-20304	-1000000	-1000000	PASS
-1020304	-1020304	0	0	PASS
-0.10203	-4.00000E-7	-0.10203	-0.10203	PASS
0.10203	3.04000E-5	0.102	0.102	PASS
-0.10203	-2.03040E-3	-0.1	-0.1	PASS
-0.10203	-0.10203	0	0	PASS
-0.10203	-0.10203	-1.00000E-7	-1.00583E-7	PASS
-0.10203	-0.10203	1.00000E-7	9.87202E-8	PASS

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 40

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 41: ACCURACY OF MULTIPLICATION"
20 PRINT "      ANSI STANDARD 7.2, 7.4, 7.6"
30 PRINT
40 PRINT "*** NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'!"
50 PRINT "      WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
60 PRINT "      THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE"
70 PRINT "      VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
80 PRINT
100 PRINT "SECTION 41.1 ACCURACY OF MULTIPLICATION."
110 PRINT

```

```

120 PRINT "THIS SECTION TESTS THE ACCURACY OF MULTIPLICATION. THE "
130 PRINT "COMPUTED RESULT IS COMPARED WITH A RANGE ESTABLISHED BY"
140 PRINT "PERTURBING EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT"
150 PRINT "PASSES IF IT FALLS WITHIN THE EXTREME VALUES GENERATED BY"
160 PRINT "THIS PERTURBATION (ACCURATE TO SIX DIGITS)."
220 PRINT
230 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
240 PRINT "DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS."
250 PRINT
260 PRINT "           BEGIN TEST"
270 PRINT
280 PRINT "FIRST", "SECOND", "TRUE", "COMPUTED", "TEST"
290 PRINT "OPERAND", "OPERAND", "VALUE", "VALUE", "RESULT"
300 PRINT
310 LET F=0
320 READ A1,A2,T,L,H
330 IF A1=999 THEN 480
335 LET M$="PASS"
340 LET C=A1*A2
350 IF C<L THEN 380
360 IF C>H THEN 380
370 GOTO 400
380 LET M$="FAIL"
390 LET F=F+1
400 PRINT A1,A2,T,C,M$
410 GOTO 320
480 PRINT
490 IF F=0 THEN 520
500 PRINT "**** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ****"
510 GOTO 530
520 PRINT "**** INFORMATIVE TEST PASSED ****"
530 PRINT
540 PRINT "           END TEST"
550 PRINT
560 PRINT "END PROGRAM 41"
2000 DATA 0,0,0,0,0
2010 DATA -3.12345,0,0,0,0
2020 DATA -0.6.68589E37,0,0,0
3000 DATA 0.390625000E+01, 0.524288000E+01, 0.204800000E+02
3010 DATA 0.204798585E+02, 0.204801415E+02
3020 DATA 0.646469000E-10, 0.336466000E+11, 0.217514839E+01
3030 DATA 0.217513355E+01, 0.217516322E+01
3040 DATA 0.646469000E-37, 0.336466000E+38, 0.217514839E+01
3050 DATA 0.217513355E+01, 0.217516322E+01
3060 DATA 0.465728000E+00, 0.447578000E+38, 0.208449607E+38
3070 DATA 0.208448193E+38, 0.208451021E+38
3080 DATA 0.348799000E+01, 0.658589000E+37, 0.229715185E+38
3090 DATA 0.229713677E+38, 0.229716692E+38
3100 DATA 0.448588000E+11, 0.888898000E+27, 0.398748976E+38
3110 DATA 0.398747138E+38, 0.398750814E+38
3120 DATA 0.995855000E+19, 0.944647000E+19, 0.940731438E+38
3130 DATA 0.940728997E+38, 0.940733879E+38
3140 DATA 0.995855000E-19, 0.944647000E-18, 0.940731438E-37
3150 DATA 0.940728997E-37, 0.940733879E-37
3160 DATA 0.995855000E-29, 0.944647000E-08, 0.940731438E-37

```

```

3170 DATA 0.940728997E-37, 0.940733879E-37
3180 DATA 0.995855000E-37, 0.944647000E-00, 0.940731438E-37
3190 DATA 0.940728997E-37, 0.940733879E-37
3200 DATA 0.999000000E+00, 0.999000000E+00, 0.998001000E+00
3210 DATA 0.997998502E+00, 0.998003499E+00
3220 DATA 0.774758000E+06, 0.758675000E+06, 0.587789526E+12
3230 DATA 0.587787492E+12, 0.587791560E+12
3240 DATA 0.365341000E+05,-0.338383000E-02,-0.123625184E+03
3250 DATA -0.123626388E+03,-0.123623979E+03
3260 DATA -0.989878000E-01, 0.985342000E-01,-0.975368368E-02
3270 DATA -0.975370844E-02,-0.975365893E-02
3280 DATA -0.228373000E+00,-0.955664000E+03, 0.218247855E+03
3290 DATA 0.218246170E+03, 0.218249539E+03
3300 DATA 0.390625000E+00,-0.248864000E+00,-0.972125000E-01
3310 DATA -0.972131895E-01,-0.972118105E-01
3320 DATA -0.390625000E-29, 0.248864000E+33,-0.972125000E+03
3330 DATA -0.972131895E+03,-0.972118105E+03
5000 DATA 999,0,0,0,0
5010 END

```

---

PROGRAM FILE 41: ACCURACY OF MULTIPLICATION  
ANSI STANDARD 7.2, 7.4, 7.6

\*\*\* NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'  
WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW  
THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE  
VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

SECTION 41.1 ACCURACY OF MULTIPLICATION.

THIS SECTION TESTS THE ACCURACY OF MULTIPLICATION. THE COMPUTED RESULT IS COMPARED WITH A RANGE ESTABLISHED BY PERTURBING EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT PASSES IF IT FALLS WITHIN THE EXTREME VALUES GENERATED BY THIS PERTURBATION (ACCURATE TO SIX DIGITS).

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS.

BEGIN TEST

FIRST OPERAND	SECOND OPERAND	TRUE VALUE	COMPUTED VALUE	TEST RESULT
0	0	0	0	PASS
-3.12345	0	0	0	PASS
0	6.68589E+37	0	0	PASS
3.90625	5.24288	20.48	20.48	PASS
6.46469E-11	3.36466E+10	2.17515	2.17515	PASS
6.46469E-38	3.36466E+37	2.17515	2.17515	PASS
0.465728	4.47578E+37	2.08450E+37	2.08450E+37	PASS
3.48799	6.58589E+36	2.29715E+37	2.29715E+37	PASS
4.48588E+10	8.88898E+26	3.98749E+37	3.98749E+37	PASS

				PASS
9.95855E+18	9.44647E+18	9.40731E+37	9.40731E+37	PASS
9.95855E-20	9.44647E-19	9.40731E-38	9.40731E-38	PASS
9.95855E-30	9.44647E-9	9.40731E-38	9.40731E-38	PASS
9.95855E-38	0.944647	9.40731E-38	9.40731E-38	PASS
0.999	0.999	0.998001	0.998001	PASS
774758	758675	5.87790E+11	5.87790E+11	PASS
36534.1	-3.38383E-3	-123.625	-123.625	PASS
-9.89878E-2	9.85342E-2	-9.75368E-3	-9.75368E-3	PASS
-0.228373	-955.664	218.248	218.248	PASS
0.390625	-0.248864	-9.72125E-2	-9.72125E-2	PASS
-3.90625E-30	2.48864E+32	-972.125	-972.125	PASS

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 41

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 42: ACCURACY OF DIVISION"
20 PRINT "      ANSI STANDARD 7.2, 7.4, 7.6"
30 PRINT
40 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'"
50 PRINT " WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
60 PRINT " THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE"
70 PRINT " VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
80 PRINT
100 PRINT "SECTION 42.1 ACCURACY OF DIVISION."
110 PRINT
120 PRINT "THIS SECTION TESTS THE ACCURACY OF DIVISION. THE "
130 PRINT "COMPUTED RESULT IS COMPARED WITH A RANGE ESTABLISHED BY"
140 PRINT "PERTURBING EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT"
150 PRINT "PASSES IF IT FALLS WITHIN THE EXTREME VALUES GENERATED BY"
160 PRINT "THIS PERTURBATION (ACCURATE TO SIX DIGITS)."
220 PRINT
230 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
240 PRINT "DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS."
250 PRINT
260 PRINT "          BEGIN TEST"
270 PRINT
280 PRINT "FIRST","SECOND","TRUE","COMPUTED","TEST"
290 PRINT "OPERAND","OPERAND","VALUE","VALUE","RESULT"
300 PRINT
310 LET F=0
320 READ A1,A2,T,L,H
330 IF A1=999 THEN 480
335 LET M$="PASS"
340 LET C=A1/A2
350 IF C<L THEN 380
360 IF C>H THEN 380
370 GOTO 400
380 LET M$="FAIL"
```

```

390 LET F=F+1
400 PRINT A1,A2,T,C,M$
410 GOTO 320
480 PRINT
490 IF F=0 THEN 520
500 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
510 GOTO 530
520 PRINT "*** INFORMATIVE TEST PASSED ***"
530 PRINT
540 PRINT "           END TEST"
550 PRINT
560 PRINT "END PROGRAM 42"
2000 DATA 0,-74.7589,0,0,0
3000 DATA 0.972125000E+13, 0.248864000E-02, 0.390625000E+16
3010 DATA 0.390622528E+16, 0.390627472E+16
3020 DATA -0.972125000E-04,-0.248864000E+14, 0.390625000E-17
3030 DATA 0.390622528E-17, 0.390627472E-17
3040 DATA 0.936467000E-01, 0.954647000E-01, 0.980956312E+00
3050 DATA 0.980953736E+00, 0.980958887E+00
3060 DATA 0.376465000E+38, 0.895857000E+38, 0.420228898E+00
3070 DATA 0.420226812E+00, 0.420230984E+00
3080 DATA 0.958764000E+18, 0.558589000E+16, 0.171640329E+03
3090 DATA 0.171639343E+03, 0.171641316E+03
3100 DATA -0.457858000E-16,-0.879796000E-13, 0.520413823E-03
3110 DATA 0.520411595E-03, 0.520416052E-03
3120 DATA 0.237847000E-37, 0.117265000E-37, 0.202828636E+01
3130 DATA 0.202825553E+01, 0.202831719E+01
3140 DATA 0.684776000E+38,-0.697536000E+00,-0.981707037E+38
3150 DATA -0.981710379E+38,-0.981703696E+38
3160 DATA -0.346756000E+21, 0.222736000E-16,-0.155680267E+38
3170 DATA -0.155681916E+38,-0.155678619E+38
3200 DATA 0.987645000E+00, 0.987897000E-37, 0.999744913E+37
3210 DATA 0.999742388E+37, 0.999747437E+37
3220 DATA -0.447563000E-36, 0.668575000E+01,-0.669428262E-37
3230 DATA -0.669431259E-37,-0.669425264E-37
3240 DATA 0.857653000E-19,-0.889763000E+18,-0.963911738E-37
3250 DATA -0.963914446E-37,-0.963909031E-37
3260 DATA -0.557645000E+01,-0.786453000E+38, 0.709063352E-37
3270 DATA 0.709060678E-37, 0.709066025E-37
5000 DATA 999,0,0,0,0
5010 END

```

---

PROGRAM FILE 42: ACCURACY OF DIVISION  
ANSI STANDARD 7.2, 7.4, 7.6

\*\*\* NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'  
WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW  
THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE  
VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

SECTION 42.1 ACCURACY OF DIVISION.

THIS SECTION TESTS THE ACCURACY OF DIVISION. THE COMPUTED RESULT IS COMPARED WITH A RANGE ESTABLISHED BY PERTURBING EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT PASSES IF IT FALLS WITHIN THE EXTREME VALUES GENERATED BY THIS PERTURBATION (ACCURATE TO SIX DIGITS).

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS.

BEGIN TEST

FIRST OPERAND	SECOND OPERAND	TRUE VALUE	COMPUTED VALUE	TEST RESULT
0	-74.7589	0	0	PASS
9.72125E+12	2.48864E-3	3.90625E+15	3.90625E+15	PASS
-9.72125E-5	-2.48864E+13	3.90625E-18	3.90625E-18	PASS
9.36467E-2	9.54647E-2	0.980956	0.980956	PASS
3.76465E+37	8.95857E+37	0.420229	0.420229	PASS
9.58764E+17	5.58589E+15	171.64	171.64	PASS
-4.57858E-17	-8.79796E-14	5.20414E-4	5.20414E-4	PASS
2.37847E-38	1.17265E-38	2.02829	2.02829	PASS
6.84776E+37	-0.697536	-9.81707E+37	-9.81707E+37	PASS
-3.46756E+20	2.22736E-17	-1.55680E+37	-1.55680E+37	PASS
0.987645	9.87897E-38	9.99745E+36	9.99745E+36	PASS
-4.47563E-37	6.68575	-6.69428E-38	-6.69428E-38	PASS
8.57653E-20	-8.89763E+17	-9.63912E-38	-9.63912E-38	PASS
-5.57645	-7.86453E+37	7.09063E-38	7.09063E-38	PASS

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 42

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 43: ACCURACY OF INVOLUTION"
20 PRINT "      ANSI STANDARD 7.2, 7.4, 7.6"
30 PRINT
40 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'!"
50 PRINT "      WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
60 PRINT "      THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE"
70 PRINT "      VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
80 PRINT
100 PRINT "SECTION 43.1 ACCURACY OF INVOLUTION."
110 PRINT
120 PRINT "THIS SECTION TESTS THE ACCURACY OF INVOLUTION. THE "
130 PRINT "COMPUTED RESULT IS COMPARED WITH A RANGE ESTABLISHED BY"
140 PRINT "PERTURBING EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT"
150 PRINT "PASSES IF IT FALLS WITHIN THE EXTREME VALUES GENERATED BY"
160 PRINT "THIS PERTURBATION (ACCURATE TO SIX DIGITS)."
220 PRINT

```

```

230 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
240 PRINT "DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS."
250 PRINT
260 PRINT "           BEGIN TEST"
270 PRINT
280 PRINT "FIRST","SECOND","TRUE","COMPUTED","TEST"
290 PRINT "OPERAND","OPERAND","VALUE","VALUE","RESULT"
300 PRINT
310 LET F=0
320 READ A1,A2,T,L,H
330 IF A1=999 THEN 480
335 LET M$="PASS"
340 LET C=A1^A2
350 IF C<L THEN 380
360 IF C>H THEN 380
370 GOTO 400
380 LET M$="FAIL"
390 LET F=F+1
400 PRINT A1,A2,T,C,M$
410 GOTO 320
480 PRINT
490 IF F=0 THEN 520
500 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
510 GOTO 530
520 PRINT "*** INFORMATIVE TEST PASSED ***"
530 PRINT
540 PRINT "           END TEST"
550 PRINT
560 PRINT "END PROGRAM 43"
1990 REM TEST X^0=1
2000 DATA -3.46478E33,0,1,1,1
2010 DATA -2.36339E-36,0,1,1,1
2020 DATA 0,0,1,1,1
2030 DATA 3.98475E-37,0,1,1,1
2040 DATA 9.88373E37,0,1,1,1
2045 REM TEST 0^X=0, FOR X>0
2050 DATA 0,3.47474E-33,0,0,0
2060 DATA 0,8.98675E33,0,0,0
2990 REM TEST A^X, A<-1
3000 DATA -0.376354000E+01,-0.440000000E+02, 0.471793993E-25
3010 DATA 0.471738338E-25, 0.471849655E-25
3020 DATA -0.687465000E+01,-0.110000000E+02,-0.616940788E-09
3030 DATA -0.616951160E-09,-0.616930416E-09
3040 DATA -0.736443000E+04,-0.100000000E+01,-0.135787834E-03
3050 DATA -0.135788519E-03,-0.135787149E-03
3060 DATA -0.333874000E+02, 0.100000000E+01,-0.333874000E+02
3070 DATA -0.333875500E+02,-0.333872500E+02
3080 DATA -0.984343000E+03, 0.700000000E+01,-0.895416722E+21
3090 DATA -0.895423590E+21,-0.895409854E+21
3100 DATA -0.336353000E+01, 0.120000000E+02, 0.209673559E+07
3110 DATA 0.209665578E+07, 0.209681540E+07
3115 REM TEST -1^X
3120 DATA -0.100000000E+01,-0.330000000E+02,-0.100000000E+01
3130 DATA -0.100033506E+01,-0.999669556E+00
3140 DATA -0.100000000E+01,-0.600000000E+01, 0.100000000E+01

```

3150 DATA 0.999939502E+00, 0.100006501E+01  
3160 DATA -0.100000000E+01, -0.100000000E+01, -0.100000000E+01  
3170 DATA -0.100001501E+01, -0.999989500E+00  
3180 DATA -0.100000000E+01, 0.100000000E+01, -0.100000000E+01  
3190 DATA -0.100001500E+01, -0.999989500E+00  
3200 DATA -0.100000000E+01, 0.800000000E+01, 0.100000000E+01  
3210 DATA 0.999919502E+00, 0.100008501E+01  
3220 DATA -0.100000000E+01, 0.900000000E+01, -0.100000000E+01  
3230 DATA -0.100009501E+01, -0.999909503E+00  
3235 REM TEST A^X, -1<A<0  
3240 DATA -0.474653000E+00, -0.220000000E+02, 0.131741601E+08  
3250 DATA 0.131734994E+08, 0.131748208E+08  
3260 DATA -0.225242000E-02, 0.300000000E+01, -0.114274183E-07  
3270 DATA -0.114276205E-07, -0.114272160E-07  
3280 DATA -0.534278000E-04, 0.600000000E+01, 0.232596807E-25  
3290 DATA 0.232593695E-25, 0.232599920E-25  
3300 DATA -0.272625000E+00, 0.900000000E+01, -0.831938279E-05  
3310 DATA -0.831966244E-05, -0.831910314E-05  
3315 REM TEST A^X, 0<A<1  
3320 DATA 0.363544000E+00, -0.363353000E+02, 0.927457281E+16  
3330 DATA 0.927270258E+16, 0.927644343E+16  
3340 DATA 0.463542000E-02, -0.100000000E+01, 0.215730182E+03  
3350 DATA 0.215717623E+03, 0.215742742E+03  
3380 DATA 0.298746000E-04, -0.263545000E-02, 0.102783787E+01  
3390 DATA 0.102783275E+01, 0.102784299E+01  
3400 DATA 0.226399000E+00, 0.847574000E+00, 0.283928118E+00  
3410 DATA 0.283926132E+00, 0.283930103E+00  
3420 DATA 0.998476000E-01, 0.100000000E+01, 0.998476000E-01  
3430 DATA 0.998451494E-01, 0.998500507E-01  
3440 DATA 0.974653000E-01, 0.143654000E+02, 0.298144041E-14  
3450 DATA 0.298069739E-14, 0.298218360E-14  
3455 REM TEST 1^X  
3460 DATA 0.100000000E+01, -0.222222000E+02, 0.100000000E+01  
3470 DATA 0.999777302E+00, 0.100022725E+01  
3480 DATA 0.100000000E+01, -0.464533000E-01, 0.100000000E+01  
3490 DATA 0.999999035E+00, 0.100000547E+01  
3500 DATA 0.100000000E+01, 0.993764000E-01, 0.100000000E+01  
3510 DATA 0.999998506E+00, 0.100000600E+01  
3520 DATA 0.100000000E+01, 0.187564000E+02, 0.100000000E+01  
3530 DATA 0.999811951E+00, 0.100019259E+01  
3535 REM TEST A^X, A>1  
3540 DATA 0.847447000E+03, -0.985743000E+01, 0.136878595E-28  
3550 DATA 0.136867274E-28, 0.136889917E-28  
3560 DATA 0.228398000E+01, -0.937362000E+00, 0.461079137E+00  
3570 DATA 0.461076363E+00, 0.461081910E+00  
3580 DATA 0.933847000E+02, 0.928273000E-01, 0.152368565E+01  
3590 DATA 0.152367980E+01, 0.152369149E+01  
3600 DATA 0.837363000E+09, 0.298383000E+01, 0.421170346E+27  
3610 DATA 0.421081821E+27, 0.421258889E+27  
3615 REM TEST A^X, WHERE ABS(A) IS CLOSE TO 1, X IS LARGE  
3620 DATA 0.999988000E+00, 0.958576000E+06, 0.100999406E-04  
3630 DATA 0.387260445E-05, 0.263410738E-04  
3640 DATA -0.999982000E+00, 0.947563000E+06, -0.391334786E-07  
3650 DATA -0.100943693E-06, -0.151711334E-07  
3660 DATA -0.999937000E+00, 0.976352000E+06, 0.193031627E-26

```

3670 DATA 0.727071227E-27, 0.512483260E-26
3680 DATA 0.100002000E+01, 0.209746500E+07, 0.165259840E+19
3690 DATA 0.128553554E+10, 0.212444587E+28
3700 DATA -0.100005000E+01, 0.353533000E+06,-0.474984719E+08
3710 DATA -0.162918665E+10,-0.138475470E+07
3720 DATA -0.100005000E+01, 0.958572000E+06, 0.652538506E+21
3730 DATA 0.448505410E+17, 0.949298339E+25
3735 REM TEST A^X, WHERE A^X CLOSE TO LIMITS OF 1E38, 1E-38
3740 DATA 0.576352000E+00, 0.111000000E+03, 0.273115473E-26
3750 DATA 0.272911951E-26, 0.273319144E-26
3760 DATA 0.578746000E+00,-0.145432000E+03, 0.348212882E+35
3770 DATA 0.347934556E+35, 0.348491430E+35
3780 DATA 0.227364000E+01, 0.944756000E+02, 0.502917503E+34
3790 DATA 0.502666781E+34, 0.503168349E+34
3792 DATA 0.233333000E+01,-0.991234000E+02, 0.334927971E-36
3795 DATA 0.334752354E-36, 0.335103676E-36
3799 REM TEST A^X, WHERE A IS CLOSE TO LIMITS OF 1E38, 1E-38
3800 DATA 0.558574000E+38, 0.976453000E-03, 0.108857473E+01
3810 DATA 0.108856963E+01, 0.108857984E+01
3820 DATA 0.576363000E-29, 0.172728000E-01, 0.312575800E+00
3830 DATA 0.312573186E+00, 0.312578414E+00
3840 DATA 0.333737000E+34,-0.953423000E-01, 0.636504568E-03
3850 DATA 0.636498972E-03, 0.636510163E-03
3860 DATA 0.623147000E-37,-0.936218000E-02, 0.223009693E+01
3870 DATA 0.223004499E+01, 0.223014888E+01
5000 DATA 999,0,0,0,0
5010 END

```

---

PROGRAM FILE 43: ACCURACY OF INVOLUTION  
ANSI STANDARD 7.2, 7.4, 7.6

\*\*\* NOTE: THIS PROGRAM MAKES USE OF 'READ' AND 'DATA'  
WHICH HAVE NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW  
THESE FEATURES TO BE INCORRECTLY IMPLEMENTED, THEN THE  
VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL.

SECTION 43.1 ACCURACY OF INVOLUTION.

THIS SECTION TESTS THE ACCURACY OF INVOLUTION. THE  
COMPUTED RESULT IS COMPARED WITH A RANGE ESTABLISHED BY  
PERTURBING EACH OPERAND BY 1 IN ITS 6TH DIGIT. THE RESULT  
PASSES IF IT FALLS WITHIN THE EXTREME VALUES GENERATED BY  
THIS PERTURBATION (ACCURATE TO SIX DIGITS).

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD  
DOES NOT MANDATE ANY ACCURACY FOR NUMERIC EXPRESSIONS.

BEGIN TEST

FIRST OPERAND	SECOND OPERAND	TRUE VALUE	COMPUTED VALUE	TEST RESULT
------------------	-------------------	---------------	-------------------	----------------

-3.46478E+33	0	1	1	PASS
-2.36339E-36	0	1	1	PASS
0	0	1	1	PASS
3.98475E-37	0	1	1	PASS
9.88373E+37	0	1	1	PASS
0	3.47474E-33	0	0	PASS
0	8.98675E+33	0	0	PASS
-3.76354	-44	4.71794E-26	4.71794E-26	PASS
-6.87465	-11	-6.16941E-10	-6.16941E-10	PASS
-7364.43	-1	-1.35788E-4	-1.35788E-4	PASS
-33.3874	1	-33.3874	-33.3874	PASS
-984.343	7	-8.95417E+20	-8.95417E+20	PASS
-3.36353	12	2.09674E+6	2.09674E+6	PASS
-1	-33	-1	-1	PASS
-1	-6	1	1	PASS
-1	-1	-1	-1	PASS
-1	1	-1	-1	PASS
-1	8	1	1	PASS
-1	9	-1	-1	PASS
-0.474653	-22	13174160	1.31742E+7	PASS
-2.25242E-3	3	-1.14274E-8	-1.14274E-8	PASS
-5.34278E-5	6	2.32597E-26	2.32597E-26	PASS
-0.272625	9	-8.31938E-6	-8.31938E-6	PASS
0.363544	-36.3353	9.27457E+15	9.27458E+15	PASS
4.63542E-3	-1	215.73	215.73	PASS
2.98746E-5	-2.63545E-3	1.02784	1.02784	PASS
0.226399	0.847574	0.283928	0.283928	PASS
9.98476E-2	1	9.98476E-2	9.98476E-2	PASS
9.74653E-2	14.3654	2.98144E-15	2.98144E-15	PASS
1	-22.2222	1	1	PASS
1	-4.64533E-2	1	1	PASS
1	9.93764E-2	1	1	PASS
1	18.7564	1	1	PASS
847.447	-9.85743	1.36879E-29	1.36879E-29	PASS
2.28398	-0.937362	0.461079	0.461079	PASS
93.3847	9.28273E-2	1.52369	1.52369	PASS
8.37363E+8	2.98383	4.21170E+26	4.21170E+26	PASS
0.999988	958576	1.00999E-5	9.99099E-6	PASS
-0.999982	947563	-3.91335E-8	-3.90496E-8	PASS
-0.999937	976352	1.93032E-27	1.92079E-27	PASS
1.00002	2097465	1.65260E+18	1.63435E+18	PASS
-1.00005	353533	-47498471	-47279728	PASS
-1.00005	958572	6.52539E+20	6.44422E+20	PASS
0.576352	111	2.73115E-27	2.73115E-27	PASS
0.578746	-145.432	3.48213E+34	3.48213E+34	PASS
2.27364	94.4756	5.02917E+33	5.02918E+33	PASS
2.33333	-99.1234	3.34928E-37	3.34928E-37	PASS
5.58574E+37	9.76453E-4	1.08857	1.08857	PASS
5.76363E-30	1.72728E-2	0.312576	0.312576	PASS
3.33737E+33	-9.53423E-2	6.36505E-4	6.36505E-4	PASS
6.23147E-38	-9.36218E-3	2.2301	2.2301	PASS

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 43

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 44: ELEMENTARY USE OF THE FOR-STATEMENT."  
20 PRINT "      ANSI STANDARD 11.2, 11.4"  
30 PRINT  
40 PRINT "SECTION 44.1: ELEMENTARY USE OF THE FOR-STATEMENT."  
50 PRINT  
60 PRINT "THIS SECTION TESTS A SIMPLE FOR-NEXT CONSTRUCTION WITH"  
70 PRINT "A VARIETY OF VALUES FOR THE INITIAL-VALUE, LIMIT, AND"  
80 PRINT "INCREMENT. TO PASS THE TEST, THE CONTROL-VARIABLE MUST"  
90 PRINT "TAKE ON THE APPROPRIATE VALUES AND THE LOOP MUST BE EXECUTED"  
100 PRINT "THE CORRECT NUMBER OF TIMES."  
110 PRINT  
120 PRINT "                  BEGIN TEST."  
130 LET F=0  
140 PRINT  
150 LET A1=.0987789  
160 LET B1=.09878  
170 LET C1=3E-7  
180 LET T2=4  
190 GOSUB 3000  
200 LET A1=5  
210 LET B1=8  
220 LET C1=-1  
230 LET T2=0  
240 GOSUB 3000  
250 LET A1=8  
260 LET B1=5  
270 LET C1=1  
280 LET T2=0  
290 GOSUB 3000  
300 LET A1=8  
310 LET B1=-5  
320 LET C1=-3  
330 LET T2=5  
340 GOSUB 3000  
350 LET A1=-1.7  
360 LET B1=2.6  
370 LET C1=.27  
380 LET T2=16  
390 GOSUB 3000  
400 LET A1=11  
410 LET B1=-7  
420 LET C1=-3  
430 LET T2=7  
440 GOSUB 3000  
450 LET A1=2  
460 LET B1=2  
470 LET C1=20  
480 LET T2=1  
490 GOSUB 3000
```

```
500 LET A1=1.234E20
510 LET B1=-4.321E20
520 LET C1=-8.7E19
530 LET T2=7
540 GOSUB 3000
550 LET A1=0
560 LET B1=0
570 LET C1=-2
580 LET T2=1
590 GOSUB 3000
2000 PRINT
2010 IF F=0 THEN 2040
2020 PRINT "*** TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2050
2040 PRINT "*** TEST PASSED ***"
2050 PRINT
2060 PRINT " END TEST."
2070 PRINT
2080 PRINT "END PROGRAM 44"
2090 STOP
3000 REM SUBROUTINE TO TEST VARIOUS VALUES IN THE FOR-LOOP
3010 PRINT
3020 PRINT "LOOP IS: FOR I1=";A1;" TO ";B1;" STEP ";C1
3030 PRINT
3040 PRINT "CONTROL VARIABLE:"
3050 PRINT "SHOULD BE","ACTUAL","RESULT"
3060 LET A2=A1
3070 LET B2=B1
3080 LET C2=C1
3090 LET I2=A2
3100 REM S1 IS ERROR SWITCH
3110 LET S1=0
3120 REM T1 IS LOOP COUNT
3130 LET T1=0
3140 FOR I1=A1 TO B1 STEP C1
3150 LET T1=T1+1
3160 LET M$=" OK "
3170 IF I1=0 THEN 3220
3180 LET R=(I2-I1)/I1
3190 IF R<-1E-6 THEN 3230
3200 IF R>1E-6 THEN 3230
3210 GOTO 3250
3220 IF I2=0 THEN 3250
3230 LET M$="FAILED"
3240 LET S1=1
3250 PRINT I2,I1,M$
3260 LET I2=I2+C2
3270 NEXT I1
3280 PRINT "LOOP EXITED"
3290 LET M$=" OK "
3300 IF I1=0 THEN 3350
3310 LET R=(I2-I1)/I1
3320 IF R<-1E-6 THEN 3360
3330 IF R>1E-6 THEN 3360
3340 GOTO 3380
```

```

3350 IF I2=0 THEN 3380
3360 LET M$="FAILED"
3370 LET S1=1
3380 PRINT I2,I1,M$
3390 LET M$=" OK "
3400 IF T1=T2 THEN 3425
3410 LET M$="FAILED"
3420 LET S1=1
3425 PRINT
3430 PRINT "NO. TIMES THROUGH LOOP:"
3440 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"      RESULT: ";M$
3450 PRINT
3455 PRINT -----
3460 LET F=F+S1
3470 RETURN
3480 END

```

---

## PROGRAM FILE 44: ELEMENTARY USE OF THE FOR-STATEMENT.

ANSI STANDARD 11.2, 11.4

## SECTION 44.1: ELEMENTARY USE OF THE FOR-STATEMENT.

THIS SECTION TESTS A SIMPLE FOR-NEXT CONSTRUCTION WITH A VARIETY OF VALUES FOR THE INITIAL-VALUE, LIMIT, AND INCREMENT. TO PASS THE TEST, THE CONTROL-VARIABLE MUST TAKE ON THE APPROPRIATE VALUES AND THE LOOP MUST BE EXECUTED THE CORRECT NUMBER OF TIMES.

BEGIN TEST.

LOOP IS: FOR I1= 9.87789E-2 TO 9.87800E-2 STEP 3.00000E-7

## CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
9.87789E-2	9.87789E-2	OK
9.87792E-2	9.87792E-2	OK
9.87795E-2	9.87795E-2	OK
9.87798E-2	9.87798E-2	OK

LOOP EXITED

9.87801E-2	9.87801E-2	OK
------------	------------	----

## NO. TIMES THROUGH LOOP:

SHOULD BE: 4 ACTUAL: 4 RESULT: OK

LOOP IS: FOR I1= 5 TO 8 STEP -1

## CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
5	5	OK

LOOP EXITED

NO. TIMES THROUGH LOOP:  
SHOULD BE: 0 ACTUAL: 0 RESULT: OK

---

LOOP IS: FOR I1= 8 TO 5 STEP 1

CONTROL VARIABLE:  
SHOULD BE ACTUAL RESULT  
LOOP EXITED  
8 8 OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 0 ACTUAL: 0 RESULT: OK

---

LOOP IS: FOR I1= 8 TO -5 STEP -3

CONTROL VARIABLE:  
SHOULD BE ACTUAL RESULT  
8 8 OK  
5 5 OK  
2 2 OK  
-1 -1 OK  
-4 -4 OK  
LOOP EXITED  
-7 -7 OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 5 ACTUAL: 5 RESULT: OK

---

LOOP IS: FOR I1=-1.7 TO 2.6 STEP 0.27

CONTROL VARIABLE:  
SHOULD BE ACTUAL RESULT  
-1.7 -1.7 OK  
-1.43 -1.43 OK  
-1.16 -1.16 OK  
-0.89 -0.89 OK  
-0.62 -0.62 OK  
-0.35 -0.35 OK  
-0.08 -0.08 OK  
0.19 0.19 OK  
0.46 0.46 OK  
0.73 0.73 OK  
1 1 OK  
1.27 1.27 OK  
1.54 1.54 OK  
1.81 1.81 OK  
2.08 2.08 OK

2.35            2.35            OK  
LOOP EXITED  
2.62            2.62            OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 16        ACTUAL: 16        RESULT: OK

---

LOOP IS: FOR I1= 11 TO -7 STEP -3

CONTROL VARIABLE:  
SHOULD BE     ACTUAL     RESULT  
11              11            OK  
8                8            OK  
5                5            OK  
2                2            OK  
-1              -1           OK  
-4              -4           OK  
-7              -7           OK  
LOOP EXITED  
-10             -10          OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 7        ACTUAL: 7        RESULT: OK

---

LOOP IS: FOR I1= 2 TO 2 STEP 20

CONTROL VARIABLE:  
SHOULD BE     ACTUAL     RESULT  
2               2            OK  
LOOP EXITED  
22              22           OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 1        ACTUAL: 1        RESULT: OK

---

LOOP IS: FOR I1= 1.23400E+20 TO -4.32100E+20 STEP -8.70000E+19

CONTROL VARIABLE:  
SHOULD BE     ACTUAL     RESULT  
1.23400E+20    1.23400E+20    OK  
3.64000E+19    3.64000E+19    OK  
-5.06000E+19   -5.06000E+19   OK  
-1.37600E+20   -1.37600E+20   OK  
-2.24600E+20   -2.24600E+20   OK  
-3.11600E+20   -3.11600E+20   OK  
-3.98600E+20   -3.98600E+20   OK  
LOOP EXITED  
-4.85600E+20   -4.85600E+20   OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 7 ACTUAL: 7 RESULT: OK

---

LOOP IS: FOR I1= 0 TO 0 STEP -2

CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
0	0	OK
LOOP EXITED		
-2	-2	OK

NO. TIMES THROUGH LOOP:

SHOULD BE: 1 ACTUAL: 1 RESULT: OK

---

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 44

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 45: ALTERING THE CONTROL-VARIABLE WITHIN"
20 PRINT "      A FOR-BLOCK."
30 PRINT "      ANSI STANDARD 11.2, 11.4"
40 PRINT
50 PRINT "SECTION 45.1: ALTERING THE CONTROL-VARIABLE WITHIN"
60 PRINT "      A FOR-BLOCK."
70 PRINT
80 PRINT "THIS SECTION TESTS THAT THE VALUE OF THE CONTROL-VARIABLE"
90 PRINT "MAY BE ALTERED BY STATEMENTS WITHIN THE FOR-BLOCK, AS"
100 PRINT "WELL AS UNDER THE DIRECT CONTROL OF THE FOR-STATEMENT."
110 PRINT
120 PRINT "      BEGIN TEST."
130 LET A1=3
140 LET B1=21
150 LET C1=3
160 LET T2=4
170 PRINT
180 PRINT "CONTROL VARIABLE:"
190 PRINT "SHOULD BE","ACTUAL","RESULT"
200 LET I2=A1
210 REM S1 IS ERROR SWITCH
220 LET S1=0
230 REM T1 IS LOOP COUNT
240 LET T1=0
250 FOR I1=A1 TO B1 STEP C1
260   LET T1=T1+1
270   LET M$="  OK  "
```

```
280 IF I1=I2 THEN 310
290 LET M$="FAILED"
300 LET S1=1
310 PRINT I2,I1,M$
320 IF I2 <> 9 THEN 360
330 LET I1=17
340 LET I2=17
350 PRINT "CONTROL VARIABLE ALTERED."
360 REM RE-JOIN PATHS
370 LET I2=I2+C1
380 NEXT I1
390 PRINT "LOOP EXITED"
400 LET M$=" OK "
410 IF I1=I2 THEN 440
420 LET M$="FAILED"
430 LET S1=1
440 PRINT I2,I1,M$
450 LET M$=" OK "
460 IF T1=T2 THEN 490
470 LET M$="FAILED"
480 LET S1=1
490 PRINT
500 PRINT "NO. TIMES THROUGH LOOP:"
510 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"      RESULT: ";M$
520 PRINT
530 IF S1=0 THEN 560
540 PRINT "*** TEST FAILED ***"
550 GOTO 570
560 PRINT "*** TEST PASSED ***"
570 PRINT
580 PRINT "          END TEST."
590 PRINT
600 PRINT "END PROGRAM 45"
610 END
```

---

PROGRAM FILE 45: ALTERING THE CONTROL-VARIABLE WITHIN  
A FOR-BLOCK.  
ANSI STANDARD 11.2, 11.4

SECTION 45.1: ALTERING THE CONTROL-VARIABLE WITHIN  
A FOR-BLOCK.

THIS SECTION TESTS THAT THE VALUE OF THE CONTROL-VARIABLE  
MAY BE ALTERED BY STATEMENTS WITHIN THE FOR-BLOCK, AS  
WELL AS UNDER THE DIRECT CONTROL OF THE FOR-STATEMENT.

BEGIN TEST.

CONTROL VARIABLE:  
 SHOULD BE      ACTUAL      RESULT  
 3                3             OK  
 6                6             OK  
 9                9             OK

CONTROL VARIABLE ALTERED.

20	20	OK
LOOP EXITED		
23	23	OK

NO. TIMES THROUGH LOOP:

SHOULD BE: 4	ACTUAL: 4	RESULT: OK
--------------	-----------	------------

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 45

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

```

10 PRINT "PROGRAM FILE 46: INTERACTION OF CONTROL STATEMENTS WITH"
20 PRINT "      THE FOR-STATEMENT."
30 PRINT "      ANSI STANDARD 11.2, 11.4, 10.2, 10.4"
40 PRINT
50 PRINT "SECTION 46.1: INTERACTION OF GOSUB-STATEMENT WITH"
60 PRINT "      THE FOR-STATEMENT."
70 PRINT
80 PRINT "THIS SECTION TESTS THAT A GOSUB ROUTINE CAN BE EXECUTED"
90 PRINT "FROM WITHIN A FOR-BLOCK."
100 PRINT
110 PRINT "                  BEGIN TEST."
120 LET A1=8
130 LET B1=-5
140 LET C1=-3
150 LET T2=4
160 PRINT
170 PRINT "CONTROL VARIABLE:"
180 PRINT "SHOULD BE","ACTUAL","RESULT"
190 LET A2=A1
200 LET B2=B1
210 LET C2=C1
220 LET I2=A2
230 REM S1 IS ERROR SWITCH
240 LET S1=0
250 REM T1 IS LOOP COUNT
260 LET T1=0
270 FOR I1=A1 TO B1 STEP C1
280   LET T1=T1+1
290   LET M$=" OK "
300   IF I1=I2 THEN 330
310   LET M$="FAILED"
320   LET S1=1

```

```
330 PRINT I2,I1,M$  
340 LET T9=99  
350 GOSUB 650  
360 IF T9=9 THEN 390  
370 PRINT "GOSUB FAILED TO TAKE EFFECT"  
380 LET S1=1  
390 REM END GOSUB TEST  
400 LET I2=I2-1  
410 LET I2=I2+C2  
420 NEXT I1  
430 PRINT "LOOP EXITED"  
450 GOSUB 6000  
490 LET M$=" OK "  
500 IF T1=T2 THEN 530  
510 LET M$="FAILED"  
520 LET S1=1  
530 PRINT  
540 PRINT "NO. TIMES THROUGH LOOP:"  
550 PRINT "SHOULD BE: ";T2;" ACTUAL: ";T1;" RESULT: ";M$  
560 PRINT  
570 IF S1=0 THEN 600  
580 PRINT "*** TEST FAILED ***"  
590 GOTO 610  
600 PRINT "*** TEST PASSED ***"  
610 PRINT  
620 PRINT " END TEST."  
630 PRINT  
640 GOTO 690  
650 REM SUBROUTINE FROM FIRST SECTION  
660 LET T9=9  
670 LET I1=I1-1  
680 RETURN  
690 REM END OF CALLED-FROM-FOR-BLOCK SUBROUTINE  
700 PRINT "SECTION 46.2: FOR-LOOPS WITH SAME CONTROL-VARIABLE"  
710 PRINT " NESTED DYNAMICALLY WITH GOSUB."  
720 PRINT  
730 PRINT "THIS SECTION TESTS THAT A GOSUB ROUTINE CAN BE EXECUTED"  
740 PRINT "FROM WITHIN A FOR-BLOCK AND THAT THE SUBROUTINE MAY ITSELF"  
750 PRINT "CONTAIN A FOR-BLOCK. IF THE TWO FOR-BLOCKS USE THE SAME"  
760 PRINT "CONTROL-VARIABLE, THE ORIGINAL VALUE IS LOST."  
770 PRINT  
780 PRINT " BEGIN TEST."  
790 LET A1=8  
800 LET B1=-5  
810 LET C1=-3  
820 LET T2=1  
830 LET T4=2  
840 PRINT  
850 PRINT "CONTROL VARIABLE:"  
860 PRINT "SHOULD BE","ACTUAL","RESULT"  
870 LET A2=A1  
880 LET B2=B1  
890 LET C2=C1  
900 LET I2=A2  
910 REM S1 IS ERROR SWITCH
```

```
920 LET S1=0
930 REM T1 IS LOOP COUNT
940 LET T1=0
950 PRINT "ENTERING OUTER LOOP"
960 FOR I1=A1 TO B1 STEP C1
970     LET T1=T1+1
990     GOSUB 6000
1030     GOSUB 1360
1040     LET I2=I2+C2
1050 NEXT I1
1060 PRINT "OUTER LOOP EXITED"
1080 GOSUB 6000
1120 LET M$=" OK "
1130 IF T3=T4 THEN 1160
1140 LET M$="FAILED"
1150 LET S1=1
1160 PRINT
1170 PRINT "NO. TIMES THROUGH INNER LOOP:"
1180 PRINT "SHOULD BE: ";T4;"      ACTUAL: ";T3;"    RESULT: ";M$
1190 PRINT
1200 LET M$=" OK "
1210 IF T1=T2 THEN 1240
1220 LET M$="FAILED"
1230 LET S1=1
1240 PRINT
1250 PRINT "NO. TIMES THROUGH OUTER LOOP:"
1260 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"    RESULT: ";M$
1270 PRINT
1280 IF S1=0 THEN 1310
1290 PRINT "*** TEST FAILED ***"
1300 GOTO 1320
1310 PRINT "*** TEST PASSED ***"
1320 PRINT
1330 PRINT "           END TEST."
1340 PRINT
1350 GOTO 1570
1360 REM SUBROUTINE FROM SECOND SECTION
1370 LET I2=-10
1380 REM T3 IS LOOP COUNT
1390 LET T3=0
1400 PRINT "ENTERING INNER LOOP"
1410 FOR I1=-10 TO -7 STEP 2
1420     LET T3=T3+1
1440     GOSUB 6000
1480     LET I2=I2+2
1490 NEXT I1
1500 PRINT "INNER LOOP EXITED"
1520 GOSUB 6000
1560 RETURN
1570 REM END OF CALLED-FROM-FOR-BLOCK SUBROUTINE
1580 PRINT
1590 PRINT "SECTION 46.3 CONTROL-STATEMENT JUMPS OUT OF FOR-BLOCK."
1600 PRINT
```

```
1610 PRINT "THIS SECTION TESTS THAT YOU CAN JUMP OUT OF A FOR-BLOCK"
1620 PRINT "AND THAT THE CONTROL-VARIABLE RETAINS ITS VALUE WHEN"
1625 PRINT "THIS HAPPENS."
1630 PRINT
1640 PRINT "               BEGIN TEST."
1650 PRINT
1655 LET S1=0
1660 LET A1=33
1670 LET B1=66
1680 LET C1=4
1690 LET I2=A1
1700 LET T2=8
1703 PRINT "LOOP IS: FOR I1=";A1;" TO ";B1;" STEP ";C1
1707 PRINT
1710 PRINT "CONTROL VARIABLE:"
1720 PRINT "SHOULD BE","ACTUAL","RESULT"
1740 LET T1=0
1750 FOR I1=A1 TO B1 STEP C1
1760 LET T1=T1+1
1770 GOSUB 6000
1780 IF I2 < 60 THEN 1790
1785 GO TO 1850
1790 LET I2=I2+C1
1800 NEXT I1
1810 PRINT "NORMAL EXIT FROM FOR-BLOCK: INCORRECT."
1820 LET S1=1
1830 GOTO 1855
1850 PRINT "JUMPED OUT OF LOOP"
1855 GOSUB 6000
1860 LET M$=" OK "
1870 IF T1=T2 THEN 1900
1880 LET M$="FAILED"
1890 LET S1=1
1900 PRINT
1910 PRINT "NO. TIMES THROUGH LOOP:"
1920 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"      RESULT: ";M$
1930 PRINT
1940 PRINT "-----"
1950 PRINT
2000 LET A1=7
2010 LET B1=7
2020 LET C1=0
2030 LET I2=A1
2040 LET T2=5
2050 PRINT "LOOP IS: FOR I1=";A1;" TO ";B1;" STEP ";C1
2060 PRINT
2070 PRINT "CONTROL VARIABLE:"
2080 PRINT "SHOULD BE","ACTUAL","RESULT"
2090 LET T1=0
2100 FOR I1=A1 TO B1 STEP C1
2110 LET T1=T1+1
2120 GOSUB 6000
2130 IF T1 > 4.5 THEN 2190
2140 LET I2=I2+C1
2150 NEXT I1
```

```

2160 PRINT "NORMAL EXIT FROM FOR-BLOCK: INCORRECT."
2170 LET S1=1
2180 GOTO 2200
2190 PRINT "JUMPED OUT OF LOOP"
2200 GOSUB 6000
2210 LET M$=" OK "
2220 IF T1=T2 THEN 2250
2230 LET M$="FAILED"
2240 LET S1=1
2250 PRINT
2260 PRINT "NO. TIMES THROUGH LOOP:"
2270 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"      RESULT: ";M$
2280 PRINT
3000 IF S1=0 THEN 3030
3010 PRINT "*** TEST FAILED ***"
3020 GOTO 3040
3030 PRINT "*** TEST PASSED ***"
3040 PRINT
3050 PRINT "           END TEST."
3060 PRINT
3070 PRINT "END PROGRAM 46"
3080 STOP
6000 REM SUBROUTINE TO CHECK THAT I1=I2
6010 LET M$=" OK "
6020 IF I1=I2 THEN 6050
6030 LET M$="FAILED"
6040 LET S1=1
6050 PRINT I2,I1,M$
6060 RETURN
6070 END

```

---

PROGRAM FILE 46: INTERACTION OF CONTROL STATEMENTS WITH  
THE FOR-STATEMENT.

ANSI STANDARD 11.2, 11.4, 10.2, 10.4

SECTION 46.1: INTERACTION OF GOSUB-STATEMENT WITH  
THE FOR-STATEMENT.

THIS SECTION TESTS THAT A GOSUB ROUTINE CAN BE EXECUTED  
FROM WITHIN A FOR-BLOCK.

BEGIN TEST.

CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
8	8	OK
4	4	OK
0	0	OK
-4	-4	OK
LOOP EXITED		
-8	-8	OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 4      ACTUAL: 4      RESULT: OK

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 46.2: FOR-LOOPS WITH SAME CONTROL-VARIABLE  
NESTED DYNAMICALLY WITH GOSUB.

THIS SECTION TESTS THAT A GOSUB ROUTINE CAN BE EXECUTED  
FROM WITHIN A FOR-BLOCK AND THAT THE SUBROUTINE MAY ITSELF  
CONTAIN A FOR-BLOCK. IF THE TWO FOR-BLOCKS USE THE SAME  
CONTROL-VARIABLE, THE ORIGINAL VALUE IS LOST.

BEGIN TEST.

CONTROL VARIABLE:

SHOULD BE      ACTUAL      RESULT

ENTERING OUTER LOOP

8                8                OK

ENTERING INNER LOOP

-10              -10             OK

-8                -8                OK

INNER LOOP EXITED

-6                -6                OK

OUTER LOOP EXITED

-9                -9                OK

NO. TIMES THROUGH INNER LOOP:

SHOULD BE: 2      ACTUAL: 2      RESULT: OK

NO. TIMES THROUGH OUTER LOOP:

SHOULD BE: 1      ACTUAL: 1      RESULT: OK

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 46.3 CONTROL-STATEMENT JUMPS OUT OF FOR-BLOCK.

THIS SECTION TESTS THAT YOU CAN JUMP OUT OF A FOR-BLOCK  
AND THAT THE CONTROL-VARIABLE RETAINS ITS VALUE WHEN  
THIS HAPPENS.

BEGIN TEST.

LOOP IS: FOR I1= 33 TO 66 STEP 4

## CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
33	33	OK
37	37	OK
41	41	OK
45	45	OK
49	49	OK
53	53	OK
57	57	OK
61	61	OK
JUMPED OUT OF LOOP		
61	61	OK

## NO. TIMES THROUGH LOOP:

SHOULD BE: 8 ACTUAL: 8 RESULT: OK

LOOP IS: FOR I1= 7 TO 7 STEP 0

## CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
7	7	OK
JUMPED OUT OF LOOP		
7	7	OK

## NO. TIMES THROUGH LOOP:

SHOULD BE: 5 ACTUAL: 5 RESULT: OK

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 46

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 47: INCREMENT IN THE STEP CLAUSE OF THE"
20 PRINT "      FOR-STATEMENT DEFAULTS TO A VALUE OF ONE."
30 PRINT "      ANSI STANDARD 11.2, 11.4"
40 PRINT
50 PRINT "SECTION 47.1: INCREMENT IN THE STEP CLAUSE OF THE"
60 PRINT "      FOR-STATEMENT DEFAULTS TO A VALUE OF ONE."
70 PRINT
```

```
80 REM S1 IS ERROR SWITCH
90 LET S1=0
110 PRINT "                  BEGIN TEST."
120 LET A1=8
130 LET B1=5
150 LET T2=0
160 GOSUB 3000
170 LET A1=5
180 LET B1=8
190 LET T2=4
200 GOSUB 3000
210 LET A1=88
220 LET B1=88
230 LET T2=1
240 GOSUB 3000
1000 IF S1=0 THEN 1030
1010 PRINT "*** TEST FAILED ***"
1020 GOTO 1040
1030 PRINT "*** TEST PASSED ***"
1040 PRINT
1050 PRINT "                  END TEST."
1060 PRINT
1070 PRINT "END PROGRAM 47"
1080 STOP
3000 REM SUBROUTINE TO TEST VARIOUS LOOPS
3002 PRINT
3004 PRINT "LOOP IS: FOR I1=";A1;" TO ";B1
3006 PRINT
3010 PRINT "CONTROL VARIABLE:"
3020 PRINT "SHOULD BE","ACTUAL","RESULT"
3030 LET I2=A1
3090 REM T1 IS LOOP COUNT
3100 LET T1=0
3110 FOR I1=A1 TO B1
3120     LET T1=T1+1
3130     GOSUB 6000
3140     LET I2=I2+1
3150 NEXT I1
3160 PRINT "LOOP EXITED"
3170 GOSUB 6000
3180 LET M$=" OK "
3190 IF T1=T2 THEN 3220
3200 LET M$="FAILED"
3210 LET S1=1
3220 PRINT
3230 PRINT "NO. TIMES THROUGH LOOP:"
3240 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"      RESULT: ";M$
3250 PRINT
3260 PRINT "-----"
3270 PRINT
```

```
3280 RETURN
6000 REM SUBROUTINE TO CHECK THAT I1=I2
6010 LET M$=" OK "
6020 IF I1=I2 THEN 6050
6030 LET M$="FAILED"
6040 LET S1=1
6050 PRINT I2,I1,M$
6060 RETURN
6070 END
```

---

PROGRAM FILE 47: INCREMENT IN THE STEP CLAUSE OF THE  
FOR-STATEMENT DEFAULTS TO A VALUE OF ONE.  
ANSI STANDARD 11.2, 11.4

SECTION 47.1: INCREMENT IN THE STEP CLAUSE OF THE  
FOR-STATEMENT DEFAULTS TO A VALUE OF ONE.

BEGIN TEST.

LOOP IS: FOR I1= 8 TO 5

CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
-----------	--------	--------

LOOP EXITED

8	8	OK
---	---	----

NO. TIMES THROUGH LOOP:

SHOULD BE: 0	ACTUAL: 0	RESULT: OK
--------------	-----------	------------

---

LOOP IS: FOR I1= 5 TO 8

CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
-----------	--------	--------

5	5	OK
6	6	OK
7	7	OK
8	8	OK

LOOP EXITED

9	9	OK
---	---	----

NO. TIMES THROUGH LOOP:

SHOULD BE: 4	ACTUAL: 4	RESULT: OK
--------------	-----------	------------

---

LOOP IS: FOR I1= 88 TO 88

CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
88	88	OK
LOOP EXITED		
89	89	OK

NO. TIMES THROUGH LOOP:

SHOULD BE: 1 ACTUAL: 1 RESULT: OK

---

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 47

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 48: LIMIT AND INCREMENT IN THE FOR-STATEMENT"  
20 PRINT "      ARE EVALUATED ONCE UPON ENTERING THE LOOP."  
30 PRINT "      ANSI STANDARD 11.2, 11.4"  
40 PRINT  
50 PRINT "SECTION 48.1: LIMIT AND INCREMENT IN THE FOR-STATEMENT"  
60 PRINT "      ARE EVALUATED ONCE UPON ENTERING THE LOOP."  
70 PRINT  
80 REM S1 IS ERROR SWITCH  
90 LET S1=0  
100 PRINT "          BEGIN TEST."  
110 LET I1=-20  
120 LET I2=-20  
130 LET T2=1  
140 PRINT  
150 PRINT "I1 SET TO -20"  
160 PRINT "LOOP IS: FOR I1=I1 TO I1 STEP -3"  
170 PRINT  
180 PRINT "CONTROL VARIABLE:"  
190 PRINT "SHOULD BE","ACTUAL","RESULT"  
200 LET T1=0  
210 FOR I1=I1 TO I1 STEP -3  
220 LET T1=T1+1  
230 GOSUB 3120  
240 IF T1>20 THEN 290  
250 LET I2=I2-3  
260 NEXT I1  
270 PRINT "LOOP EXITED"  
280 GOTO 310  
290 PRINT "JUMPED OUT OF LOOP: INCORRECT"  
300 LET S1=1  
310 GOSUB 3120  
320 GOSUB 3000
```

```
400 LET I1=-3
410 LET I2=-3
420 LET T2=6
430 PRINT
440 PRINT "I1 SET TO -3"
450 PRINT "LOOP IS: FOR I1=I1 TO -20 STEP I1"
460 PRINT
470 PRINT "CONTROL VARIABLE:"
480 PRINT "SHOULD BE","ACTUAL","RESULT"
490 LET T1=0
500 FOR I1=I1 TO -20 STEP I1
510 LET T1=T1+1
520 GOSUB 3120
530 IF T1>20 THEN 580
540 LET I2=I2-3
550 NEXT I1
560 PRINT "LOOP EXITED"
570 GOTO 600
580 PRINT "JUMPED OUT OF LOOP: INCORRECT"
590 LET S1=1
600 GOSUB 3120
610 GOSUB 3000
700 LET I1=-2
710 LET I2=9
720 LET T2=6
730 PRINT
740 PRINT "I1 SET TO -2"
750 PRINT "LOOP IS: FOR I1=9 TO I1 STEP I1"
760 PRINT
770 PRINT "CONTROL VARIABLE:"
780 PRINT "SHOULD BE","ACTUAL","RESULT"
790 LET T1=0
800 FOR I1=9 TO I1 STEP I1
810 LET T1=T1+1
820 GOSUB 3120
830 IF T1>20 THEN 880
840 LET I2=I2-2
850 NEXT I1
860 PRINT "LOOP EXITED"
870 GOTO 900
880 PRINT "JUMPED OUT OF LOOP: INCORRECT"
890 LET S1=1
900 GOSUB 3120
910 GOSUB 3000
2000 IF S1=0 THEN 2030
2010 PRINT "*** TEST FAILED ***"
2020 GOTO 2040
2030 PRINT "*** TEST PASSED ***"
2040 PRINT
2050 PRINT "END TEST."
```

```

2060 PRINT
2070 PRINT "END PROGRAM 48"
2080 STOP
3000 REM SUBROUTINE TO CHECK NUMBER OF TIMES THROUGH LOOP
3010 LET M$=" OK "
3020 IF T1=T2 THEN 3050
3030 LET M$="FAILED"
3040 LET S1=1
3050 PRINT
3060 PRINT "NO. TIMES THROUGH LOOP:"
3070 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"    RESULT: ";M$
3080 PRINT
3090 PRINT "-----"
3100 PRINT
3110 RETURN
3120 REM SUBROUTINE TO CHECK THAT I1=I2
3130 LET M$=" OK "
3140 IF I1=I2 THEN 3170
3150 LET M$="FAILED"
3160 LET S1=1
3170 PRINT I2,I1,M$
3180 RETURN
3190 END

```

---

PROGRAM FILE 48: LIMIT AND INCREMENT IN THE FOR-STATEMENT  
ARE EVALUATED ONCE UPON ENTERING THE LOOP.  
ANSI STANDARD 11.2, 11.4

SECTION 48.1: LIMIT AND INCREMENT IN THE FOR-STATEMENT  
ARE EVALUATED ONCE UPON ENTERING THE LOOP.

BEGIN TEST.

I1 SET TO -20  
LOOP IS: FOR I1=I1 TO I1 STEP -3

CONTROL VARIABLE:  
SHOULD BE ACTUAL RESULT  
-20 -20 OK  
LOOP EXITED  
-23 -23 OK

NO. TIMES THROUGH LOOP:  
SHOULD BE: 1 ACTUAL: 1 RESULT: OK

---

I1 SET TO -3  
LOOP IS: FOR I1=I1 TO -20 STEP I1

## CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
-3	-3	OK
-6	-6	OK
-9	-9	OK
-12	-12	OK
-15	-15	OK
-18	-18	OK
LOOP EXITED		
-21	-21	OK

## NO. TIMES THROUGH LOOP:

SHOULD BE: 6      ACTUAL: 6      RESULT: OK

---

I1 SET TO -2

LOOP IS: FOR I1=9 TO I1 STEP I1

## CONTROL VARIABLE:

SHOULD BE	ACTUAL	RESULT
9	9	OK
7	7	OK
5	5	OK
3	3	OK
1	1	OK
-1	-1	OK
LOOP EXITED		
-3	-3	OK

## NO. TIMES THROUGH LOOP:

SHOULD BE: 6      ACTUAL: 6      RESULT: OK

---

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 48

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 49: NESTED FOR-BLOCKS."
20 PRINT "      ANSI STANDARD 11.2, 11.4"
30 PRINT
40 PRINT "SECTION 49.1: NESTED FOR-BLOCKS."
50 PRINT
60 PRINT "THIS SECTION TESTS THAT FOR-BLOCKS MAY BE NESTED IN THE"
70 PRINT "TEXT OF THE PROGRAM, WITH SEPARATE CONTROL MAINTAINED"
80 PRINT "FOR EACH CONTROL-VARIABLE."
90 PRINT

```

```
100 LET S1=0
110 PRINT "                  BEGIN TEST."
120 LET T2=204
130 LET E$="ENTERING FOR-BLOCK"
140 LET F$="EXITING FOR-BLOCK"
150 PRINT
160 PRINT "EACH DETAIL LINE REPORTS:"
170 PRINT "    1) DEPTH OF NESTING"
180 PRINT "    2) CORRECT VALUE FOR CONTROL VARIABLE"
190 PRINT "    3) ACTUAL VALUE OF CONTROL VARIABLE"
200 PRINT "    4) RESULT (OK OR FAILED)"
210 PRINT
220 PRINT "DETAIL LINES ARE INDENTED TO SHOW DEPTH OF NESTING."
230 PRINT
240 LET T1=0
250 LET I2=12
260     PRINT E$;" 1"
270     FOR I1=12 TO 0 STEP -1
280     LET T1=T1+1
290     LET J2=-1
300         PRINT E$;" 2"
310         FOR J1=-1 TO 0
320         LET T1=T1+1
330         GOSUB 1070
340         LET J2=J2+1
350         NEXT J1
360         PRINT F$;" 2"
370     GOSUB 1070
380     LET K2=1
390     PRINT E$;" 3"
400     FOR K1=1 TO 3
410     LET T1=T1+1
420     LET J2=I2
430         PRINT E$;" 4"
440         FOR J1=I1 TO 1 STEP -2
450         LET T1=T1+1
460         GOSUB 1070
470         LET J2=J2-2
480         NEXT J1
490         PRINT F$;" 4"
500     GOSUB 1070
510     GOSUB 1140
520     LET K2=K2+1
530     NEXT K1
540     PRINT F$;" 3"
550     GOSUB 1140
560     GOSUB 1000
570     LET I2=I2-1
580     NEXT I1
590     PRINT F$;" 1"
600 GOSUB 1000
610 LET M$=" OK "
620 IF T1=T2 THEN 650
```

```
630 LET M$="FAILED"
640 LET S1=1
650 PRINT
660 PRINT "NO. TIMES THROUGH LOOPS:"
670 PRINT "SHOULD BE: ";T2;"      ACTUAL: ";T1;"    RESULT: ";M$
680 PRINT
690 IF S1=0 THEN 720
700 PRINT "*** TEST FAILED ***"
710 GOTO 730
720 PRINT "*** TEST PASSED ***"
730 PRINT
740 PRINT "                  END TEST."
750 PRINT
760 PRINT "END PROGRAM 49"
770 STOP
1000 REM SUBROUTINE TO CHECK THAT I1=I2
1010 LET M$=" OK "
1020 IF I1=I2 THEN 1050
1030 LET M$="FAILED"
1040 LET S1=1
1050 PRINT 1;TAB(10);I2;TAB(20);I1;TAB(30);M$
1060 RETURN
1070 REM SUBROUTINE TO CHECK THAT J1=J2
1080 LET M$=" OK "
1090 IF J1=J2 THEN 1120
1100 LET M$="FAILED"
1110 LET S1=1
1120 PRINT TAB(2);2;TAB(12);J2;TAB(22);J1;TAB(32);M$
1130 RETURN
1140 REM SUBROUTINE TO CHECK THAT K1=K2
1150 LET M$=" OK "
1160 IF K1=K2 THEN 1190
1170 LET M$="FAILED"
1180 LET S1=1
1190 PRINT TAB(4);3;TAB(14);K2;TAB(24);K1;TAB(34);M$
1200 RETURN
1210 END
```

---

PROGRAM FILE 49: NESTED FOR-BLOCKS.

ANSI STANDARD 11.2, 11.4

#### SECTION 49.1: NESTED FOR-BLOCKS.

THIS SECTION TESTS THAT FOR-BLOCKS MAY BE NESTED IN THE TEXT OF THE PROGRAM, WITH SEPARATE CONTROL MAINTAINED FOR EACH CONTROL-VARIABLE.

BEGIN TEST.

EACH DETAIL LINE REPORTS:

- 1) DEPTH OF NESTING
- 2) CORRECT VALUE FOR CONTROL VARIABLE
- 3) ACTUAL VALUE OF CONTROL VARIABLE
- 4) RESULT (OK OR FAILED)

DETAIL LINES ARE INDENTED TO SHOW DEPTH OF NESTING.

ENTERING FOR-BLOCK 1

ENTERING FOR-BLOCK 2

2	-1	-1	OK
2	0	0	OK

EXITING FOR-BLOCK 2

2	1	1	OK
---	---	---	----

ENTERING FOR-BLOCK 3

ENTERING FOR-BLOCK 4

2	12	12	OK
2	10	10	OK
2	8	8	OK
2	6	6	OK
2	4	4	OK
2	2	2	OK

EXITING FOR-BLOCK 4

2	0	0	OK
3	1	1	OK

ENTERING FOR-BLOCK 4

2	12	12	OK
2	10	10	OK
2	8	8	OK
2	6	6	OK
2	4	4	OK
2	2	2	OK

EXITING FOR-BLOCK 4

2	0	0	OK
3	2	2	OK

ENTERING FOR-BLOCK 4

2	12	12	OK
2	10	10	OK
2	8	8	OK
2	6	6	OK
2	4	4	OK
2	2	2	OK

EXITING FOR-BLOCK 4

2	0	0	OK
3	3	3	OK

EXITING FOR-BLOCK 3

3	4	4	OK
---	---	---	----

1	12	12	OK
---	----	----	----

ENTERING FOR-BLOCK 2

2	-1	-1	OK
2	0	0	OK

EXITING FOR-BLOCK 2

2	1	1	OK
---	---	---	----

ENTERING FOR-BLOCK 3

ENTERING FOR-BLOCK 4

```

2      11      11      OK
2      9       9       OK
2      7       7       OK
2      5       5       OK
2      3       3       OK
2      1       1       OK
EXITING FOR-BLOCK 4
2      -1      -1      OK
3      1       1       OK
ENTERING FOR-BLOCK 4
2      11      11      OK
2      9       9       OK
2      7       7       OK
2      5       5       OK
2      3       3       OK
2      1       1       OK
EXITING FOR-BLOCK 4
2      -1      -1      OK
3      2       2       OK
ENTERING FOR-BLOCK 4
2      11      11      OK
2      9       9       OK
2      7       7       OK
2      5       5       OK
2      3       3       OK
2      1       1       OK
EXITING FOR-BLOCK 4
2      -1      -1      OK
3      3       3       OK
EXITING FOR-BLOCK 3
3      4       4       OK
1      11      11      OK
ENTERING FOR-BLOCK 2
2      -1      -1      OK
2      0       0       OK
EXITING FOR-BLOCK 2
2      1       1       OK
ENTERING FOR-BLOCK 3
ENTERING FOR-BLOCK 4
2      10      10      OK
2      8       8       OK
2      6       6       OK
2      4       4       OK
2      2       2       OK
EXITING FOR-BLOCK 4
2      0       0       OK
3      1       1       OK
ENTERING FOR-BLOCK 4
2      10      10      OK
2      8       8       OK
2      6       6       OK
2      4       4       OK
2      2       2       OK
EXITING FOR-BLOCK 4
2      0       0       OK

```

```

      3      2      2      OK
ENTERING FOR-BLOCK 4
      2      10     10      OK
      2       8      8      OK
      2       6      6      OK
      2       4      4      OK
      2       2      2      OK
EXITING FOR-BLOCK 4
      2       0      0      OK
      3      3      3      OK
EXITING FOR-BLOCK 3
      3      4      4      OK
      1      10     10      OK
ENTERING FOR-BLOCK 2
      2      -1     -1      OK
      2       0      0      OK
EXITING FOR-BLOCK 2
      2       1      1      OK
ENTERING FOR-BLOCK 3
ENTERING FOR-BLOCK 4
      2       9      9      OK
      2       7      7      OK
      2       5      5      OK
      2       3      3      OK
      2       1      1      OK
EXITING FOR-BLOCK 4
      2      -1     -1      OK
      3      1      1      OK
ENTERING FOR-BLOCK 4
      2       9      9      OK
      2       7      7      OK
      2       5      5      OK
      2       3      3      OK
      2       1      1      OK
EXITING FOR-BLOCK 4
      2      -1     -1      OK
      3      2      2      OK
ENTERING FOR-BLOCK 4
      2       9      9      OK
      2       7      7      OK
      2       5      5      OK
      2       3      3      OK
      2       1      1      OK
EXITING FOR-BLOCK 4
      2      -1     -1      OK
      3      3      3      OK
EXITING FOR-BLOCK 3
      3      4      4      OK
      1      9      9      OK
ENTERING FOR-BLOCK 2
      2      -1     -1      OK
      2       0      0      OK
EXITING FOR-BLOCK 2
      2       1      1      OK
ENTERING FOR-BLOCK 3

```

ENTERING FOR-BLOCK 4

2	8	8	OK
2	6	6	OK
2	4	4	OK
2	2	2	OK

EXITING FOR-BLOCK 4

2	0	0	OK
3	1	1	OK

ENTERING FOR-BLOCK 4

2	8	8	OK
2	6	6	OK
2	4	4	OK
2	2	2	OK

EXITING FOR-BLOCK 4

2	0	0	OK
3	2	2	OK

ENTERING FOR-BLOCK 4

2	8	8	OK
2	6	6	OK
2	4	4	OK
2	2	2	OK

EXITING FOR-BLOCK 4

2	0	0	OK
3	3	3	OK

EXITING FOR-BLOCK 3

3	4	4	OK
1	8	8	OK

ENTERING FOR-BLOCK 2

2	-1	-1	OK
2	0	0	OK

EXITING FOR-BLOCK 2

2	1	1	OK
---	---	---	----

ENTERING FOR-BLOCK 3

ENTERING FOR-BLOCK 4

2	7	7	OK
2	5	5	OK
2	3	3	OK
2	1	1	OK

EXITING FOR-BLOCK 4

2	-1	-1	OK
3	1	1	OK

ENTERING FOR-BLOCK 4

2	7	7	OK
2	5	5	OK
2	3	3	OK
2	1	1	OK

EXITING FOR-BLOCK 4

2	-1	-1	OK
3	2	2	OK

ENTERING FOR-BLOCK 4

2	7	7	OK
2	5	5	OK
2	3	3	OK
2	1	1	OK

EXITING FOR-BLOCK 4

2	-1	-1	OK
3	3	3	OK
EXITING FOR-BLOCK 3			
3	4	4	OK
1	7	7	OK
ENTERING FOR-BLOCK 2			
2	-1	-1	OK
2	0	0	OK
EXITING FOR-BLOCK 2			
2	1	1	OK
ENTERING FOR-BLOCK 3			
ENTERING FOR-BLOCK 4			
2	6	6	OK
2	4	4	OK
2	2	2	OK
EXITING FOR-BLOCK 4			
2	0	0	OK
3	1	1	OK
ENTERING FOR-BLOCK 4			
2	6	6	OK
2	4	4	OK
2	2	2	OK
EXITING FOR-BLOCK 4			
2	0	0	OK
3	2	2	OK
ENTERING FOR-BLOCK 4			
2	6	6	OK
2	4	4	OK
2	2	2	OK
EXITING FOR-BLOCK 4			
2	0	0	OK
3	3	3	OK
EXITING FOR-BLOCK 3			
3	4	4	OK
1	6	6	OK
ENTERING FOR-BLOCK 2			
2	-1	-1	OK
2	0	0	OK
EXITING FOR-BLOCK 2			
2	1	1	OK
ENTERING FOR-BLOCK 3			
ENTERING FOR-BLOCK 4			
2	5	5	OK
2	3	3	OK
2	1	1	OK
EXITING FOR-BLOCK 4			
2	-1	-1	OK
3	1	1	OK
ENTERING FOR-BLOCK 4			
2	5	5	OK
2	3	3	OK
2	1	1	OK
EXITING FOR-BLOCK 4			
2	-1	-1	OK
3	2	2	OK

ENTERING FOR-BLOCK 4  
2 5 5 OK  
2 3 3 OK  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 3 3 OK  
EXITING FOR-BLOCK 3  
3 4 4 OK  
1 5 5 OK  
ENTERING FOR-BLOCK 2  
2 -1 -1 OK  
2 0 0 OK  
EXITING FOR-BLOCK 2  
2 1 1 OK  
ENTERING FOR-BLOCK 3  
ENTERING FOR-BLOCK 4  
2 4 4 OK  
2 2 2 OK  
EXITING FOR-BLOCK 4  
2 0 0 OK  
3 1 1 OK  
ENTERING FOR-BLOCK 4  
2 4 4 OK  
2 2 2 OK  
EXITING FOR-BLOCK 4  
2 0 0 OK  
3 2 2 OK  
ENTERING FOR-BLOCK 4  
2 4 4 OK  
2 2 2 OK  
EXITING FOR-BLOCK 4  
2 0 0 OK  
3 3 3 OK  
EXITING FOR-BLOCK 3  
3 4 4 OK  
1 4 4 OK  
ENTERING FOR-BLOCK 2  
2 -1 -1 OK  
2 0 0 OK  
EXITING FOR-BLOCK 2  
2 1 1 OK  
ENTERING FOR-BLOCK 3  
ENTERING FOR-BLOCK 4  
2 3 3 OK  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 1 1 OK  
ENTERING FOR-BLOCK 4  
2 3 3 OK  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 2 2 OK

ENTERING FOR-BLOCK 4  
2 3 3 OK  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 3 3 OK  
EXITING FOR-BLOCK 3  
3 4 4 OK  
1 3 3 OK  
ENTERING FOR-BLOCK 2  
2 -1 -1 OK  
2 0 0 OK  
EXITING FOR-BLOCK 2  
2 1 1 OK  
ENTERING FOR-BLOCK 3  
ENTERING FOR-BLOCK 4  
2 2 2 OK  
EXITING FOR-BLOCK 4  
2 0 0 OK  
3 1 1 OK  
ENTERING FOR-BLOCK 4  
2 2 2 OK  
EXITING FOR-BLOCK 4  
2 0 0 OK  
3 2 2 OK  
ENTERING FOR-BLOCK 4  
2 2 2 OK  
EXITING FOR-BLOCK 4  
2 0 0 OK  
3 3 3 OK  
EXITING FOR-BLOCK 3  
3 4 4 OK  
1 2 2 OK  
ENTERING FOR-BLOCK 2  
2 -1 -1 OK  
2 0 0 OK  
EXITING FOR-BLOCK 2  
2 1 1 OK  
ENTERING FOR-BLOCK 3  
ENTERING FOR-BLOCK 4  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 1 1 OK  
ENTERING FOR-BLOCK 4  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 2 2 OK  
ENTERING FOR-BLOCK 4  
2 1 1 OK  
EXITING FOR-BLOCK 4  
2 -1 -1 OK  
3 3 3 OK  
EXITING FOR-BLOCK 3

3	4	4	OK
1	1	1	OK
ENTERING FOR-BLOCK 2			
2	-1	-1	OK
2	0	0	OK
EXITING FOR-BLOCK 2			
2	1	1	OK
ENTERING FOR-BLOCK 3			
ENTERING FOR-BLOCK 4			
EXITING FOR-BLOCK 4			
2	0	0	OK
3	1	1	OK
ENTERING FOR-BLOCK 4			
EXITING FOR-BLOCK 4			
2	0	0	OK
3	2	2	OK
ENTERING FOR-BLOCK 4			
EXITING FOR-BLOCK 4			
2	0	0	OK
3	3	3	OK
EXITING FOR-BLOCK 3			
3	4	4	OK
1	0	0	OK
EXITING FOR-BLOCK 1			
1	-1	-1	OK

NO. TIMES THROUGH LOOPS:

SHOULD BE: 204      ACTUAL: 204      RESULT: OK

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 49

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 50: ERROR - FOR-STATEMENT WITHOUT A"
15 PRINT "           MATCHING NEXT-STATEMENT."
20 PRINT "           ANSI STANDARD 11.2, 11.4"
30 PRINT
40 PRINT "SECTION 50.1: ERROR - FOR-STATEMENT WITHOUT A"
50 PRINT "           MATCHING NEXT-STATEMENT."
60 PRINT
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT
110 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCOMPANIED BY DOCUMENTATION"
120 PRINT "           ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT "           BY THE PROCESSOR, OR"
140 PRINT

```

```
150 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT "                                BEGIN TEST."
210 PRINT
220 PRINT "ENTERING FOR-BLOCK; LOOP IS: FOR I=1 TO 5"
230 FOR I=1 TO 5
240 PRINT "CONTROL-VARIABLE = ";I
310 PRINT
320 PRINT "                                END TEST."
330 PRINT
340 PRINT "END PROGRAM 50"
350 END
```

---

? FOR WITHOUT NEXT IN LINE 230

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 51: ERROR - NEXT-STATEMENT WITHOUT A"
15 PRINT "      MATCHING FOR-STATEMENT."
20 PRINT "      ANSI STANDARD 11.2, 11.4"
30 PRINT
40 PRINT "SECTION 51.1: ERROR - NEXT-STATEMENT WITHOUT A"
50 PRINT "      MATCHING FOR-STATEMENT."
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
110 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "      BY THE PROCESSOR, OR"
130 PRINT
140 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 PRINT "ENTERING FOR-BLOCK; LOOP IS: FOR I=1 TO 5"
220 FOR I=1 TO 5
230 PRINT "CONTROL-VARIABLE = ";I
240 NEXT I
270 PRINT
280 PRINT "                                END TEST."
```

```
290 PRINT
300 PRINT "END PROGRAM 51"
303 STOP
306 NEXT I
310 END
```

---

? NEXT WITHOUT FOR IN LINE 306

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 52: ERROR - MISMATCHED CONTROL-VARIABLES ON"
15 PRINT "          FOR-STATEMENT AND NEXT-STATEMENT."
20 PRINT "          ANSI STANDARD 11.4"
30 PRINT
40 PRINT "SECTION 52.1: ERROR - MISMATCHED CONTROL-VARIABLES ON"
50 PRINT "          FOR-STATEMENT AND NEXT-STATEMENT."
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 PRINT "ENTERING FOR-BLOCK; LOOP IS: FOR I=1 TO 5 ... NEXT J"
220 FOR I=1 TO 5
230 PRINT "I = ";I;"      J = ";J
240 NEXT J
270 PRINT
280 PRINT "                                END TEST."
290 PRINT
300 PRINT "END PROGRAM 52"
310 END
```

---

? NEXT WITHOUT FOR IN LINE 240

? FOR WITHOUT NEXT IN LINE 220

```
*****
*****  
10 PRINT "PROGRAM FILE 53: ERROR - INTERLEAVED FOR-BLOCKS."  
20 PRINT "      ANSI STANDARD 11.4"  
30 PRINT  
40 PRINT "SECTION 53.1: ERROR - INTERLEAVED FOR-BLOCKS."  
50 PRINT  
60 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
70 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:  
80 PRINT  
90 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
100 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
110 PRINT "          BY THE PROCESSOR, OR"  
120 PRINT  
130 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
140 PRINT  
150 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
160 PRINT "FOR DETAILED CRITERIA."  
170 PRINT  
180 PRINT "                                BEGIN TEST."  
190 PRINT  
200 PRINT "ENTERING FOR-BLOCKS; LOOPS ARE:"  
205 PRINT "FOR I=1 TO 3 / FOR J=1 TO 5 / NEXT I / NEXT J"  
210 FOR I=1 TO 3  
220 FOR J=1 TO 5  
230 PRINT "I = ";I;"        J = ";J  
270 NEXT I  
280 NEXT J  
310 PRINT  
320 PRINT "                                END TEST."  
330 PRINT  
340 PRINT "END PROGRAM 53"  
350 END
```

---

```
? NEXT WITHOUT FOR IN LINE 270
```

```
? FOR WITHOUT NEXT IN LINE 210
```

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 54: ERROR - NESTED FOR-BLOCKS"  
15 PRINT "      WITH THE SAME CONTROL VARIABLE."  
20 PRINT "      ANSI STANDARD 11.4"  
30 PRINT  
40 PRINT "SECTION 54.1: ERROR - NESTED FOR-BLOCKS WITH"  
50 PRINT "      THE SAME CONTROL VARIABLE."  
60 PRINT
```

```

70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 PRINT "ENTERING FOR-BLOCKS; LOOPS ARE:"
220 PRINT "FOR I=1 TO 3 / FOR J=2 TO 4 / FOR I=3 TO 5 / ";
230 PRINT "NEXT I / NEXT J / NEXT I"
240 PRINT
260 FOR I=1 TO 3
270 FOR J=2 TO 4
280 FOR I=3 TO 5
290 PRINT "I = ";I;"      J = ";J
300 NEXT I
305 NEXT J
310 NEXT I
350 PRINT
360 PRINT "                                END TEST."
370 PRINT
380 PRINT "END PROGRAM 54"
390 END

```

---

? FOR USING PREVIOUS CONTROL-VARIABLE IN LINE 280

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 55: ERROR - JUMP INTO FOR-BLOCK."
20 PRINT "    ANSI STANDARD 11.4"
30 PRINT
40 PRINT "SECTION 55.1: ERROR - JUMP INTO FOR-BLOCK."
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT

```

```

160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "                                BEGIN TEST."
192 PRINT "SETTING I = 5"
195 LET I=5
200 PRINT
210 PRINT "ABOUT TO JUMP INTO FOR-BLOCK AROUND FOR-STATEMENT;"
220 PRINT "LOOP IS: FOR I=1 TO 9 ... NEXT I"
240 PRINT
250 GOTO 270
260 FOR I=1 TO 9
270 PRINT "I = ";I
310 NEXT I
350 PRINT
360 PRINT "                                END TEST."
370 PRINT
380 PRINT "END PROGRAM 55"
390 END

```

---

? CONTROL TRANSFER INTO FOR-BLOCK IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 56: ARRAY ASSIGNMENT WITHOUT"
15 PRINT "      THE OPTION-STATEMENT."
20 PRINT "      ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 15.2, 15.4"
30 PRINT
90 PRINT "SECTION 56.1: ONE-DIMENSIONAL ARRAYS WITHOUT THE"
95 PRINT "      DIM-STATEMENT."
100 PRINT
110 PRINT "                                BEGIN TEST."
120 PRINT
130 FOR I=0 TO 10
140 LET A(I)=I
150 NEXT I
160 LET A1=0
170 LET C=0
180 FOR I=10 TO 0 STEP -1
190 LET Y=A(I)
200 LET A1=A1+Y
210 IF Y<>I THEN 230
220 LET C=C+1
230 NEXT I
240 IF C<>11 THEN 260
250 IF A1=55 THEN 280
260 PRINT "*** TEST FAILED ***"
270 GOTO 290
280 PRINT "*** TEST PASSED ***"
290 PRINT
300 PRINT "                                END TEST."

```

```
310 PRINT
320 PRINT "SECTION 56.2: TWO-DIMENSIONAL ARRAYS WITHOUT THE"
325 PRINT "           DIM-STATEMENT."
330 PRINT
340 PRINT "                               BEGIN TEST."
350 PRINT
360 FOR I=0 TO 10
370 FOR J=0 TO 10
380 LET Z=I+J
390 LET B(I,J)=Z
400 NEXT J
410 NEXT I
420 LET B1=0
430 LET C=0
440 FOR I=10 TO 0 STEP -1
450 FOR J=10 TO 0 STEP -1
460 LET W=B(I,J)
470 LET B1=B1+W
480 LET Z=I+J
490 IF W<>Z THEN 510
500 LET C=C+1
510 NEXT J
520 NEXT I
530 IF C<>121 THEN 550
540 IF B1=1210 THEN 570
550 PRINT "*** TEST FAILED ***"
560 GOTO 580
570 PRINT "*** TEST PASSED ***"
580 PRINT
590 PRINT "                               END TEST."
600 PRINT
640 PRINT "SECTION 56.3: ONE-DIMENSIONAL ARRAYS WITH THE"
645 PRINT "           DIM-STATEMENT."
650 PRINT
660 PRINT "                               BEGIN TEST."
670 PRINT
680 DIM D(20)
690 FOR I=0 TO 20
700 LET H=I+1
710 LET D(I)=H
720 NEXT I
730 LET D1=0
740 LET C=0
750 FOR I=0 TO 20
760 LET M=D(I)
770 LET D1=D1+M
780 LET H=I+1
790 IF M<>H THEN 810
800 LET C=C+1
810 NEXT I
```

```
820 IF C<>21 THEN 840
830 IF D1=231 THEN 860
840 PRINT "**** TEST FAILED ****"
850 GOTO 870
860 PRINT "**** TEST PASSED ****"
870 PRINT
880 PRINT " END TEST."
890 PRINT
900 PRINT
910 PRINT "SECTION 56.4: TWO-DIMENSIONAL ARRAYS WITH THE"
915 PRINT " DIM-STATEMENT."
920 PRINT
930 PRINT " BEGIN TEST."
940 PRINT
950 DIM N(20,2),P(2,20),R(20,20)
960 FOR I=0 TO 20
970 LET I1=I+1
980 LET N(I,1)=I1
990 LET N(I,2)=-I
1000 LET N(I,0)=1
1010 FOR J=2 TO 0 STEP -1
1020 LET J1=I*j
1030 LET P(J,I)=J1
1040 NEXT J
1050 NEXT I
1060 FOR K=0 TO 20
1070 FOR L=0 TO 20
1080 LET K1=K+L
1090 LET R(K,L)=K1
1100 NEXT L
1110 NEXT K
1120 LET N1=0
1130 LET P1=0
1140 LET R1=0
1150 LET C=0
1160 LET T=0
1170 FOR I=0 TO 20
1180 LET I1=I+1
1190 LET T7=N(I,1)
1200 LET T8=N(I,2)
1210 LET T9=N(I,0)
1220 IF T7 <> I1 THEN 1490
1230 IF T8 <> -I THEN 1490
1240 IF T9 <> 1 THEN 1490
1250 FOR J=0 TO 2
1260 LET F1=N(I,J)
1270 LET N1=N1+F1
1280 LET G1=P(J,I)
1290 LET P1=P1+G1
1300 LET J1=I*j
1310 IF G1 <> J1 THEN 1330
1320 LET C=C+1
1330 NEXT J
1340 NEXT I
1350 IF C<>63 THEN 1490
```

```
1360 IF N1<>42 THEN 1490
1370 IF P1<>630 THEN 1490
1380 FOR K=20 TO 0 STEP -1
1390 FOR L=0 TO 20
1400 LET Q1=R(K,L)
1410 LET R1=R1+Q1
1420 LET K1=K+L
1430 IF Q1 <> K1 THEN 1450
1440 LET T=T+1
1450 NEXT L
1460 NEXT K
1470 IF T<>441 THEN 1490
1480 IF R1=8820 THEN 1510
1490 PRINT "*** TEST FAILED ***"
1500 GOTO 1520
1510 PRINT "*** TEST PASSED ***"
1520 PRINT
1530 PRINT "END TEST."
1540 PRINT
1550 PRINT
1560 PRINT "END PROGRAM 56"
1570 END
```

---

PROGRAM FILE 56: ARRAY ASSIGNMENT WITHOUT  
THE OPTION-STATEMENT.

ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 15.2, 15.4

SECTION 56.1: ONE-DIMENSIONAL ARRAYS WITHOUT THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 56.2: TWO-DIMENSIONAL ARRAYS WITHOUT THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 56.3: ONE-DIMENSIONAL ARRAYS WITH THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 56.4: TWO-DIMENSIONAL ARRAYS WITH THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 56

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 57: ARRAY ASSIGNMENT WITH OPTION BASE 0."
20 PRINT "      ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 15.2, 15.4"
30 PRINT
50 OPTION BASE 0
90 PRINT "SECTION 57.1: ONE-DIMENSIONAL ARRAYS WITHOUT THE"
95 PRINT "      DIM-STATEMENT."
100 PRINT
110 PRINT "                                BEGIN TEST."
120 PRINT
130 FOR I=0 TO 10
140 LET A(I)=I
150 NEXT I
160 LET A1=0
170 LET C=0
180 FOR I=10 TO 0 STEP -1
190 LET Y=A(I)
200 LET A1=A1+Y
210 IF Y<>I THEN 230
220 LET C=C+1
230 NEXT I
240 IF C<>11 THEN 260
250 IF A1=55 THEN 280
260 PRINT "*** TEST FAILED ***"
270 GOTO 290
280 PRINT "*** TEST PASSED ***"
290 PRINT
300 PRINT "                                END TEST."
310 PRINT
320 PRINT "SECTION 57.2: TWO-DIMENSIONAL ARRAYS WITHOUT THE"
325 PRINT "      DIM-STATEMENT."
330 PRINT
340 PRINT "                                BEGIN TEST."
350 PRINT
360 FOR I=0 TO 10
370 FOR J=0 TO 10
380 LET Z=I+J
390 LET B(I,J)=Z
400 NEXT J
410 NEXT I
```

```
420 LET B1=0
430 LET C=0
440 FOR I=10 TO 0 STEP -1
450 FOR J=10 TO 0 STEP -1
460 LET W=B(I,J)
470 LET B1=B1+W
480 LET Z=I+J
490 IF W<>Z THEN 510
500 LET C=C+1
510 NEXT J
520 NEXT I
530 IF C<>121 THEN 550
540 IF B1=1210 THEN 570
550 PRINT "*** TEST FAILED ***"
560 GOTO 580
570 PRINT "*** TEST PASSED ***"
580 PRINT
590 PRINT "                                END TEST."
600 PRINT
640 PRINT "SECTION 57.3: ONE-DIMENSIONAL ARRAYS WITH THE"
645 PRINT "                  DIM-STATEMENT."
650 PRINT
660 PRINT "                                BEGIN TEST."
670 PRINT
680 DIM D(20)
690 FOR I=0 TO 20
700 LET H=I+1
710 LET D(I)=H
720 NEXT I
730 LET D1=0
740 LET C=0
750 FOR I=0 TO 20
760 LET M=D(I)
770 LET D1=D1+M
780 LET H=I+1
790 IF M<>H THEN 810
800 LET C=C+1
810 NEXT I
820 IF C<>21 THEN 840
830 IF D1=231 THEN 860
840 PRINT "*** TEST FAILED ***"
850 GOTO 870
860 PRINT "*** TEST PASSED ***"
870 PRINT
880 PRINT "                                END TEST."
890 PRINT
900 PRINT
910 PRINT "SECTION 57.4: TWO-DIMENSIONAL ARRAYS WITH THE"
915 PRINT "                  DIM-STATEMENT."
920 PRINT
930 PRINT "                                BEGIN TEST."
940 PRINT
```

```
950 DIM N(20,2),P(2,20),R(20,20)
960 FOR I=0 TO 20
970 LET I1=I+1
980 LET N(I,1)=I1
990 LET N(I,2)=-I
1000 LET N(I,0)=1
1010 FOR J=2 TO 0 STEP -1
1020 LET J1=I*j
1030 LET P(J,I)=J1
1040 NEXT J
1050 NEXT I
1060 FOR K=0 TO 20
1070 FOR L=0 TO 20
1080 LET K1=K+L
1090 LET R(K,L)=K1
1100 NEXT L
1110 NEXT K
1120 LET N1=0
1130 LET P1=0
1140 LET R1=0
1150 LET C=0
1160 LET T=0
1170 FOR I=0 TO 20
1180 LET I1=I+1
1190 LET T7=N(I,1)
1200 LET T8=N(I,2)
1210 LET T9=N(I,0)
1220 IF T7 <> I1 THEN 1490
1230 IF T8 <> -I THEN 1490
1240 IF T9 <> 1 THEN 1490
1250 FOR J=0 TO 2
1260 LET F1=N(I,J)
1270 LET N1=N1+F1
1280 LET G1=P(J,I)
1290 LET P1=P1+G1
1300 LET J1=I*j
1310 IF G1 <> J1 THEN 1330
1320 LET C=C+1
1330 NEXT J
1340 NEXT I
1350 IF C<>63 THEN 1490
1360 IF N1<>42 THEN 1490
1370 IF P1<>630 THEN 1490
1380 FOR K=20 TO 0 STEP -1
1390 FOR L=0 TO 20
1400 LET Q1=R(K,L)
1410 LET R1=R1+Q1
1420 LET K1=K+L
1430 IF Q1 <> K1 THEN 1450
1440 LET T=T+1
1450 NEXT L
1460 NEXT K
1470 IF T<>441 THEN 1490
```

```
1480 IF R1=8820 THEN 1510
1490 PRINT "*** TEST FAILED ***"
1500 GOTO 1520
1510 PRINT "*** TEST PASSED ***"
1520 PRINT
1530 PRINT "                                END TEST."
1540 PRINT
1550 PRINT
1560 PRINT "END PROGRAM 57"
1570 END
```

---

PROGRAM FILE 57: ARRAY ASSIGNMENT WITH OPTION BASE 0.

ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 15.2, 15.4

SECTION 57.1: ONE-DIMENSIONAL ARRAYS WITHOUT THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 57.2: TWO-DIMENSIONAL ARRAYS WITHOUT THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 57.3: ONE-DIMENSIONAL ARRAYS WITH THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 57.4: TWO-DIMENSIONAL ARRAYS WITH THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 57

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 58: ARRAY ASSIGNMENT WITH OPTION BASE 1."  
20 PRINT "      ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 15.2, 15.4"  
30 PRINT  
45 GOTO 60  
50 OPTION BASE 1  
60 PRINT  
100 PRINT "SECTION 58.1: ONE-DIMENSIONAL ARRAYS WITHOUT THE"  
105 PRINT "      DIM-STATEMENT."  
110 PRINT  
120 PRINT "                                BEGIN TEST."  
130 PRINT  
140 FOR I=1 TO 10  
150 LET A(I)=I  
160 NEXT I  
170 LET A1=0  
180 LET C=0  
190 FOR I=10 TO 1 STEP -1  
200 LET Y=A(I)  
210 LET A1=A1+Y  
220 IF Y<>I THEN 240  
230 LET C=C+1  
240 NEXT I  
250 IF C<>10 THEN 270  
260 IF A1=55 THEN 290  
270 PRINT "*** TEST FAILED ***"  
280 GOTO 300  
290 PRINT "*** TEST PASSED ***"  
300 PRINT  
310 PRINT "                                END TEST."  
320 PRINT  
330 PRINT "SECTION 58.2: TWO-DIMENSIONAL ARRAYS WITHOUT THE "  
335 PRINT "      DIM-STATEMENT."  
340 PRINT  
350 PRINT "                                BEGIN TEST."  
360 PRINT  
370 FOR I=1 TO 10  
380 FOR J=1 TO 10  
390 LET Z=I+J  
400 LET B(I,J)=Z  
410 NEXT J  
420 NEXT I  
430 LET B1=0  
440 LET C=0  
450 FOR I=10 TO 1 STEP -1  
460 FOR J=10 TO 1 STEP -1  
470 LET W=B(I,J)  
480 LET B1=B1+W
```

```
490 LET Z=I+J
500 IF W>Z THEN 520
510 LET C=C+1
520 NEXT J
530 NEXT I
540 IF C<>100 THEN 560
550 IF B1=1100 THEN 580
560 PRINT "*** TEST FAILED ***"
570 GOTO 590
580 PRINT "*** TEST PASSED ***"
590 PRINT
600 PRINT " END TEST."
610 PRINT
650 PRINT "SECTION 58.3: ONE-DIMENSIONAL ARRAYS WITH THE"
655 PRINT " DIM-STATEMENT."
660 PRINT
670 PRINT " BEGIN TEST."
680 PRINT
690 DIM E(1)
700 LET E(1)=6
710 LET T9=E(1)
720 IF 6<>T9 THEN 890
730 DIM D(20)
740 FOR I=1 TO 20
750 LET H=I+1
760 LET D(I)=H
770 NEXT I
780 LET D1=0
790 LET C=0
800 FOR I=1 TO 20
810 LET M=D(I)
820 LET D1=D1+M
830 LET H=I+1
840 IF M<>H THEN 860
850 LET C=C+1
860 NEXT I
870 IF C<>20 THEN 890
880 IF D1=230 THEN 910
890 PRINT "*** TEST FAILED ***"
900 GOTO 920
910 PRINT "*** TEST PASSED ***"
920 PRINT
930 PRINT " END TEST."
940 PRINT
950 PRINT
960 PRINT "SECTION 58.4: TWO-DIMENSIONAL ARRAYS WITH THE"
965 PRINT " DIM-STATEMENT."
970 PRINT
980 PRINT " BEGIN TEST."
990 PRINT
```

```
1000 DIM N(20,2),P(2,20),R(20,20)
1010 FOR I=1 TO 20
1020 LET I1=I+1
1030 LET N(I,1)=I1
1040 LET N(I,2)=-I
1050 FOR J=2 TO 1 STEP -1
1060 LET J1=I*j
1070 LET P(J,I)=J1
1080 NEXT J
1090 NEXT I
1100 FOR K=1 TO 20
1110 FOR L=1 TO 20
1120 LET K1=K+L
1130 LET R(K,L)=K1
1140 NEXT L
1150 NEXT K
1160 LET N1=0
1170 LET P1=0
1180 LET R1=0
1190 LET C=0
1200 LET T=0
1210 FOR I=1 TO 20
1220 LET I1=I+1
1230 LET T8=N(I,1)
1240 LET T9=N(I,2)
1250 IF T8 <> I1 THEN 1510
1260 IF T9 <> -I THEN 1510
1270 FOR J=1 TO 2
1280 LET F1=N(I,J)
1290 LET N1=N1+F1
1300 LET G1=P(J,I)
1310 LET P1=P1+G1
1320 LET J1=I*j
1330 IF G1 <> J1 THEN 1350
1340 LET C=C+1
1350 NEXT J
1360 NEXT I
1370 IF C<>40 THEN 1510
1380 IF N1<>20 THEN 1510
1390 IF P1<>630 THEN 1510
1400 FOR K=20 TO 1 STEP -1
1410 FOR L=1 TO 20
1420 LET Q1=R(K,L)
1430 LET R1=R1+Q1
1440 LET K1=K+L
1450 IF Q1 <> K1 THEN 1470
1460 LET T=T+1
1470 NEXT L
1480 NEXT K
1490 IF T<>400 THEN 1510
1500 IF R1=8400 THEN 1530
1510 PRINT "*** TEST FAILED ***"
1520 GOTO 1540
1530 PRINT "*** TEST PASSED ***"
1540 PRINT
```

```
1550 PRINT "END TEST."
1560 PRINT
1570 PRINT
1580 PRINT "END PROGRAM 58"
1590 END
```

---

PROGRAM FILE 58: ARRAY ASSIGNMENT WITH OPTION BASE 1.  
ANSI STANDARD 6.2, 6.4, 9.2, 9.4, 15.2, 15.4

SECTION 58.1: ONE-DIMENSIONAL ARRAYS WITHOUT THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 58.2: TWO-DIMENSIONAL ARRAYS WITHOUT THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 58.3: ONE-DIMENSIONAL ARRAYS WITH THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 58.4: TWO-DIMENSIONAL ARRAYS WITH THE  
DIM-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 58

```
*****
*****  
10 PRINT "PROGRAM FILE 59: ARRAY NAMED 'A' IS DISTINCT FROM 'A$'."  
20 PRINT "    ANSI STANDARD 6.2, 6.4"  
30 PRINT  
40 PRINT "SECTION 59.1: ARRAY NAMED 'A' IS DISTINCT FROM 'A$'."  
50 PRINT  
60 PRINT "                BEGIN TEST"  
70 PRINT  
80 FOR J=0 TO 10  
90 LET A(J)=11-J  
100 NEXT J  
110 LET E9=0  
120 LET A$="OK"  
130 FOR I=10 TO 0 STEP -1  
140 IF A(I)=11-I THEN 170  
150 LET E9=1  
160 PRINT "LOST CORRECT VALUE IN ARRAY ELEMENT #";I  
170 NEXT I  
180 IF A$="OK" THEN 210  
190 PRINT "LOST CORRECT VALUE IN A$"  
200 LET E9=1  
210 PRINT  
220 IF E9<>0 THEN 250  
230 PRINT "*** TEST PASSED ***"  
240 GOTO 260  
250 PRINT "*** TEST FAILED ***"  
260 PRINT  
270 PRINT "                END TEST"  
280 PRINT  
290 PRINT "END PROGRAM 59"  
340 END
```

---

PROGRAM FILE 59: ARRAY NAMED 'A' IS DISTINCT FROM 'A\$'.  
ANSI STANDARD 6.2, 6.4

SECTION 59.1: ARRAY NAMED 'A' IS DISTINCT FROM 'A\$'.

BEGIN TEST

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 59

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 60: NUMERIC CONSTANTS USED AS SUBSCRIPTS"  
20 PRINT "          ARE ROUNDED TO NEAREST INTEGER."  
30 PRINT "          ANSI STANDARD 6.4, 5.4"  
40 PRINT  
50 PRINT "SECTION 60.2: NUMERIC CONSTANTS USED AS SUBSCRIPTS"  
60 PRINT "          ARE ROUNDED TO NEAREST INTEGER."  
70 PRINT  
80 PRINT "          BEGIN TEST"  
90 PRINT  
100 FOR I=0 TO 10  
110 LET A(I)=11-I  
120 NEXT I  
130 LET Y$="3"  
140 LET X$="3"  
150 IF A(3)<>8 THEN 280  
160 LET X$="2.50001"  
170 IF A(2.50001)<>8 THEN 280  
180 LET X$="3.49999"  
190 IF A(3.49999)<>8 THEN 280  
200 LET Y$="0"  
210 LET X$="-.49"  
220 IF A(-.49) <> 11 THEN 280  
230 LET Y$="10"  
240 LET X$="10.49"  
250 IF A(10.49)<>1 THEN 280  
260 PRINT " *** TEST PASSED *** "  
270 GOTO 300  
280 PRINT "SUBSCRIPT OF ";X$;" DID NOT EVALUATE TO ";Y$  
290 PRINT " *** TEST FAILED *** "  
300 PRINT  
310 PRINT "          END TEST"  
320 PRINT  
330 PRINT "END PROGRAM 60"  
340 END
```

PROGRAM FILE 60: NUMERIC CONSTANTS USED AS SUBSCRIPTS  
ARE ROUNDED TO NEAREST INTEGER.  
ANSI STANDARD 6.4, 5.4

SECTION 60.2: NUMERIC CONSTANTS USED AS SUBSCRIPTS  
ARE ROUNDED TO NEAREST INTEGER.

BEGIN TEST

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 60

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 61: NUMERIC EXPRESSIONS CONTAINING"  
20 PRINT "      SUBSCRIPTED VARIABLES."  
30 PRINT "      ANSI STANDARD 6.2, 6.4, 7.2, 7.4"  
40 PRINT  
50 PRINT "SECTION 61.1: NUMERIC EXPRESSIONS CONTAINING"  
60 PRINT "      SUBSCRIPTED VARIABLES."  
70 PRINT  
80 PRINT "THIS SECTION TESTS WHETHER SUBSCRIPTED VARIABLES"  
90 PRINT "CAN BE USED AS PRIMARIES IN NUMERIC EXPRESSIONS."  
100 PRINT  
110 PRINT "      BEGIN TEST"  
120 PRINT  
130 PRINT "CASE #","SHOULD BE","ACTUAL","OUTCOME"  
140 PRINT  
150 LET F=0  
160 LET C1=0  
180 REM ARRAY B(I) CONTAINS FIBONACCI SERIES  
200 LET B(1)=+001.00  
210 LET B(2)=1  
220 FOR I=3 TO 10  
230 LET J=I-1  
240 LET K=I-2  
250 LET B(I)=B(J)+B(K)  
260 NEXT I  
270 LET A=B(5)+B(8)  
280 LET Y=26  
290 GOSUB 9000  
300 LET C(7)=47.23/B(7)+23.E-1  
310 LET H9=7.0  
320 LET A=C(H9)+H9*B(H9)  
330 LET Y=96.933076923  
340 GOSUB 9000  
350 LET H9=6  
360 LET H9=B(H9)  
370 LET H9=B(H9)  
380 LET Z(0)=(H9/7)^B(4)*20/(B(7)-B(4))-B(10)-.02E+02  
390 LET Z(1)=Z(0)+Z(0)  
400 LET Z(2)=-Z(0)-Z(0)  
410 LET Z(3)=Z(0)*Z(0)  
420 LET Z(4)=Z(0)/Z(0)  
430 LET Z(5)=Z(0)^Z(0)  
440 LET A=Z(0)+Z(1)  
450 LET Y=-9  
460 GOSUB 9000  
470 LET A=Z(2)+Z(3)+Z(4)+Z(5)  
480 LET Y=15.962962963  
490 GOSUB 9000  
500 LET A=-2000.00E-03-B(10)-(B(4)-B(7))/2E+1*B(4)^(H9/7)  
510 LET Y=-43.5
```

```

520 GOSUB 9000
530 FOR I=1 TO 7
540 FOR J=1 TO 7
550 LET Q(I,J)=I/J
560 NEXT J
570 NEXT I
580 LET A=Q(6,3)+Q(6,7)*Q(5,4)*(-Q(2,3))*Q(7,2)/Q(5,6)/Q(3,4)
590 LET Y=-2
600 GOSUB 9000
2000 PRINT
2010 IF F=0 THEN 2040
2020 PRINT "**** TEST FAILED IN ";F;" CASE(S). ****"
2030 GOTO 2050
2040 PRINT "**** TEST PASSED ****"
2050 PRINT
2060 PRINT " END TEST"
2070 PRINT
2080 PRINT "END PROGRAM 61"
2090 STOP
9000 REM HANDLE REPORT
9002 REM THIS SUBROUTINE ADOPTS A VERY LOOSE CRITERION OF CORRECTNESS
9004 REM (RELATIVE ERROR < .01). ITS PURPOSE IS NOT TO MEASURE ACCURACY
9006 REM BUT ONLY TO ASSURE THAT THE SEMANTICS OF THE EXPRESSION
9008 REM HAVE BEEN CORRECTLY IMPLEMENTED.
9010 LET C1=C1+1
9020 PRINT C1,Y,A,
9030 LET P$="FAILS"
9040 LET M=0.01
9050 LET T=(A-Y)/Y
9060 IF T>M THEN 9110
9070 LET T=-T
9080 IF T>M THEN 9110
9090 LET P$="PASSES"
9100 GOTO 9120
9110 LET F=F+1
9120 PRINT "TEST ";P$
9130 RETURN
9140 END

```

---

PROGRAM FILE 61: NUMERIC EXPRESSIONS CONTAINING  
SUBSCRIPTED VARIABLES.

ANSI STANDARD 6.2, 6.4, 7.2, 7.4

SECTION 61.1: NUMERIC EXPRESSIONS CONTAINING  
SUBSCRIPTED VARIABLES.

THIS SECTION TESTS WHETHER SUBSCRIPTED VARIABLES  
CAN BE USED AS PRIMARIES IN NUMERIC EXPRESSIONS.

BEGIN TEST

CASE #	SHOULD BE	ACTUAL	OUTCOME
--------	-----------	--------	---------

1	26	26	TEST PASSES
2	96.9331	96.9331	TEST PASSES
3	-9	-9	TEST PASSES
4	15.963	15.963	TEST PASSES
5	-43.5	-43.5	TEST PASSES
6	-2	-2.	TEST PASSES

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 61

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 62: GENERAL SYNTACTIC AND SEMANTIC"
20 PRINT "      PROPERTIES OF ARRAY CONTROL STATEMENTS:"
30 PRINT "      OPTION AND DIM."
40 PRINT "      ANSI STANDARD 15.2, 15.4"
50 PRINT
60 PRINT "SECTION 62.1: GENERAL SYNTACTIC AND SEMANTIC"
70 PRINT "      PROPERTIES OF ARRAY CONTROL STATEMENTS:"
80 PRINT "      OPTION AND DIM."
90 PRINT
100 PRINT "THIS SECTION ASSURES THAT OPTION-STATEMENTS MAY PRECEDE"
110 PRINT "DIM-STATEMENTS, THAT REFERENCES TO IMPLICITLY-DIMENSIONED"
120 PRINT "ARRAYS MAY ALSO PRECEDE DIM-STATEMENTS, AND THAT FLOW"
130 PRINT "OF CONTROL MAY PASS THROUGH OPTION AND DIM STATEMENTS."
140 PRINT
150 PRINT "      BEGIN TEST"
160 LET F=0
170 LET T=1
180 OPTION BASE 1
190 REM CAN FALL THROUGH OR JUMP TO OPTION WITH NO EFFECT
200 IF T<>1 THEN 230
210 LET T=2
220 GOTO 180
230 LET A(1)=1
240 REM CAN REFERENCE ARRAY BEFORE ANY DIM-STATEMENT
250 FOR I=2 TO 10
260 LET J=I-1
270 LET A(I)=A(J)+A(J)
280 NEXT I
290 IF A(10)=512 THEN 320
300 LET M$="IMPLICIT ARRAY"
310 GOSUB 1000
320 LET T=1
330 DIM D(3),C(2,4),B(12)
340 REM CAN DIMENSION SEVERAL ARRAYS IN NON-ALPHA ORDER AND
350 REM OF DIFFERENT DIMENSIONALITY
360 REM CAN FALL THROUGH OR JUMP TO DIM
370 IF T<>1 THEN 400
380 LET T=2

```

```

390 GOTO 330
400 LET B(12)=A(10)
410 FOR I=11 TO 1 STEP -1
420 LET J=I+1
430 LET B(I)=B(J)
440 NEXT I
450 IF B(1)=512 THEN 480
460 LET M$="EXPLICIT ARRAY"
470 GOSUB 1000
480 GOTO 510
490 DIM E(14)
500 REM DIM SETS ARRAY SIZE EVEN IF JUMPED OVER
510 LET A(10)=666
520 LET B(12)=A(10)
530 LET C(2,4)=B(12)
540 LET D(3)=C(2,4)
550 LET E(14)=D(3)
560 IF E(14)=666 THEN 590
570 LET M$="SKIPPED-OVER DIM"
580 GOSUB 1000
590 PRINT
600 IF F=0 THEN 630
610 PRINT " *** TEST FAILED IN ";F;" CASE(S) *** "
620 GOTO 640
630 PRINT " *** TEST PASSED *** "
640 PRINT
650 PRINT " END TEST"
660 PRINT
670 PRINT "END PROGRAM 62"
680 STOP
1000 PRINT "FAILURE BECAUSE OF: ";M$
1010 LET F=F+1
1020 RETURN
1030 END

```

---

PROGRAM FILE 62: GENERAL SYNTACTIC AND SEMANTIC  
 PROPERTIES OF ARRAY CONTROL STATEMENTS:  
 OPTION AND DIM.  
 ANSI STANDARD 15.2, 15.4

SECTION 62.1: GENERAL SYNTACTIC AND SEMANTIC  
 PROPERTIES OF ARRAY CONTROL STATEMENTS:  
 OPTION AND DIM.

THIS SECTION ASSURES THAT OPTION-STATEMENTS MAY PRECEDE  
 DIM-STATEMENTS, THAT REFERENCES TO IMPLICITLY-DIMENSIONED  
 ARRAYS MAY ALSO PRECEDE DIM-STATEMENTS, AND THAT FLOW  
 OF CONTROL MAY PASS THROUGH OPTION AND DIM STATEMENTS.

BEGIN TEST

\*\*\* TEST PASSED \*\*\*

```
END TEST
```

```
END PROGRAM 62
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 63: EXCEPTION - SUBSCRIPT TOO LARGE FOR"  
15 PRINT "      ONE-DIMENSIONAL ARRAY."  
20 PRINT "      ANSI STANDARD 6.5"  
30 PRINT  
40 PRINT "SECTION 63.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR"  
50 PRINT "      ONE-DIMENSIONAL ARRAY."  
60 PRINT  
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"  
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS"  
90 PRINT "THE IMPLICIT UPPER BOUND OF AN ARRAY."  
100 PRINT  
110 PRINT "TO PASS THIS TEST:"  
120 PRINT  
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
140 PRINT "          DISPLAYED, AND"  
150 PRINT  
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL"  
170 PRINT "          SUBSCRIPT."  
180 PRINT  
190 PRINT "                                BEGIN TEST."  
200 PRINT  
210 FOR I=7 TO 12  
220 PRINT "ABOUT TO ASSIGN TO A(";I;"). ";  
230 IF I<11 THEN 260  
240 PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"  
250 GOTO 270  
260 PRINT "ASSIGNMENT SHOULD BE OK."  
270 LET A(I) = 20 - I  
280 PRINT "HAVE ASSIGNED ";20-I;" TO A(";I;")."  
290 IF I<11 THEN 310  
300 PRINT "ASSIGNMENT TO SUBSCRIPT > 10 - *** TEST FAILS ***"  
310 PRINT  
320 NEXT I  
330 PRINT  
340 PRINT "                                END TEST."  
350 PRINT  
360 PRINT "END PROGRAM 63"  
370 END
```

---

```
PROGRAM FILE 63: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
      ONE-DIMENSIONAL ARRAY.  
      ANSI STANDARD 6.5
```

SECTION 63.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
ONE-DIMENSIONAL ARRAY.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS  
THE IMPLICIT UPPER BOUND OF AN ARRAY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO A( 7 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 13 TO A( 7 ).

ABOUT TO ASSIGN TO A( 8 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 12 TO A( 8 ).

ABOUT TO ASSIGN TO A( 9 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 11 TO A( 9 ).

ABOUT TO ASSIGN TO A( 10 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 10 TO A( 10 ).

ABOUT TO ASSIGN TO A( 11 ). \*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 270

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 64: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
15 PRINT "      TWO-DIMENSIONAL ARRAY."
20 PRINT "      ANSI STANDARD 6.5"
30 PRINT
40 PRINT "SECTION 64.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
50 PRINT "      TWO-DIMENSIONAL ARRAY."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS"
90 PRINT "THE IMPLICIT LOWER BOUND OF AN ARRAY."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
```

```

160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL"
170 PRINT "          SUBSCRIPT."
180 PRINT
190 PRINT "                      BEGIN TEST."
200 PRINT
210 FOR I=3 TO -2 STEP -1
220   PRINT "ABOUT TO ASSIGN TO B(7,";I;").";
230   IF I>-1 THEN 260
240   PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
250 GOTO 270
260 PRINT " ASSIGNMENT SHOULD BE OK."
270 LET B(7,I) = 20 - I
280 PRINT "HAVE ASSIGNED ";20-I;" TO B(7,";I;")."
290 IF I>-1 THEN 310
300 PRINT "ASSIGNMENT TO SUBSCRIPT < 0 - *** TEST FAILS ***"
310 PRINT
320 NEXT I
330 PRINT
340 PRINT "                      END TEST."
350 PRINT
360 PRINT "END PROGRAM 64"
370 END

```

---

PROGRAM FILE 64: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
TWO-DIMENSIONAL ARRAY.

ANSI STANDARD 6.5

SECTION 64.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
TWO-DIMENSIONAL ARRAY.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS  
THE IMPLICIT LOWER BOUND OF AN ARRAY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO B(7, 3 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 17 TO B(7, 3 ).

ABOUT TO ASSIGN TO B(7, 2 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 18 TO B(7, 2 ).

ABOUT TO ASSIGN TO B(7, 1 ). ASSIGNMENT SHOULD BE OK  
HAVE ASSIGNED 19 TO B(7, 1 ).

ABOUT TO ASSIGN TO B(7, 0). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 20 TO B(7, 0).

ABOUT TO ASSIGN TO B(7,-1).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 270

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 65: EXCEPTION - SUBSCRIPT TOO SMALL FOR"  
15 PRINT "          ONE-DIMENSIONAL ARRAY, WITH DIM."  
20 PRINT "          ANSI STANDARD 6.5, 15.2, 15.4"  
30 PRINT  
40 PRINT "SECTION 65.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR"  
50 PRINT "          ONE-DIMENSIONAL ARRAY, WITH DIM."  
60 PRINT  
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"  
80 PRINT "RECOGNIZES USE OF A NEGATIVE SUBSCRIPT AS A FATAL EXCEPTION"  
90 PRINT "WHEN NO OPTION HAS BEEN SPECIFIED."  
100 PRINT  
110 PRINT "TO PASS THIS TEST:"  
120 PRINT  
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
140 PRINT "          DISPLAYED, AND"  
150 PRINT  
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE"  
170 PRINT "          NEGATIVE SUBSCRIPT."  
180 PRINT  
190 PRINT "                                BEGIN TEST."  
200 PRINT  
210 DIM A(8)  
220 FOR I=3 TO -2 STEP -1  
230 PRINT "ABOUT TO ASSIGN TO A(";I;").";  
240 IF I>-1 THEN 270  
250 PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"  
260 GOTO 280  
270 PRINT " ASSIGNMENT SHOULD BE OK."  
280 LET A(I) = 20 - I  
290 PRINT "HAVE ASSIGNED ";20-I;" TO A(";I;")."  
300 IF I>-1 THEN 320  
310 PRINT "ASSIGNMENT TO SUBSCRIPT < 1 - *** TEST FAILS ***"  
320 PRINT  
330 NEXT I  
340 PRINT  
350 PRINT "                                END TEST."  
360 PRINT  
370 PRINT "END PROGRAM 65"  
380 END
```

PROGRAM FILE 65: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
ONE-DIMENSIONAL ARRAY, WITH DIM.  
ANSI STANDARD 6.5, 15.2, 15.4

SECTION 65.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
ONE-DIMENSIONAL ARRAY, WITH DIM.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES USE OF A NEGATIVE SUBSCRIPT AS A FATAL EXCEPTION  
WHEN NO OPTION HAS BEEN SPECIFIED.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE  
NEGATIVE SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO A( 3 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 17 TO A( 3 ).

ABOUT TO ASSIGN TO A( 2 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 18 TO A( 2 ).

ABOUT TO ASSIGN TO A( 1 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 19 TO A( 1 ).

ABOUT TO ASSIGN TO A( 0 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 20 TO A( 0 ).

ABOUT TO ASSIGN TO A(-1 ).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 280

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 66: EXCEPTION - SUBSCRIPT TOO LARGE FOR"  
15 PRINT "      TWO-DIMENSIONAL ARRAY, WITH DIM."  
20 PRINT "      ANSI STANDARD 6.5, 15.2, 15.4"  
30 PRINT  
40 PRINT "SECTION 66.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR"  
50 PRINT "      TWO-DIMENSIONAL ARRAY, WITH DIM."  
60 PRINT  
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"  
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS"  
90 PRINT "THE EXPLICIT UPPER BOUND OF AN ARRAY."  
100 PRINT  
110 PRINT "TO PASS THIS TEST:"  
120 PRINT
```

```
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
140 PRINT "          DISPLAYED, AND"  
150 PRINT  
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL"  
170 PRINT "          SUBSCRIPT."  
180 PRINT  
190 PRINT "                                BEGIN TEST."  
200 PRINT  
210 DIM B(3,12)  
220 FOR I=9 TO 14  
230   PRINT "ABOUT TO ASSIGN TO B(0,";I;").";  
240   IF I<13 THEN 270  
250   PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"  
260   GOTO 280  
270   PRINT " ASSIGNMENT SHOULD BE OK."  
280   LET B(0,I) = 20 - I  
290   PRINT "HAVE ASSIGNED ";20-I;" TO B(0,";I;")."  
300   IF I<13 THEN 320  
310   PRINT "ASSIGNMENT TO SUBSCRIPT > 12 - *** TEST FAILS ***"  
320   PRINT  
330 NEXT I  
340 PRINT  
350 PRINT "                                END TEST."  
360 PRINT  
370 PRINT "END PROGRAM 66"  
380 END
```

---

PROGRAM FILE 66: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM.  
ANSI STANDARD 6.5, 15.2, 15.4

SECTION 66.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS  
THE EXPLICIT UPPER BOUND OF AN ARRAY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO B(0, 9 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 11 TO B(0, 9 ).

ABOUT TO ASSIGN TO B(0, 10). ASSIGNMENT SHOULD BE OK.  
 HAVE ASSIGNED 10 TO B(0, 10).

ABOUT TO ASSIGN TO B(0, 11). ASSIGNMENT SHOULD BE OK.  
 HAVE ASSIGNED 9 TO B(0, 11).

ABOUT TO ASSIGN TO B(0, 12). ASSIGNMENT SHOULD BE OK.  
 HAVE ASSIGNED 8 TO B(0, 12).

ABOUT TO ASSIGN TO B(0, 13).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 280

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

```

10 PRINT "PROGRAM FILE 67: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
15 PRINT "      ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 1."
20 PRINT "      ANSI STANDARD 6.5, 15.2, 15.4"
30 PRINT
40 PRINT "SECTION 67.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
50 PRINT "      ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 1."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES USE OF A ZERO SUBSCRIPT AS A FATAL EXCEPTION WHEN"
90 PRINT "OPTION BASE 1 HAS BEEN SPECIFIED."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ZERO"
170 PRINT "          SUBSCRIPT."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 OPTION BASE 1
220 FOR I=3 TO -2 STEP -1
230   PRINT "ABOUT TO ASSIGN TO A(";I;").";
240   IF I>0 THEN 270
250   PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
260   GOTO 280
270   PRINT " ASSIGNMENT SHOULD BE OK."
280   LET A(I) = 20 - I
290   PRINT "HAVE ASSIGNED ";20-I;" TO A(";I;")."
300   IF I>0 THEN 320
310   PRINT "ASSIGNMENT TO SUBSCRIPT < 1 - *** TEST FAILS ***"
320   PRINT
330 NEXT I
340 PRINT
350 PRINT "                                END TEST."
```

```
360 PRINT  
370 PRINT "END PROGRAM 67"  
380 END
```

---

PROGRAM FILE 67: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 1.  
ANSI STANDARD 6.5, 15.2, 15.4

SECTION 67.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 1.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES USE OF A ZERO SUBSCRIPT AS A FATAL EXCEPTION WHEN  
OPTION BASE 1 HAS BEEN SPECIFIED.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ZERO  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO A( 3 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 17 TO A( 3 ).

ABOUT TO ASSIGN TO A( 2 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 18 TO A( 2 ).

ABOUT TO ASSIGN TO A( 1 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 19 TO A( 1 ).

ABOUT TO ASSIGN TO A( 0 ).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 280

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 68: EXCEPTION - SUBSCRIPT TOO LARGE FOR"  
15 PRINT "          ONE-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1."  
20 PRINT "          ANSI STANDARD 6.5, 15.2, 15.4"  
30 PRINT  
40 PRINT "SECTION 68.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR"  
50 PRINT "          ONE-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1."  
60 PRINT
```

```

70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS"
90 PRINT "THE EXPLICIT UPPER BOUND OF AN ARRAY, WITH AN OPTION"
100 PRINT "STATEMENT PRESENT."
110 PRINT
120 PRINT "TO PASS THIS TEST:"
130 PRINT
140 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
150 PRINT "          DISPLAYED, AND"
160 PRINT
170 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL"
180 PRINT "          SUBSCRIPT."
190 PRINT
200 PRINT "                                BEGIN TEST."
210 OPTION BASE 1
220 DIM A(7)
230 PRINT
240 FOR I=5 TO 9
250   PRINT "ABOUT TO ASSIGN TO A(";I;"). ";
260   IF I<8 THEN 290
270   PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
280   GOTO 300
290   PRINT "ASSIGNMENT SHOULD BE OK."
300   LET A(I) = 20 - I
310   PRINT "HAVE ASSIGNED ";20-I;" TO A(";I;")."
320   IF I<8 THEN 340
330   PRINT "ASSIGNMENT TO SUBSCRIPT > 7 - *** TEST FAILS ***"
340   PRINT
350 NEXT I
360 PRINT
370 PRINT "                                END TEST."
380 PRINT
390 PRINT "END PROGRAM 68"
400 END

```

---

PROGRAM FILE 68: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
 ONE-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1.  
 ANSI STANDARD 6.5, 15.2, 15.4

SECTION 68.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
 ONE-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
 RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS  
 THE EXPLICIT UPPER BOUND OF AN ARRAY, WITH AN OPTION  
 STATEMENT PRESENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
 DISPLAYED, AND

- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO A( 5 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 15 TO A( 5 ).

ABOUT TO ASSIGN TO A( 6 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 14 TO A( 6 ).

ABOUT TO ASSIGN TO A( 7 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 13 TO A( 7 ).

ABOUT TO ASSIGN TO A( 8 ). \*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 300

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 69: EXCEPTION - SUBSCRIPT TOO LARGE FOR"
15 PRINT "      TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0."
20 PRINT "      ANSI STANDARD 6.5, 15.2, 15.4"
30 PRINT
40 PRINT "SECTION 69.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR"
50 PRINT "      TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS"
90 PRINT "THE EXPLICIT UPPER BOUND OF AN ARRAY, WITH AN OPTION"
100 PRINT "STATEMENT PRESENT."
110 PRINT
120 PRINT "TO PASS THIS TEST:"
130 PRINT
140 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
150 PRINT "          DISPLAYED, AND"
160 PRINT
170 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL"
180 PRINT "          SUBSCRIPT."
190 PRINT
200 PRINT "                                BEGIN TEST."
210 PRINT
220 OPTION BASE 0
230 DIM B(3,12)
240 FOR I=9 TO 14
250 PRINT "ABOUT TO ASSIGN TO B(0,";I;")."
260 IF I<13 THEN 290
270 PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
280 GOTO 300
290 PRINT " ASSIGNMENT SHOULD BE OK."
300 LET B(0,I) = 20 - I
```

```
310 PRINT "HAVE ASSIGNED ";20-I;" TO B(0,";I;")."
320 IF I<13 THEN 340
330 PRINT "ASSIGNMENT TO SUBSCRIPT > 12 - *** TEST FAILS ***"
340 PRINT
350 NEXT I
360 PRINT
370 PRINT "END TEST."
380 PRINT
390 PRINT "END PROGRAM 69"
400 END
```

---

PROGRAM FILE 69: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0.  
ANSI STANDARD 6.5, 15.2, 15.4

SECTION 69.1: EXCEPTION - SUBSCRIPT TOO LARGE FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS  
THE EXPLICIT UPPER BOUND OF AN ARRAY, WITH AN OPTION  
STATEMENT PRESENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO B(0, 9). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 11 TO B(0, 9).

ABOUT TO ASSIGN TO B(0, 10). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 10 TO B(0, 10).

ABOUT TO ASSIGN TO B(0, 11). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 9 TO B(0, 11).

ABOUT TO ASSIGN TO B(0, 12). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 8 TO B(0, 12).

ABOUT TO ASSIGN TO B(0, 13).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*
? DIMENSION ERROR IN LINE 300

```
10 PRINT "PROGRAM FILE 70: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
15 PRINT "          ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 0."
20 PRINT "      ANSI STANDARD 6.5, 15.2, 15.4"
30 PRINT
40 PRINT "SECTION 70.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
50 PRINT "          ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 0."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES USE OF A NEGATIVE SUBSCRIPT AS A FATAL EXCEPTION"
90 PRINT "WHEN OPTION BASE 0 HAS BEEN SPECIFIED."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE"
170 PRINT "          NEGATIVE SUBSCRIPT."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 OPTION BASE 0
220 FOR I=3 TO -2 STEP -1
230 PRINT "ABOUT TO ASSIGN TO A(";I;").";
240 IF I>-1 THEN 270
250 PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
260 GOTO 280
270 PRINT " ASSIGNMENT SHOULD BE OK."
280 LET A(I) = 20 - I
290 PRINT "HAVE ASSIGNED ";20-I;" TO A(";I;")."
300 IF I>-1 THEN 320
310 PRINT "ASSIGNMENT TO SUBSCRIPT < 0 - *** TEST FAILS ***"
320 PRINT
330 NEXT I
340 PRINT
350 PRINT "                                END TEST."
360 PRINT
370 PRINT "END PROGRAM 70"
380 END
```

PROGRAM FILE 70: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 0.  
ANSI STANDARD 6.5, 15.2, 15.4

## SECTION 70.1: EXCEPTION – SUBSCRIPT TOO SMALL FOR ONE-DIMENSIONAL ARRAY, WITH OPTION BASE 0.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR RECOGNIZES USE OF A NEGATIVE SUBSCRIPT AS A FATAL EXCEPTION WHEN OPTION BASE 0 HAS BEEN SPECIFIED.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE NEGATIVE SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO A( 3 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 17 TO A( 3 ).

ABOUT TO ASSIGN TO A( 2 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 18 TO A( 2 ).

ABOUT TO ASSIGN TO A( 1 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 19 TO A( 1 ).

ABOUT TO ASSIGN TO A( 0 ). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 20 TO A( 0 ).

ABOUT TO ASSIGN TO A(-1 ).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 280

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 71: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
15 PRINT "      TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0."
20 PRINT "      ANSI STANDARD 6.5, 15.2, 15.4"
30 PRINT
40 PRINT "SECTION 71.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
50 PRINT "      TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0."
70 PRINT
80 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
90 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS"
100 PRINT "THE EXPLICIT LOWER BOUND OF AN ARRAY."
110 PRINT
120 PRINT "TO PASS THIS TEST:"
130 PRINT
140 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
150 PRINT "          DISPLAYED, AND"
160 PRINT
170 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL"
180 PRINT "          SUBSCRIPT."
190 PRINT
```

```

200 PRINT "
210 OPTION BASE 0
220 DIM B(11,3)
230 PRINT
240 FOR I=3 TO -2 STEP -1
250   PRINT "ABOUT TO ASSIGN TO B(";I;");3).";
260   IF I>-1 THEN 290
270   PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
280 GOTO 300
290 PRINT " ASSIGNMENT SHOULD BE OK."
300 LET B(I,3) = 20 - I
310 PRINT "HAVE ASSIGNED ";20-I;" TO B(";I;";3)."
320 IF I>-1 THEN 340
330 PRINT "ASSIGNMENT TO SUBSCRIPT < 0 - *** TEST FAILS ***"
340 PRINT
350 NEXT I
360 PRINT
370 PRINT "
380 PRINT "
390 PRINT "END PROGRAM 71"
400 END

```

---

PROGRAM FILE 71: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0.  
ANSI STANDARD 6.5, 15.2, 15.4

SECTION 71.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 0.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A SUBSCRIPT EXCEEDS  
THE EXPLICIT LOWER BOUND OF AN ARRAY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ILLEGAL  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO B( 3 ,3). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 17 TO B( 3 ,3).

ABOUT TO ASSIGN TO B( 2 ,3). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 18 TO B( 2 ,3).

ABOUT TO ASSIGN TO B( 1 ,3). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 19 TO B( 1 ,3).

ABOUT TO ASSIGN TO B( 0 ,3). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 20 TO B( 0 ,3).

ABOUT TO ASSIGN TO B(-1 ,3).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 300

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 72: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
15 PRINT "           TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1."
20 PRINT "           ANSI STANDARD 6.5, 15.2, 15.4"
30 PRINT
40 PRINT "SECTION 72.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR"
50 PRINT "           TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1."
70 PRINT
80 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
90 PRINT "RECOGNIZES USE OF A ZERO SUBSCRIPT AS A FATAL EXCEPTION WHEN"
100 PRINT "OPTION BASE 1 HAS BEEN SPECIFIED ALONG WITH A"
110 PRINT "TWO-DIMENSIONAL ARRAY SET UP WITH A DIM-STATEMENT."
120 PRINT
130 PRINT "TO PASS THIS TEST:"
140 PRINT
150 PRINT "    1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
160 PRINT "        DISPLAYED, AND"
170 PRINT
180 PRINT "    2) THE PROGRAM MUST TERMINATE UPON USE OF THE ZERO"
190 PRINT "        SUBSCRIPT."
200 PRINT
210 PRINT "                               BEGIN TEST."
220 PRINT
230 OPTION BASE 1
240 DIM B(12,4)
250 FOR I=4 TO -2 STEP -1
260 PRINT "ABOUT TO ASSIGN TO B(12,";I;").";
270 IF I>0 THEN 300
280 PRINT "*** EXCEPTION SHOULD OCCUR NOW ***"
290 GOTO 310
300 PRINT " ASSIGNMENT SHOULD BE OK."
310 LET B(12,I)=10-I
320 PRINT "HAVE ASSIGNED ";10-I;" TO B(12,";I;")."
330 IF I>0 THEN 350
340 PRINT "ASSIGNMENT TO SUBSCRIPT < 1 - *** TEST FAILS ***"
350 PRINT
360 NEXT I
370 PRINT
380 PRINT "                               END TEST."
```

```
390 PRINT
400 PRINT "END PROGRAM 72"
410 END
```

---

PROGRAM FILE 72: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1.  
ANSI STANDARD 6.5, 15.2, 15.4

SECTION 72.1: EXCEPTION - SUBSCRIPT TOO SMALL FOR  
TWO-DIMENSIONAL ARRAY, WITH DIM AND OPTION BASE 1.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES USE OF A ZERO SUBSCRIPT AS A FATAL EXCEPTION WHEN  
OPTION BASE 1 HAS BEEN SPECIFIED ALONG WITH A  
TWO-DIMENSIONAL ARRAY SET UP WITH A DIM-STATEMENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ZERO  
SUBSCRIPT.

BEGIN TEST.

ABOUT TO ASSIGN TO B(12, 4). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 6 TO B(12, 4).

ABOUT TO ASSIGN TO B(12, 3). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 7 TO B(12, 3).

ABOUT TO ASSIGN TO B(12, 2). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 8 TO B(12, 2).

ABOUT TO ASSIGN TO B(12, 1). ASSIGNMENT SHOULD BE OK.  
HAVE ASSIGNED 9 TO B(12, 1).

ABOUT TO ASSIGN TO B(12, 0).\*\*\* EXCEPTION SHOULD OCCUR NOW \*\*\*

? DIMENSION ERROR IN LINE 310

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 73: ERROR - DIM SETS UPPER BOUND OF ZERO WITH"
20 PRINT "          OPTION BASE 1."
30 PRINT "          ANSI STANDARD 15.4"
40 PRINT
```

```
50 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF THE INPUT STATEMENT"
60 PRINT " WHICH HAS NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
70 PRINT " THIS FEATURE TO BE INCORRECTLY IMPLEMENTED, THEN THE "
80 PRINT " VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
90 PRINT
100 PRINT "SECTION 73.1: ERROR - DIM SETS UPPER BOUND OF ZERO WITH"
110 PRINT " OPTION BASE 1."
120 PRINT
130 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
140 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
150 PRINT
160 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
170 PRINT " ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
180 PRINT " BY THE PROCESSOR, OR"
190 PRINT
200 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
210 PRINT
220 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
230 PRINT "FOR DETAILED CRITERIA."
240 PRINT
250 PRINT " BEGIN TEST"
260 PRINT
270 OPTION BASE 1
280 DIM A(0)
300 PRINT "TO HELP IN CHECKING INTERPRETATION, YOU MAY ENTER"
310 PRINT "VALUES TO BE USED AS SUBSCRIPTS ON THE ARRAY. THE PROGRAM"
320 PRINT "WILL ASSIGN THE VALUE 777 TO THE INDICATED ELEMENT AND"
330 PRINT "THEN ECHO THE RESULT. IN THIS WAY YOU CAN CHECK WHICH"
340 PRINT "SUBSCRIPTS (0 OR 1) ARE WITHIN THE ARRAY AND WHICH CAUSE"
350 PRINT "EXCEPTIONS. ENTER 100 WHEN YOU WISH TO STOP."
360 PRINT
370 PRINT "ENTER VALUE TO BE USED AS SUBSCRIPT"
380 INPUT I
390 IF I=100 THEN 440
400 LET A(I)=777
410 PRINT "A(";I;") = ";A(I)
420 PRINT
430 GOTO 370
440 PRINT
450 PRINT " END TEST"
460 PRINT
470 PRINT "END PROGRAM 73"
480 END
```

---

? DIM INCOMPATIBLE WITH OPTION IN LINE 280

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 74: ERROR - DIM SETS ARRAY TO ONE DIMENSION"
13 PRINT "          AND REFERENCE IS MADE TO TWO-DIMENSIONAL"
16 PRINT "          VARIABLE OF SAME NAME."
20 PRINT "      ANSI STANDARD 15.4, 6.4"
30 PRINT
40 PRINT "SECTION 74.1: ERROR - DIM SETS ARRAY TO ONE DIMENSION "
50 PRINT "          AND REFERENCE IS MADE TO TWO-DIMENSIONAL"
60 PRINT "          VARIABLE OF SAME NAME."
70 PRINT
80 PRINT
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
110 PRINT
120 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
130 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
140 PRINT "          BY THE PROCESSOR, OR"
150 PRINT
160 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
170 PRINT
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
190 PRINT "FOR DETAILED CRITERIA."
200 PRINT
210 PRINT "          BEGIN TEST"
220 PRINT
230 DIM A(150)
240 FOR I=0 TO 10
250 FOR J=0 TO 10
260 LET A(I,J)=10000+(100*I)+J
270 NEXT J
280 NEXT I
300 PRINT "VARIABLE A(I,J) WAS ASSIGNED 10000+(100*I)+J."
310 PRINT "RESULTS BELOW:"
320 PRINT
330 PRINT "ARRAY VALUES FOR A(I,J):"
340 PRINT "EACH LINE HAS ALL THE ELEMENTS FOR A GIVEN VALUE OF I,"
350 PRINT "WITH J GOING FROM 0 TO 10."
360 PRINT
370 FOR I = 0 TO 10
380 PRINT "I=";I;"";
390 FOR J=0 TO 10
400 PRINT A(I,J);
410 NEXT J
420 PRINT
430 NEXT I
440 PRINT
450 PRINT "          END TEST"
460 PRINT
470 PRINT "END PROGRAM 74"
480 END
```

---

```
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS DIM IN LINE 260
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS DIM IN LINE 400
```

```
*****
*****  

10 PRINT "PROGRAM FILE 75: ERROR - DIM SETS ARRAY TO ONE DIMENSION"  

13 PRINT "          AND REFERENCE IS MADE TO SIMPLE VARIABLE"  

16 PRINT "          OF SAME NAME."  

20 PRINT "      ANSI STANDARD 15.4, 6.4"  

30 PRINT  

40 PRINT "SECTION 75.1: ERROR - DIM SETS ARRAY TO ONE DIMENSION"  

50 PRINT "          AND REFERENCE IS MADE TO SIMPLE VARIABLE"  

60 PRINT "          OF SAME NAME."  

70 PRINT  

80 PRINT  

90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  

100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  

110 PRINT  

120 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  

130 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  

140 PRINT "    BY THE PROCESSOR, OR"  

150 PRINT  

160 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  

170 PRINT  

180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  

190 PRINT "FOR DETAILED CRITERIA."  

200 PRINT  

210 PRINT "          BEGIN TEST"  

220 PRINT  

230 DIM A(47)  

240 LET A=777  

300 PRINT "VARIABLE A WAS ASSIGNED 777. RESULTS BELOW:"  

320 PRINT  

400 PRINT "VARIABLE A = ";A  

440 PRINT  

450 PRINT "          END TEST"  

460 PRINT  

470 PRINT "END PROGRAM 75"  

480 END
```

---

? VARIABLE REFERENCE INCOMPATIBLE WITH PREVIOUS DIM IN LINE 240

```
*****
*****  

10 PRINT "PROGRAM FILE 76: ERROR - DIM SETS ARRAY TO TWO DIMENSIONS"  

13 PRINT "          AND REFERENCE IS MADE TO ONE-DIMENSIONAL"  

16 PRINT "          VARIABLE OF SAME NAME."  

20 PRINT "      ANSI STANDARD 15.4, 6.4"  

30 PRINT
```

```

40 PRINT "SECTION 76.1: ERROR - DIM SETS ARRAY TO TWO DIMENSIONS"
50 PRINT "          AND REFERENCE IS MADE TO ONE-DIMENSIONAL"
60 PRINT "          VARIABLE OF SAME NAME."
70 PRINT
80 PRINT
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
110 PRINT
120 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
130 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
140 PRINT "          BY THE PROCESSOR, OR"
150 PRINT
160 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
170 PRINT
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
190 PRINT "FOR DETAILED CRITERIA."
200 PRINT
210 PRINT "          BEGIN TEST"
220 PRINT
230 DIM A(3,5)
240 FOR I=0 TO 10
250 LET A(I)=111*I+5000
260 NEXT I
280 PRINT "VARIABLE A(I) WAS ASSIGNED 111*I + 5000."
290 PRINT "RESULTS BELOW:"
300 PRINT
310 FOR I = 0 TO 10
320 PRINT "A(";I;") = ";A(I)
330 NEXT I
340 PRINT
350 PRINT "          END TEST"
360 PRINT
370 PRINT "END PROGRAM 76"
380 END

```

---

? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS DIM IN LINE 250  
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS DIM IN LINE 320

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 77: ERROR - REFERENCE TO ARRAY AND SIMPLE "
15 PRINT "          VARIABLE OF SAME NAME."
20 PRINT "          ANSI STANDARD 6.4"
30 PRINT
40 PRINT "SECTION 77.1: ERROR - REFERENCE TO ARRAY AND SIMPLE "
50 PRINT "          VARIABLE OF SAME NAME."
60 PRINT
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT

```

```

110 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
120 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT "          BY THE PROCESSOR, OR"
140 PRINT
150 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT "                  BEGIN TEST"
210 PRINT
220 LET A=777
230 FOR I=0 TO 10
240 LET A(I)=111*I+5000
250 NEXT I
270 PRINT "VARIABLE A(I) WAS ASSIGNED 111*I + 5000,"
280 PRINT "AND SIMPLE VARIABLE A WAS ASSIGNED 777."
290 PRINT "RESULTS BELOW:"
300 PRINT
310 PRINT "A = ";A
320 IF A=777 THEN 350
330 PRINT
340 PRINT "***** ORIGINAL VALUE OF A WAS LOST *****"
350 PRINT
360 LET C=0
370 FOR I = 0 TO 10
380 PRINT "A(";I;") = ";A(I)
390 IF A(I)=111*I+5000 THEN 410
400 LET C=9
410 NEXT I
420 IF C=0 THEN 450
430 PRINT
440 PRINT "***** SOME VALUES IN ARRAY WERE LOST *****"
450 PRINT
460 PRINT "                  END TEST"
470 PRINT
480 PRINT "END PROGRAM 77"
490 END

```

---

? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS VARIABLE IN LINE 240  
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS VARIABLE IN LINE 380  
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS VARIABLE IN LINE 390

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 78: ERROR - REFERENCE TO ONE-DIMENSIONAL AND "
15 PRINT "          TWO-DIMENSIONAL VARIABLE OF SAME NAME."
20 PRINT "          ANSI STANDARD 6.4"
30 PRINT

```

```
40 PRINT "SECTION 78.1: ERROR - REFERENCE TO ONE-DIMENSIONAL AND "
50 PRINT "           TWO-DIMENSIONAL VARIABLE OF SAME NAME."
60 PRINT
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT
110 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
120 PRINT "           ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT "           BY THE PROCESSOR, OR"
140 PRINT
150 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT "           BEGIN TEST"
210 PRINT
220 FOR I=0 TO 10
230 LET A(I)=11*I+5000
240 NEXT I
250 FOR I=0 TO 10
260 FOR J=0 TO 10
270 LET A(I,J)=10000+(100*I)+J
280 NEXT J
290 NEXT I
310 PRINT "VARIABLE A(I) WAS ASSIGNED 11*I + 5000,"
320 PRINT "AND VARIABLE A(I,J) WAS ASSIGNED 10000+(100*I)+J."
330 PRINT "RESULTS BELOW:"
340 PRINT
350 PRINT "ARRAY VALUES FOR A(I,J):"
360 PRINT "EACH LINE HAS ALL THE ELEMENTS FOR A GIVEN VALUE OF I,"
370 PRINT "WITH J GOING FROM 0 TO 10."
380 PRINT
390 LET C1=0
400 LET C2=0
410 FOR I = 0 TO 10
420 PRINT "I=";I;"";
430 FOR J=0 TO 10
440 PRINT A(I,J);
450 IF A(I,J)=10000+(100*I)+J THEN 470
460 LET C2=9
470 NEXT J
480 PRINT
490 NEXT I
500 PRINT
510 PRINT "ARRAY VALUES FOR A(I):"
520 PRINT
530 FOR I = 0 TO 10
540 PRINT "A(";I;") = ";A(I)
550 IF A(I)=11*I+5000 THEN 570
560 LET C1=9
570 NEXT I
580 IF C1=0 THEN 610
590 PRINT
```

```
600 PRINT "***** SOME VALUES IN ONE-DIMENSIONAL ARRAY WERE LOST *****"
610 IF C2=0 THEN 640
620 PRINT
630 PRINT "***** SOME VALUES IN TWO-DIMENSIONAL ARRAY WERE LOST *****"
640 PRINT
650 PRINT "           END TEST"
660 PRINT
670 PRINT "END PROGRAM 78"
680 END
```

---

```
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS VARIABLE IN LINE 270
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS VARIABLE IN LINE 440
? ARRAY REFERENCE INCOMPATIBLE WITH PREVIOUS VARIABLE IN LINE 450
```

```
*****
```

```
10 PRINT "PROGRAM FILE 79: ERROR - REFERENCE TO ARRAY"
15 PRINT "           WITH LETTER-DIGIT NAME."
20 PRINT "           ANSI STANDARD 6.2"
30 PRINT
40 PRINT "SECTION 79.1: ERROR - REFERENCE TO ARRAY"
50 PRINT "           WITH LETTER-DIGIT NAME."
60 PRINT
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT
110 PRINT "   1) ACCEPT THE PROGRAM AND BE ACCOMPANIED BY DOCUMENTATION"
120 PRINT "       ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT "       BY THE PROCESSOR, OR"
140 PRINT
150 PRINT "   2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT "           BEGIN TEST"
210 PRINT
230 FOR I=0 TO 10
240 LET A9(I)=111*I+5000
250 NEXT I
270 PRINT "VARIABLE A9(I) WAS ASSIGNED 111*I + 5000."
290 PRINT "RESULTS BELOW:"
300 PRINT
360 LET C=0
370 FOR I = 0 TO 10
380 PRINT "A9(";I;") = ";A9(I)
390 IF A9(I)=111*I+5000 THEN 410
400 LET C=9
410 NEXT I
```

```
420 IF C=0 THEN 450
430 PRINT
440 PRINT "***** SOME VALUES IN ARRAY WERE LOST ****"
450 PRINT
460 PRINT "           END TEST"
470 PRINT
480 PRINT "END PROGRAM 79"
490 END
```

---

```
? ILLEGAL ARRAY NAME IN LINE 240
? ILLEGAL ARRAY NAME IN LINE 380
? ILLEGAL ARRAY NAME IN LINE 390
```

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 80: ERROR - MULTIPLE OPTION STATEMENTS."
20 PRINT "      ANSI STANDARD 15.4"
30 PRINT
90 PRINT "SECTION 80.1: ERROR - MULTIPLE OPTION STATEMENTS."
100 PRINT
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
130 PRINT
140 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
150 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
160 PRINT "          BY THE PROCESSOR, OR"
170 PRINT
180 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
190 PRINT
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
210 PRINT "FOR DETAILED CRITERIA."
220 PRINT
230 PRINT "           BEGIN TEST"
240 PRINT
250 OPTION BASE 1
260 OPTION BASE 0
340 PRINT "YOU MAY WISH TO REVERSE THE ORDER OF THE OPTION"
350 PRINT "STATEMENTS IN THE SOURCE CODE TO TEST PROCESSOR"
360 PRINT "INTERPRETATION."
370 PRINT
380 PRINT "ABOUT TO EXECUTE 'LET A(0) = 777'"
410 LET A(0)=777
420 PRINT "A(";I;") = .";A(I)
430 PRINT
460 PRINT "           END TEST"
```

```
470 PRINT  
480 PRINT "END PROGRAM 80"  
490 END
```

? DUPLICATE OPTION STATEMENT IN LINE 260

```
10 PRINT "PROGRAM FILE 81: ERROR - DIM-STATEMENT PRECEDES"
20 PRINT "          OPTION-STATEMENT."
30 PRINT "      ANSI STANDARD 15.4"
40 PRINT
50 PRINT "**** NOTE: THIS PROGRAM MAKES USE OF THE INPUT STATEMENT"
60 PRINT "      WHICH HAS NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"
70 PRINT "      THIS FEATURE TO BE INCORRECTLY IMPLEMENTED, THEN THE "
80 PRINT "      VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."
90 PRINT
100 PRINT "SECTION 81.1: ERROR - DIM-STATEMENT PRECEDES "
110 PRINT "          OPTION-STATEMENT."
120 PRINT
130 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
140 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
150 PRINT
160 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
170 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
180 PRINT "          BY THE PROCESSOR, OR"
190 PRINT
200 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
210 PRINT
220 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
230 PRINT "FOR DETAILED CRITERIA."
240 PRINT
250 PRINT "          BEGIN TEST"
260 PRINT
270 DIM A(5)
280 OPTION BASE 1
290 PRINT "TO HELP IN CHECKING INTERPRETATION, YOU MAY ENTER"
300 PRINT "VALUES TO BE USED AS SUBSCRIPTS ON THE ARRAY. THE PROGRAM"
310 PRINT "WILL ASSIGN THE VALUE 777 TO THE INDICATED ELEMENT AND"
320 PRINT "THEN ECHO THE RESULT. IN THIS WAY YOU CAN CHECK WHICH"
330 PRINT "SUBSCRIPTS ARE WITHIN THE ARRAY AND WHICH CAUSE EXCEPTIONS."
340 PRINT "THE ARRAY IS SET WITH 'DIM A(5)', FOLLOWED BY OPTION BASE 1."
350 PRINT "YOU SHOULD AT LEAST TEST SUBSCRIPT VALUES OF ZERO"
360 PRINT "AND SIX. THE ARRAY HAS BEEN DIMENSIONED AS A(5)."
370 PRINT
380 PRINT "ENTER 100 WHEN YOU WISH TO STOP."
390 PRINT
```

```
400 PRINT "ENTER VALUE TO BE USED AS SUBSCRIPT:"  
410 INPUT I  
420 IF I=100 THEN 470  
430 LET A(I)=777  
440 PRINT "A(";I;") = ";A(I)  
450 PRINT  
460 GOTO 400  
470 PRINT  
480 PRINT " END TEST"  
490 PRINT  
500 PRINT "END PROGRAM 81"  
510 END
```

---

? OPTION PRECEDED BY DIM IN LINE 280

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 82: ERROR - ARRAY-REFERENCE PRECEDES"  
20 PRINT " OPTION-STATEMENT."  
30 PRINT " ANSI STANDARD 15.4"  
40 PRINT  
50 PRINT "SECTION 82.1: ERROR - ARRAY-REFERENCE PRECEDES"  
60 PRINT " OPTION-STATEMENT."  
70 PRINT  
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
100 PRINT  
110 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
120 PRINT " ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
130 PRINT " BY THE PROCESSOR, OR"  
140 PRINT  
150 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
160 PRINT  
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
180 PRINT "FOR DETAILED CRITERIA."  
190 PRINT  
200 PRINT " BEGIN TEST"  
210 PRINT  
220 PRINT "ABOUT TO EXECUTE 'LET A(0) = 105'"  
230 PRINT "'OPTION BASE 1' FOLLOWS THIS STATEMENT."  
240 LET A(0) = 105  
250 OPTION BASE 1  
260 PRINT  
270 PRINT "A(0) = ";A(0)  
280 PRINT  
290 PRINT " END TEST"
```

```
300 PRINT  
310 PRINT "END PROGRAM 82"  
320 END
```

---

```
? OPTION PRECEDED BY ARRAY REFERENCE IN LINE 250
```

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 83: ERROR - ARRAY-REFERENCE PRECEDES"  
15 PRINT "      DIM-STATEMENT."  
20 PRINT "      ANSI STANDARD 15.4"  
30 PRINT  
90 PRINT "SECTION 83.1: ERROR - ARRAY-REFERENCE PRECEDES"  
95 PRINT "      DIM-STATEMENT."  
100 PRINT  
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
130 PRINT  
140 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
150 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
160 PRINT "          BY THE PROCESSOR, OR"  
170 PRINT  
180 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
190 PRINT  
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
210 PRINT "FOR DETAILED CRITERIA."  
220 PRINT  
230 PRINT "      BEGIN TEST"  
240 PRINT  
260 PRINT "ABOUT TO EXECUTE 'LET A(6) = 777'"  
270 PRINT "'DIM A(5)' FOLLOWS THIS STATEMENT."  
280 PRINT  
400 LET A(6)=777  
410 PRINT "A(6) = ";A(6)  
440 PRINT  
450 PRINT "      END TEST"  
460 PRINT  
470 PRINT "END PROGRAM 83"  
480 STOP  
490 DIM A(5)  
500 END
```

---

```
? DIM PRECEDED BY ARRAY REFERENCE IN LINE 490
```

```
*****
*****
*****  
10 PRINT "PROGRAM FILE 84: ERROR - DIMENSIONING THE SAME ARRAY "  
20 PRINT "           MORE THAN ONCE."  
30 PRINT "           ANSI STANDARD 15.4"  
40 PRINT  
50 PRINT "*** NOTE: THIS PROGRAM MAKES USE OF THE INPUT STATEMENT"  
60 PRINT "           WHICH HAS NOT YET BEEN TESTED. IF SUBSEQUENT TESTS SHOW"  
70 PRINT "           THIS FEATURE TO BE INCORRECTLY IMPLEMENTED, THEN THE "  
80 PRINT "           VALIDITY OF THE RESULTS OF THIS TEST ROUTINE IS DOUBTFUL."  
90 PRINT  
100 PRINT "SECTION 84.1: ERROR - DIMENSIONING THE SAME ARRAY "  
110 PRINT "           MORE THAN ONCE."  
120 PRINT  
130 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
140 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
150 PRINT  
160 PRINT "     1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
170 PRINT "           ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
180 PRINT "           BY THE PROCESSOR, OR"  
190 PRINT  
200 PRINT "     2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
210 PRINT  
220 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
230 PRINT "FOR DETAILED CRITERIA."  
240 PRINT  
250 PRINT "           BEGIN TEST"  
260 PRINT  
270 PRINT "IN ORDER TO DETERMINE WHAT INTERPRETATION IS GIVEN TO A"  
280 PRINT "DOUBLY-DIMENSIONED ARRAY, YOU MUST EXECUTE THIS PROGRAM"  
290 PRINT "THREE TIMES, AND NOTE EACH TIME WHETHER A SUBSCRIPT-OUT-"  
300 PRINT "OF-RANGE EXCEPTION OCCURS. THE TABLE BELOW GIVES THE"  
310 PRINT "EXPECTED RESULTS FOR EACH OF FIVE POSSIBLE INTERPRETATIONS."  
320 PRINT  
330 PRINT "RESULTS WHICH DO NOT MATCH ANY OF THE ROWS IN THE TABLE"  
340 PRINT "INDICATE THAT SOME OTHER INTERPRETATION IS BEING USED."  
350 PRINT  
360 PRINT "TABLE OF RESULTS: THE THREE COLUMNS STAND FOR 1ST, 2ND, AND"  
370 PRINT "3RD EXECUTION, AND THE FIVE ROWS STAND FOR FIVE POSSIBLE"  
380 PRINT "INTERPRETATIONS."  
390 PRINT  
400 PRINT "           DID EXCEPTION OCCUR?"  
410 PRINT  
420 PRINT "           1      2      3"  
430 PRINT "INT NO. 1    YES    YES    YES"  
440 PRINT "INT NO. 2    YES    YES    NO"  
450 PRINT "INT NO. 3    NO     NO     NO"  
460 PRINT "INT NO. 4    YES    NO     NO"  
470 PRINT "INT NO. 5    NO     YES    NO"  
480 PRINT  
490 PRINT "INTERPRETATION NUMBER 1 IS THAT THE FIRST DIM-STATEMENT IN"  
500 PRINT "THE SOURCE CODE GOVERNS THE SIZE OF THE ARRAY."  
510 PRINT
```

```

520 PRINT "INTERPRETATION NUMBER 2 IS THAT THE DIM-STATEMENT MOST"
530 PRINT "RECENTLY PRECEDING AN ARRAY REFERENCE IN THE SOURCE CODE"
540 PRINT "DETERMINES THE SIZE OF THE ARRAY AT THAT REFERENCE."
550 PRINT
560 PRINT "INTERPRETATION NUMBER 3 IS THAT THE LAST DIM-STATEMENT IN"
570 PRINT "THE SOURCE CODE DETERMINES THE SIZE OF THE ARRAY."
580 PRINT
590 PRINT "INTERPRETATION NUMBER 4 IS THAT THE FIRST DIM-STATEMENT"
600 PRINT "EXECUTED DETERMINES THE SIZE OF THE ARRAY."
610 PRINT
620 PRINT "INTERPRETATION NUMBER 5 IS THAT THE DIM-STATEMENT MOST"
630 PRINT "RECENTLY EXECUTED DETERMINES THE SIZE OF THE ARRAY."
640 LET S=13
650 PRINT
660 PRINT "IS THIS THE 1ST, 2ND, OR 3RD EXECUTION (RESPOND 1, 2, OR 3)?"
670 INPUT I
680 IF I=1 THEN 730
690 IF I=2 THEN 770
700 IF I=3 THEN 770
710 PRINT "INVALID RESPONSE - RE-ENTER."
720 GOTO 650
730 DIM A(12)
740 IF I=1 THEN 770
750 LET A(S)=7
760 GOTO 810
770 DIM A(14)
780 IF I=1 THEN 750
790 IF I=2 THEN 730
800 LET A(S)=8
810 PRINT
820 PRINT
830 PRINT "           END TEST"
840 PRINT
850 PRINT "END PROGRAM 84"
860 END

```

---

? VARIABLE DIMENSIONED TWICE IN LINE 770

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 85: GENERAL CAPABILITIES OF GOSUB/RETURN."
20 PRINT "      ANSI STANDARD 10.4"
30 PRINT
40 PRINT "SECTION 85.1: SIMPLE NESTING OF GOSUBS."
50 PRINT
60 PRINT "THIS SECTION TESTS THAT SUBROUTINES MAY CALL ONE ANOTHER"
70 PRINT "AND THAT A SUBROUTINE MAY BE CALLED FROM SEVERAL PLACES."
80 PRINT
90 PRINT "           BEGIN TEST."
100 PRINT

```

```
400 LET F=0
500 LET X=0
510 GOSUB 570
520 IF X=11 THEN 530
525 LET F=1
530 LET X=0
540 GOSUB 600
550 IF X=8 THEN 560
555 LET F=1
560 GOTO 700
570 LET X=X+8
580 GOSUB 640
590 RETURN
600 GOSUB 640
610 GOSUB 670
620 LET X=X+4
630 RETURN
640 LET X=X+2
650 GOSUB 670
660 RETURN
670 LET X=X+1
680 RETURN
700 PRINT
710 IF F=0 THEN 740
720 PRINT "*** TEST FAILED ***"
730 GOTO 750
740 PRINT "*** TEST PASSED ***"
750 PRINT
1000 PRINT " END TEST."
1010 PRINT
1020 PRINT
1030 PRINT "SECTION 85.2: STACKING OF RETURN ADDRESSES."
1040 PRINT
1050 PRINT "THIS SECTION TESTS THAT RETURN ADDRESSES ARE PROPERLY"
1060 PRINT "STACKED BY GOSUBS (INCLUDING SEVERAL EXECUTIONS OF THE"
1070 PRINT "SAME GOSUB) AND UNSTACKED BY CORRESPONDING RETURNS."
1080 PRINT
1090 PRINT " BEGIN TEST."
1100 PRINT
1150 LET F=0
1200 LET I0=3
1210 LET J0=4
1220 GOSUB 1500
1230 IF K0=4 THEN 1250
1240 LET F=1
1250 LET I0=2
1260 LET J0=7
1270 GOSUB 1500
1280 IF K0=21 THEN 1300
1290 LET F=1
1300 LET I0=4
1310 LET J0=8
1320 GOSUB 1500
1330 IF K0=70 THEN 1350
1340 LET F=1
```

```
1350 GOTO 1900
1500 REM TEST RECURSIVE USE OF GOSUB
1510 LET L9=0
1520 LET M0=0
1540 GOSUB 1570
1560 RETURN
1570 REM BINOMIAL COEFFICIENT SUBROUTINE
1580 REM FINDS K0=BIN(I0,J0)
1590 REM ADD 1 TO LEVEL-COUNTER (L9) AND SAVE STATE
1600 LET L9=L9+1
1610 LET M(L9)=M0
1620 LET I(L9)=I0
1630 LET J(L9)=J0
1640 IF I0=0 THEN 1670
1650 IF I0=J0 THEN 1670
1660 GOTO 1740
1670 LET K0=1
1680 REM RESTORE STATE
1690 LET I0=I(L9)
1700 LET J0=J(L9)
1710 LET M0=M(L9)
1720 LET L9=L9-1
1730 RETURN
1740 REM BIN(I0,J0)=BIN(I0,J0-1)+BIN(I0-1,J0-1)
1750 LET J0=J0-1
1760 GOSUB 1570
1770 LET M0=K0
1780 LET I0=I0-1
1790 GOSUB 1570
1800 LET K0=K0+M0
1810 REM RESTORE STATE
1820 LET I0=I(L9)
1830 LET J0=J(L9)
1840 LET M0=M(L9)
1850 LET L9=L9-1
1860 RETURN
1900 PRINT
1910 IF F=0 THEN 1940
1920 PRINT "*** TEST FAILED ***"
1930 GOTO 1950
1940 PRINT "*** TEST PASSED ***"
1950 PRINT
2000 PRINT " END TEST."
2010 PRINT
2020 PRINT
2030 PRINT "SECTION 85.3: GOSUB WITHOUT RETURN."
2040 PRINT
2050 PRINT "THIS SECTION TESTS THAT PROGRAM EXECUTION MAY VALIDLY"
2060 PRINT "TERMINATE EVEN THOUGH MORE GOSUBS HAVE BEEN EXECUTED "
2070 PRINT "THAN RETURNS."
2080 PRINT
2090 PRINT " BEGIN TEST."
2100 PRINT
```

```
2110 GOSUB 2200
2120 PRINT "*** TEST FAILED ***"
2130 GOTO 2990
2200 PRINT "IF PROGRAM TERMINATES NORMALLY, THEN"
2220 PRINT "*** TEST PASSED ***"
2990 PRINT
3000 PRINT " END TEST."
3010 PRINT
3020 PRINT "END PROGRAM 85"
4000 END
```

---

PROGRAM FILE 85: GENERAL CAPABILITIES OF GOSUB/RETURN.

ANSI STANDARD 10.4

#### SECTION 85.1: SIMPLE NESTING OF GOSUBS.

THIS SECTION TESTS THAT SUBROUTINES MAY CALL ONE ANOTHER AND THAT A SUBROUTINE MAY BE CALLED FROM SEVERAL PLACES.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 85.2: STACKING OF RETURN ADDRESSES.

THIS SECTION TESTS THAT RETURN ADDRESSES ARE PROPERLY STACKED BY GOSUBS (INCLUDING SEVERAL EXECUTIONS OF THE SAME GOSUB) AND UNSTACKED BY CORRESPONDING RETURNS.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 85.3: GOSUB WITHOUT RETURN.

THIS SECTION TESTS THAT PROGRAM EXECUTION MAY VALIDLY TERMINATE EVEN THOUGH MORE GOSUBS HAVE BEEN EXECUTED THAN RETURNS.

BEGIN TEST.

IF PROGRAM TERMINATES NORMALLY, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 85

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 86: EXCEPTION - RETURN WITHOUT GOSUB."  
20 PRINT "    ANSI STANDARD 10.5"  
30 PRINT  
40 PRINT "SECTION 86.1: EXCEPTION - RETURN WITHOUT GOSUB."  
50 PRINT  
60 PRINT "ENCOUNTERING A RETURN-STATEMENT WITHOUT HAVING EXECUTED A"  
70 PRINT "CORRESPONDING GOSUB-STATEMENT SHOULD RESULT IN THE "  
80 PRINT "TERMINATION OF THE PROGRAM AND THE GENERATION OF A MESSAGE"  
90 PRINT "IDENTIFYING THE EXCEPTION."  
100 PRINT  
110 PRINT "TO PASS THIS TEST:"  
120 PRINT  
130 PRINT "    1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
140 PRINT "        DISPLAYED, AND"  
150 PRINT  
160 PRINT "    2) THE PROCESSOR MUST TERMINATE THIS PROGRAM WHEN"  
170 PRINT "        THE RETURN-STATEMENT IS ENCOUNTERED."  
180 PRINT  
190 PRINT "                BEGIN TEST."  
200 PRINT  
210 LET I=0  
220 GOSUB 310  
230 IF I>1 THEN 330  
240 GOTO 280  
250 GOSUB 300  
260 GOTO 330  
280 REM SKIP GOSUB  
290 GOTO 310  
300 PRINT "*** TEST FAILED ***"  
310 LET I=I+1  
320 RETURN  
330 PRINT "*** TEST FAILED ***"  
340 PRINT  
350 PRINT "                END TEST."  
360 PRINT  
370 PRINT "END PROGRAM 86"  
380 END
```

---

PROGRAM FILE 86: EXCEPTION - RETURN WITHOUT GOSUB.  
ANSI STANDARD 10.5

SECTION 86.1: EXCEPTION - RETURN WITHOUT GOSUB.

ENCOUNTERING A RETURN-STATEMENT WITHOUT HAVING EXECUTED A  
CORRESPONDING GOSUB-STATEMENT SHOULD RESULT IN THE  
TERMINATION OF THE PROGRAM AND THE GENERATION OF A MESSAGE  
IDENTIFYING THE EXCEPTION.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) THE PROCESSOR MUST TERMINATE THIS PROGRAM WHEN THE RETURN-STATEMENT IS ENCOUNTERED.

BEGIN TEST.

? RETURN BEFORE GOSUB IN LINE 320

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 87: ERROR - TRANSFER TO NON-EXISTING LINE"  
15 PRINT "      NUMBER USING THE GOSUB-STATEMENT."  
20 PRINT "      ANSI STANDARD 10.4"  
30 PRINT  
40 PRINT "SECTION 87.1: ERROR - TRANSFER TO NON-EXISTING LINE "  
50 PRINT "      NUMBER USING THE GOSUB-STATEMENT."  
60 PRINT  
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
90 PRINT  
100 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
110 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
120 PRINT "          BY THE PROCESSOR, OR"  
130 PRINT  
140 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
150 PRINT  
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
170 PRINT "FOR DETAILED CRITERIA."  
180 PRINT  
190 PRINT "                                BEGIN TEST."  
200 PRINT  
210 LET X=0  
220 LET Y=0  
230 GOSUB 285  
240 IF X=1 THEN 310  
250 IF Y=1 THEN 330  
260 PRINT "THE PROCESSOR IGNORED THE GOSUB STATEMENT."  
270 GOTO 340  
280 LET X=1  
290 LET Y=1  
300 RETURN  
310 PRINT " THE PROCESSOR SUBSTITUTED AN EXISTING, LOWER LINE NUMBER."  
320 GOTO 340  
330 PRINT "THE PROCESSOR SUBSTITUTED AN EXISTING, HIGHER LINE NUMBER."  
340 PRINT  
350 PRINT "                                END TEST."
```

```
360 PRINT  
370 PRINT "END PROGRAM 87"  
380 END
```

? UNDEFINED LINE NUMBER 285 IN LINE 230

```
10 PRINT "PROGRAM FILE 88: THE ON-GOTO-STATEMENT."
20 PRINT "      ANSI STANDARD 10.2, 10.4"
30 PRINT
60 PRINT "SECTION 88.1: THE ON-GOTO EXPRESSION WITH INTEGER VALUES."
80 PRINT
90 PRINT "
100 PRINT
120 LET N=0
130 FOR I=1 TO 5
140 REM 3RD DIGIT OF LINE NUMBER IS ITS POSITION IN LIST
150 ON I GOTO 221, 252, 213, 244, 235
160 PRINT "CONTROL FELL THROUGH ON-GOTO"
170 GOTO 280
213 IF I=3 THEN 290
218 LET K=3
219 GOTO 270
221 IF I=1 THEN 290
228 LET K=1
229 GOTO 270
235 IF I=5 THEN 290
238 LET K=5
239 GOTO 270
244 IF I=4 THEN 290
248 LET K=4
249 GOTO 270
252 IF I=2 THEN 290
258 LET K=2
259 GOTO 270
270 PRINT "CONTROL WENT TO LINE #";K;" IN LIST EVEN THOUGH ON-GOTO";
275 PRINT " EXPRESSION = ";I
280 LET N=1
290 NEXT I
300 PRINT
310 IF N=0 THEN 340
320 PRINT "*** TEST FAILED ***"
330 GOTO 350
340 PRINT "*** TEST PASSED ***"
350 PRINT
360 PRINT "
END TEST."
```

```

370 PRINT
375 PRINT
380 PRINT "SECTION 88.2: THE ON-GOTO EXPRESSION WITH FRACTIONAL VALUES."
400 PRINT
410 PRINT "THIS SECTION TESTS THAT FRACTIONAL VALUES ARE ROUNDED"
420 PRINT "TO THE NEAREST INTEGER BY THE ON-GOTO BEFORE SELECTION"
425 PRINT "OF THE LINE-NUMBER FROM THE LIST."
430 PRINT
440 PRINT "                                BEGIN TEST."
450 PRINT
500 LET N=0
510 FOR I=0.6 TO 3.4 STEP .2
530 ON I GOTO 600,700,800
540 PRINT "CONTROL FELL THROUGH ON-GOTO."
550 GOTO 950
600 LET K=1
610 IF I<.5 THEN 900
620 IF I>1.5 THEN 900
630 GOTO 1000
700 LET K=2
710 IF I<1.5 THEN 900
720 IF I>2.5 THEN 900
730 GOTO 1000
800 LET K=3
810 IF I<2.5 THEN 900
820 IF I>3.5 THEN 900
830 GOTO 1000
900 PRINT "CONTROL WENT TO LINE #";K;" IN LIST, BUT ON-GOTO";
910 PRINT " EXPRESSION = ";I
950 LET N=1
1000 NEXT I
1010 PRINT
1020 IF N=0 THEN 1050
1030 PRINT "*** TEST FAILED ***"
1040 GOTO 1060
1050 PRINT "*** TEST PASSED ***"
1060 PRINT
1070 PRINT "                                END TEST."
1260 PRINT
1270 PRINT "END PROGRAM 88"
1280 END

```

---

PROGRAM FILE 88: THE ON-GOTO-STATEMENT.  
ANSI STANDARD 10.2, 10.4

SECTION 88.1: THE ON-GOTO EXPRESSION WITH INTEGER VALUES.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

## SECTION 88.2: THE ON-GOTO EXPRESSION WITH FRACTIONAL VALUES.

THIS SECTION TESTS THAT FRACTIONAL VALUES ARE ROUNDED TO THE NEAREST INTEGER BY THE ON-GOTO BEFORE SELECTION OF THE LINE-NUMBER FROM THE LIST.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 88

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 89: EXCEPTION - ON-GOTO CONTROL EXPRESSION"  
20 PRINT "      LESS THAN 1."  
30 PRINT "      ANSI STANDARD 10.5"  
40 PRINT  
50 PRINT "SECTION 89.1: EXCEPTION - ON-GOTO CONTROL EXPRESSION "  
60 PRINT "      LESS THAN 1."  
70 PRINT  
80 PRINT "TO PASS THIS TEST:"  
90 PRINT  
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
110 PRINT "          DISPLAYED, AND"  
120 PRINT  
130 PRINT "      2) EXECUTION MUST TERMINATE."  
140 PRINT  
150 PRINT "          BEGIN TEST."  
160 PRINT  
170 LET X=.3  
180 ON X GOTO 210,230  
190 LET A$="Z"  
200 GOTO 250  
210 LET A$="FIRST"  
220 GOTO 250  
230 LET A$="SECOND"  
240 GOTO 250  
250 IF A$="Z" THEN 280  
260 PRINT "ON GOTO JUMPED TO ";A$;" LINE NUMBER IN LIST."  
270 GOTO 290  
280 PRINT "CONTROL FELL THROUGH ON-GOTO TO NEXT STATEMENT."  
290 PRINT "*** TEST FAILED ***"  
300 PRINT  
310 PRINT "          END TEST."
```

```
320 PRINT
330 PRINT "END PROGRAM 89"
340 END
```

---

PROGRAM FILE 89: EXCEPTION - ON-GOTO CONTROL EXPRESSION  
LESS THAN 1.

ANSI STANDARD 10.5

SECTION 89.1: EXCEPTION - ON-GOTO CONTROL EXPRESSION  
LESS THAN 1.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE.

BEGIN TEST.

? ON EVALUATED OUT OF RANGE IN LINE 180

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 90: EXCEPTION - ON-GOTO CONTROL EXPRESSION"
20 PRINT "          GREATER THAN NUMBER OF LINE-NUMBERS IN LIST."
30 PRINT "          ANSI STANDARD 10.5"
40 PRINT
50 PRINT "SECTION 90.1: EXCEPTION - ON-GOTO CONTROL EXPRESSION"
60 PRINT "          GREATER THAN NUMBER OF LINE-NUMBERS IN LIST."
70 PRINT
80 PRINT "TO PASS THIS TEST:"
90 PRINT
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
110 PRINT "          DISPLAYED, AND"
120 PRINT
130 PRINT "      2) EXECUTION MUST TERMINATE."
140 PRINT
150 PRINT "          BEGIN TEST."
160 PRINT
170 LET X=2.7
180 ON X GOTO 210,230
190 LET A$="Z"
200 GOTO 250
210 LET A$="FIRST"
220 GOTO 250
230 LET A$="SECOND"
240 GOTO 250
250 IF A$="Z" THEN 280
260 PRINT "ON GOTO JUMPED TO ";A$;" LINE NUMBER IN LIST."
```

```

270 GOTO 290
280 PRINT "CONTROL FELL THROUGH ON-GOTO TO NEXT STATEMENT."
290 PRINT "*** TEST FAILED ***"
300 PRINT
310 PRINT " END TEST."
320 PRINT
330 PRINT "END PROGRAM 90"
340 END

```

---

PROGRAM FILE 90: EXCEPTION - ON-GOTO CONTROL EXPRESSION  
GREATER THAN NUMBER OF LINE-NUMBERS IN LIST.  
ANSI STANDARD 10.5

SECTION 90.1: EXCEPTION - ON-GOTO CONTROL EXPRESSION  
GREATER THAN NUMBER OF LINE-NUMBERS IN LIST.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE.

BEGIN TEST.

? ON EVALUATED OUT OF RANGE IN LINE 180

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 91: ERROR - TRANSFER TO NON-EXISTING LINE"
15 PRINT "      NUMBER USING THE ON-GOTO-STATEMENT."
20 PRINT "      ANSI STANDARD 10.4"
30 PRINT
40 PRINT "SECTION 91.1: ERROR - TRANSFER TO NON-EXISTING LINE"
50 PRINT "      NUMBER USING THE ON-GOTO-STATEMENT."
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "      BEGIN TEST."
200 PRINT

```

```
230 LET A$="LINE 300."
240 LET X=1
250 ON X GOTO 295
260 PRINT "CONTROL FELL THROUGH THE ON-GOTO STATEMENT."
280 GOTO 310
290 LET A$="LINE 290."
300 PRINT "'ON X GOTO 295' JUMPED TO ";A$
310 PRINT
320 PRINT "
330 PRINT END TEST."
340 PRINT "END PROGRAM 91."
350 END
```

---

? UNDEFINED LINE NUMBER 295 IN LINE 250

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 92: READ AND DATA STATEMENTS FOR NUMERIC DATA."
20 PRINT "      ANSI STANDARD 5.2, 14.2, 14.4"
30 PRINT
40 PRINT "SECTION 92.1: READ AND DATA STATEMENTS FOR NUMERIC DATA."
50 PRINT
60 PRINT "THIS SECTION TESTS THE PROCESSOR'S CAPABILITY TO HANDLE"
70 PRINT "INTERNAL NUMERIC DATA USING THE DATA AND READ STATEMENTS."
80 PRINT "THE DATA-VALUES WILL BE COMPARED TO THOSE ASSIGNED TO AN"
90 PRINT "ARRAY AND THE RESULTS DISPLAYED. IF THE ACCURACY IS LESS"
120 PRINT "THAN SIX DIGITS, THE TEST WILL BE COUNTED AS A FAILURE."
130 PRINT
132 PRINT "          BEGIN TEST"
134 PRINT
138 DIM A(110)
140 LET A(1)=9.99998E37
145 LET A(2)=1E38
150 LET A(3)=-1E38
155 LET A(4)=-9.99998E37
160 LET A(5)=9.87653E37
165 LET A(6)=9.87655E37
170 LET A(7)=9.87653E-38
175 LET A(8)=9.87655E-38
180 LET A(9)=1.00000E-38
185 LET A(10)=1.00002E-38
190 LET A(11)=-1.00002E-38
195 LET A(12)=-1.00000E-38
200 LET A(13)=9.99998E-38
205 LET A(14)=10E-38
210 LET A(15)=123.455
215 LET A(16)=123.457
220 LET A(17)=123.455
225 LET A(18)=123.457
230 LET A(19)=9.87653E37
235 LET A(20)=9.87655E37
```

```
240 LET A(21)=123455
245 LET A(22)=123457
250 LET A(23)=123455
255 LET A(24)=123457
260 LET A(25)=123455
265 LET A(26)=123457
270 LET A(27)=987.653
275 LET A(28)=987.655
280 LET A(29)=1.23455E+6
285 LET A(30)=1.23457E+6
290 LET A(31)=1.23455E+8
295 LET A(32)=1.23457E+8
300 LET A(33)=1.23455E-2
305 LET A(34)=1.23457E-2
310 LET A(35)=1.23455E-4
315 LET A(36)=1.23457E-4
320 LET A(37)=.119999
325 LET A(38)=.120001
330 LET A(39)=.119999
335 LET A(40)=.120001
340 LET A(41)=-.120001
345 LET A(42)=-.119999
350 LET A(43)=-.119999
355 LET A(44)=.120001
360 LET A(45)=0
365 LET A(46)=0
370 LET A(47)=0
375 LET A(48)=0
380 LET A(49)=0
385 LET A(50)=0
390 LET A(51)=1.22999E9
395 LET A(52)=1.23001E9
400 LET A(53)=1.22999E9
405 LET A(54)=1.23001E9
410 LET A(55)=1.22999E9
415 LET A(56)=1.23001E9
420 LET A(57)=1.22999E-9
425 LET A(58)=1.23001E-9
430 LET A(59)=1.22999E-9
435 LET A(60)=1.23001E-9
440 LET A(61)=1.22999E-9
445 LET A(62)=1.23001E-9
450 LET A(63)=1.22999E-9
455 LET A(64)=1.23001E-9
460 LET A(65)=0
465 LET A(66)=0
470 LET A(67)=0
475 LET A(68)=0
480 LET A(69)=0
485 LET A(70)=0
490 LET A(71)=0
495 LET A(72)=0
500 LET A(73)=122.999
505 LET A(74)=123.001
510 LET A(75)=122.999
```

```
515 LET A(76)=123.001
520 LET A(77)=122.999
525 LET A(78)=123.001
530 LET A(79)=122.999
535 LET A(80)=123.001
540 LET A(81)=1.2345578E19
545 LET A(82)=1.2345779E19
550 LET A(83)=1.23455E15
555 LET A(84)=1.23457E15
560 LET A(85)=1.23455E-15
565 LET A(86)=1.23457E-15
570 LET A(87)=123455
575 LET A(88)=123457
580 LET A(89)=123455
585 LET A(90)=123457
1000 LET I=-1
1001 LET J=0
1002 PRINT "TRIAL #", "DATA VALUE", "RESULTS"
1005 LET F=0
1010 READ X
1020 IF X=-1 THEN 5000
1025 LET J=J+1
1030 LET I=I+2
1035 LET K=I+1
1040 IF X<A(I) THEN 1100
1050 IF X>A(K) THEN 1100
1060 PRINT J,X," OK "
1070 GOTO 1010
1100 PRINT J,X,"FAIL"
1110 LET F=F+1
1120 GOTO 1010
4000 DATA +.999999E38,-.999999E38,.00987654E40,987.654E-40
4010 DATA +1.00001E-38,-1.00001E-38,9.99999E-38
4015 DATA 123456.E-3,.123456E3
4020 DATA 9.87654E37,123456,123456.,123456.0,987.654
4030 DATA 1234560,123456000,.0123456,.000123456
4040 DATA .12,+.12,-.12,0.12
4050 DATA 0.0,+0,-.000
4060 DATA 1.23E9,1.23E09,1.23E+9,1.23E-9,1.23E-09,1.23E-0009
4070 DATA 000001.2300000E-000009,0E0,000.000E22,+000E55,0.0E-000
4080 DATA 123E0,123E000,123E-00,123E+0
4090 DATA 12345678901234567890,123456E10,0.0000123456E-10
4100 DATA 123456000000000E-9,0.00000000123456E15
4900 DATA -1
5000 PRINT
5010 IF F=0 THEN 5050
5020 PRINT "***** TEST FAILED ***** ";F;" CASE(S) HANDLED IMPROPERLY."
5030 GOTO 5100
5050 PRINT "***** TEST PASSED *****"
5100 PRINT
5110 PRINT " END TEST"
```

```
5120 PRINT
5130 PRINT "END PROGRAM 92"
5140 END
```

---

PROGRAM FILE 92: READ AND DATA STATEMENTS FOR NUMERIC DATA.  
ANSI STANDARD 5.2, 14.2, 14.4

SECTION 92.1: READ AND DATA STATEMENTS FOR NUMERIC DATA.

THIS SECTION TESTS THE PROCESSOR'S CAPABILITY TO HANDLE INTERNAL NUMERIC DATA USING THE DATA AND READ STATEMENTS. THE DATA-VALUES WILL BE COMPARED TO THOSE ASSIGNED TO AN ARRAY AND THE RESULTS DISPLAYED. IF THE ACCURACY IS LESS THAN SIX DIGITS, THE TEST WILL BE COUNTED AS A FAILURE.

BEGIN TEST

TRIAL #	DATA VALUE	RESULTS
1	9.99999E+37	OK
2	-9.99999E+37	OK
3	9.87654E+37	OK
4	9.87654E-38	OK
5	1.00001E-38	OK
6	-1.00001E-38	OK
7	9.99999E-38	OK
8	123.456	OK
9	123.456	OK
10	9.87654E+37	OK
11	123456	OK
12	123456	OK
13	123456	OK
14	987.654	OK
15	1.23456E+6	OK
16	1.23456E+8	OK
17	1.23456E-2	OK
18	1.23456E-4	OK
19	.12	OK
20	.12	OK
21	-.12	OK
22	.12	OK
23	0	OK
24	0	OK
25	0	OK
26	1.23000E+9	OK
27	1.23000E+9	OK
28	1.23000E+9	OK
29	1.23000E-9	OK
30	1.23000E-9	OK
31	1.23000E-9	OK
32	1.23000E-9	OK
33	0	OK
34	0	OK
35	0	OK

36	0	OK
37	123	OK
38	123	OK
39	123	OK
40	123	OK
41	1.23457E+19	OK
42	1.23456E+15	OK
43	1.23456E-15	OK
44	123456	OK
45	123456	OK

\*\*\*\*\* TEST PASSED \*\*\*\*\*

END TEST

END PROGRAM 92

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 93: READ AND DATA STATEMENTS FOR STRING DATA."
20 PRINT "      ANSI STANDARD 3.2, 5.2, 14.2, 14.4"
30 PRINT
40 PRINT "SECTION 93.1: READ AND DATA STATEMENTS FOR STRING DATA"
50 PRINT
60 PRINT "THIS SECTION TESTS THE ABILITY TO PICK UP STRING (CHARACTER)"
70 PRINT "DATA USING THE READ STATEMENT. VARIOUS SYNTACTIC VARIATIONS"
80 PRINT "ARE TESTED, ESPECIALLY FOR DATA FORMATS."
90 PRINT
100 PRINT "PLEASE NOTE THAT THIS ROUTINE IS STRONGLY DEPENDENT"
110 PRINT "ON THE CORRECT OPERATION OF THE STRING COMPARISON"
120 PRINT "FEATURE. IF THAT FEATURE IS INCORRECT, THIS ROUTINE"
130 PRINT "WILL LIKELY GIVE INVALID RESULTS."
140 PRINT
150 PRINT "THE VALUES FROM THE DATA STATEMENT ARE COMPARED TO THOSE"
160 PRINT "ASSIGNED TO STRING VARIABLES. THE EXPECTED AND ACTUAL"
170 PRINT "RESULTS ARE DISPLAYED, AND AN ERROR COUNT MAINTAINED."
180 PRINT "THE READ STATEMENT IS USED WITH SEVERAL VARIABLES IN"
190 PRINT "THE VARIABLE LIST."
200 PRINT
210 PRINT "                  BEGIN TEST"
220 PRINT
230 LET I=0
240 LET F=0
250 READ N
255 IF N=0 THEN 1200
260 LET I=I+1
270 LET I$="#"
280 LET J$="#"
290 LET K$="#"
300 LET L$="#"
310 ON I GOTO 410,420,430,440,450,460,470,480,490,500,510
410 LET I$="A"
411 LET J$="B"

```

```
412 LET K$="C"
413 LET L$="D"
414 GOTO 600
420 LET I$="EF"
421 LET J$="GHI"
422 LET K$="JKLM"
423 GOTO 600
430 LET I$="ABCDEFGHIJKLM"
431 LET J$="NOPQRSTUVWXYZ"
432 LET K$="0123456789"
433 LET L$="+. -"
434 GOTO 600
440 LET I$="EIGHTEEN POSITIONS"
441 GOTO 600
450 LET I$="ABC"
451 LET J$="DEF"
452 LET K$="GHI"
453 LET L$="2.1E3"
454 GOTO 600
460 LET I$="ABC   "
461 LET J$="    DEF   "
462 LET K$="    GHI"
463 GOTO 600
470 LET I$="A   B"
471 LET J$="C   D"
472 GOTO 600
480 LET I$="A   B"
481 LET J$="   C   D"
482 LET K$="E   F"
483 GOTO 600
490 LET I$="A"
491 LET J$="B,C"
492 LET K$="''"
493 LET L$="D"
494 GOTO 600
500 LET I$="ABCDEFGHIJKLM"
501 LET J$="NOPQRSTUVWXYZ"
502 LET K$="0123456789"
503 GOTO 600
510 LET I$="!#$%&!'()*+, -"
511 LET J$=".:/;<=>?^_"
512 GOTO 600
600 REM COMPARAND VALUES INITIALIZED
630 LET A$="#"
640 LET B$="#"
650 LET C$="#"
660 LET D$="#"
670 ON N GOTO 710,720,730,740
710 READ A$
715 GOTO 800
720 READ A$,B$
725 GOTO 800
730 READ A$,B$,C$
735 GOTO 800
740 READ A$,B$,C$,D$
```

```
745 GOTO 800
800 REM TEST VALUES NOW SET
810 IF A$<>I$ THEN 900
820 IF B$<>J$ THEN 900
830 IF C$<>K$ THEN 900
840 IF D$<>L$ THEN 900
850 LET S=0
860 GOTO 910
900 LET S=1
910 ON N GOTO 920,930,940,950
920 LET B$=""
925 LET J$=""
930 LET C$=""
935 LET K$=""
940 LET D$=""
945 LET L$=""
950 REM NOW SET UP FOR PRINT
960 PRINT
970 PRINT
980 PRINT "SHOULD BE: ***";I$;J$;K$;L$;"***"
990 PRINT "ACTUAL:    ***";A$;B$;C$;D$;"***"
1000 IF S=0 THEN 1100
1010 PRINT "TEST FAILED"
1020 LET F=F+1
1030 GOTO 250
1100 PRINT "TEST OK"
1110 GOTO 250
1200 PRINT
1210 PRINT
1220 IF F=0 THEN 1270
1230 PRINT "*** TEST FAILED: ";F;" CASE(S) HANDLED IMPROPERLY. ***"
1240 GOTO 1300
1270 PRINT "*** TEST PASSED ***"
1300 PRINT
1310 PRINT "                      END TEST"
1320 PRINT
1330 PRINT "END PROGRAM 93"
2010 DATA 4,"A",B,"C",D
2020 DATA 3,EF,"GHI",JKLM
2030 DATA 4,ABCDEFGHIJKLM,NOPQRSTUVWXYZ,0123456789,+. -
2040 DATA 1,EIGHTEEN POSITIONS
2050 DATA 4,ABC      , DEF      , GHI,2.1E3
2060 DATA 3,"ABC      ,," DEF      ,," GHI"
2070 DATA 2, A      B      , C      D
2080 DATA 3, A      B      , " C      D      " , E      F
2090 DATA 4,A,"B,C","", D
2100 DATA 3,"ABCDEFGHIJKLM","NOPQRSTUVWXYZ","0123456789"
2110 DATA 2,!#$%&'()*+, -", ".:/; <=>?^_
2120 DATA 0
2130 END
```

PROGRAM FILE 93: READ AND DATA STATEMENTS FOR STRING DATA.

ANSI STANDARD 3.2, 5.2, 14.2, 14.4

### SECTION 93.1: READ AND DATA STATEMENTS FOR STRING DATA

THIS SECTION TESTS THE ABILITY TO PICK UP STRING (CHARACTER) DATA USING THE READ STATEMENT. VARIOUS SYNTACTIC VARIATIONS ARE TESTED, ESPECIALLY FOR DATA FORMATS.

PLEASE NOTE THAT THIS ROUTINE IS STRONGLY DEPENDENT ON THE CORRECT OPERATION OF THE STRING COMPARISON FEATURE. IF THAT FEATURE IS INCORRECT, THIS ROUTINE WILL LIKELY GIVE INVALID RESULTS.

THE VALUES FROM THE DATA STATEMENT ARE COMPARED TO THOSE ASSIGNED TO STRING VARIABLES. THE EXPECTED AND ACTUAL RESULTS ARE DISPLAYED, AND AN ERROR COUNT MAINTAINED. THE READ STATEMENT IS USED WITH SEVERAL VARIABLES IN THE VARIABLE LIST.

BEGIN TEST

SHOULD BE: \*\*\*ABCD\*\*\*

ACTUAL: \*\*\*ABCD\*\*\*

TEST OK

SHOULD BE: \*\*\*EFGHIJKLM\*\*\*

ACTUAL: \*\*\*EFGHIJKLM\*\*\*

TEST OK

SHOULD BE: \*\*\*ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789+. -\*\*\*

ACTUAL: \*\*\*ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789+. -\*\*\*

TEST OK

SHOULD BE: \*\*\*EIGHTEEN POSITIONS\*\*\*

ACTUAL: \*\*\*EIGHTEEN POSITIONS\*\*\*

TEST OK

SHOULD BE: \*\*\*ABCDEFGH12.1E3\*\*\*

ACTUAL: \*\*\*ABCDEFGH12.1E3\*\*\*

TEST OK

SHOULD BE: \*\*\*ABC DEF GHI\*\*\*

ACTUAL: \*\*\*ABC DEF GHI\*\*\*

TEST OK

SHOULD BE: \*\*\*A BC D\*\*\*

ACTUAL: \*\*\*A BC D\*\*\*

TEST OK

SHOULD BE: \*\*\*A B C D E F\*\*\*

ACTUAL: \*\*\*A B C D E F\*\*\*

TEST OK

SHOULD BE: \*\*\*AB,CD\*\*\*  
ACTUAL: \*\*\*AB,CD\*\*\*  
TEST OK

SHOULD BE: \*\*\*ABCDEFGHIJKLMNPQRSTUVWXYZ0123456789\*\*\*  
ACTUAL: \*\*\*ABCDEFGHIJKLMNPQRSTUVWXYZ0123456789\*\*\*  
TEST OK

SHOULD BE: \*\*\*!#\$%&'()\*+,./:;<=>?^\_\*\*\*  
ACTUAL: \*\*\*!#\$%&'()\*+,./:;<=>?^\_\*\*\*  
TEST OK

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 93

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 94: READING DATA INTO SUBSCRIPTED VARIABLES."  
20 PRINT "      ANSI STANDARD 14.2, 14.4"  
30 PRINT  
40 PRINT "SECTION 94.1: READING DATA INTO SUBSCRIPTED VARIABLES."  
50 PRINT  
60 PRINT "THIS SECTION TESTS THAT THE NUMERIC DATA CAN BE READ INTO"  
70 PRINT "ARRAYS AND THAT SUBSCRIPTS ARE EVALUATED AFTER READING"  
80 PRINT "INTO PREVIOUS VARIABLES."  
90 PRINT  
100 PRINT "          BEGIN TEST"  
110 PRINT  
120 REM ARRAYS A AND C ARE GIVEN DATA THROUGH THE DATA AND READ  
130 REM AND COMPARED WITH B AND D WHICH ARE ASSIGNED THE RIGHT  
140 REM VALUES. IF THE ARRAYS MATCH TEST PASSES.  
150 FOR I=1 TO 5  
160 LET A(I)=I  
170 LET B(I)=I  
180 FOR J=1 TO 3  
190 LET C(I,J)=1000*I+J  
200 LET D(I,J)=1000*I+J  
210 NEXT J  
220 NEXT I  
230 LET J=2  
240 READ I,A(I),I,A(I),I,C(I,J),J,I,C(I,J)  
250 DATA 3,199,2,299,4,399,1,2,499  
260 LET B(3)=199  
270 LET B(2)=299  
280 LET D(4,2)=399  
290 LET D(2,1)=499  
300 PRINT "VALUES FOR A(I)"  
310 PRINT "I","A(I)","A(I)"  
320 PRINT ,,"SHOULD BE","ACTUAL"  
330 LET M$="PASSED. ***"
```

```

340 FOR I=1 TO 5
350 LET S$=""
360 IF A(I)=B(I) THEN 390
370 LET M$="FAILED. ***"
380 LET S$=" ***"
390 PRINT I,B(I),A(I),S$
400 NEXT I
410 PRINT
420 PRINT "*** TEST FOR ONE-DIMENSIONAL ARRAY ";M$
430 LET M$="PASSED. ***"
440 PRINT
450 PRINT "VALUES FOR C(I,J)"
460 PRINT "I=,,J=1", "J=2", "J=3"
470 FOR I=1 TO 5
480 PRINT I,"SHOULD BE:",D(I,1),D(I,2),D(I,3)
490 PRINT , "ACTUAL:",C(I,1),C(I,2),C(I,3)
500 PRINT ,
510 FOR J=1 TO 3
520 IF C(I,J)=D(I,J) THEN 560
530 LET M$="FAILED. ***"
540 PRINT " ***",
550 GOTO 570
560 PRINT "   ",
570 NEXT J
580 PRINT
590 NEXT I
600 PRINT
610 PRINT "*** TEST FOR TWO-DIMENSIONAL ARRAY ";M$
620 PRINT
630 PRINT "           END TEST"
640 PRINT
650 PRINT "END PROGRAM 94"
660 END

```

---

PROGRAM FILE 94: READING DATA INTO SUBSCRIPTED VARIABLES.

ANSI STANDARD 14.2, 14.4

#### SECTION 94.1: READING DATA INTO SUBSCRIPTED VARIABLES.

THIS SECTION TESTS THAT THE NUMERIC DATA CAN BE READ INTO ARRAYS AND THAT SUBSCRIPTS ARE EVALUATED AFTER READING INTO PREVIOUS VARIABLES.

BEGIN TEST

VALUES FOR A(I)

I	A(I)	A(I)
	SHOULD BE	ACTUAL
1	1	1
2	299	299
3	199	199

4                  4                  4  
 5                  5                  5

\*\*\* TEST FOR ONE-DIMENSIONAL ARRAY PASSED. \*\*\*

VALUES FOR C(I,J)

		J=1	J=2	J=3
I=	SHOULD BE:	1001	1002	1003
1	ACTUAL:	1001	1002	1003
2	SHOULD BE:	499	2002	2003
	ACTUAL:	499	2002	2003
3	SHOULD BE:	3001	3002	3003
	ACTUAL:	3001	3002	3003
4	SHOULD BE:	4001	399	4003
	ACTUAL:	4001	399	4003
5	SHOULD BE:	5001	5002	5003
	ACTUAL:	5001	5002	5003

\*\*\* TEST FOR TWO-DIMENSIONAL ARRAY PASSED. \*\*\*

END TEST

END PROGRAM 94

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

```

10 PRINT "PROGRAM FILE 95: GENERAL USE OF THE READ, DATA,"
15 PRINT "      AND RESTORE STATEMENTS."
20 PRINT "      ANSI STANDARD 14.2, 14.4"
30 PRINT
40 REM THE FOLLOWING LINES DESCRIBE THE DATA AND READ STRUCTURE OF
50 REM THIS PROGRAM. LINE #1 REPRESENTS THE DATA STRING. LINE #2
60 REM REPRESENTS THE FIRST SERIES OF READ STATEMENTS AND THE TYPE
70 REM (STRING OR NUMERIC) OF VARIABLE. LINE #3 REPRESENTS THE
80 REM SECOND SERIES OF READ STATEMENTS, AFTER THE FIRST RESTORE,
90 REM AND LINE #4, THE THIRD SERIES OF READS. SLASHES DELIMIT
100 REM THE RANGE OF INDIVIDUAL STATEMENTS
110 REM
120 REM LINE #1:   1 A 2 B 3 4/5 C D E/F 6/7 8 9 10/11
130 REM LINE #2:   N S N S/N N N S/S S S S S/N/S/ N   N
140 REM LINE #3:   S S S S S S/S S S S N N S
150 REM LINE #4:   N S S S N
160 REM

```

```
170 PRINT "SECTION 95.1: GENERAL SYNTAX AND SEMANTICS OF READ AND DATA"
180 PRINT
190 PRINT "THIS SECTION EXERCISES SOME OF THE GENERAL CAPABILITIES"
200 PRINT "AND PROPERTIES OF THE READ AND DATA STATEMENTS, AS LISTED:"
210 PRINT " 1) READ AND DATA CAN MIX STRING AND NUMERIC DATA"
220 PRINT " 2) CONTROL FALLS THROUGH DATA STATEMENTS WITHOUT EFFECT"
230 PRINT " 3) DATA STATEMENTS MAY BE SCATTERED THROUGHOUT SOURCE"
240 PRINT " 4) READ AND DATA LISTS CAN OVERLAP"
250 PRINT " 5) READ CAN ACCESS DATA ANYWHERE IN SOURCE"
260 PRINT
270 PRINT "          BEGIN TEST"
280 PRINT
290 LET F=0
300 LET W$="READ OK"
310 LET X$="READ FAILED"
320 LET Y$="SHOULD BE:"
330 LET Z$="ACTUAL:    "
340 DATA 1,A,2,B,3,4
350 READ A,A$,B,B$
360 PRINT
370 PRINT Y$,1,"A",2,"B"
380 PRINT Z$,A,A$,B,B$
390 IF A<>1 THEN 450
400 IF A$<>"A" THEN 450
410 IF B<>2 THEN 450
420 IF B$<>"B" THEN 450
430 PRINT W$
440 GOTO 470
450 PRINT X$
460 LET F=F+1
470 READ A,B,C,A$
480 PRINT
490 PRINT Y$,3,4,5,"C"
500 PRINT Z$,A,B,C,A$
510 IF A<>3 THEN 570
520 IF B<>4 THEN 570
530 IF C<>5 THEN 570
540 IF A$<>"C" THEN 570
550 PRINT W$
560 GOTO 590
570 PRINT X$
580 LET F=F+1
590 GOTO 620
600 REM TEST BRANCHING TO A DATA STATEMENT
610 DATA 5,C,D,E
620 DATA F,6
630 DATA 7,8,9,10
640 READ A$,B$,C$,D$,E$,A
650 PRINT
```

```
660 PRINT Y$,"D","E","F","6","7",8
670 PRINT Z$,A$,B$,C$,D$,E$,A
680 IF A$<>"D" THEN 760
690 IF B$<>"E" THEN 760
700 IF C$<>"F" THEN 760
710 IF D$<>"6" THEN 760
720 IF E$<>"7" THEN 760
730 IF A<>8 THEN 760
740 PRINT W$
750 GOTO 780
760 PRINT X$
770 LET F=F+1
780 READ A$
790 PRINT
800 PRINT Y$,"9"
810 PRINT Z$,A$
820 IF A$<>"9" THEN 850
830 PRINT W$
840 GOTO 870
850 PRINT X$
860 LET F=F+1
870 READ A,B
880 PRINT
890 PRINT Y$,10,11
900 PRINT Z$,A,B
910 IF A<>10 THEN 950
920 IF B<>11 THEN 950
930 PRINT W$
940 GOTO 970
950 PRINT X$
960 LET F=F+1
970 DATA 11
980 PRINT
990 IF F=0 THEN 1020
1000 PRINT "**** TEST FAILED IN ";F;" CASE(S). ****"
1010 GOTO 1030
1020 PRINT "**** TEST PASSED ****"
1030 PRINT
1040 PRINT "           END TEST"
1050 PRINT
1060 PRINT "SECTION 95.2: RESTORE STATEMENT"
1070 PRINT
1080 PRINT "THIS SECTION TESTS THE CAPABILITY OF THE RESTORE STATEMENT"
1090 PRINT "TO ENABLE READING FROM THE BEGINNING OF THE DATA SEQUENCE."
1100 PRINT "FURTHERMORE, DATA VALUES WHICH WERE ORIGINALLY READ AS"
1110 PRINT "NUMERIC MAY BE SUBSEQUENTLY READ AS STRINGS, AND "
1120 PRINT "VICE-VERSA, AS LONG AS DATA-TYPE IS COMPATIBLE WITH"
1130 PRINT "VARIABLE-TYPE."
1140 PRINT
1150 PRINT "           BEGIN TEST"
1160 LET F=0
1170 PRINT
```

```
1180 PRINT "EXECUTE FIRST RESTORE"
1190 RESTORE
1200 READ A$,B$,C$,D$,E$,F$,G$
1210 PRINT
1220 PRINT Y$,"1","A","2","B","3","4","5"
1230 PRINT Z$,A$,B$,C$,D$,E$,F$,G$
1240 IF A$<>"1" THEN 1330
1250 IF B$<>"A" THEN 1330
1260 IF C$<>"2" THEN 1330
1270 IF D$<>"B" THEN 1330
1280 IF E$<>"3" THEN 1330
1290 IF F$<>"4" THEN 1330
1300 IF G$<>"5" THEN 1330
1310 PRINT W$
1320 GOTO 1350
1330 PRINT X$
1340 LET F=F+1
1350 READ A$,B$,C$,D$,A,B,E$
1360 PRINT
1370 PRINT Y$,"C","D","E","F",6,7,"8"
1380 PRINT Z$,A$,B$,C$,D$,A,B,E$
1390 IF A$<>"C" THEN 1480
1400 IF B$<>"D" THEN 1480
1410 IF C$<>"E" THEN 1480
1420 IF D$<>"F" THEN 1480
1430 IF A<>6 THEN 1480
1440 IF B<>7 THEN 1480
1450 IF E$<>"8" THEN 1480
1460 PRINT W$
1470 GOTO 1500
1480 PRINT X$
1490 LET F=F+1
1500 PRINT
1510 PRINT "EXECUTE SECOND RESTORE"
1520 RESTORE
1530 PRINT
1540 READ A,A$,B$,C$,B
1550 PRINT
1560 PRINT Y$,1,"A","2","B",3
1570 PRINT Z$,A,A$,B$,C$,B
1580 IF A<>1 THEN 1650
1590 IF A$<>"A" THEN 1650
1600 IF B$<>"2" THEN 1650
1610 IF C$<>"B" THEN 1650
1620 IF B<>3 THEN 1650
1630 PRINT W$
1640 GOTO 1670
1650 PRINT X$
1660 LET F=F+1
1670 PRINT
1680 PRINT
```

```

1690 IF F=0 THEN 1720
1700 PRINT "*** TEST FAILED IN ";F;" CASE(S). ***"
1710 GOTO 1730
1720 PRINT "*** TEST PASSED ***"
1730 PRINT
1740 PRINT "           END TEST"
1750 PRINT
1760 PRINT "END PROGRAM 95"
1770 END

```

---

PROGRAM FILE 95: GENERAL USE OF THE READ, DATA,  
AND RESTORE STATEMENTS.

ANSI STANDARD 14.2, 14.4

#### SECTION 95.1: GENERAL SYNTAX AND SEMANTICS OF READ AND DATA

THIS SECTION EXERCISES SOME OF THE GENERAL CAPABILITIES  
AND PROPERTIES OF THE READ AND DATA STATEMENTS, AS LISTED:

- 1) READ AND DATA CAN MIX STRING AND NUMERIC DATA
- 2) CONTROL FALLS THROUGH DATA STATEMENTS WITHOUT EFFECT
- 3) DATA STATEMENTS MAY BE SCATTERED THROUGHOUT SOURCE
- 4) READ AND DATA LISTS CAN OVERLAP
- 5) READ CAN ACCESS DATA ANYWHERE IN SOURCE

BEGIN TEST

SHOULD BE:	1	A	2	B
ACTUAL:	1	A	2	B
READ OK				

SHOULD BE:	3	4	5	C
ACTUAL:	3	4	5	C
READ OK				

SHOULD BE:	D	E	F	6
7	8			
ACTUAL:	D	E	F	6
7	8			
READ OK				

SHOULD BE:	9			
ACTUAL:	9			
READ OK				

SHOULD BE:	10	11		
ACTUAL:	10	11		
READ OK				

\*\*\* TEST PASSED \*\*\*

END TEST

## SECTION 95.2: RESTORE STATEMENT

THIS SECTION TESTS THE CAPABILITY OF THE RESTORE STATEMENT TO ENABLE READING FROM THE BEGINNING OF THE DATA SEQUENCE. FURTHERMORE, DATA VALUES WHICH WERE ORIGINALLY READ AS NUMERIC MAY BE SUBSEQUENTLY READ AS STRINGS, AND VICE-VERSA, AS LONG AS DATA-TYPE IS COMPATIBLE WITH VARIABLE-TYPE.

BEGIN TEST

EXECUTE FIRST RESTORE

SHOULD BE:	1	A	2	B
3	4	5		
ACTUAL:	1	A	2	B
3	4	5		

READ OK

SHOULD BE:	C	D	E	F
6	7	8		
ACTUAL:	C	D	E	F
6	7	8		

READ OK

EXECUTE SECOND RESTORE

SHOULD BE:	1	A	2	B
3				
ACTUAL:	1	A	2	B
3				

READ OK

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 95

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 96: EXCEPTION - NUMERIC UNDERFLOW WHEN"
20 PRINT "      READING DATA CAUSES REPLACEMENT BY ZERO."
30 PRINT "      ANSI STANDARD 5.6, 14.4"
40 PRINT
50 PRINT "SECTION 96.1: EXCEPTION - NUMERIC UNDERFLOW WHEN"
60 PRINT "      READING DATA CAUSES REPLACEMENT BY ZERO."
70 PRINT
80 PRINT "TO PASS THIS TEST:"
90 PRINT
100 PRINT "    1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE"
110 PRINT "      DISPLAYED (NOT MANDATORY), AND"
120 PRINT

```

```
130 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"
140 PRINT "          OF THE CONSTANT"
150 PRINT
160 PRINT "                  BEGIN TEST."
170 PRINT
180 DATA 1E-99999
190 READ A
200 PRINT "RESULTING VALUE =";A
210 PRINT
220 IF A=0 THEN 260
230 PRINT "RESULTING VALUE <> 0. IF VALUE <> 1E-99999, THEN,"
240 PRINT "*** TEST FAILED ***"
250 GOTO 270
260 PRINT "*** TEST PASSED ***"
270 PRINT
280 PRINT "                  END TEST."
290 PRINT
300 PRINT "END PROGRAM 96"
310 END
```

---

PROGRAM FILE 96: EXCEPTION - NUMERIC UNDERFLOW WHEN  
READING DATA CAUSES REPLACEMENT BY ZERO.  
ANSI STANDARD 5.6, 14.4

SECTION 96.1: EXCEPTION - NUMERIC UNDERFLOW WHEN  
READING DATA CAUSES REPLACEMENT BY ZERO.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE  
DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE  
OF THE CONSTANT

BEGIN TEST.

RESULTING VALUE = 0

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 96

```
*****  
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 97: EXCEPTION - INSUFFICIENT DATA FOR READ."
20 PRINT "    ANSI STANDARD 14.5"
30 PRINT
```

```
40 PRINT "SECTION 97.1: EXCEPTION - INSUFFICIENT DATA FOR READ."
50 PRINT
60 PRINT "THIS TEST CAUSES A READ OF THREE VARIABLES WHEN ONLY TWO"
70 PRINT "DATA ARE AVAILABLE."
80 PRINT
90 PRINT "TO PASS THIS TEST:"
100 PRINT
110 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
120 PRINT "          DISPLAYED, AND"
130 PRINT
140 PRINT "      2) THE PROGRAM MUST TERMINATE."
150 PRINT
160 PRINT "                  BEGIN TEST"
170 PRINT
180 DATA 5,6
190 LET A=1
200 LET B=2
210 LET C=3
220 PRINT "ABOUT TO EXECUTE READ - "
230 READ A,B,C
240 PRINT
245 PRINT "*** TEST FAILED ***"
260 PRINT "A, B, AND C WERE INITIALIZED TO 1, 2, AND 3"
270 PRINT "RESPECTIVELY, AND THEN 'READ A,B,C' WAS ATTEMPTED"
280 PRINT "FROM A DATA-LIST CONTAINING 5,6. RESULTS BELOW:"
290 PRINT
300 PRINT "A=";A;"      B=";B;"      C=";C
310 PRINT
320 PRINT "                  END TEST"
330 PRINT
340 PRINT "END PROGRAM 97"
350 END
```

---

PROGRAM FILE 97: EXCEPTION - INSUFFICIENT DATA FOR READ.  
ANSI STANDARD 14.5

SECTION 97.1: EXCEPTION - INSUFFICIENT DATA FOR READ.

THIS TEST CAUSES A READ OF THREE VARIABLES WHEN ONLY TWO  
DATA ARE AVAILABLE.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE.

BEGIN TEST

ABOUT TO EXECUTE READ -

? OUT OF DATA IN LINE 230

A grid of small black asterisks arranged in a regular pattern on a white background.

```
10 PRINT "PROGRAM FILE 98: EXCEPTION - READING UNQUOTED STRING"
15 PRINT "           DATA INTO A NUMERIC VARIABLE."
20 PRINT "           ANSI STANDARD 14.5"
30 PRINT
40 PRINT "SECTION 98.1: EXCEPTION - READING UNQUOTED STRING"
50 PRINT "           DATA INTO A NUMERIC VARIABLE."
60 PRINT
70 PRINT "THIS TEST CAUSES A READ TO BE ATTEMPTED FROM STRING-TYPE"
80 PRINT "DATA INTO A NUMERIC VARIABLE."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "           DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST TERMINATE."
200 PRINT
210 PRINT "           BEGIN TEST"
220 PRINT
230 LET A=1
240 LET B=2
250 LET C=3
260 DATA 5,6,2D3
270 PRINT "ABOUT TO READ -"
290 READ A,B,C
300 PRINT
310 PRINT "*** TEST FAILED ***"
320 PRINT "VARIABLES A, B, AND C WERE INITIALIZED TO 1, 2, AND 3,"
330 PRINT "RESPECTIVELY, AND THEN READ INTO FROM A DATA-LIST"
340 PRINT "CONTAINING 5, 6, AND '2D3'. RESULTS BELOW:"
350 PRINT
360 PRINT "A=";A;"     B=";B;"     C=";C
370 PRINT
380 PRINT "           END TEST"
390 PRINT
400 PRINT "END PROGRAM 98"
410 END
```

PROGRAM FILE 98: EXCEPTION - READING UNQUOTED STRING  
DATA INTO A NUMERIC VARIABLE.  
ANSI STANDARD 14.5

## SECTION 98.1: EXCEPTION - READING UNQUOTED STRING DATA INTO A NUMERIC VARIABLE.

THIS TEST CAUSES A READ TO BE ATTEMPTED FROM STRING-TYPE DATA INTO A NUMERIC VARIABLE.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE.

BEGIN TEST

ABOUT TO READ -

? DATA NOT IN CORRECT FORM IN LINE 260

```
*****
*****
*****
```

```

10 PRINT "PROGRAM FILE 99: EXCEPTION - READING QUOTED STRING"
15 PRINT "          DATA INTO A NUMERIC VARIABLE."
20 PRINT "          ANSI STANDARD 14.5"
30 PRINT
40 PRINT "SECTION 99.1: EXCEPTION - READING QUOTED STRING"
50 PRINT "          DATA INTO A NUMERIC VARIABLE."
60 PRINT
70 PRINT "THIS TEST CAUSES A READ TO BE ATTEMPTED FROM STRING-TYPE"
80 PRINT "DATA INTO A NUMERIC VARIABLE."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST TERMINATE."
170 PRINT
180 PRINT "          BEGIN TEST"
190 PRINT
200 PRINT
210 PRINT "          BEGIN TEST"
220 PRINT
230 LET A=1
240 LET B=2
250 LET C=3
260 DATA 5,6,"7"
270 PRINT "ABOUT TO READ -"
280 READ A,B,C
290 PRINT
300 PRINT
310 PRINT "*** TEST FAILED ***"
320 PRINT "VARIABLES A, B, AND C WERE INITIALIZED TO 1, 2, AND 3,"
330 PRINT "RESPECTIVELY, AND THEN READ INTO FROM A DATA-LIST"
340 PRINT "CONTAINING 5, 6, AND A QUOTED '7'. RESULTS BELOW:"
350 PRINT
360 PRINT "A=";A;"      B=";B;"      C=";C
370 PRINT
380 PRINT "          END TEST"
```

```
390 PRINT
400 PRINT "END PROGRAM 99"
410 END
```

---

PROGRAM FILE 99: EXCEPTION - READING QUOTED STRING  
DATA INTO A NUMERIC VARIABLE.

ANSI STANDARD 14.5

SECTION 99.1: EXCEPTION - READING QUOTED STRING  
DATA INTO A NUMERIC VARIABLE.

THIS TEST CAUSES A READ TO BE ATTEMPTED FROM STRING-TYPE  
DATA INTO A NUMERIC VARIABLE.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE.

BEGIN TEST

ABOUT TO READ -

? DATA NOT IN CORRECT FORM IN LINE 260

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
1 PRINT "PROGRAM FILE 100: EXCEPTION - STRING OVERFLOW ON READ."
2 PRINT "      ANSI STANDARD 14.5"
3 PRINT
5 DATA ABC1234567890123456789012345678901234567890123456789XYZ
45 PRINT "SECTION 100.1 EXCEPTION - STRING OVERFLOW ON READ."
55 PRINT
65 PRINT "THIS TEST READS A LONG STRING (65 CHARACTERS) INTO A"
75 PRINT "VARIABLE. IN ORDER TO PASS, THE PROCESSOR MUST EITHER:"
85 PRINT
95 PRINT "1) CORRECTLY ACCEPT THE STRING AND RETAIN ITS VALUE IN"
105 PRINT "   THE VARIABLE, AS DEMONSTRATED IN THE PRINTING, OR"
110 PRINT
115 PRINT "2) DETECT STRING OVERFLOW, REPORT THE EXCEPTION AND"
125 PRINT "   TERMINATE THE PROGRAM, AS THIS EXCEPTION IS FATAL."
135 PRINT
145 PRINT "               BEGIN TEST"
155 PRINT
165 LET A$="INITIALIZED"
175 PRINT "ABOUT TO ATTEMPT READ. FATAL EXCEPTION MAY OCCUR NOW:"
185 PRINT
```

```

195 READ A$
205 PRINT "IF THIS LINE PRINTS, NO EXCEPTION SHOULD HAVE BEEN REPORTED"
215 PRINT "AND THE TWO LINES BELOW (BEGINNING WITH 'ABC' AND ENDING"
220 PRINT "WITH 'XYZ') SHOULD BE IDENTICAL. OTHERWISE,"
225 PRINT "*** TEST FAILS ***"
230 PRINT
235 PRINT "(FIRST LINE IS CONSTANT, SECOND IS FROM THE VARIABLE.)"
245 PRINT
255 PRINT "ABC123456789012345678901234567890";
265 PRINT "12345678901234567890123456789XYZ"
275 PRINT A$
285 PRINT
295 PRINT "           END TEST"
305 PRINT
315 PRINT "END PROGRAM 100"
325 END

```

---

PROGRAM FILE 100: EXCEPTION - STRING OVERFLOW ON READ.  
ANSI STANDARD 14.5

#### SECTION 100.1 EXCEPTION - STRING OVERFLOW ON READ.

THIS TEST READS A LONG STRING (65 CHARACTERS) INTO A VARIABLE. IN ORDER TO PASS, THE PROCESSOR MUST EITHER:

- 1) CORRECTLY ACCEPT THE STRING AND RETAIN ITS VALUE IN THE VARIABLE, AS DEMONSTRATED IN THE PRINTING, OR
- 2) DETECT STRING OVERFLOW, REPORT THE EXCEPTION AND TERMINATE THE PROGRAM, AS THIS EXCEPTION IS FATAL.

BEGIN TEST

ABOUT TO ATTEMPT READ. FATAL EXCEPTION MAY OCCUR NOW:

% STRING OVERFLOW IN LINE 195

```
*****
*****
*****
```

```

10 PRINT "PROGRAM FILE 101: EXCEPTION - NUMERIC OVERFLOW ON READ."
20 PRINT "      ANSI STANDARD 14.5"
30 PRINT
40 PRINT "SECTION 101.1: EXCEPTION - POSITIVE OVERFLOW ON READ."
50 PRINT
60 PRINT "THIS TEST CAUSES 9.9E99999 TO BE READ INTO A NUMERIC"
70 PRINT "VARIABLE. IN ORDER TO PASS, THE PROCESSOR MUST EITHER:"
80 PRINT
90 PRINT "1) CORRECTLY ACCEPT THE NUMBER AND RETAIN ITS VALUE IN"
100 PRINT "     THE VARIABLE, AS DEMONSTRATED IN THE PRINTING, OR"
110 PRINT

```

```
120 PRINT "2) DETECT NUMERIC OVERFLOW, REPORT THE EXCEPTION,"  
130 PRINT "    SUPPLY POSITIVE MACHINE INFINITY AND CONTINUE."  
140 PRINT  
150 PRINT "                BEGIN TEST"  
160 PRINT  
170 PRINT "ABOUT TO EXECUTE READ."  
180 DATA 9.9E99999  
190 READ A  
200 PRINT  
210 PRINT "RESULTING VALUE IN VARIABLE = ";A  
220 PRINT  
225 IF A <= .99E38 THEN 250  
230 PRINT "IF EXCEPTION OCCURRED, VARIABLE SHOULD = POSITIVE INFINITY."  
240 PRINT "IF NOT, IT SHOULD = 9.9E99999; OTHERWISE,"  
250 PRINT "*** TEST FAILED ***"  
260 PRINT  
270 PRINT "                END TEST"  
280 PRINT  
290 PRINT "SECTION 101.2: EXCEPTION - NEGATIVE OVERFLOW ON READ"  
300 PRINT  
310 PRINT "THIS SECTION IS IDENTICAL TO THE PREVIOUS, EXCEPT THAT"  
320 PRINT "-9.9E99999 IS USED TO CAUSE OVERFLOW."  
330 PRINT  
340 PRINT "                BEGIN TEST"  
350 PRINT  
360 PRINT "ABOUT TO EXECUTE READ."  
370 DATA -9.9E99999  
380 READ A  
390 PRINT  
400 PRINT "RESULTING VALUE IN VARIABLE = ";A  
410 PRINT  
415 IF A >= -.99E38 THEN 440  
420 PRINT "IF EXCEPTION OCCURRED, VARIABLE SHOULD = NEGATIVE INFINITY."  
430 PRINT "IF NOT, IT SHOULD = -9.9E99999; OTHERWISE,"  
440 PRINT "*** TEST FAILED ***"  
450 PRINT  
460 PRINT "                END TEST"  
470 PRINT  
480 PRINT "END PROGRAM 101"  
490 END
```

---

PROGRAM FILE 101: EXCEPTION - NUMERIC OVERFLOW ON READ.  
ANSI STANDARD 14.5

SECTION 101.1: EXCEPTION - POSITIVE OVERFLOW ON READ.

THIS TEST CAUSES 9.9E99999 TO BE READ INTO A NUMERIC VARIABLE. IN ORDER TO PASS, THE PROCESSOR MUST EITHER:

- 1) CORRECTLY ACCEPT THE NUMBER AND RETAIN ITS VALUE IN THE VARIABLE, AS DEMONSTRATED IN THE PRINTING, OR

2) DETECT NUMERIC OVERFLOW, REPORT THE EXCEPTION,  
SUPPLY POSITIVE MACHINE INFINITY AND CONTINUE.

BEGIN TEST

ABOUT TO EXECUTE READ.

% OVERFLOW IN LINE 190  
RESULTING VALUE IN VARIABLE = 1.70141E+38

IF EXCEPTION OCCURRED, VARIABLE SHOULD = POSITIVE INFINITY.  
IF NOT, IT SHOULD = 9.9E99999; OTHERWISE,  
\*\*\* TEST FAILED \*\*\*

END TEST

#### SECTION 101.2: EXCEPTION - NEGATIVE OVERFLOW ON READ

THIS SECTION IS IDENTICAL TO THE PREVIOUS, EXCEPT THAT  
-9.9E99999 IS USED TO CAUSE OVERFLOW.

BEGIN TEST

ABOUT TO EXECUTE READ.

% OVERFLOW IN LINE 380  
RESULTING VALUE IN VARIABLE = -1.70141E+38

IF EXCEPTION OCCURRED, VARIABLE SHOULD = NEGATIVE INFINITY.  
IF NOT, IT SHOULD = -9.9E99999; OTHERWISE,  
\*\*\* TEST FAILED \*\*\*

END TEST

END PROGRAM 101

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 102: ERROR - ILLEGAL CHARACTER IN UNQUOTED"
15 PRINT "      STRING IN DATA STATEMENT."
20 PRINT "      ANSI STANDARD 3.2, 14.2"
30 PRINT
40 PRINT "SECTION 102.1: ERROR - ILLEGAL CHARACTER IN UNQUOTED"
45 PRINT "      STRING IN DATA STATEMENT."
50 PRINT
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT

```

```
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
183 PRINT "                  BEGIN TEST"
187 PRINT
188 PRINT "A$, B$, AND C$ ARE SET TO 'AAAAAA', 'BBBBBB', AND 'CCCCCC'."
190 PRINT "THE TEST THEN TRIES TO READ THE UNQUOTED STRING 'ABC' INTO"
200 PRINT "A$, 'D?F' INTO B$, AND 'GHI' INTO C$. THE SECOND OF THESE"
210 PRINT "OPERATIONS IS SYNTACTICALLY ILLEGAL, AS '?' IS NOT A"
220 PRINT "PLAIN-STRING-CHARACTER."
250 PRINT
260 LET A$="AAAAAA"
270 LET B$="BBBBBB"
280 LET C$="CCCCCC"
290 DATA ABC,D?F,GHI
300 READ A$,B$,C$
305 PRINT "RESULTING VALUES:"
310 PRINT "A$=";A$
320 PRINT "B$=";B$
330 PRINT "C$=";C$
340 PRINT
350 PRINT "                  END TEST"
360 PRINT
370 PRINT "END PROGRAM 102"
380 END
```

? DATA NOT IN CORRECT FORM IN LINE 290

```
10 PRINT "PROGRAM FILE 103: ERROR - READING QUOTED STRINGS CONTAINING"
15 PRINT "           SINGLE QUOTE."
20 PRINT "           ANSI STANDARD 3.2, 14.2"
30 PRINT
40 PRINT "SECTION 103.1: ERROR - READING QUOTED STRINGS CONTAINING "
45 PRINT "           SINGLE QUOTE."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING A SINGLE OCCURRENCE OF THE QUOTE CHARACTER WITHIN"
80 PRINT "THE QUOTED STRING OF A DATA-STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "           ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "           BY THE PROCESSOR, OR"
160 PRINT
```



```
240 READ A$  
250 PRINT "A READ STATEMENT HAS ATTEMPTED TO ASSOCIATE THE "  
260 PRINT "STRING CONSISTING OF AN ASTERISK, TWO CONSECUTIVE"  
270 PRINT "QUOTE CHARACTERS AND A QUESTION MARK WITH A STRING "  
280 PRINT "VARIABLE."  
290 PRINT  
300 PRINT "VARIABLE=";A$  
310 PRINT  
315 DATA "*""?"  
320 PRINT " END TEST."  
330 PRINT  
340 PRINT "END PROGRAM 104."  
350 END
```

---

? DATA NOT IN CORRECT FORM IN LINE 315

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 105: ERROR - NULL DATUM IN DATA-LIST."  
20 PRINT " ANSI STANDARD 14.2"  
30 PRINT  
40 PRINT "SECTION 105.1: ERROR - NULL DATUM IN DATA-LIST."  
50 PRINT  
60 PRINT  
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
90 PRINT  
100 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
110 PRINT " ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
120 PRINT " BY THE PROCESSOR, OR"  
130 PRINT  
140 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
150 PRINT  
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
170 PRINT "FOR DETAILED CRITERIA."  
180 PRINT  
240 PRINT " BEGIN TEST"  
250 PRINT  
260 LET A$="AAAAAA"  
270 LET B$="BBBBBB"  
280 LET C$="CCCCCC"  
281 PRINT "A$, B$, AND C$ INITIALIZED TO 'AAAAAA', 'BBBBBB', AND"  
282 PRINT "'CCCCCC', RESPECTIVELY. ABOUT TO READ A$, B$, C$ FROM"  
283 PRINT "'DATA ABC,,GHI,JKL'."  
285 PRINT
```

```
290 DATA ABC,,GHI,JKL
300 READ A$,B$,C$
303 PRINT "RESULTING VALUES:"
310 PRINT "A$=";A$
320 PRINT "B$=";B$
330 PRINT "C$=";C$
340 PRINT
350 PRINT "                END TEST"
360 PRINT
370 PRINT "END PROGRAM 105"
380 END
```

{

---

```
? DATA NOT IN CORRECT FORM IN LINE 290
```

```
*****  
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 106: ERROR - NULL ENTRY IN READ'S VARIABLE-LIST."
20 PRINT "      ANSI STANDARD 14.2"
30 PRINT
40 PRINT "SECTION 106.1: ERROR - NULL ENTRY IN READ'S VARIABLE-LIST."
50 PRINT
60 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
70 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
80 PRINT
90 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
100 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
110 PRINT "          BY THE PROCESSOR, OR"
120 PRINT
130 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
140 PRINT
150 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
160 PRINT "FOR DETAILED CRITERIA."
170 PRINT
180 PRINT "                BEGIN TEST"
190 PRINT
200 LET A$="AAAAAA"
210 LET B$="BBBBBB"
220 LET C$="CCCCCC"
230 PRINT "A$, B$, AND C$ INITIALIZED TO 'AAAAAA', 'BBBBBB', AND"
240 PRINT "'CCCCCC', RESPECTIVELY. ABOUT TO EXECUTE 'READ A$,,C$'"
250 PRINT "FROM 'DATA ABC,DEF,GHI'."'
260 DATA ABC,DEF,GHI
270 READ A$,,C$
280 PRINT
290 PRINT "RESULTING VALUES:"
300 PRINT "A$=";A$
310 PRINT "B$=";B$
320 PRINT "C$=";C$
330 PRINT
340 PRINT "                END TEST"
```

```

350 PRINT
360 PRINT "END PROGRAM 106"
370 END

```

---

? , WAS SEEN WHERE A LETTER WAS EXPECTED IN LINE 270

```
*****  
*****  
*****
```

```

10 PRINT "PROGRAM FILE 107: INPUT OF NUMERIC CONSTANTS."
20 PRINT "      ANSI STANDARD 5.2, 13.2, 13.4"
30 PRINT
40 PRINT "SECTION 107.1: INPUT OF NUMERIC CONSTANTS."
50 PRINT
60 PRINT " THIS SECTION TESTS THE PROCESSOR'S CAPABILITY TO INPUT"
70 PRINT " NUMERIC VALUES CORRECTLY. IF AN INPUT VALUE IS NOT ACCURATE"
80 PRINT " TO AT LEAST SIX DIGITS, IT IS COUNTED AS A FAILURE."
90 PRINT
100 PRINT " ENTER THE NUMBERS BELOW EXACTLY AS REQUESTED."
110 PRINT " FOR USER CONVENIENCE, THE NUMBER TO BE ENTERED IS INDENTED"
120 PRINT " TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO"
130 PRINT " POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE"
140 PRINT " ALIGNED WITH THE PROMPT STRING."
150 PRINT
160 PRINT "ALL THE REPLIES PROMPTED FOR ARE VALID, THEREFORE "
170 PRINT "IF THE PROCESSOR REJECTS ANY OF THEM,"
180 PRINT "*** TEST FAILED ***"
190 PRINT
200 REM   FORMAT OF DATA:
210 REM   1 - CORRECT NUMERIC VALUE
220 REM   2 - LOWER LIMIT
230 REM   3 - UPPER LIMIT
240 REM   4 - PROMPT (HOW NUMBER IS TO BE ENTERED)
250 REM   5 - STRING FOR CORRECT VALUE
260 DATA +.999999E38,.999998E38,1E38,"+.999999E38"," 9.99999E+37"
270 DATA -.999999E38,-1E38,-.999998E38,"-.999999E38","-9.99999E+37"
280 DATA +1.00001E-38,1E-38,1.00002E-38,"+1.00001E-38"," 1.00001E-38"
290 DATA -1.00001E-38,-1.00002E-38,-1E-38,"-1.00001E-38","-1.00001E-38"
300 DATA 9.99999E-38,9.99998E-38,10E-38,"9.99999E-38"," 9.99999E-38"
310 DATA 9.87654E37,9.87653E37,9.87655E37,"9.87654E37"," 9.87654E+37"
320 DATA 123456,123455,123457,"123456"," 123456"
330 DATA 123456,123455,123457,"123456."," 123456"
340 DATA 123456,123455,123457,"123456.0"," 123456"
350 DATA 987.654,987.653,987.655,"987.654"," 987.654"
360 DATA 1.23456E+6,1.23455E6,1.23457E6,"1234560"," 1.23456E+6"
370 DATA 1.23456E+8,1.23455E8,1.23457E8,"123456000"," 1.23456E+8"
380 DATA 1.23456E-2,1.23455E-2,1.23457E-2,".0123456"," 1.23456E-2"
390 DATA 1.23456E-4,1.23455E-4,1.23457E-4,".000123456"," 1.23456E-4"
400 DATA .12,.119999,.120001,".12"," .12"
410 DATA .12,.119999,.120001,"+.12"," .12"
420 DATA -.12,-.120001,-.119999,"-.12","-.12"
430 DATA .12,.119999,.120001,"0.12"," .12"

```

```

440 DATA 0,0,0,"0.0"," 0"
450 DATA 0,0,0,"+0"," 0"
460 DATA 0,0,0,"-.000"," 0"
470 DATA 1.23E9,1.22999E9,1.23001E9,"1.23E9"," 1.23000E+9"
480 DATA 1.23E9,1.22999E9,1.23001E9,"1.23E09"," 1.23000E+9"
490 DATA 1.23E9,1.22999E9,1.23001E9,"1.23E+9"," 1.23000E+9"
500 DATA 1.23E-9,1.22999E-9,1.23001E-9,"1.23E-9"," 1.23000E-9"
510 DATA 1.23E-9,1.22999E-9,1.23001E-9,"1.23E-09"," 1.23000E-9"
520 DATA 1.23E-9,1.22999E-9,1.23001E-9,"1.23E-0009"," 1.23000E-9"
530 DATA 1.23E-9,1.22999E-9,1.23001E-9
540 DATA      "000001.2300000","X","E-000009"," 1.23000E-9"
550 DATA 0,0,0,"0E0"," 0"
560 DATA 0,0,0,"000.000E22"," 0"
570 DATA 0,0,0,"+000E55"," 0"
580 DATA 0,0,0,"0.0E-000"," 0"
590 DATA 123,122.999,123.001,"123E0"," 123"
600 DATA 123,122.999,123.001,"123E000"," 123"
610 DATA 123,122.999,123.001,"123E-00"," 123"
620 DATA 123,122.999,123.001,"123E+0"," 123"
630 DATA 1.2345678E19,1.2345578E19,1.2345779E19
640 DATA      "1234567890","X","1234567890"," 1.23457E+19"
650 DATA 1.23456E15,1.23455E15,1.23457E15,"123456E10"," 1.23456E+15"
660 DATA 1.23456E-15,1.23455E-15,1.23457E-15
670 DATA      "0.0000123456E-10"," 1.23456E-15"
680 DATA 123456,123455,123457,"1234560000","X","00000E-9"," 123456"
690 DATA 123456,123455,123457,"0.000000000","X","123456E15"," 123456"
700 DATA 9.87654E37,9.87653E37,9.87655E37,".00987654E40"," 9.87654E+37"
710 DATA 9.87654E-38,9.87653E-38,9.87655E-38,"987.654E-40"
720 DATA      " 9.87654E-38"
730 DATA 123.456,123.455,123.457,"123456.E-3"," 123.456"
740 DATA 123.456,123.455,123.457,".123456E3"," 123.456"
750 DATA -1,-1,-1,"","",""
760 PRINT "                  BEGIN TEST"
770 LET F1=0
780 READ V,L9,H9,A$,B$
790 IF V=-1 THEN 1000
800 LET C$ = ""
810 IF B$<>"X" THEN 830
820 READ C$,B$
830 PRINT
840 PRINT
850 PRINT "PLEASE ENTER: "
860 PRINT " ";A$;C$
870 INPUT W
880 LET M$="PASS"
890 IF W<L9 THEN 940
900 IF W>H9 THEN 940
910 PRINT "SHOULD BE","ACTUAL","RESULT"
920 PRINT B$,W,M$
930 GOTO 780
940 PRINT "APPARENT FAILURE - ENTER 1 TO RE-TRY INPUT, 0 IF NOT"
950 INPUT Z
960 IF Z<>0 THEN 830

```

```
970 LET M$="FAIL"
980 LET F1=F1+1
990 GOTO 910
1000 PRINT
1010 PRINT
1020 IF F1=0 THEN 1050
1030 PRINT "*** TEST FAILED: ";F1;" NUMBER(S) HANDLED IMPROPERLY.***"
1040 GOTO 1060
1050 PRINT "***** TEST PASSED. *****"
1060 REM END OF MAIN LINE FOR 107.1
1070 PRINT
1080 PRINT "           END TEST"
1090 PRINT
1100 PRINT "END PROGRAM 107"
1110 STOP
1120 END
```

---

#### PROGRAM FILE 107: INPUT OF NUMERIC CONSTANTS.

ANSI STANDARD 5.2, 13.2, 13.4

#### SECTION 107.1: INPUT OF NUMERIC CONSTANTS.

THIS SECTION TESTS THE PROCESSOR'S CAPABILITY TO INPUT NUMERIC VALUES CORRECTLY. IF AN INPUT VALUE IS NOT ACCURATE TO AT LEAST SIX DIGITS, IT IS COUNTED AS A FAILURE.

ENTER THE NUMBERS BELOW EXACTLY AS REQUESTED.  
FOR USER CONVENIENCE, THE NUMBER TO BE ENTERED IS INDENTED TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE ALIGNED WITH THE PROMPT STRING.

ALL THE REPLIES PROMPTED FOR ARE VALID, THEREFORE  
IF THE PROCESSOR REJECTS ANY OF THEM,  
\*\*\* TEST FAILED \*\*\*

BEGIN TEST

PLEASE ENTER:

+.999999E38  
? +.999999E38

SHOULD BE	ACTUAL	RESULT
9.99999E+37	9.99999E+37	PASS

PLEASE ENTER:

-.999999E38  
? -.999999E38

SHOULD BE	ACTUAL	RESULT
-9.99999E+37	-9.99999E+37	PASS

PLEASE ENTER:

+1.00001E-38

? +1.00001E-38

SHOULD BE	ACTUAL	RESULT
1.00001E-38	1.00001E-38	PASS

PLEASE ENTER:

-1.00001E-38

? -1.00001E-38

SHOULD BE	ACTUAL	RESULT
-1.00001E-38	-1.00001E-38	PASS

PLEASE ENTER:

9.99999E-38

? 9.99999E-38

SHOULD BE	ACTUAL	RESULT
9.99999E-38	9.99999E-38	PASS

PLEASE ENTER:

9.87654E37

? 9.87654E37

SHOULD BE	ACTUAL	RESULT
9.87654E+37	9.87654E+37	PASS

PLEASE ENTER:

123456

? 123456

SHOULD BE	ACTUAL	RESULT
123456	123456	PASS

PLEASE ENTER:

123456.

? 123456.

SHOULD BE	ACTUAL	RESULT
123456	123456	PASS

PLEASE ENTER:

123456.0

? 123456.0

SHOULD BE	ACTUAL	RESULT
123456	123456	PASS

PLEASE ENTER:

987.654

? 987.654

SHOULD BE	ACTUAL	RESULT
987.654	987.654	PASS

PLEASE ENTER:

1234560

? 1234560

SHOULD BE	ACTUAL	RESULT
1.23456E+6	1.23456E+6	PASS

PLEASE ENTER:

123456000		
? 123456000		
SHOULD BE	ACTUAL	RESULT
1.23456E+8	1.23456E+8	PASS

PLEASE ENTER:

.0123456		
? .0123456		
SHOULD BE	ACTUAL	RESULT
1.23456E-2	1.23456E-2	PASS

PLEASE ENTER:

.000123456		
? .000123456		
SHOULD BE	ACTUAL	RESULT
1.23456E-4	1.23456E-4	PASS

PLEASE ENTER:

.12		
? .12		
SHOULD BE	ACTUAL	RESULT
.12	.12	PASS

PLEASE ENTER:

+.12		
? +.12		
SHOULD BE	ACTUAL	RESULT
.12	.12	PASS

PLEASE ENTER:

-.12		
? -.12		
SHOULD BE	ACTUAL	RESULT
-.12	-.12	PASS

PLEASE ENTER:

0.12		
? 0.12		
SHOULD BE	ACTUAL	RESULT
.12	.12	PASS

PLEASE ENTER:

0.0		
? 0.0		
SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

+0		
? +0		
SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

- .000

? - .000

SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

1.23E9

? 1.23E9

SHOULD BE	ACTUAL	RESULT
1.23000E+9	1.23000E+9	PASS

PLEASE ENTER:

1.23E09

? 1.23E09

SHOULD BE	ACTUAL	RESULT
1.23000E+9	1.23000E+9	PASS

PLEASE ENTER:

1.23E+9

? 1.23E+9

SHOULD BE	ACTUAL	RESULT
1.23000E+9	1.23000E+9	PASS

PLEASE ENTER:

1.23E-9

? 1.23E-9

SHOULD BE	ACTUAL	RESULT
1.23000E-9	1.23000E-9	PASS

PLEASE ENTER:

1.23E-09

? 1.23E-09

SHOULD BE	ACTUAL	RESULT
1.23000E-9	1.23000E-9	PASS

PLEASE ENTER:

1.23E-0009

? 1.23E-0009

SHOULD BE	ACTUAL	RESULT
1.23000E-9	1.23000E-9	PASS

PLEASE ENTER:

000001.2300000E-000009

? 000001.2300000E-000009

SHOULD BE	ACTUAL	RESULT
1.23000E-9	1.23000E-9	PASS

PLEASE ENTER:

OEO

? OEO

SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

000.000E22

? 000.000E22

SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

+000E55

? +000E55

SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

0.0E-000

? 0.0E-000

SHOULD BE	ACTUAL	RESULT
0	0	PASS

PLEASE ENTER:

123E0

? 123E0

SHOULD BE	ACTUAL	RESULT
123	123	PASS

PLEASE ENTER:

123E000

? 123E000

SHOULD BE	ACTUAL	RESULT
123	123	PASS

PLEASE ENTER:

123E-00

? 123E-00

SHOULD BE	ACTUAL	RESULT
123	123	PASS

PLEASE ENTER:

123E+0

? 123E+0

SHOULD BE	ACTUAL	RESULT
123	123	PASS

PLEASE ENTER:

12345678901234567890

? 12345678901234567890

SHOULD BE	ACTUAL	RESULT
1.23457E+19	1.23457E+19	PASS

PLEASE ENTER:

123456E10

? 123456E10

SHOULD BE	ACTUAL	RESULT
1.23456E+15	1.23456E+15	PASS

PLEASE ENTER:

0.0000123456E-10

? 0.0000123456E-10

SHOULD BE	ACTUAL	RESULT
1.23456E-15	1.23456E-15	PASS

PLEASE ENTER:

1234560000000000E-9

? 1234560000000000E-9

SHOULD BE	ACTUAL	RESULT
123456	123456	PASS

PLEASE ENTER:

0.00000000123456E15

? 0.00000000123456E15

SHOULD BE	ACTUAL	RESULT
123456	123456	PASS

PLEASE ENTER:

.00987654E40

? .00987654E40

SHOULD BE	ACTUAL	RESULT
9.87654E+37	9.87654E+37	PASS

PLEASE ENTER:

987.654E-40

? 987.654E-40

SHOULD BE	ACTUAL	RESULT
9.87654E-38	9.87654E-38	PASS

PLEASE ENTER:

123456.E-3

? 123456.E-3

SHOULD BE	ACTUAL	RESULT
123.456	123.456	PASS

PLEASE ENTER:

.123456E3

? .123456E3

SHOULD BE	ACTUAL	RESULT
123.456	123.456	PASS

\*\*\*\*\* TEST PASSED. \*\*\*\*\*

END TEST

END PROGRAM 107

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 108: INPUT TO SUBSCRIPTED VARIABLES."  
20 PRINT "      ANSI STANDARD 13.2, 13.4"  
30 PRINT  
40 PRINT "SECTION 108.1: SIMPLE INPUT TO FILL ARRAY."  
50 PRINT  
60 PRINT "                  BEGIN TEST."  
70 PRINT  
80 DATA 7,3,2,1,5,9,10,0,6,4,8  
90 DATA 7,3,2,1,9,4,8,0,10,5,6  
100 FOR I=0 TO 10  
110 READ J  
120 PRINT "PLEASE ENTER:";I  
130 INPUT A(J)  
140 NEXT I  
150 FOR I=0 TO 10  
160 READ T(I)  
170 NEXT I  
180 GOSUB 3000  
190 PRINT "                  END TEST."  
200 PRINT  
210 PRINT  
220 PRINT "SECTION 108.2: INPUT OF SUBSCRIPT FOLLOWED BY SUBSCRIPTED"  
230 PRINT "      VARIABLE."  
240 PRINT  
250 PRINT "                  BEGIN TEST."  
260 PRINT  
270 PRINT "FOR THIS REPLY, STATEMENT IS: INPUT A(I),I,A(I),I,A(I)"  
280 PRINT "PLEASE ENTER: 500,6,600,2,200"  
290 FOR I=0 TO 10  
300 LET A(I)=10+I  
310 LET T(I)=A(I)  
320 NEXT I  
330 LET I=5  
340 INPUT A(I),I,A(I),I,A(I)  
350 LET T(5)=500  
360 LET T(6)=600  
370 LET T(2)=200  
380 GOSUB 3000  
390 PRINT "                  END TEST."  
400 PRINT  
410 PRINT  
420 PRINT "SECTION 108.3: NO ASSIGNMENT BEFORE VALIDATION OF INPUT."  
430 PRINT  
440 PRINT "                  BEGIN TEST"  
450 PRINT  
460 PRINT "THIS SECTION TESTS THAT NO ASSIGNMENT IS DONE BEFORE"  
470 PRINT "THE ENTIRE REPLY IS VALIDATED."  
480 FOR I=0 TO 10  
490 LET A(I)=I*100  
500 LET T(I)=A(I)  
510 NEXT I
```

```
520 LET T(1)=8
530 PRINT
540 PRINT "PLEASE ENTER LINE NO. 1 FIRST. IT SHOULD BE REJECTED"
550 PRINT "BECAUSE OF INSUFFICIENT DATA AND YOU SHOULD BE ALLOWED"
560 PRINT "TO RE-SUPPLY INPUT. THEN ENTER LINE NO. 2 (NOTE THE SECOND"
570 PRINT "CHARACTER IS A COMMA RATHER THAN A PERIOD)."
580 PRINT
590 PRINT "*** IF NOT ALLOWED TO RE-SUPPLY INPUT, TEST FAILS ***"
600 PRINT
610 PRINT "THE ASSOCIATED INPUT STATEMENT IS: INPUT H,I,J,A(I),K,L"
620 PRINT "THEREFORE, AN 8 SHOULD BE PUT INTO A(1), AND NO OTHER"
630 PRINT "CHANGE MADE TO THE ARRAY."
640 PRINT
650 PRINT "LINE NO. 1:    3,1,6,8,9,11"
660 PRINT "LINE NO. 2:    3,1,6,8,9,11"
670 INPUT H,I,J,A(I),K,L
680 GOSUB 3000
690 PRINT "                      END TEST"
700 PRINT
710 PRINT "SECTION 108.4: INPUT TO TWO-DIMENSIONAL ARRAY."
720 PRINT
730 FOR I=1 TO 5
740 FOR J=1 TO 3
750 LET X(I,J)=1000*I+J
760 NEXT J
770 NEXT I
780 PRINT "                      BEGIN TEST"
790 PRINT
800 PRINT "STATEMENT FOR THIS REPLY: INPUT I,J,X(I,J)"
810 PRINT "PLEASE ENTER: 2,3,999"
820 LET E=0
830 INPUT I,J,X(I,J)
840 PRINT
850 PRINT "X(I,J)", "J=1", "J=2", "J=3"
860 FOR I=1 TO 5
870 PRINT "I=";I,X(I,1),X(I,2),X(I,3)
880 FOR J=1 TO 3
890 IF I<>2 THEN 920
900 IF J<>3 THEN 920
910 GOTO 940
920 IF X(I,J)= 1000*I+J THEN 940
930 LET E=1
940 NEXT J
950 NEXT I
960 IF X(2,3)=999 THEN 980
970 LET E=1
980 PRINT
990 IF E=0 THEN 1040
1000 PRINT "*** TEST FAILED ***"
1010 PRINT "EITHER X(2,3) <> 999 OR SOME OTHER "
1020 PRINT "X(I,J) <> 1000*I + J AS INITIALIZED."
1030 GOTO 1050
1040 PRINT "*** TEST PASSED ***"
1050 PRINT
1060 PRINT "                      END TEST"
```

```
1070 PRINT
1080 PRINT "END PROGRAM 108"
1090 STOP
3000 REM SUBROUTINE TO CHECK THAT A(I)=T(I)
3010 LET F=0
3020 PRINT
3025 PRINT " ,," CONTENTS OF ARRAY"
3030 PRINT "SUBSCRIPT","SHOULD BE","ACTUAL"
3040 FOR I=0 TO 10
3050 IF A(I)=T(I) THEN 3070
3060 LET F=F+1
3070 PRINT I,T(I),A(I)
3080 NEXT I
3090 PRINT
3100 IF F=0 THEN 3130
3110 PRINT "*** TEST FAILED IN ";F;"CASE(S) ***"
3120 GOTO 3140
3130 PRINT "*** TEST PASSED ***"
3140 PRINT
3150 RETURN
3160 END
```

---

PROGRAM FILE 108: INPUT TO SUBSCRIPTED VARIABLES.  
ANSI STANDARD 13.2, 13.4

#### SECTION 108.1: SIMPLE INPUT TO FILL ARRAY.

BEGIN TEST.

```
PLEASE ENTER: 0
? 0
PLEASE ENTER: 1
? 1
PLEASE ENTER: 2
? 2
PLEASE ENTER: 3
? 3
PLEASE ENTER: 4
? 4
PLEASE ENTER: 5
? 5
PLEASE ENTER: 6
? 6
PLEASE ENTER: 7
? 7
PLEASE ENTER: 8
? 8
PLEASE ENTER: 9
? 9
PLEASE ENTER: 10
? 10
```

CONTENTS OF ARRAY		
SUBSCRIPT	SHOULD BE	ACTUAL
0	7	7
1	3	3
2	2	2
3	1	1
4	9	9
5	4	4
6	8	8
7	0	0
8	10	10
9	5	5
10	6	6

\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 108.2: INPUT OF SUBSCRIPT FOLLOWED BY SUBSCRIPTED VARIABLE.

BEGIN TEST.

FOR THIS REPLY, STATEMENT IS: INPUT A(I),I,A(I),I,A(I)  
 PLEASE ENTER: 500,6,600,2,200  
 ? 500,6,600,2,200

CONTENTS OF ARRAY		
SUBSCRIPT	SHOULD BE	ACTUAL
0	10	10
1	11	11
2	200	200
3	13	13
4	14	14
5	500	500
6	600	600
7	17	17
8	18	18
9	19	19
10	20	20

\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 108.3: NO ASSIGNMENT BEFORE VALIDATION OF INPUT.

BEGIN TEST

THIS SECTION TESTS THAT NO ASSIGNMENT IS DONE BEFORE THE ENTIRE REPLY IS VALIDATED.

PLEASE ENTER LINE NO. 1 FIRST. IT SHOULD BE REJECTED BECAUSE OF INSUFFICIENT DATA AND YOU SHOULD BE ALLOWED TO RE-SUPPLY INPUT. THEN ENTER LINE NO. 2 (NOTE THE SECOND CHARACTER IS A COMMA RATHER THAN A PERIOD).

\*\*\* IF NOT ALLOWED TO RE-SUPPLY INPUT, TEST FAILS \*\*\*

THE ASSOCIATED INPUT STATEMENT IS: INPUT H,I,J,A(I),K,L THEREFORE, AN 8 SHOULD BE PUT INTO A(1), AND NO OTHER CHANGE MADE TO THE ARRAY.

LINE NO. 1: 3.1,6,8,9,11

LINE NO. 2: 3,1,6,8,9,11

? 3.1,6,8,9,11

? INSUFFICIENT DATA IN INPUT-REPLY IN LINE 670; PLEASE RE-SUPPLY  
? 3,1,6,8,9,11

CONTENTS OF ARRAY		
SUBSCRIPT	SHOULD BE	ACTUAL
0	0	0
1	8	8
2	200	200
3	300	300
4	400	400
5	500	500
6	600	600
7	700	700
8	800	800
9	900	900
10	1000	1000

\*\*\* TEST PASSED \*\*\*

END TEST

SECTION 108.4: INPUT TO TWO-DIMENSIONAL ARRAY.

BEGIN TEST

STATEMENT FOR THIS REPLY: INPUT I,J,X(I,J)

PLEASE ENTER: 2,3,999

? 2,3,999

X(I,J)	J=1	J=2	J=3
I= 1	1001	1002	1003
I= 2	2001	2002	999
I= 3	3001	3002	3003
I= 4	4001	4002	4003
I= 5	5001	5002	5003

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 108

```
*****
*****
*****
```

10 PRINT "PROGRAM FILE 109: STRING INPUT."  
20 PRINT " ANSI STANDARD 3.2, 13.2, 13.4"  
30 PRINT  
40 PRINT "SECTION 109.1 INPUT OF QUOTED AND UNQUOTED STRINGS"  
50 PRINT  
60 PRINT "PLEASE ENTER YOUR INPUT-REPLIES EXACTLY AS REQUESTED"  
70 PRINT "BY THE PROMPT, INCLUDING ALL SPACING, COMMAS AND OTHER"  
80 PRINT "PUNCTUATION."  
90 PRINT  
100 PRINT " - EXCEPT - WHEREVER THE PROMPT CONTAINS AN EQUALS (=)"  
110 PRINT "SUBSTITUTE A BLANK SPACE, AND WHEREVER THE PROMPT"  
120 PRINT "CONTAINS A NUMBER-SIGN (#) SUBSTITUTE A QUOTE."  
130 PRINT "THUS, THE PROMPT:"  
140 PRINT " ==#ER#=="  
150 PRINT "SHOULD CAUSE YOU TO STRIKE THE KEYS:"  
160 PRINT " 1. SPACE"  
170 PRINT " 2. SPACE"  
180 PRINT " 3. QUOTE"  
190 PRINT " 4. LETTER E"  
200 PRINT " 5. LETTER R"  
210 PRINT " 6. QUOTE"  
220 PRINT " 7. SPACE"  
230 PRINT " 8. SPACE"  
240 PRINT  
250 PRINT "THESE CONVENTIONS ARE NECESSARY BECAUSE THE QUOTE IS"  
260 PRINT "UNPRINTABLE IN MINIMAL BASIC, AND SPACES ARE DIFFICULT"  
270 PRINT "TO COUNT ON A TERMINAL."  
280 PRINT  
290 PRINT "PLEASE NOTE THAT THIS ROUTINE IS STRONGLY DEPENDENT"  
300 PRINT "ON THE CORRECT OPERATION OF THE STRING COMPARISON"  
310 PRINT "FEATURE. IF THAT FEATURE IS INCORRECT, THIS ROUTINE"  
320 PRINT "WILL LIKELY GIVE INVALID RESULTS."  
330 PRINT  
340 PRINT "ALL THE REPLIES PROMPTED FOR ARE VALID, THEREFORE "  
350 PRINT "IF THE PROCESSOR REJECTS ANY OF THEM,"  
360 PRINT "\*\*\* TEST FAILED \*\*\*"  
370 PRINT  
380 PRINT "FOR USER CONVENIENCE, THE STRING TO BE ENTERED IS INDENTED"  
390 PRINT "TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO"  
400 PRINT "POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE"  
410 PRINT "ALIGNED WITH THE PROMPT STRING."  
420 PRINT  
430 PRINT " BEGIN TEST"  
440 LET F=0  
450 READ N,P\$  
460 LET Q\$=""  
470 IF N<100 THEN 500  
480 LET N=N-100  
490 READ Q\$

```
500 IF N=0 THEN 1380
510 LET A$=""
520 LET B$=""
530 LET C$=""
540 LET D$=""
550 LET I$=""
560 LET J$=""
570 LET K$=""
580 LET L$=""
590 READ I$
600 IF N<=1 THEN 660
610 READ J$
620 IF N<=2 THEN 660
630 READ K$
640 IF N<=3 THEN 660
650 READ L$
660 PRINT
670 PRINT
680 PRINT "PLEASE ENTER:"
690 PRINT " ";P$;Q$
700 ON N GOTO 710,730,750,770
710 INPUT A$
720 GOTO 790
730 INPUT A$,B$
740 GOTO 790
750 INPUT A$,B$,C$
760 GOTO 790
770 INPUT A$,B$,C$,D$
780 GOTO 790
790 REM COMMON EXIT POINT
800 PRINT
810 PRINT "SHOULD BE: ***";I$;J$;K$;L$;"***"
820 PRINT "ACTUAL:      ***";A$;B$;C$;D$;"***"
830 IF A$<>I$ THEN 890
840 IF B$<>J$ THEN 890
850 IF C$<>K$ THEN 890
860 IF D$<>L$ THEN 890
870 PRINT "TEST OK"
880 GOTO 450
890 PRINT "INPUT NOT EQUAL EXPECTED VALUE. RE-TRY (Y OR N)?"
900 INPUT T$
910 IF T$<>"N" THEN 660
920 PRINT "TEST FAILED."
930 LET F=F+1
940 GOTO 450
950 REM DATA ITEM FORMAT:
960 REM 1 - NUMBER OF VARIABLES (+100 IF LONG PROMPT)
970 REM 2 - PROMPT
980 REM 3 - CORRECT RESULT
990 REM      OR
1000 REM 2,3 - LONG PROMPT
1010 REM 4 - CORRECT RESULT
1020 DATA 1,"ABC","ABC"
1030 DATA 1,"#ABC#", "ABC"
1040 DATA 2,"ABC,DEF","ABC","DEF"
```

```

1050 DATA 2,"#ABC#",DEF#,ABC",DEF"
1060 DATA 2,"#ABC#,DEF",ABC",DEF"
1070 DATA 2,ABC,DEF#,ABC",DEF"
1080 REM TEST ALL CHARACTERS LEGAL IN UNQUOTED STRING
1090 DATA 1,"ABCDEFGHIJKLMNOP","ABCDEFGHIJKLMNOP"
1100 DATA 1,"NOPQRSTUVWXYZ","NOPQRSTUVWXYZ"
1110 DATA 1,"+=====+,-"
1120 DATA 1,"----5---10---15-18","----5---10---15-18"
1130 DATA 1,"==ABC","ABC"
1140 DATA 1,"ABC==","ABC"
1150 DATA 1,"==ABC==","ABC"
1160 DATA 1,"==ABC#",ABC"
1170 DATA 1,"#ABC==#",ABC"
1180 DATA 1,"#==ABC==#",ABC"
1190 DATA 101,"==#=ABC","==#===",ABC"
1200 DATA 103,"==ABC==,==#DEF#","==,==GHI==",ABC,DEF,GHI"
1210 DATA 3,"=1=,==2==,==3==","1","2","3"
1220 DATA 1,A==B,A B"
1230 DATA 1,"==A==B==","A B"
1240 DATA 101,"==EIGHTEEN",=POSITIONS==,"EIGHTEEN POSITIONS"
1250 DATA 103,"==A==B==,==C==,==D==,==E==F==",A B,C D,E F"
1260 DATA 103,"==A==B==,==#D==,==E==F==",A B,D,E F"
1270 DATA 3,"#A#=,=B=C,=#D#=","A","B C","D"
1280 DATA 103,"==#==A==B==#==,==#C=D#=,=E=F="," A B "
1290 DATA " C D ",E F"
1300 DATA 4,"A,B,#C,D#,E#","A","B","C,D","E"
1310 DATA 1,"#","
1320 DATA 3,"A,##,B","A","","B"
1330 DATA 3,"==A==,==#==,==B==","A","","B"
1340 DATA 1,"AB+3-5.6B","AB+3-5.6B"
1350 DATA 1,"-1.23",-1.23"
1360 DATA 1,"+3-5=-8+6","+3-5 -8+6"
1370 DATA 0,"Q"
1380 PRINT
1390 PRINT
1400 IF F=0 THEN 1430
1410 PRINT "***TEST FAILED: ";F;" CASE(S) HANDLED IMPROPERLY***"
1420 GOTO 1440
1430 PRINT "*** TEST PASSED ***"
1440 PRINT
1450 PRINT END TEST"
1460 PRINT
1470 PRINT "SECTION 109.2 TEST QUOTED-STRING-CHARACTERS"
1480 PRINT
1490 PRINT "THIS PART OF THE TEST IS INTENDED TO DETERMINE WHETHER"
1500 PRINT "QUOTED STRINGS ON INPUT CAN CONTAIN ANY OF THE LEGAL"
1510 PRINT "QUOTED STRING CHARACTERS."
1520 PRINT
1530 PRINT "PLEASE RESPOND TO THE PROMPT BY ENTERING THE REQUESTED"
1540 PRINT "CHARCTERS IMMEDIATELY PRECEDED AND FOLLOWED BY THE"
1550 PRINT "QUOTE CHARACTER.  THUS THE PROMPT:"
1560 PRINT ABC"
1570 PRINT "SHOULD CAUSE YOU TO STRIKE THE KEYS:"
1580 PRINT 1. QUOTE"
1590 PRINT 2. LETTER A"

```

```
1600 PRINT " 3. LETTER B"
1610 PRINT " 4. LETTER C"
1620 PRINT " 5. QUOTE"
1630 PRINT "THUS THE QUOTE TO BE ENTERED IS NOW IMPLICIT AND THE"
1640 PRINT "NUMBER SIGN (#) AND EQUALS SIGN (=) HAVE NO SPECIAL"
1650 PRINT "SIGNIFICANCE."
1660 PRINT
1670 PRINT "               BEGIN TEST"
1680 LET F=0
1690 READ A$
1700 IF A$="Q" THEN 1900
1710 PRINT
1720 PRINT
1730 PRINT "PLEASE ENTER:"
1740 PRINT "    ";A$
1750 INPUT B$
1760 PRINT "SHOULD BE: ***";A$;"***"
1770 PRINT "ACTUAL:      ***";B$;"***"
1780 IF A$<>B$ THEN 1810
1790 PRINT "TEST OK"
1800 GOTO 1690
1810 PRINT "INPUT NOT EQUAL TO EXPECTED VALUE. RE-TRY (Y OR N)?"
1820 INPUT C$
1830 IF C$<>"N" THEN 1710
1840 PRINT "TEST FAILED."
1850 LET F=F+1
1860 GOTO 1690
1870 DATA "ABCDEFGHIJKLM","NOPQRSTUVWXYZ","0123456789"
1880 DATA "!#%&'()*+, -,. /:;<=>?^_","EMBEDDED SPACE"
1890 DATA "Q"
1900 PRINT
1910 IF F=0 THEN 1940
1920 PRINT "*** TEST FAILED. ";F;" CASE(S) HANDLED IMPROPERLY. ***"
1930 GOTO 1950
1940 PRINT "***** TEST PASSED *****"
1950 PRINT
1960 PRINT "               END TEST"
1970 PRINT
1980 PRINT "END PROGRAM 109"
1990 END
```

---

PROGRAM FILE 109: STRING INPUT.  
ANSI STANDARD 3.2, 13.2, 13.4

#### SECTION 109.1 INPUT OF QUOTED AND UNQUOTED STRINGS

PLEASE ENTER YOUR INPUT-REPLIES EXACTLY AS REQUESTED  
BY THE PROMPT, INCLUDING ALL SPACING, COMMAS AND OTHER  
PUNCTUATION.

- EXCEPT - WHEREVER THE PROMPT CONTAINS AN EQUALS (=) SUBSTITUTE A BLANK SPACE, AND WHEREVER THE PROMPT CONTAINS A NUMBER-SIGN (#) SUBSTITUTE A QUOTE.  
THUS, THE PROMPT:

==#ER#==

SHOULD CAUSE YOU TO STRIKE THE KEYS:

1. SPACE
2. SPACE
3. QUOTE
4. LETTER E
5. LETTER R
6. QUOTE
7. SPACE
8. SPACE

THESE CONVENTIONS ARE NECESSARY BECAUSE THE QUOTE IS UNPRINTABLE IN MINIMAL BASIC, AND SPACES ARE DIFFICULT TO COUNT ON A TERMINAL.

PLEASE NOTE THAT THIS ROUTINE IS STRONGLY DEPENDENT ON THE CORRECT OPERATION OF THE STRING COMPARISON FEATURE. IF THAT FEATURE IS INCORRECT, THIS ROUTINE WILL LIKELY GIVE INVALID RESULTS.

ALL THE REPLIES PROMPTED FOR ARE VALID, THEREFORE IF THE PROCESSOR REJECTS ANY OF THEM,

\*\*\* TEST FAILED \*\*\*

FOR USER CONVENIENCE, THE STRING TO BE ENTERED IS INDENTED TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE ALIGNED WITH THE PROMPT STRING.

BEGIN TEST

PLEASE ENTER:

ABC  
? ABC

SHOULD BE: \*\*\*ABC\*\*\*

ACTUAL:      \*\*\*ABC\*\*\*

TEST OK

PLEASE ENTER:

#ABC#  
? "ABC"

SHOULD BE: \*\*\*ABC\*\*\*

ACTUAL:      \*\*\*ABC\*\*\*

TEST OK

PLEASE ENTER:

ABC,DEF  
? ABC,DEF

SHOULD BE: \*\*\*ABCDEF\*\*\*  
ACTUAL: \*\*\*ABCDEF\*\*\*  
TEST OK

PLEASE ENTER:  
#ABC#, #DEF#  
? "ABC", "DEF"

SHOULD BE: \*\*\*ABCDEF\*\*\*  
ACTUAL: \*\*\*ABCDEF\*\*\*  
TEST OK

PLEASE ENTER:  
#ABC#, DEF  
? "ABC", DEF

SHOULD BE: \*\*\*ABCDEF\*\*\*  
ACTUAL: \*\*\*ABCDEF\*\*\*  
TEST OK

PLEASE ENTER:  
ABC, #DEF#  
? ABC, "DEF"

SHOULD BE: \*\*\*ABCDEF\*\*\*  
ACTUAL: \*\*\*ABCDEF\*\*\*  
TEST OK

PLEASE ENTER:  
ABCDEFGHIJKLM  
? ABCDEFGHIJKLM

SHOULD BE: \*\*\*ABCDEFGHIJKLM\*\*\*  
ACTUAL: \*\*\*ABCDEFGHIJKLM\*\*\*  
TEST OK

PLEASE ENTER:  
NOPQRSTUVWXYZ  
? NOPQRSTUVWXYZ

SHOULD BE: \*\*\*NOPQRSTUVWXYZ\*\*\*  
ACTUAL: \*\*\*NOPQRSTUVWXYZ\*\*\*  
TEST OK

PLEASE ENTER:  
+.=====.  
? +. -

SHOULD BE: \*\*\*+. -\*\*\*  
ACTUAL: \*\*\*+. -\*\*\*  
TEST OK

PLEASE ENTER:

----5---10---15-18

? ----5---10---15-18

SHOULD BE: \*\*\*-----5---10---15-18\*\*\*

ACTUAL:        \*\*\*-----5---10---15-18\*\*\*

TEST OK

PLEASE ENTER:

==ABC

?      ABC

SHOULD BE: \*\*\*ABC\*\*\*

ACTUAL:        \*\*\*ABC\*\*\*

TEST OK

PLEASE ENTER:

ABC==

?      ABC

SHOULD BE: \*\*\*ABC\*\*\*

ACTUAL:        \*\*\*ABC\*\*\*

TEST OK

PLEASE ENTER:

==ABC==

?      ABC

SHOULD BE: \*\*\*ABC\*\*\*

ACTUAL:        \*\*\*ABC\*\*\*

TEST OK

PLEASE ENTER:

#==ABC#

? "      ABC"

SHOULD BE: \*\*\*     ABC\*\*\*

ACTUAL:        \*\*\*     ABC\*\*\*

TEST OK

PLEASE ENTER:

#ABC==#

? "      ABC" "

SHOULD BE: \*\*\*ABC    \*\*\*

ACTUAL:        \*\*\*ABC    \*\*\*

TEST OK

PLEASE ENTER:

#==ABC==#

? "      ABC" "

SHOULD BE: \*\*\*     ABC    \*\*\*

ACTUAL:        \*\*\*     ABC    \*\*\*

TEST OK

PLEASE ENTER:

====ABC====#====  
? " ABC "

SHOULD BE: \*\*\* ABC \*\*\*  
ACTUAL: \*\*\* ABC \*\*\*  
TEST OK

PLEASE ENTER:

==ABC==, ==#DEF#==, ==GHI==  
? ABC , "DEF" , GHI

SHOULD BE: \*\*\*ABCDEFGHI\*\*\*  
ACTUAL: \*\*\*ABCDEFGHI\*\*\*  
TEST OK

PLEASE ENTER:

=1=, ==2==, ===3===  
? 1 , 2 , 3

SHOULD BE: \*\*\*123\*\*\*  
ACTUAL: \*\*\*123\*\*\*  
TEST OK

PLEASE ENTER:

A==B  
? A B

SHOULD BE: \*\*\*A B\*\*\*  
ACTUAL: \*\*\*A B\*\*\*  
TEST OK

PLEASE ENTER:

==A==B==  
? A B

SHOULD BE: \*\*\*A B\*\*\*  
ACTUAL: \*\*\*A B\*\*\*  
TEST OK

PLEASE ENTER:

==EIGHTEEN=POSITIONS===  
? EIGHTEEN POSITIONS

SHOULD BE: \*\*\*EIGHTEEN POSITIONS\*\*\*  
ACTUAL: \*\*\*EIGHTEEN POSITIONS\*\*\*  
TEST OK

PLEASE ENTER:

--A==B==, ==C==D==, ==E==F==  
? A B , C D , E F

SHOULD BE: \*\*\*A BC DE F\*\*\*  
ACTUAL: \*\*\*A BC DE F\*\*\*  
TEST OK

PLEASE ENTER:

==A==B==, ==#D==, ==E==F==  
? A B , "D" , E F

SHOULD BE: \*\*\*A BDE F\*\*\*

ACTUAL: \*\*\*A BDE F\*\*\*

TEST OK

PLEASE ENTER:

=#A#=, =B=C=, =#D#=  
? "A" , B C , "D"

SHOULD BE: \*\*\*AB CD\*\*\*

ACTUAL: \*\*\*AB CD\*\*\*

TEST OK

PLEASE ENTER:

==#==A==B==#=, ==#C=D==#, ==E=F=  
? " A B " , " C D " , E F

SHOULD BE: \*\*\* A B C D E F\*\*\*

ACTUAL: \*\*\* A B C D E F\*\*\*

TEST OK

PLEASE ENTER:

A,B,#C,D#,#E#  
? A,B,"C,D","E"

SHOULD BE: \*\*\*ABC,DE\*\*\*

ACTUAL: \*\*\*ABC,DE\*\*\*

TEST OK

PLEASE ENTER:

##  
? ""

SHOULD BE: \*\*\*\*\*

ACTUAL: \*\*\*\*\*

TEST OK

PLEASE ENTER:

A,##,B  
? A,"",B

SHOULD BE: \*\*\*AB\*\*\*

ACTUAL: \*\*\*AB\*\*\*

TEST OK

PLEASE ENTER:

==A==, ==##=, ==B==  
? A , " " , B

SHOULD BE: \*\*\*AB\*\*\*

ACTUAL: \*\*\*AB\*\*\*

TEST OK

PLEASE ENTER:

AB+3-5.6B  
? AB+3-5.6B

SHOULD BE: \*\*\*AB+3-5.6B\*\*\*  
ACTUAL: \*\*\*AB+3-5.6B\*\*\*  
TEST OK

PLEASE ENTER:

-1.23  
? -1.23

SHOULD BE: \*\*\*-1.23\*\*\*  
ACTUAL: \*\*\*-1.23\*\*\*  
TEST OK

PLEASE ENTER:

+3-5=-8+6  
? +3-5 -8+6

SHOULD BE: \*\*\*+3-5 -8+6\*\*\*  
ACTUAL: \*\*\*+3-5 -8+6\*\*\*  
TEST OK

\*\*\* TEST PASSED \*\*\*

END TEST

## SECTION 109.2 TEST QUOTED-STRING-CHARACTERS

THIS PART OF THE TEST IS INTENDED TO DETERMINE WHETHER QUOTED STRINGS ON INPUT CAN CONTAIN ANY OF THE LEGAL QUOTED STRING CHARACTERS.

PLEASE RESPOND TO THE PROMPT BY ENTERING THE REQUESTED CHARCTERS IMMEDIATELY PRECEDED AND FOLLOWED BY THE QUOTE CHARACTER. THUS THE PROMPT:

ABC

SHOULD CAUSE YOU TO STRIKE THE KEYS:

1. QUOTE
2. LETTER A
3. LETTER B
4. LETTER C
5. QUOTE

THUS THE QUOTE TO BE ENTERED IS NOW IMPLICIT AND THE NUMBER SIGN (#) AND EQUALS SIGN (=) HAVE NO SPECIAL SIGNIFICANCE.

BEGIN TEST

PLEASE ENTER:  
 ABCDEFGHIJKLM  
 ? "ABCDEFGHIJKLM"  
 SHOULD BE: \*\*\*ABCDEFGHIJKLM\*\*\*  
 ACTUAL: \*\*\*ABCDEFGHIJKLM\*\*\*  
 TEST OK

PLEASE ENTER:  
 NOPQRSTUVWXYZ  
 ? "NOPQRSTUVWXYZ"  
 SHOULD BE: \*\*\*NOPQRSTUVWXYZ\*\*\*  
 ACTUAL: \*\*\*NOPQRSTUVWXYZ\*\*\*  
 TEST OK

PLEASE ENTER:  
 0123456789  
 ? "0123456789"  
 SHOULD BE: \*\*\*0123456789\*\*\*  
 ACTUAL: \*\*\*0123456789\*\*\*  
 TEST OK

PLEASE ENTER:  
 !#\$%&'()\*+,-  
 ? "!#\$%&'()\*+,-"  
 SHOULD BE: \*\*\*!#\$%&'()\*+,-\*\*\*  
 ACTUAL: \*\*\*!#\$%&'()\*+,-\*\*\*  
 TEST OK

PLEASE ENTER:  
 ./:;<=>?^  
 ? "./:;<=>?^"  
 SHOULD BE: \*\*\*./:;<=>?^\*\*\*  
 ACTUAL: \*\*\*./:;<=>?^\*\*\*  
 TEST OK

PLEASE ENTER:  
 EMBEDDED SPACE  
 ? "EMBEDDED SPACE"  
 SHOULD BE: \*\*\*EMBEDDED SPACE\*\*\*  
 ACTUAL: \*\*\*EMBEDDED SPACE\*\*\*  
 TEST OK

\*\*\*\*\* TEST PASSED \*\*\*\*\*

END TEST

END PROGRAM 109

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

5 PRINT "PROGRAM FILE 110: MIXED INPUT OF STRINGS AND NUMBERS."  
 10 PRINT " ANSI STANDARD 13.2, 13.4"  
 20 PRINT

```
25 PRINT "SECTION 110.1: MIXED INPUT OF STRINGS AND NUMBERS."
30 PRINT
45 PRINT "PLEASE ENTER YOUR INPUT-REPLIES EXACTLY AS REQUESTED"
50 PRINT "BY THE PROMPT, INCLUDING ALL SPACING, COMMAS AND OTHER"
55 PRINT "PUNCTUATION."
60 PRINT
65 PRINT " - EXCEPT - WHEREVER THE PROMPT CONTAINS AN EQUALS (=)"
70 PRINT "SUBSTITUTE A BLANK SPACE, AND WHEREVER THE PROMPT"
75 PRINT "CONTAINS A NUMBER-SIGN (#) SUBSTITUTE A QUOTE."
80 PRINT "THUS, THE PROMPT:"
85 PRINT "           ==#ER#=="
90 PRINT "SHOULD CAUSE YOU TO STRIKE THE KEYS:"
95 PRINT " 1. SPACE"
100 PRINT " 2. SPACE"
105 PRINT " 3. QUOTE"
110 PRINT " 4. LETTER E"
115 PRINT " 5. LETTER R"
120 PRINT " 6. QUOTE"
125 PRINT " 7. SPACE"
130 PRINT " 8. SPACE"
135 PRINT
150 PRINT "THESE CONVENTIONS ARE NECESSARY BECAUSE THE QUOTE IS"
155 PRINT "UNPRINTABLE IN MINIMAL BASIC, AND SPACES ARE DIFFICULT"
160 PRINT "TO COUNT ON A TERMINAL."
165 PRINT
170 PRINT "PLEASE NOTE THAT THIS ROUTINE IS STRONGLY DEPENDENT"
175 PRINT "ON THE CORRECT OPERATION OF THE STRING COMPARISON"
180 PRINT "FEATURE. IF THAT FEATURE IS INCORRECT, THIS ROUTINE"
185 PRINT "WILL LIKELY GIVE INVALID RESULTS."
195 PRINT
200 PRINT "ALL THE REPLIES PROMPTED FOR ARE VALID, THEREFORE "
205 PRINT "IF THE PROCESSOR REJECTS ANY OF THEM,"
210 PRINT "*** TEST FAILED ***"
215 PRINT
225 PRINT "FOR USER CONVENIENCE, THE STRING TO BE ENTERED IS INDENTED"
230 PRINT "TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO"
235 PRINT "POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE"
240 PRINT "ALIGNED WITH THE PROMPT STRING."
241 PRINT
242 PRINT "           BEGIN TEST"
245 LET F=0
247 LET Y$="SHOULD BE: ***"
248 LET Z$="ACTUAL : ***"
250 READ M,C(1),C(2),C(3),P$
255 LET Q$=""
257 LET R$=""
260 IF M<100 THEN 275
265 LET M=M-100
270 READ Q$,R$
275 IF M=0 THEN 850
280 LET A$=""
285 LET B$=""
290 LET C$=""
295 LET I$=""
300 LET J$=""
```

```
305 LET K$=""  
310 LET A1=0  
315 LET B1=0  
320 LET C1=0  
325 LET I1=0  
330 LET J1=0  
335 LET K1=0  
345 LET Q=0  
350 FOR I2=1 TO M  
355 LET Q=Q+Q+C(I2)  
360 NEXT I2  
362 LET Q=Q+1  
365 IF M<=1 THEN 380  
370 LET Q=Q+2  
372 IF M<=2 THEN 380  
375 LET Q=Q+4  
380 LET Q1=Q  
382 IF C(1)=0 THEN 395  
385 READ I$  
390 GOTO 400  
395 READ I1  
400 IF M<2 THEN 450  
405 IF C(2)=0 THEN 420  
410 READ J$  
415 GOTO 425  
420 READ J1  
425 IF M<3 THEN 450  
430 IF C(3)=0 THEN 445  
435 READ K$  
440 GOTO 450  
445 READ K1  
450 PRINT  
455 PRINT  
460 PRINT "PLEASE ENTER:"  
465 PRINT " ";P$;Q$;R$  
475 ON Q1 GOTO 480,500,520,540,560,580,600,620,640,660,680,700,720,740  
480 INPUT A1  
485 PRINT Y$;I1;"***"  
490 PRINT Z$;A1;"***"  
495 GOTO 760  
500 INPUT A$  
505 PRINT Y$;I$;"***"  
510 PRINT Z$;A$;"***"  
515 GOTO 760  
520 INPUT A1,B1  
525 PRINT Y$;I1;J1;"***"  
530 PRINT Z$;A1;B1;"***"  
535 GOTO 760  
540 INPUT A1,B$  
545 PRINT Y$;I1;J$;"***"  
550 PRINT Z$;A1;B$;"***"  
555 GOTO 760  
560 INPUT A$,B1  
565 PRINT Y$;I$;J1;"***"  
570 PRINT Z$;A$;B1;"***"
```

```
575 GOTO 760
580 INPUT A$,B$
585 PRINT Y$;I$;J$;"****"
590 PRINT Z$;A$;B$;"****"
595 GOTO 760
600 INPUT A1,B1,C1
605 PRINT Y$;I1;J1;K1;"****"
610 PRINT Z$;A1;B1;C1;"****"
615 GOTO 760
620 INPUT A1,B1,C$
625 PRINT Y$;I1;J1;K$;"****"
630 PRINT Z$;A1;B1;C$;"****"
635 GOTO 760
640 INPUT A1,B$,C1
645 PRINT Y$;I1;J$;K1;"****"
650 PRINT Z$;A1;B$;C1;"****"
655 GOTO 760
660 INPUT A1,B$,C$
665 PRINT Y$;I1;J$;K$;"****"
670 PRINT Z$;A1;B$;C$;"****"
675 GOTO 760
680 INPUT A$,B1,C1
685 PRINT Y$;I$;J1;K1;"****"
690 PRINT Z$;A$;B1;C1;"****"
695 GOTO 760
700 INPUT A$,B1,C$
705 PRINT Y$;I$;J1;K$;"****"
710 PRINT Z$;A$;B1;C$;"****"
715 GOTO 760
720 INPUT A$,B$,C1
725 PRINT Y$;I$;J$;K1;"****"
730 PRINT Z$;A$;B$;C1;"****"
735 GOTO 760
740 INPUT A$,B$,C$
745 PRINT Y$;I$;J$;K$;"****"
750 PRINT Z$;A$;B$;C$;"****"
755 GOTO 760
760 REM RESULTS DISPLAYED
765 IF A$<>I$ THEN 820
770 IF B$<>J$ THEN 820
771 IF C$<>K$ THEN 820
772 LET D9=A1
774 LET E9=I1
776 GOSUB 3000
778 IF F9<>0 THEN 820
782 LET D9=B1
784 LET E9=J1
786 GOSUB 3000
788 IF F9<>0 THEN 820
792 LET D9=C1
794 LET E9=K1
796 GOSUB 3000
798 IF F9<>0 THEN 820
799 PRINT "TEST OK"
800 GOTO 250
```

```

820 PRINT "INPUT NOT EQUAL EXPECTED VALUE. RE-TRY (Y OR N)?"
825 INPUT S$
830 IF S$<>"N" THEN 450
835 PRINT "TEST FAILED."
840 LET F=F+1
845 GOTO 250
850 PRINT
855 IF F=0 THEN 870
860 PRINT "*** TEST FAILED: ";F;" CASE(S) HANDLED IMPROPERLY ***"
865 GOTO 875
870 PRINT "*** TEST PASSED ***"
875 PRINT
880 PRINT " END TEST"
885 PRINT
890 PRINT "END PROGRAM 110"
895 STOP
900 REM FORMAT OF DATA PARMs:
910 REM 1 - # OF DATA ITEMS (+100 IF LONG PROMPT)
920 REM 2-4 - DATA FORMAT, X=0 => NUMERIC
930 REM X=1 => ALPHA
940 REM 5 - PROMPT (1ST OF THREE, IF LONG)
950 REM 6-N (OR 8-N) - COMPARAND VALUES
1000 DATA 3,0,0,0,"==1==,==2==,==3==",1,2,3
1010 DATA 103,0,0,0,"==+987999E32==,", "==-1.00000E-37==,", "==3.E37=="
1020 DATA 9.87999E37,-1E-37,3E37
1030 DATA 103,0,0,0,"==222222",,111111==,, "==333333"
1040 DATA 222222,111111,333333
1050 DATA 2,0,0,0,"5,6",5,6
1060 DATA 2,0,0,0,"-05.34,345.567E-11",-5.34,3.45567E-9
1070 DATA 2,0,0,0,"2E2,-3.45",,200,-3.45
1080 DATA 102,0,0,0,"-",,0000.00012345,"6E-11,+1E37",-1.23456E-15,1E37
1090 DATA 103,0,0,0,"-00",,0.E-00,,+.000,0E22,,0,0,0
1100 DATA 103,0,0,0,"-",,999.E-00,,+.999,9E22,-999,+.999,9E22
1110 DATA 3,1,1,1,"ABC",##,DEF#,,ABC,"",DEF"
1120 DATA 103,0,1,0,"==4.56789E-11==,, ==MIDDLE=ITEM==,, ==9=="
1130 DATA 4.56789E-11,"MIDDLE ITEM",9
1140 DATA 103,0,1,0,"==987654==,, ==#==MIDDLE=ITE"
1150 DATA "M==#==,, ==656565==",987654," MIDDLE ITEM ",656565
1160 DATA 103,1,0,1,"==AN=UNQUOTED=STR","ING==,, ==3.14159"
1170 DATA "==,, ==#EQUALS=PI#=="
1180 DATA "AN UNQUOTED STRING",3.14159,"EQUALS PI"
1190 DATA 103,0,1,1,"07676760000000E00",,00022==,, ====="
1200 DATA "==X=====,X"
1210 DATA 7.67676E34,"X",,X"
1220 DATA 103,1,0,1,"=====#======,==5",,==,,=====THIRD=",,ITEM===="
1230 DATA "",5,"THIRD ITEM"
1240 DATA 103,1,0,1,"=====#=",,#=,,==0==,,==,,=====#===",, " ,0,,""
1250 DATA 102,0,1,0,"==+333.",,333E-33==,, "==+333.333E-33=="
1260 DATA 3.33333E-31,"+333.333E-33"
1270 DATA 103,0,1,0,"1,2",,=====,, "=====3,4"
1280 DATA 1,"2",,3",,4
2000 DATA 0,0,0,0,"Q"
3000 REM SUBROUTINE TO CHECK NUMERIC VALUES
3010 REM ONLY A LOOSE CRITERION OF EQUALITY IS USED
3020 LET E7=.999*E9

```

```
3030 LET E8=1.001*E9
3040 IF E9>=0 THEN 3100
3050 LET E6=E7
3060 LET E7=E8
3070 LET E8=E6
3100 LET F9=0
3110 IF E7>D9 THEN 3200
3120 IF E8<D9 THEN 3200
3130 RETURN
3200 LET F9=1
3210 RETURN
4000 END
```

---

#### PROGRAM FILE 110: MIXED INPUT OF STRINGS AND NUMBERS.

ANSI STANDARD 13.2, 13.4

#### SECTION 110.1: MIXED INPUT OF STRINGS AND NUMBERS.

PLEASE ENTER YOUR INPUT-REPLIES EXACTLY AS REQUESTED BY THE PROMPT, INCLUDING ALL SPACING, COMMAS AND OTHER PUNCTUATION.

- EXCEPT - WHEREVER THE PROMPT CONTAINS AN EQUALS (=) SUBSTITUTE A BLANK SPACE, AND WHEREVER THE PROMPT CONTAINS A NUMBER-SIGN (#) SUBSTITUTE A QUOTE.  
THUS, THE PROMPT:

==#ER#==

SHOULD CAUSE YOU TO STRIKE THE KEYS:

1. SPACE
2. SPACE
3. QUOTE
4. LETTER E
5. LETTER R
6. QUOTE
7. SPACE
8. SPACE

THESE CONVENTIONS ARE NECESSARY BECAUSE THE QUOTE IS UNPRINTABLE IN MINIMAL BASIC, AND SPACES ARE DIFFICULT TO COUNT ON A TERMINAL.

PLEASE NOTE THAT THIS ROUTINE IS STRONGLY DEPENDENT ON THE CORRECT OPERATION OF THE STRING COMPARISON FEATURE. IF THAT FEATURE IS INCORRECT, THIS ROUTINE WILL LIKELY GIVE INVALID RESULTS.

ALL THE REPLIES PROMPTED FOR ARE VALID, THEREFORE IF THE PROCESSOR REJECTS ANY OF THEM,

\*\*\* TEST FAILED \*\*\*

FOR USER CONVENIENCE, THE STRING TO BE ENTERED IS INDENTED TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE ALIGNED WITH THE PROMPT STRING.

BEGIN TEST

PLEASE ENTER:

```
==1==,==2==,==3==  
? 1 , 2 , 3  
SHOULD BE: *** 1 2 3 ***  
ACTUAL : *** 1 2 3 ***  
TEST OK
```

PLEASE ENTER:

```
==+987999E32==,== -1.00000E-37 ==,== 3.E37 ==  
? +987999E32 , -1.00000E-37 , 3.E37  
SHOULD BE: *** 9.87999E+37 -1.00000E-37 3.00000E+37 ***  
ACTUAL : *** 9.87999E+37 -1.00000E-37 3.00000E+37 ***  
TEST OK
```

PLEASE ENTER:

```
==222222,111111==,== 333333  
? 222222,111111 , 333333  
SHOULD BE: *** 222222 111111 333333 ***  
ACTUAL : *** 222222 111111 333333 ***  
TEST OK
```

PLEASE ENTER:

```
5,6  
? 5,6  
SHOULD BE: *** 5 6 ***  
ACTUAL : *** 5 6 ***  
TEST OK
```

PLEASE ENTER:

```
-05.34,345.567E-11  
? -05.34,345.567E-11  
SHOULD BE: ***-5.34 3.45567E-9 ***  
ACTUAL : ***-5.34 3.45567E-9 ***  
TEST OK
```

PLEASE ENTER:

```
2E2,-3.45  
? 2E2,-3.45  
SHOULD BE: *** 200 -3.45 ***  
ACTUAL : *** 200 -3.45 ***  
TEST OK
```

PLEASE ENTER:

```
-0000.000123456E-11,+1E37  
? -0000.000123456E-11,+1E37  
SHOULD BE: ***-1.23456E-15 1.00000E+37 ***  
ACTUAL : ***-1.23456E-15 1.00000E+37 ***  
TEST OK
```

PLEASE ENTER:  
-000.E-00,+.000,0E22  
? -000.E-00,+.000,0E22  
SHOULD BE: \*\*\* 0 0 0 \*\*\*  
ACTUAL : \*\*\* 0 0 0 \*\*\*  
TEST OK

PLEASE ENTER:  
-999.E-00,+.999,9E22  
? -999.E-00,+.999,9E22  
SHOULD BE: \*\*\*-999 0.999 9.00000E+22 \*\*\*  
ACTUAL : \*\*\*-999 0.999 9.00000E+22 \*\*\*  
TEST OK

PLEASE ENTER:  
ABC,##,#DEF#  
? ABC,"","DEF"  
SHOULD BE: \*\*\*ABCDEF\*\*\*  
ACTUAL : \*\*\*ABCDEF\*\*\*  
TEST OK

PLEASE ENTER:  
==4.56789E-11==,==MIDDLE=ITEM==,==9==  
? 4.56789E-11 , MIDDLE ITEM , 9  
SHOULD BE: \*\*\* 4.56789E-11 MIDDLE ITEM 9 \*\*\*  
ACTUAL : \*\*\* 4.56789E-11 MIDDLE ITEM 9 \*\*\*  
TEST OK

PLEASE ENTER:  
==987654==,==#=MIDDLE=ITEM==#=,==656565==  
? 987654 , " MIDDLE ITEM " , 656565  
SHOULD BE: \*\*\* 987654 MIDDLE ITEM 656565 \*\*\*  
ACTUAL : \*\*\* 987654 MIDDLE ITEM 656565 \*\*\*  
TEST OK

PLEASE ENTER:  
==AN=UNQUOTED=STRING==,==3.14159==,==#EQUALS=PI==  
? AN UNQUOTED STRING , 3.14159 , "EQUALS PI"  
SHOULD BE: \*\*\*AN UNQUOTED STRING 3.14159 EQUALS PI\*\*\*  
ACTUAL : \*\*\*AN UNQUOTED STRING 3.14159 EQUALS PI\*\*\*  
TEST OK

PLEASE ENTER:  
07676760000000E0000022==,=====X=====,X  
? 07676760000000E0000022 , X , X  
SHOULD BE: \*\*\* 7.67676E+34 XXX\*\*  
ACTUAL : \*\*\* 7.67676E+34 XXX\*\*  
TEST OK

PLEASE ENTER:  
=====##=====,==5==,====THIRD=ITEM=====  
? "" , 5 , THIRD ITEM  
SHOULD BE: \*\*\* 5 THIRD ITEM\*\*\*  
ACTUAL : \*\*\* 5 THIRD ITEM\*\*\*  
TEST OK

PLEASE ENTER:

=====#=#=, ==0====, ===##===

? " " , 0 , "

SHOULD BE: \*\*\* 0 \*\*\*

ACTUAL : \*\*\* 0 \*\*\*

TEST OK

PLEASE ENTER:

==+333.333E-33==, ==+333.333E-33==

? +333.333E-33 , +333.333E-33

SHOULD BE: \*\*\* 3.33333E-31 +333.333E-33\*\*\*

ACTUAL : \*\*\* 3.33333E-31 +333.333E-33\*\*\*

TEST OK

PLEASE ENTER:

1,2=====3,4

? 1,2 3,4

SHOULD BE: \*\*\* 1 2 3 4 \*\*\*

ACTUAL : \*\*\* 1 2 3 4 \*\*\*

TEST OK

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 110

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 111: EXCEPTION - NUMERIC UNDERFLOW ON INPUT"
20 PRINT "      CAUSES REPLACEMENT BY ZERO."
30 PRINT "      ANSI STANDARD 5.6, 13.4"
40 PRINT
50 PRINT "SECTION 111.1: EXCEPTION - NUMERIC UNDERFLOW ON INPUT"
60 PRINT "      CAUSES REPLACEMENT BY ZERO."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE"
140 PRINT "          DISPLAYED (NOT MANDATORY), AND"
150 PRINT
160 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE"
170 PRINT "          NUMERIC CONSTANT IN THE INPUT-REPLY."
180 PRINT
200 PRINT "AFTER THE INPUT-PROMPT, ENTER 1E-99999. THIS IS INTENDED"
210 PRINT "TO CAUSE NUMERIC UNDERFLOW. IF IT DOES, THE PROCESSOR"
220 PRINT "MUST SET THE VALUE OF THE VARIABLE TO ZERO."
260 PRINT
270 PRINT "SINCE '1E-99999' IS A SYNTACTICALLY LEGAL NUMBER,"
280 PRINT "IF THE PROCESSOR DOES NOT ACCEPT 1E-99999 AS INPUT,"
290 PRINT "*** TEST FAILS ***"
300 PRINT

```

```
310 PRINT "                                BEGIN TEST."
320 PRINT
330 PRINT "ENTER 1E-99999"
340 INPUT A
350 PRINT
360 PRINT "RESULTING VALUE =";A
370 PRINT
380 IF A=0 THEN 420
390 PRINT "RESULTING VALUE <> 0. IF VALUE <> 1E-99999, THEN,"
400 PRINT "*** TEST FAILED ***"
410 GOTO 430
420 PRINT "*** TEST PASSED ***"
430 PRINT
440 PRINT "                                END TEST."
450 PRINT
460 PRINT "END PROGRAM 111"
470 END
```

---

PROGRAM FILE 111: EXCEPTION - NUMERIC UNDERFLOW ON INPUT  
CAUSES REPLACEMENT BY ZERO.  
ANSI STANDARD 5.6, 13.4

SECTION 111.1: EXCEPTION - NUMERIC UNDERFLOW ON INPUT  
CAUSES REPLACEMENT BY ZERO.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MAY BE  
DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE  
NUMERIC CONSTANT IN THE INPUT-REPLY.

AFTER THE INPUT-PROMPT, ENTER 1E-99999. THIS IS INTENDED  
TO CAUSE NUMERIC UNDERFLOW. IF IT DOES, THE PROCESSOR  
MUST SET THE VALUE OF THE VARIABLE TO ZERO.

SINCE '1E-99999' IS A SYNTACTICALLY LEGAL NUMBER,  
IF THE PROCESSOR DOES NOT ACCEPT 1E-99999 AS INPUT,  
\*\*\* TEST FAILS \*\*\*

BEGIN TEST.

ENTER 1E-99999  
? 1E-99999

? UNDERFLOW ON NUMERIC-CONSTANT IN LINE 340; SUBSTITUTING ZERO

RESULTING VALUE = 0

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 111

```
*****  
*****  
*****  
5 PRINT "PROGRAM FILE 112: EXCEPTION - INPUT-REPLY INCONSISTENT WITH"  
7 PRINT "           INPUT VARIABLE-LIST."  
10 PRINT "     ANSI STANDARD 13.4, 13.5, 3.2, 5.2"  
20 PRINT  
25 PRINT "SECTION 112.1: EXCEPTION - INPUT-REPLY INCONSISTENT WITH"  
27 PRINT "           INPUT VARIABLE-LIST."  
30 PRINT  
35 PRINT "PLEASE ENTER YOUR INPUT-REPLIES EXACTLY AS REQUESTED"  
40 PRINT "BY THE PROMPT, INCLUDING ALL SPACING, COMMAS AND OTHER"  
45 PRINT "PUNCTUATION."  
50 PRINT  
55 PRINT " - EXCEPT - WHEREVER THE PROMPT CONTAINS AN EQUALS (=)"  
60 PRINT "SUBSTITUTE A BLANK SPACE, AND WHEREVER THE PROMPT"  
65 PRINT "CONTAINS A NUMBER-SIGN (#) SUBSTITUTE A QUOTE."  
70 PRINT "THUS, THE PROMPT:  
75 PRINT "      ==#ER#=="  
80 PRINT "SHOULD CAUSE YOU TO STRIKE THE KEYS:"  
85 PRINT " 1. SPACE"  
90 PRINT " 2. SPACE"  
95 PRINT " 3. QUOTE"  
100 PRINT " 4. LETTER E"  
105 PRINT " 5. LETTER R"  
110 PRINT " 6. QUOTE"  
115 PRINT " 7. SPACE"  
120 PRINT " 8. SPACE"  
125 PRINT  
130 PRINT "THESE CONVENTIONS ARE NECESSARY BECAUSE THE QUOTE IS"  
135 PRINT "UNPRINTABLE IN MINIMAL BASIC, AND SPACES ARE DIFFICULT"  
140 PRINT "TO COUNT ON A TERMINAL."  
145 PRINT  
150 PRINT "AFTER ENTERING THE REQUESTED REPLY, THE PROCESSOR SHOULD"  
155 PRINT "REPORT AN EXCEPTION OF THE KIND INDICATED IN THE PROMPT."  
160 PRINT "(ALTHOUGH SOME RESPONSES MAY VALIDLY CAUSE ONE OF SEVERAL"  
165 PRINT "DIFFERENT KINDS OF EXCEPTION TO BE REPORTED BY THE SYSTEM.)"  
170 PRINT  
175 PRINT "YOU SHOULD THEN BE GIVEN A CHANCE TO RE-SUPPLY INPUT AND"  
185 PRINT "SHOULD REPLY WITH AS MANY ZEROS AS INDICATED IN THE PROMPT."  
190 PRINT "(E.G., FOR 3 ITEMS, RESPOND: 0,0,0 TO BYPASS EXCEPTION.)"  
192 PRINT  
193 PRINT "*** IF NOT ALLOWED TO RE-SUPPLY INPUT, TEST FAILS ***"  
194 PRINT  
195 PRINT "IF THE PROCESSOR ACCEPTS AN INPUT-REPLY, IT FAILS THE"  
200 PRINT "TEST UNLESS 1) IT HAS A DOCUMENTED ENHANCED DEFINITION OF"  
205 PRINT "NUMERIC-CONSTANT, UNQUOTED-STRING, QUOTED-STRING, OR"  
210 PRINT "INPUT-REPLY SUCH AS TO RENDER MEANINGFUL A PARTICULAR"  
215 PRINT "NON-STANDARD RESPONSE, OR 2) ITS DOCUMENTED IMPLEMENTATION-"  
217 PRINT "DEFINED LIMITS FOR OVERFLOW (STRING OR NUMERIC) EXCEED"
```

```
218 PRINT "THE MINIMUM LIMITS ESTABLISHED BY THE STANDARD."
220 PRINT "NOT ALL THE INPUT-REPLIES ARE NON-STANDARD, BUT THEY"
225 PRINT "SHOULD ALL CAUSE EXCEPTIONS, SAVE AS NOTED ABOVE."
230 PRINT
235 PRINT "FOR USER CONVENIENCE, THE STRING TO BE ENTERED IS INDENTED"
240 PRINT "TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO"
245 PRINT "POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE"
250 PRINT "ALIGNED WITH THE PROMPT STRING."
255 PRINT
260 PRINT "               BEGIN TEST"
265 LET F=0
270 READ M,C(1),C(2),C(3),O,P$
275 LET Q$=""
280 LET R$=""
285 IF M<100 THEN 300
290 LET M=M-100
295 READ Q$,R$
300 IF M=0 THEN 2000
304 LET Q=0
308 FOR I2=1 TO M
312 LET Q=Q+Q+C(I2)
316 NEXT I2
320 LET Q=Q+1
324 IF M<=1 THEN 340
328 LET Q=Q+2
332 IF M<=2 THEN 340
336 LET Q=Q+4
340 LET Q1=Q
350 LET A1=1
355 LET B1=1
360 LET C1=1
365 LET A$="1"
370 LET B$="1"
375 LET C$="1"
380 PRINT
385 PRINT
390 PRINT "STATEMENT FOR THIS INPUT-REPLY: INPUT ";
395 IF C(1)=0 THEN 410
400 PRINT "A$";
405 GOTO 415
410 PRINT "A1";
415 IF M<=1 THEN 490
420 IF C(2)=0 THEN 435
425 PRINT ",B$";
430 GOTO 440
435 PRINT ",B1";
440 IF M<=2 THEN 490
445 IF C(3)=0 THEN 460
450 PRINT ",C$";
455 GOTO 490
460 PRINT ",C1";
490 PRINT
```

```
500 PRINT "EXCEPTION: ";
505 ON O GOTO 510,520,530,540,550
510 PRINT "WRONG TYPE";
515 GOTO 560
520 PRINT "TOO MUCH DATA";
525 GOTO 560
530 PRINT "INSUFFICIENT DATA";
535 GOTO 560
540 PRINT "NUMERIC OVERFLOW";
545 GOTO 560
550 PRINT "STRING OVERFLOW";
555 GOTO 560
560 PRINT "; SHOULD BE ";M;" ITEM(S)."
565 PRINT "PLEASE ENTER:"
570 PRINT " ";P$;Q$;R$
580 ON Q1 GOTO 585,595,605,615,625,635,645,655,665,675,685,695,705,715
585 INPUT A1
590 GOTO 725
595 INPUT A$
600 GOTO 725
605 INPUT A1,B1
610 GOTO 725
615 INPUT A1,B$
620 GOTO 725
625 INPUT A$,B1
630 GOTO 725
635 INPUT A$,B$
640 GOTO 725
645 INPUT A1,B1,C1
650 GOTO 725
655 INPUT A1,B1,C$
660 GOTO 725
665 INPUT A1,B$,C1
670 GOTO 725
675 INPUT A1,B$,C$
680 GOTO 725
685 INPUT A$,B1,C1
690 GOTO 725
695 INPUT A$,B1,C$
700 GOTO 725
705 INPUT A$,B$,C1
710 GOTO 725
715 INPUT A$,B$,C$
720 GOTO 725
725 REM RESULTS DISPLAYED
728 REM Y2 IS ERROR SWITCH, Y IS ITEM COUNT.
730 LET Y2=0
735 LET Y=1
740 LET Y1=A1
745 LET Z$=A$
750 GOSUB 1000
755 IF M<=1 THEN 795
760 LET Y1=B1
765 LET Z$=B$
770 GOSUB 1000
```

```
775 IF M<=2 THEN 795
780 LET Y1=C1
785 LET Z$=C$
790 GOSUB 1000
795 IF Y2<>0 THEN 810
800 PRINT "TEST OK."
805 GOTO 270
810 PRINT "NOT ALL VARIABLES EQUAL TO ZERO."
820 PRINT "POSSIBILITIES:"
825 PRINT "1 - PROCESSOR FAILED TO TREAT REPLY AS EXCEPTION"
835 PRINT "2 - USER REPLY TO BYPASS EXCEPTION NOT ALL ZEROS"
840 PRINT "3 - PROCESSOR DID NOT ALLOW RE-ENTRY OF INPUT-REPLY."
845 PRINT "RE-TRY (Y OR N)?"
850 INPUT W$
855 IF W$<>"N" THEN 350
860 PRINT "TEST FAILS, UNLESS DOCUMENTED SYNTACTIC ENHANCEMENT."
865 LET F=F+1
870 GOTO 270
1000 REM SUBROUTINE TO CHECK AND DISPLAY ONE ITEM
1002 PRINT "ITEM#";Y;":";
1005 IF C(Y)=0 THEN 1025
1010 PRINT Z$
1015 IF Z$="0" THEN 1040
1020 GOTO 1035
1025 PRINT Y1
1030 IF Y1=0 THEN 1040
1035 LET Y2=9
1040 LET Y=Y+1
1045 RETURN
2000 PRINT
2010 IF F=0 THEN 2040
2020 PRINT "*** POSSIBLE TEST FAILURE IN ";F;" CASE(S). ***"
2030 GOTO 2050
2040 PRINT "*** TEST PASSED ***"
2050 PRINT
2060 PRINT "          END TEST"
2070 PRINT
2080 PRINT "END PROGRAM 112"
3000 REM 1      - #ITEMS(+100 IF LONG PROMPT),
3010 REM 2-4    - 3-DIGIT A/N CODE (0=>N, 1=>A)
3020 REM 5      - EXCP CODE
3030 REM 6(-8)  - PROMPT (3 STRINGS IF LONG)
3040 DATA 3,1,1,1,2,"M,M,M,M"
3050 DATA 3,1,1,1,3,"M,M"
3060 DATA 1,0,0,0,4,"1E99999"
3070 DATA 101,1,1,1,5,"IF=THIS=DOES=NOT=C","AUSE=STRING=OVRFLW"
3080 DATA           "=TRY=LONGER=REPLY"
3090 DATA 1,1,1,1,1,"AB?CD"
3100 DATA 1,1,1,1,1,"AB;CD"
3110 DATA 1,1,1,1,1,"K*L"
3120 DATA 3,0,0,0,1,"1,Q,1"
3125 DATA 1,0,0,0,1,"1D1"
3130 DATA 1,1,1,1,1,"AB##CD"
3140 DATA 1,1,1,1,1,"AB#CD"
3150 DATA 1,1,1,1,1,"#AB"
```

```
3160 DATA 1,1,1,1,1,"AB#"  
3170 DATA 1,1,1,1,1,"#AB##CD#"  
3180 DATA 1,1,1,1,1,"#AB#CD#"  
3190 DATA 2,1,1,1,1,"AB#CD,EF"  
3200 DATA 2,1,1,1,1,"AB,CD#EF"  
3210 DATA 2,1,1,1,1,"A#B,C#D"  
3220 DATA 3,1,1,1,1,"A,,B"  
3230 DATA 2,1,1,1,2,"X,Y,"  
3240 DATA 3,1,1,1,1,"X,Y,"  
3250 DATA 3,1,1,1,1,"A,B"  
3260 DATA 101,1,1,1,1,"      NULL REPLY ","(HIT RETURN ONLY)",""  
3270 DATA 2,0,0,0,1,"2==3"  
3280 DATA 1,0,0,0,1,"2==3"  
3290 DATA 3,1,1,1,1,"X,==:,Y"  
4000 DATA 0,0,0,0,0,"Q"  
4010 END
```

---

PROGRAM FILE 112: EXCEPTION - INPUT-REPLY INCONSISTENT WITH  
INPUT VARIABLE-LIST.

ANSI STANDARD 13.4, 13.5, 3.2, 5.2

SECTION 112.1: EXCEPTION - INPUT-REPLY INCONSISTENT WITH  
INPUT VARIABLE-LIST.

PLEASE ENTER YOUR INPUT-REPLIES EXACTLY AS REQUESTED  
BY THE PROMPT, INCLUDING ALL SPACING, COMMAS AND OTHER  
PUNCTUATION.

- EXCEPT - WHEREVER THE PROMPT CONTAINS AN EQUALS (=)  
SUBSTITUTE A BLANK SPACE, AND WHEREVER THE PROMPT  
CONTAINS A NUMBER-SIGN (#) SUBSTITUTE A QUOTE.

THUS, THE PROMPT:

==#ER#==

SHOULD CAUSE YOU TO STRIKE THE KEYS:

1. SPACE
2. SPACE
3. QUOTE
4. LETTER E
5. LETTER R
6. QUOTE
7. SPACE
8. SPACE

THESE CONVENTIONS ARE NECESSARY BECAUSE THE QUOTE IS  
UNPRINTABLE IN MINIMAL BASIC, AND SPACES ARE DIFFICULT  
TO COUNT ON A TERMINAL.

AFTER ENTERING THE REQUESTED REPLY, THE PROCESSOR SHOULD  
REPORT AN EXCEPTION OF THE KIND INDICATED IN THE PROMPT.  
(ALTHOUGH SOME RESPONSES MAY VALIDLY CAUSE ONE OF SEVERAL  
DIFFERENT KINDS OF EXCEPTION TO BE REPORTED BY THE SYSTEM.)

YOU SHOULD THEN BE GIVEN A CHANCE TO RE-SUPPLY INPUT AND SHOULD REPLY WITH AS MANY ZEROS AS INDICATED IN THE PROMPT. (E.G., FOR 3 ITEMS, RESPOND: 0,0,0 TO BYPASS EXCEPTION.)

\*\*\* IF NOT ALLOWED TO RE-SUPPLY INPUT, TEST FAILS \*\*\*

IF THE PROCESSOR ACCEPTS AN INPUT-REPLY, IT FAILS THE TEST UNLESS 1) IT HAS A DOCUMENTED ENHANCED DEFINITION OF NUMERIC-CONSTANT, UNQUOTED-STRING, QUOTED-STRING, OR INPUT-REPLY SUCH AS TO RENDER MEANINGFUL A PARTICULAR NON-STANDARD RESPONSE, OR 2) ITS DOCUMENTED IMPLEMENTATION-DEFINED LIMITS FOR OVERFLOW (STRING OR NUMERIC) EXCEED THE MINIMUM LIMITS ESTABLISHED BY THE STANDARD.

NOT ALL THE INPUT-REPLIES ARE NON-STANDARD, BUT THEY SHOULD ALL CAUSE EXCEPTIONS, SAVE AS NOTED ABOVE.

FOR USER CONVENIENCE, THE STRING TO BE ENTERED IS INDENTED TWO SPACES, SO THAT IF THE INPUT-PROMPT OCCUPIES TWO POSITIONS, AS RECOMMENDED, YOUR INPUT-REPLY WILL BE ALIGNED WITH THE PROMPT STRING.

BEGIN TEST

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$,C\$  
EXCEPTION: TOO MUCH DATA; SHOULD BE 3 ITEM(S).

PLEASE ENTER:

M,M,M,M  
? M,M,M,M

? EXCESS DATA IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 :0  
ITEM# 2 :0  
ITEM# 3 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$,C\$  
EXCEPTION: INSUFFICIENT DATA; SHOULD BE 3 ITEM(S).  
PLEASE ENTER:

M,M  
? M,M

? INSUFFICIENT DATA IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 :0  
ITEM# 2 :0  
ITEM# 3 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A1  
EXCEPTION: NUMERIC OVERFLOW; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

1E99999  
? 1E99999

? NUMERIC OVERFLOW IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 : 0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: STRING OVERFLOW; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:  
IF=THIS=DOES=NOT=CAUSE=STRING=OVRFLW=TRY=LONGER=REPLY  
? IF THIS DOES NOT CAUSE STRING OVRFLW TRY LONGER REPLY

? STRING OVERFLOW IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

AB?CD  
? AB?CD

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

AB;CD  
? AB;CD

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

K\*L  
? K\*L

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A1,B1,C1  
EXCEPTION: WRONG TYPE; SHOULD BE 3 ITEM(S).  
PLEASE ENTER:

1,Q,1  
? 1,Q,1

```
? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 : 0  
ITEM# 2 : 0  
ITEM# 3 : 0  
TEST OK.
```

```
STATEMENT FOR THIS INPUT-REPLY: INPUT A1  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:
```

```
1D1  
? 1D1
```

```
? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 : 0  
TEST OK.
```

```
STATEMENT FOR THIS INPUT-REPLY: INPUT A$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:
```

```
AB##CD  
? AB""CD
```

```
? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.
```

```
STATEMENT FOR THIS INPUT-REPLY: INPUT A$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:
```

```
AB#CD  
? AB"CD
```

```
? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.
```

```
STATEMENT FOR THIS INPUT-REPLY: INPUT A$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:
```

```
#AB  
? "AB
```

```
? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.
```

```
STATEMENT FOR THIS INPUT-REPLY: INPUT A$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:
```

```
AB#
```

? AB"

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

#AB##CD#

? "AB""CD"

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

#AB#CD#

? "AB"CD"

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$  
EXCEPTION: WRONG TYPE; SHOULD BE 2 ITEM(S).  
PLEASE ENTER:

AB#CD,EF

? AB"CD,EF

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0  
ITEM# 1 :0  
ITEM# 2 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$  
EXCEPTION: WRONG TYPE; SHOULD BE 2 ITEM(S).  
PLEASE ENTER:

AB,CD#EF

? AB,CD"EF

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0  
ITEM# 1 :0  
ITEM# 2 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$  
EXCEPTION: WRONG TYPE; SHOULD BE 2 ITEM(S).  
PLEASE ENTER:  
A#B,C#D  
? A"B,C"D

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0  
ITEM# 1 :0  
ITEM# 2 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$,C\$  
EXCEPTION: WRONG TYPE; SHOULD BE 3 ITEM(S).  
PLEASE ENTER:

A,,B  
? A,,B

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 :0  
ITEM# 2 :0  
ITEM# 3 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$  
EXCEPTION: TOO MUCH DATA; SHOULD BE 2 ITEM(S).  
PLEASE ENTER:

X,Y,  
? X,Y,

? EXCESS DATA IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0  
ITEM# 1 :0  
ITEM# 2 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$,C\$  
EXCEPTION: WRONG TYPE; SHOULD BE 3 ITEM(S).  
PLEASE ENTER:

X,Y,  
? X,Y,

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 :0  
ITEM# 2 :0  
ITEM# 3 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$,C\$  
EXCEPTION: WRONG TYPE; SHOULD BE 3 ITEM(S).  
PLEASE ENTER:

,A,B  
? ,A,B

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 :0  
ITEM# 2 :0  
ITEM# 3 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

NULL REPLY (HIT RETURN ONLY)

?

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 :0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A1,B1  
EXCEPTION: WRONG TYPE; SHOULD BE 2 ITEM(S).  
PLEASE ENTER:

2==3  
? 2 3

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0  
ITEM# 1 : 0  
ITEM# 2 : 0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A1  
EXCEPTION: WRONG TYPE; SHOULD BE 1 ITEM(S).  
PLEASE ENTER:

2==3  
? 2 3

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0  
ITEM# 1 : 0  
TEST OK.

STATEMENT FOR THIS INPUT-REPLY: INPUT A\$,B\$,C\$  
EXCEPTION: WRONG TYPE; SHOULD BE 3 ITEM(S).  
PLEASE ENTER:

X,==:,Y  
? X, ,Y

? INCORRECT DATA TYPE IN INPUT-REPLY; PLEASE RE-SUPPLY  
? 0,0,0  
ITEM# 1 :0  
ITEM# 2 :0  
ITEM# 3 :0  
TEST OK.

```
*** TEST PASSED ***
```

```
END TEST
```

```
END PROGRAM 112
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 113: ERROR - NULL ENTRY IN INPUT-LIST."  
20 PRINT "      ANSI STANDARD 13.2"  
30 PRINT  
40 PRINT "SECTION 113.1: ERROR - NULL ENTRY IN INPUT-LIST."  
50 PRINT  
60 PRINT  
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
90 PRINT  
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
110 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
120 PRINT "          BY THE PROCESSOR, OR"  
130 PRINT  
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
150 PRINT  
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
170 PRINT "FOR DETAILED CRITERIA."  
180 PRINT  
190 PRINT "                  BEGIN TEST"  
200 PRINT  
210 PRINT "VARIABLES A AND B INITIALIZED TO ZERO."  
220 PRINT  
230 LET A=0  
240 LET B=0  
250 PRINT "PLEASE ENTER: 1,2,3"  
260 PRINT "STATEMENT FOR THIS REPLY: INPUT A,,B"  
270 INPUT A,,B  
280 PRINT "A = ";A;"      B = ";B  
290 PRINT  
300 PRINT "                  END TEST"  
310 PRINT  
320 PRINT "END PROGRAM 113"  
330 END
```

---

```
? , WAS SEEN WHERE A LETTER WAS EXPECTED IN LINE 270
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 114: EVALUATION OF ABS FUNCTION."  
20 PRINT "      ANSI STANDARD 8.4"  
30 PRINT
```

```

40 PRINT "SECTION 114.1: EVALUATION OF ABS FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHETHER THE ABS NUMERIC-SUPPLIED-FUNCTION"
70 PRINT "RETURNS THE ABSOLUTE VALUE FOR A VARIETY OF ARGUMENTS."
100 PRINT
110 PRINT "               BEGIN TEST"
120 PRINT
130 LET F=0
140 PRINT " ,,"CORRECT",,"ACTUAL",,"TEST"
150 PRINT "ARGUMENT", "VALUE", "VALUE", "RESULT"
160 PRINT
170 READ A,V
180 IF A=999 THEN 290
190 LET V1=ABS(A)
200 IF V=V1 THEN 270
240 LET F=F+1
250 PRINT A,V,V1,"FAIL"
260 GOTO 170
270 PRINT A,V,V1," OK "
280 GOTO 170
290 PRINT
300 IF F=0 THEN 330
310 PRINT "*** TEST FAILED IN ";F;" CASE(S) ***"
320 GOTO 340
330 PRINT "*** TEST PASSED ***"
340 PRINT
350 PRINT "               END TEST"
360 PRINT
370 PRINT "END PROGRAM 114"
380 DATA -2,2,-1,1,+0,0,-0,0,1,1,2,2,2.1,2.1,2.9,2.9,-2.1,2.1,-2.9,2.9
390 DATA -1.23E22,1.23E22,3.21E-11,3.21E-11,9.99E37,9.99E37
400 DATA -9.99E37,9.99E37,1E38,1E38,-1E38,1E38,1E-38,1E-38
410 DATA -1E-38,1E-38,-1.01E-38,1.01E-38,.001,.001,-.001,.001
420 DATA 999,999
430 END

```

---

PROGRAM FILE 114: EVALUATION OF ABS FUNCTION.  
ANSI STANDARD 8.4

#### SECTION 114.1: EVALUATION OF ABS FUNCTION.

THIS SECTION TESTS WHETHER THE ABS NUMERIC-SUPPLIED-FUNCTION  
RETURNS THE ABSOLUTE VALUE FOR A VARIETY OF ARGUMENTS.

BEGIN TEST

ARGUMENT	CORRECT VALUE	ACTUAL VALUE	TEST RESULT
----------	------------------	-----------------	----------------

-2	2	2	OK
-1	1	1	OK
0	0	0	OK
0	0	0	OK
1	1	1	OK
2	2	2	OK
2.1	2.1	2.1	OK
2.9	2.9	2.9	OK
-2.1	2.1	2.1	OK
-2.9	2.9	2.9	OK
-1.23000E+22	1.23000E+22	1.23000E+22	OK
3.21000E-11	3.21000E-11	3.21000E-11	OK
9.99000E+37	9.99000E+37	9.99000E+37	OK
-9.99000E+37	9.99000E+37	9.99000E+37	OK
1.00000E+38	1.00000E+38	1.00000E+38	OK
-1.00000E+38	1.00000E+38	1.00000E+38	OK
1.00000E-38	1.00000E-38	1.00000E-38	OK
-1.00000E-38	1.00000E-38	1.00000E-38	OK
-1.01000E-38	1.01000E-38	1.01000E-38	OK
.001	.001	.001	OK
-.001	.001	.001	OK

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 114

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 115: EVALUATION OF INT FUNCTION."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 115.1: EVALUATION OF INT FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHETHER THE INT NUMERIC-SUPPLIED-FUNCTION"
70 PRINT "RETURNS THE GREATEST INTEGER LESS THAN OR EQUAL TO"
80 PRINT "THE VALUE OF THE ARGUMENT PASSED TO THE FUNCTION."
110 PRINT
120 PRINT "                  BEGIN TEST"
130 PRINT
140 LET F=0
150 PRINT "  ","CORRECT","ACTUAL","TEST"
160 PRINT "ARGUMENT","VALUE","VALUE","RESULT"
170 PRINT
180 READ A,V
190 IF A=999 THEN 300
200 LET V1=INT(A)
210 IF V=V1 THEN 280
250 LET F=F+1
260 PRINT A,V,V1,"FAIL"

```

```
270 GOTO 180
280 PRINT A,V,V1," OK "
290 GOTO 180
300 REM DISPLAY RESULTS
310 PRINT
320 IF F=0 THEN 350
330 PRINT "*** TEST FAILED IN ";F;" CASE(S) ***"
340 GOTO 360
350 PRINT "*** TEST PASSED ***"
360 PRINT
370 PRINT " END TEST"
380 PRINT
390 PRINT
400 PRINT "SECTION 115.2 EVALUATION OF INT FUNCTION FOR VALUES"
410 PRINT " OF LARGE MAGNITUDE."
420 PRINT
430 PRINT "THIS IS AN INFORMATIVE TEST TO EXAMINE THE BEHAVIOR OF"
440 PRINT "THE INT FUNCTION WHEN THE RESULT DEPENDS ON CALCULATION"
450 PRINT "BEYOND SIX DECIMAL DIGITS. THIS SECTION IS NOT CONSIDERED"
460 PRINT "EITHER TO PASS OR FAIL."
470 PRINT
480 PRINT " BEGIN TEST"
490 PRINT
500 PRINT " ","CORRECT","ACTUAL"
510 PRINT "ARGUMENT","VALUE","VALUE"
520 PRINT
530 READ A,A$,V$
540 IF A=999 THEN 580
550 LET V=INT(A)
560 PRINT A$,V$,V
570 GOTO 530
580 PRINT
590 PRINT " END TEST"
600 PRINT
610 PRINT "END PROGRAM 115"
620 DATA 3,3,-3,-3,2.1,2,2.9,2,-2.1,-3,-2.9,-3
630 DATA 1,1,-1,-1,0,0,.1,0,.9,0,-.1,-1,-.9,-1
640 DATA 1234.2,1234,1234.9,1234,-12345.6,-12346
650 DATA 3.99999,3,-44.0001,-45
660 DATA -123E2,-12300,-456E-2,-5,789999E-4,78
670 DATA 1E-38,0,123.45678E-3,0,123.456E-33,0
680 DATA -1E-38,-1,-123.45678E-3,-1,-123.456E-33,-1
690 DATA 999,999
700 DATA 987654.1,"987654.1","987654",987654.9,"987654.9","987654"
710 DATA 1234567.9,"1234567.9","1234567",12345678.9,"12345678.9"
720 DATA "12345678",123456789.9,"123456789.9","123456789"
730 DATA 1234567890.9,"1234567890.9","1234567890"
740 DATA -987654.1,"-987654.1","-987655"
750 DATA -987654.9,"-987654.9","-987655"
760 DATA -1234567.1,"-1234567.1","-1234568"
770 DATA -12345678.1,"-12345678.1","-12345679"
780 DATA -123456789.1,"-123456789.1","-123456790"
790 DATA -1234567890.1,"-1234567890.1","-1234567891"
2000 DATA 999,"Q","Q"
2010 END
```

PROGRAM FILE 115: EVALUATION OF INT FUNCTION.  
ANSI STANDARD 8.4

SECTION 115.1: EVALUATION OF INT FUNCTION.

THIS SECTION TESTS WHETHER THE INT NUMERIC-SUPPLIED-FUNCTION RETURNS THE GREATEST INTEGER LESS THAN OR EQUAL TO THE VALUE OF THE ARGUMENT PASSED TO THE FUNCTION.

BEGIN TEST

ARGUMENT	CORRECT VALUE	ACTUAL VALUE	TEST RESULT
3	3	3	OK
-3	-3	-3	OK
2.1	2	2	OK
2.9	2	2	OK
-2.1	-3	-3	OK
-2.9	-3	-3	OK
1	1	1	OK
-1	-1	-1	OK
0	0	0	OK
.1	0	0	OK
.9	0	0	OK
-.1	-1	-1	OK
-.9	-1	-1	OK
1234.2	1234	1234	OK
1234.9	1234	1234	OK
-12345.6	-12346	-12346	OK
3.99999	3	3	OK
-44.0001	-45	-45	OK
-12300	-12300	-12300	OK
-4.56	-5	-5	OK
78.9999	78	78	OK
1.00000E-38	0	0	OK
.123457	0	0	OK
1.23456E-31	0	0	OK
-1.00000E-38	-1	-1	OK
-.123457	-1	-1	OK
-1.23456E-31	-1	-1	OK

\*\*\* TEST PASSED \*\*\*

END TEST

SECTION 115.2 EVALUATION OF INT FUNCTION FOR VALUES OF LARGE MAGNITUDE.

THIS IS AN INFORMATIVE TEST TO EXAMINE THE BEHAVIOR OF THE INT FUNCTION WHEN THE RESULT DEPENDS ON CALCULATION BEYOND SIX DECIMAL DIGITS. THIS SECTION IS NOT CONSIDERED EITHER TO PASS OR FAIL.

BEGIN TEST

ARGUMENT	CORRECT VALUE	ACTUAL VALUE
987654.1	987654	987654
987654.9	987654	987654
1234567.9	1234567	1.23457E+6
12345678.9	12345678	1.23457E+7
123456789.9	123456789	1.23457E+8
1234567890.9	1234567890	1.23457E+9
-987654.1	-987655	-987655
-987654.9	-987655	-987655
-1234567.1	-1234568	-1.23457E+6
-12345678.1	-12345679	-1.23457E+7
-123456789.1	-123456790	-1.23457E+8
-1234567890.1	-1234567891	-1.23457E+9

END TEST

END PROGRAM 115

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 116: EVALUATION OF SGN FUNCTION."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 116.1: EVALUATION OF SGN FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHETHER THE SGN NUMERIC-SUPPLIED-FUNCTION"
70 PRINT "RETURNS THE CORRECT VALUE FOR A VARIETY OF ARGUMENTS."
100 PRINT
110 PRINT "                  BEGIN TEST"
120 PRINT
130 LET F=0
140 PRINT "  ", "CORRECT", "ACTUAL", "TEST"
150 PRINT "ARGUMENT", "VALUE", "VALUE", "RESULT"
160 PRINT
170 READ A,V
180 IF A=999 THEN 280
190 LET V1=SGN(A)
210 LET R$=" OK "
220 IF V=V1 THEN 260
240 LET R$="FAIL"
250 LET F=F+1
260 PRINT A,V,V1,R$
270 GOTO 170
280 PRINT

```

```

290 IF F=0 THEN 320
300 PRINT "*** TEST FAILED IN ";F;" CASE(S) ***"
310 GOTO 330
320 PRINT "*** TEST PASSED ***"
330 PRINT
340 PRINT " END TEST"
350 PRINT
360 PRINT "END PROGRAM 116"
370 DATA 0,0,1,1,-1,-1,3,1,-4,-1,86.1234,1,-345.678E22,-1
380 DATA +22.2222E-33,1,.0000000123,1
390 DATA 1E38,1,-1E38,-1,1E-38,1,-1E-38,-1,-0,0,+0,0
400 DATA -00.00E11,0
410 DATA 999,999
420 END

```

---

PROGRAM FILE 116: EVALUATION OF SGN FUNCTION.

ANSI STANDARD 8.4

#### SECTION 116.1: EVALUATION OF SGN FUNCTION.

THIS SECTION TESTS WHETHER THE SGN NUMERIC-SUPPLIED-FUNCTION RETURNS THE CORRECT VALUE FOR A VARIETY OF ARGUMENTS.

BEGIN TEST

ARGUMENT	CORRECT VALUE	ACTUAL VALUE	TEST RESULT
0	0	0	OK
1	1	1	OK
-1	-1	-1	OK
3	1	1	OK
-4	-1	-1	OK
86.1234	1	1	OK
-3.45678E+24	-1	-1	OK
2.2222E-32	1	1	OK
1.23000E-8	1	1	OK
1.00000E+38	1	1	OK
-1.00000E+38	-1	-1	OK
1.00000E-38	1	1	OK
-1.00000E-38	-1	-1	OK
0	0	0	OK
0	0	0	OK
0	0	0	OK

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 116

```
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 117: ACCURACY OF SQR FUNCTION."  
20 PRINT "    ANSI STANDARD 7.6, 8.4"  
30 PRINT  
40 PRINT "SECTION 117.1: ACCURACY OF SQR FUNCTION."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"  
70 PRINT "SQR FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"  
80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"  
90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"  
100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"  
110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"  
120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"  
130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"  
140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"  
150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."  
160 PRINT  
170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"  
180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."  
190 PRINT  
200 PRINT "           BEGIN TEST"  
210 PRINT  
220 PRINT "ARGUMENT","TRUE","COMPUTED","ERROR","OUTCOME"  
230 PRINT " ","VALUE","VALUE","MEASURE"  
240 PRINT  
250 LET F=0  
260 READ A,T,L,H  
270 IF A=999 THEN 2000  
280 LET C=SQR(A)  
290 IF C=T THEN 333  
300 IF C<L THEN 444  
310 IF C>H THEN 444  
320 LET O$=" OK "  
330 GOTO 500  
333 LET O$=" OK - EXACT"  
340 GOTO 500  
444 LET O$="FAIL"  
445 LET F=F+1  
450 GOTO 500  
500 LET R=H-L  
510 IF R=0 THEN 700  
520 LET E=ABS((C-T)/R)  
525 LET E=E+E  
530 PRINT A,T,C,E,O$  
540 GOTO 260  
700 PRINT A,T,C,"RANGE ZERO",O$  
710 GOTO 260  
2000 PRINT
```

```
2010 IF F=0 THEN 2100
2020 PRINT "**** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ****"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT "           END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 117"
2990 DATA 0,0,0,0
3000 DATA 1.000000000E-38, 0.100000000E-18
3010 DATA 0.999994499E-19, 0.100001000E-18
3040 DATA 0.999999000E-37, 0.316227608E-18
3050 DATA 0.316226949E-18, 0.316228267E-18
3052 DATA 0.900000000E-37, 0.300000000E-18
3055 DATA 0.299999333E-18, 0.300000667E-18
3060 DATA 0.999999000E-36, 0.999999500E-18
3070 DATA 0.9999984499E-18, 0.100000500E-17
3080 DATA 0.999999000E-30, 0.999999500E-15
3090 DATA 0.9999984499E-15, 1.000000500E-15
3100 DATA 0.123456000E-19, 0.111110756E-09
3110 DATA 0.111109805E-09, 0.111111706E-09
3120 DATA 0.654321000E-04, 0.808901106E-02
3130 DATA 0.808899988E-02, 0.808902225E-02
3140 DATA 0.500000000E+00, 0.707106781E+00
3150 DATA 0.707105574E+00, 0.707107989E+00
3160 DATA 0.100000000E+01, 0.100000000E+01
3170 DATA 0.999994499E+00, 0.100001000E+01
3180 DATA 0.999999000E+00, 0.999999500E+00
3190 DATA 0.9999984499E+00, 0.100000500E+01
3200 DATA 0.200000000E+01, 0.141421356E+01
3210 DATA 0.141420502E+01, 0.141422210E+01
3220 DATA 0.300000000E+01, 0.173205081E+01
3230 DATA 0.173204292E+01, 0.173205870E+01
3240 DATA 0.400000000E+01, 0.200000000E+01
3250 DATA 0.199999249E+01, 0.200000750E+01
3260 DATA 0.500000000E+01, 0.223606798E+01
3270 DATA 0.223606074E+01, 0.223607522E+01
3280 DATA 0.999999000E+01, 0.316227608E+01
3290 DATA 0.316226949E+01, 0.316228267E+01
3300 DATA 0.100000000E+02, 0.316227766E+01
3310 DATA 0.316225684E+01, 0.316229848E+01
3320 DATA 0.400000000E+05, 0.200000000E+03
3330 DATA 0.199999249E+03, 0.200000750E+03
3340 DATA 0.123456000E+09, 0.111110756E+05
3350 DATA 0.111109805E+05, 0.111111706E+05
3360 DATA 0.654321000E+21, 0.255796990E+11
3370 DATA 0.255796294E+11, 0.255797686E+11
3380 DATA 0.999999000E+32, 0.999999500E+16
3390 DATA 0.9999984499E+16, 1.000000500E+16
3392 DATA 0.900000000E+37, 0.300000000E+19
3394 DATA 0.299999333E+19, 0.300000667E+19
3400 DATA 0.100000000E+39, 1.000000000E+19
3410 DATA 0.999994499E+19, 0.100001000E+20
9000 DATA 999,0,0,0
9400 END
```

## PROGRAM FILE 117: ACCURACY OF SQR FUNCTION.

ANSI STANDARD 7.6, 8.4

## SECTION 117.1: ACCURACY OF SQR FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE SQR FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST  $2 * \text{THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE}$ ; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME	
0	0	0	RANGE ZERO	OK	- EXACT
1.00000E-38	1.00000E-19	1.00000E-19	1.04167E-3	OK	
9.99999E-38	3.16228E-19	3.16228E-19	0	OK	- EXACT
9.00000E-38	3.00000E-19	3.00000E-19	4.85437E-3	OK	
9.99999E-37	1.00000E-18	1.00000E-18	0	OK	- EXACT
9.99999E-31	1.00000E-15	1.00000E-15	0	OK	- EXACT
1.23456E-20	1.11111E-10	1.11111E-10	5.47945E-3	OK	
6.54321E-5	8.08901E-3	8.08901E-3	1.04167E-2	OK	
.5	.707107	.707107	6.17284E-3	OK	
1	1	1	0	OK	- EXACT
.999999	1.	1.	2.28571E-3	OK	
2	1.41421	1.41421	8.71840E-3	OK	
3	1.73205	1.73205	1.88857E-3	OK	
4	2	2	0	OK	- EXACT
5	2.23607	2.23607	1.23203E-2	OK	
9.99999	3.16228	3.16228	1.35135E-2	OK	
10	3.16228	3.16228	2.86533E-3	OK	
40000	200	200	0	OK	- EXACT
1.23456E+8	11111.1	11111.1	5.13809E-3	OK	
6.54321E+20	2.55797E+10	2.55797E+10	3.68324E-3	OK	
9.99999E+31	1.00000E+16	1.00000E+16	0	OK	- EXACT
9.00000E+36	3.00000E+18	3.00000E+18	0	OK	- EXACT
1.00000E+38	1.00000E+19	1.00000E+19	0	OK	- EXACT

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 117

A large grid of black asterisks on a white background, arranged in approximately 20 rows and 40 columns.

```
10 PRINT "PROGRAM FILE 118: EXCEPTION - SQR OF NEGATIVE ARGUMENT."
20 PRINT "      ANSI STANDARD 8.5"
30 PRINT
40 PRINT "SECTION 118.1: EXCEPTION - SQR OF NEGATIVE ARGUMENT."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A NEGATIVE ARGUMENT"
90 PRINT "IS PASSED TO THE SQR FUNCTION."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE NEGATIVE"
170 PRINT "          ARGUMENT."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 PRINT "FATAL EXCEPTION SHOULD OCCUR NOW:"
220 PRINT
230 LET A=(-3)
240 LET B=SQR(A)
255 PRINT "*** TEST FAILED ***"
257 PRINT
260 PRINT "ARGUMENT WAS SET TO -3; IT NOW EQUALS ";A
270 PRINT "RESULT OF SQR IS ";B
280 PRINT
330 PRINT
340 PRINT "                                END TEST"
350 PRINT
360 PRINT "END PROGRAM 118"
370 END
```

PROGRAM FILE 118: EXCEPTION - SQR OF NEGATIVE ARGUMENT.  
ANSI STANDARD 8.5

SECTION 118.1: EXCEPTION = SQR OF NEGATIVE ARGUMENT.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR RECOGNIZES A FATAL EXCEPTION WHEN A NEGATIVE ARGUMENT IS PASSED TO THE SQR FUNCTION.

**TO PASS THIS TEST:**

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE NEGATIVE ARGUMENT.

BEGIN TEST.

FATAL EXCEPTION SHOULD OCCUR NOW:

? SQRT OF NEGATIVE NUMBER IN LINE 240

```
*****
*****  

10 PRINT "PROGRAM FILE 119: ACCURACY OF ATN FUNCTION."  

20 PRINT "      ANSI STANDARD 7.6, 8.4"  

30 PRINT  

40 PRINT "SECTION 119.1: ACCURACY OF ATN FUNCTION."  

50 PRINT  

60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"  

70 PRINT "ATN FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"  

80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"  

90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"  

100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"  

110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"  

120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"  

130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"  

140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"  

150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."  

160 PRINT  

170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"  

180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."  

190 PRINT  

200 PRINT "          BEGIN TEST"  

210 PRINT  

220 PRINT "ARGUMENT","TRUE","COMPUTED","ERROR","OUTCOME"  

230 PRINT " ","VALUE","VALUE","MEASURE"  

240 PRINT  

250 LET F=0  

260 READ A,T,L,H  

270 IF A=999 THEN 2000  

280 LET C=ATN(A)  

290 IF C=T THEN 333  

300 IF C<L THEN 444  

310 IF C>H THEN 444  

320 LET O$=" OK "  

330 GOTO 500  

333 LET O$=" OK - EXACT"  

340 GOTO 500  

444 LET O$="FAIL"  

445 LET F=F+1  

450 GOTO 500
```

```
500 LET R=H-L
510 IF R=0 THEN 700
520 LET E=ABS((C-T)/R)
525 LET E=E+E
530 PRINT A,T,C,E,O$
540 GOTO 260
700 PRINT A,T,C,"RANGE ZERO",O$
710 GOTO 260
2000 PRINT
2010 IF F=0 THEN 2100
2020 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT "           END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 119"
3000 DATA -0.10000000E+39,-0.157079633E+01
3010 DATA -0.157080133E+01,-0.157079132E+01
3020 DATA -0.99999900E+38,-0.157079633E+01
3030 DATA -0.157080133E+01,-0.157079132E+01
3040 DATA -0.30000000E+31,-0.157079633E+01
3050 DATA -0.157080133E+01,-0.157079132E+01
3060 DATA -0.20000000E+21,-0.157079633E+01
3070 DATA -0.157080133E+01,-0.157079132E+01
3080 DATA -0.90000000E+11,-0.157079633E+01
3090 DATA -0.157080133E+01,-0.157079132E+01
3100 DATA -0.90000000E+06,-0.157079522E+01
3110 DATA -0.157080022E+01,-0.157079021E+01
3120 DATA -0.30000000E+06,-0.157079299E+01
3130 DATA -0.157079800E+01,-0.157078799E+01
3140 DATA -0.20000000E+06,-0.157079133E+01
3150 DATA -0.157079633E+01,-0.157078632E+01
3160 DATA -0.10000000E+06,-0.157078633E+01
3170 DATA -0.157079133E+01,-0.157078132E+01
3180 DATA -0.987654000E+05,-0.157078620E+01
3190 DATA -0.157079121E+01,-0.157078120E+01
3200 DATA -0.987654000E+04,-0.157069508E+01
3210 DATA -0.157070008E+01,-0.157069007E+01
3220 DATA -0.987600000E+03,-0.156978377E+01
3230 DATA -0.156978878E+01,-0.156977877E+01
3240 DATA -0.987000000E+02,-0.156066496E+01
3250 DATA -0.156066998E+01,-0.156065995E+01
3260 DATA -0.980000000E+01,-0.146910748E+01
3270 DATA -0.146911258E+01,-0.146910237E+01
3280 DATA -0.880000000E+01,-0.145764535E+01
3290 DATA -0.145765048E+01,-0.145764021E+01
3300 DATA -0.770000000E+01,-0.144164904E+01
3310 DATA -0.144165421E+01,-0.144164387E+01
3320 DATA -0.660000000E+01,-0.142042490E+01
3330 DATA -0.142043013E+01,-0.142041967E+01
3340 DATA -0.550000000E+01,-0.139094283E+01
3350 DATA -0.139094815E+01,-0.139093750E+01
3360 DATA -0.440000000E+01,-0.134731973E+01
3370 DATA -0.134732522E+01,-0.134731423E+01
```

3380 DATA -0.330000000E+01,-0.127656176E+01  
3390 DATA -0.127656761E+01,-0.127655592E+01  
3400 DATA -0.220000000E+01,-0.114416883E+01  
3410 DATA -0.114417555E+01,-0.114416212E+01  
3420 DATA -0.110000000E+01,-0.832981267E+00  
3430 DATA -0.832986292E+00,-0.832976241E+00  
3440 DATA -0.900000000E+00,-0.732815102E+00  
3450 DATA -0.732816155E+00,-0.732814049E+00  
3460 DATA -0.600000000E+00,-0.540419500E+00  
3470 DATA -0.540420736E+00,-0.540418264E+00  
3480 DATA -0.100000000E+00,-0.996686525E-01  
3490 DATA -0.996696926E-01,-0.996676123E-01  
3500 DATA -0.900000000E-01,-0.897581742E-01  
3510 DATA -0.897583234E-01,-0.897580249E-01  
3520 DATA -0.900000000E-02,-0.899975701E-02  
3530 DATA -0.899977202E-02,-0.899974201E-02  
3540 DATA -0.600000000E-02,-0.599992800E-02  
3550 DATA -0.599994301E-02,-0.599991300E-02  
3560 DATA -0.300000000E-02,-0.299999100E-02  
3570 DATA -0.300000600E-02,-0.299997600E-02  
3580 DATA -0.200000000E-02,-0.199999733E-02  
3590 DATA -0.200001234E-02,-0.199998233E-02  
3600 DATA -0.100000000E-02,-0.999999667E-03  
3610 DATA -0.100001467E-02,-0.999989166E-03  
3620 DATA -0.900000000E-03,-0.899999757E-03  
3630 DATA -0.900001257E-03,-0.899998257E-03  
3640 DATA -0.400000000E-03,-0.399999979E-03  
3650 DATA -0.400001479E-03,-0.399998478E-03  
3660 DATA -0.900000000E-04,-0.899999998E-04  
3670 DATA -0.900001498E-04,-0.899998497E-04  
3680 DATA -0.900000000E-09,-0.900000000E-09  
3690 DATA -0.900001500E-09,-0.899998500E-09  
3700 DATA -0.900000000E-19,-0.900000000E-19  
3710 DATA -0.900001500E-19,-0.899998500E-19  
3720 DATA -0.900000000E-29,-0.900000000E-29  
3730 DATA -0.900001500E-29,-0.899998500E-29  
3740 DATA -0.900000000E-37,-0.900000000E-37  
3750 DATA -0.900001500E-37,-0.899998500E-37  
3760 DATA 0,0,0,0  
3780 DATA 0.900000000E-37, 0.900000000E-37  
3790 DATA 0.899998500E-37, 0.900001500E-37  
3820 DATA 0.900000000E-29, 0.900000000E-29  
3830 DATA 0.899998500E-29, 0.900001500E-29  
3840 DATA 0.900000000E-19, 0.900000000E-19  
3850 DATA 0.899998500E-19, 0.900001500E-19  
3860 DATA 0.900000000E-09, 0.900000000E-09  
3870 DATA 0.899998500E-09, 0.900001500E-09  
3880 DATA 0.900000000E-04, 0.899999998E-04  
3890 DATA 0.899998497E-04, 0.900001498E-04  
3900 DATA 0.400000000E-03, 0.399999979E-03  
3910 DATA 0.399998478E-03, 0.400001479E-03  
3920 DATA 0.900000000E-03, 0.899999757E-03  
3930 DATA 0.899998257E-03, 0.900001257E-03  
3940 DATA 0.100000000E-02, 0.999999667E-03  
3950 DATA 0.999989166E-03, 0.100001467E-02

3960 DATA 0.200000000E-02, 0.199999733E-02  
3970 DATA 0.199998233E-02, 0.200001234E-02  
3980 DATA 0.300000000E-02, 0.299999100E-02  
3990 DATA 0.299997600E-02, 0.300000600E-02  
4000 DATA 0.600000000E-02, 0.599992800E-02  
4010 DATA 0.599991300E-02, 0.599994301E-02  
4020 DATA 0.900000000E-02, 0.899975701E-02  
4030 DATA 0.899974201E-02, 0.899977202E-02  
4040 DATA 0.900000000E-01, 0.897581742E-01  
4050 DATA 0.897580249E-01, 0.897583234E-01  
4060 DATA 0.100000000E+00, 0.996686525E-01  
4070 DATA 0.996676123E-01, 0.996696926E-01  
4080 DATA 0.600000000E+00, 0.540419500E+00  
4090 DATA 0.540418264E+00, 0.540420736E+00  
4100 DATA 0.900000000E+00, 0.732815102E+00  
4110 DATA 0.732814049E+00, 0.732816155E+00  
4120 DATA 0.110000000E+01, 0.832981267E+00  
4130 DATA 0.832976241E+00, 0.832986292E+00  
4140 DATA 0.220000000E+01, 0.114416883E+01  
4150 DATA 0.114416212E+01, 0.114417555E+01  
4160 DATA 0.330000000E+01, 0.127656176E+01  
4170 DATA 0.127655592E+01, 0.127656761E+01  
4180 DATA 0.440000000E+01, 0.134731973E+01  
4190 DATA 0.134731423E+01, 0.134732522E+01  
4200 DATA 0.550000000E+01, 0.139094283E+01  
4210 DATA 0.139093750E+01, 0.139094815E+01  
4220 DATA 0.660000000E+01, 0.142042490E+01  
4230 DATA 0.142041967E+01, 0.142043013E+01  
4240 DATA 0.770000000E+01, 0.144164904E+01  
4250 DATA 0.144164387E+01, 0.144165421E+01  
4260 DATA 0.880000000E+01, 0.145764535E+01  
4270 DATA 0.145764021E+01, 0.145765048E+01  
4280 DATA 0.990000000E+01, 0.147012767E+01  
4290 DATA 0.147012257E+01, 0.147013278E+01  
4300 DATA 0.987000000E+02, 0.156066496E+01  
4310 DATA 0.156065995E+01, 0.156066998E+01  
4320 DATA 0.987600000E+03, 0.156978377E+01  
4330 DATA 0.156977877E+01, 0.156978878E+01  
4340 DATA 0.987654000E+04, 0.157069508E+01  
4350 DATA 0.157069007E+01, 0.157070008E+01  
4360 DATA 0.987654000E+05, 0.157078620E+01  
4370 DATA 0.157078120E+01, 0.157079121E+01  
4380 DATA 0.100000000E+06, 0.157078633E+01  
4390 DATA 0.157078132E+01, 0.157079133E+01  
4400 DATA 0.200000000E+06, 0.157079133E+01  
4410 DATA 0.157078632E+01, 0.157079633E+01  
4420 DATA 0.300000000E+06, 0.157079299E+01  
4430 DATA 0.157078799E+01, 0.157079800E+01  
4440 DATA 0.900000000E+06, 0.157079522E+01  
4450 DATA 0.157079021E+01, 0.157080022E+01  
4460 DATA 0.900000000E+11, 0.157079633E+01  
4470 DATA 0.157079132E+01, 0.157080133E+01  
4480 DATA 0.200000000E+21, 0.157079633E+01  
4490 DATA 0.157079132E+01, 0.157080133E+01  
4500 DATA 0.300000000E+31, 0.157079633E+01

```

4510 DATA 0.157079132E+01, 0.157080133E+01
4520 DATA 0.999999000E+38, 0.157079633E+01
4530 DATA 0.157079132E+01, 0.157080133E+01
4540 DATA 0.100000000E+39, 0.157079633E+01
4550 DATA 0.157079132E+01, 0.157080133E+01
9000 DATA 999,0,0,0
9400 END

```

---

PROGRAM FILE 119: ACCURACY OF ATN FUNCTION.  
ANSI STANDARD 7.6, 8.4

#### SECTION 119.1: ACCURACY OF ATN FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE ATN FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST  $2 * \text{THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE}$ ; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

#### BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME
-1.00000E+38	-1.5708	-1.5708	5.95238E-3	OK
-9.99999E+37	-1.5708	-1.5708	5.95238E-3	OK
-3.00000E+30	-1.5708	-1.5708	5.95238E-3	OK
-2.00000E+20	-1.5708	-1.5708	5.95238E-3	OK
-9.00000E+10	-1.5708	-1.5708	5.95238E-3	OK
-900000	-1.5708	-1.5708	5.96125E-3	OK
-300000	-1.57079	-1.57079	2.36337E-2	OK
-200000	-1.57079	-1.57079	8.94188E-3	OK
-100000	-1.57079	-1.57079	5.96125E-3	OK
-98765.4	-1.57079	-1.57079	2.98063E-3	OK
-9876.54	-1.5707	-1.5707	1.78306E-2	OK
-987.6	-1.56978	-1.56978	1.78571E-2	OK
-98.7	-1.56066	-1.56066	1.19225E-2	OK
-9.8	-1.46911	-1.46911	1.75439E-2	OK
-8.8	-1.45765	-1.45765	1.46413E-2	OK
-7.7	-1.44165	-1.44165	8.59599E-3	OK
-6.6	-1.42042	-1.42042	2.83688E-3	OK
-5.5	-1.39094	-1.39094	8.43882E-3	OK
-4.4	-1.34732	-1.34732	8.13008E-3	OK
-3.3	-1.27656	-1.27656	1.27389E-2	OK
-2.2	-1.14417	-1.14417	6.68151E-3	OK

-1.1	-.832981	-.832981	2.96296E-3	OK
-.9	-.732815	-.732815	7.06714E-3	OK
-.6	-.54042	-.54042	0	OK - EXACT
-.1	-9.96687E-2	-9.96687E-2	8.95255E-4	OK
-.09	-8.97582E-2	-8.97582E-2	0	OK - EXACT
-.009	-8.99976E-3	-8.99976E-3	0	OK - EXACT
-.006	-5.99993E-3	-5.99993E-3	3.88350E-3	OK
-.003	-2.99999E-3	-2.99999E-3	0	OK - EXACT
-.002	-2.00000E-3	-2.00000E-3	0	OK - EXACT
-.001	-1.00000E-3	-1.00000E-3	0	OK - EXACT
-.0009	-9.00000E-4	-9.00000E-4	4.85437E-3	OK
-.0004	-.0004	-.0004	2.42718E-3	OK
-.00009	-9.00000E-5	-9.00000E-5	0	OK - EXACT
-9.00000E-10	-9.00000E-10	-9.00000E-10	0	OK - EXACT
-9.00000E-20	-9.00000E-20	-9.00000E-20	0	OK - EXACT
-9.00000E-30	-9.00000E-30	-9.00000E-30	0	OK - EXACT
-9.00000E-38	-9.00000E-38	-9.00000E-38	0	OK - EXACT
0	0	0	RANGE ZERO	OK - EXACT
9.00000E-38	9.00000E-38	9.00000E-38	0	OK - EXACT
9.00000E-30	9.00000E-30	9.00000E-30	0	OK - EXACT
9.00000E-20	9.00000E-20	9.00000E-20	0	OK - EXACT
9.00000E-10	9.00000E-10	9.00000E-10	0	OK - EXACT
.00009	9.00000E-5	9.00000E-5	0	OK - EXACT
.0004	.0004	.0004	2.42718E-3	OK
.0009	9.00000E-4	9.00000E-4	4.85437E-3	OK
.001	1.00000E-3	1.00000E-3	0	OK - EXACT
.002	2.00000E-3	2.00000E-3	0	OK - EXACT
.003	2.99999E-3	2.99999E-3	0	OK - EXACT
.006	5.99993E-3	5.99993E-3	3.88350E-3	OK
.009	8.99976E-3	8.99976E-3	0	OK - EXACT
.09	8.97582E-2	8.97582E-2	0	OK - EXACT
.1	9.96687E-2	9.96687E-2	8.95255E-4	OK
.6	.54042	.54042	0	OK - EXACT
.9	.732815	.732815	7.06714E-3	OK
1.1	.832981	.832981	2.96296E-3	OK
2.2	1.14417	1.14417	6.68151E-3	OK
3.3	1.27656	1.27656	1.27389E-2	OK
4.4	1.34732	1.34732	8.13008E-3	OK
5.5	1.39094	1.39094	8.43882E-3	OK
6.6	1.42042	1.42042	2.83688E-3	OK
7.7	1.44165	1.44165	8.59599E-3	OK
8.8	1.45765	1.45765	1.46413E-2	OK
9.9	1.47013	1.47013	1.75182E-2	OK
98.7	1.56066	1.56066	1.19225E-2	OK
987.6	1.56978	1.56978	1.78571E-2	OK
9876.54	1.5707	1.5707	1.78306E-2	OK
98765.4	1.57079	1.57079	2.98063E-3	OK
100000	1.57079	1.57079	5.96125E-3	OK
200000	1.57079	1.57079	8.94188E-3	OK
300000	1.57079	1.57079	2.36337E-2	OK
900000	1.5708	1.5708	5.96125E-3	OK
9.00000E+10	1.5708	1.5708	5.95238E-3	OK
2.00000E+20	1.5708	1.5708	5.95238E-3	OK

3.00000E+30	1.5708	1.5708	5.95238E-3	OK
9.99999E+37	1.5708	1.5708	5.95238E-3	OK
1.00000E+38	1.5708	1.5708	5.95238E-3	OK

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 119

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 120: ACCURACY OF COS FUNCTION."
20 PRINT "    ANSI STANDARD 7.6, 8.4"
30 PRINT
40 PRINT "SECTION 120.1: ACCURACY OF COS FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"
70 PRINT "COS FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"
80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"
90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"
100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"
110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"
120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"
130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"
140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"
150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."
160 PRINT
170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."
190 PRINT
200 PRINT "          BEGIN TEST"
210 PRINT
220 PRINT "ARGUMENT","TRUE","COMPUTED","ERROR","OUTCOME"
230 PRINT " ","VALUE","VALUE","MEASURE"
240 PRINT
250 LET F=0
260 READ A,T,L,H
270 IF A=999 THEN 2000
280 LET C=COS(A)
290 IF C=T THEN 333
300 IF C<L THEN 444
310 IF C>H THEN 444
320 LET O$=" OK "
330 GOTO 500
333 LET O$=" OK - EXACT"
340 GOTO 500
444 LET O$="FAIL"
445 LET F=F+1
450 GOTO 500
500 LET R=H-L
510 IF R=0 THEN 700
520 LET E=ABS((C-T)/R)

```

```
525 LET E=E+E
530 PRINT A,T,C,E,O$
540 GOTO 260
700 PRINT A,T,C,"RANGE ZERO",O$
710 GOTO 260
2000 PRINT
2010 IF F=0 THEN 2100
2020 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT " END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 120"
3000 DATA -0.987654000E+05, 0.999948424E+00
3010 DATA 0.993938411E+00, 1.000000000E+00
3020 DATA -0.987644000E+05, 0.548820645E+00
3030 DATA 0.462623568E+00, 0.629534088E+00
3040 DATA -0.942682000E+05,-0.400578628E-04
3050 DATA -0.998733243E-01, 0.997936089E-01
3060 DATA -0.942792000E+04,-0.999999900E+00
3070 DATA -1.000000000E+00,-0.999944935E+00
3080 DATA -0.942592000E+04, 0.415740724E+00
3090 DATA 0.406624756E+00, 0.424815118E+00
3100 DATA -0.942949000E+04,-0.349749758E-03
3110 DATA -0.103496150E-01, 0.965010546E-02
3120 DATA -0.986460000E+03, 0.999999996E+00
3130 DATA 0.999998902E+00, 1.000000000E+00
3140 DATA -0.999000000E+03, 0.999649853E+00
3150 DATA 0.999622392E+00, 0.999676314E+00
3160 DATA -0.994314000E+03, 0.748611695E-04
3170 DATA -0.925139199E-03, 0.107486597E-02
3180 DATA -0.879646000E+02, 1.000000000E+00
3190 DATA 0.999999494E+00, 1.000000000E+00
3200 DATA -0.990000000E+02, 0.398208804E-01
3210 DATA 0.397209095E-01, 0.399208509E-01
3220 DATA -0.895354000E+02,-0.937269090E-05
3230 DATA -0.109373191E-03, 0.906273590E-04
3240 DATA -0.500000000E+02, 0.964966028E+00
3250 DATA 0.964939286E+00, 0.964992762E+00
3260 DATA -0.200000000E+02, 0.408082062E+00
3270 DATA 0.407990265E+00, 0.408173855E+00
3280 DATA -0.590000000E+01, 0.927478431E+00
3290 DATA 0.927474191E+00, 0.927482670E+00
3300 DATA -0.471239000E+01, 0.101961531E-05
3310 DATA -0.898038969E-05, 0.110196654E-04
3320 DATA -0.314159000E+01,-1.000000000E+00
3330 DATA -1.000000000E+00,-0.999999499E+00
3340 DATA -0.230000000E+01,-0.666276021E+00
3350 DATA -0.666283979E+00,-0.666268064E+00
3360 DATA -0.157080000E+01,-0.367320510E-05
3370 DATA -0.136732552E-04, 0.632679990E-05
3380 DATA -0.120000000E+01, 0.362357754E+00
3390 DATA 0.362347934E+00, 0.362367575E+00
3400 DATA -0.100000000E+01, 0.540302306E+00
```

3410 DATA 0.540293391E+00, 0.540311221E+00  
3420 DATA -0.987654000E+00, 0.550649666E+00  
3430 DATA 0.550648331E+00, 0.550651001E+00  
3440 DATA -0.765432000E+00, 0.721083114E+00  
3450 DATA 0.721081921E+00, 0.721084307E+00  
3460 DATA -0.987654000E-01, 0.995126661E+00  
3470 DATA 0.995126151E+00, 0.995127172E+00  
3480 DATA -0.765432000E-01, 0.997071999E+00  
3490 DATA 0.997071491E+00, 0.997072507E+00  
3500 DATA -0.975319000E-02, 0.999952438E+00  
3510 DATA 0.999951937E+00, 0.999952939E+00  
3520 DATA -0.345678000E-02, 0.999994025E+00  
3530 DATA 0.999993525E+00, 0.999994526E+00  
3540 DATA -0.100001000E-02, 0.999999500E+00  
3550 DATA 0.999998999E+00, 1.000000000E+00  
3560 DATA -0.100000000E-02, 0.999999500E+00  
3570 DATA 0.999998999E+00, 1.000000000E+00  
3580 DATA -0.999999000E-03, 0.999999500E+00  
3590 DATA 0.999999000E+00, 1.000000000E+00  
3600 DATA -0.987654000E-04, 0.999999995E+00  
3610 DATA 0.999999495E+00, 1.000000000E+00  
3620 DATA -0.917359000E-05, 1.000000000E+00  
3630 DATA 0.999999499E+00, 1.000000000E+00  
3640 DATA -0.900000000E-09, 0.100000000E+01  
3650 DATA 0.999999500E+00, 0.100000000E+01  
3660 DATA -0.900000000E-19, 0.100000000E+01  
3670 DATA 0.999999500E+00, 0.100000000E+01  
3680 DATA -0.900000000E-29, 0.100000000E+01  
3690 DATA 0.999999500E+00, 0.100000000E+01  
3691 DATA -0.900000000E-37, 0.100000000E+01  
3692 DATA 0.999999500E+00, 0.100000000E+01  
3693 DATA 0,1,1,1  
3695 DATA 1.000000000E-38, 0.100000000E+01  
3697 DATA 0.999995000E+00, 0.100000000E+01  
3700 DATA 0.900000000E-29, 0.100000000E+01  
3710 DATA 0.999999500E+00, 0.100000000E+01  
3720 DATA 0.900000000E-19, 0.100000000E+01  
3730 DATA 0.999999500E+00, 0.100000000E+01  
3740 DATA 0.900000000E-09, 0.100000000E+01  
3750 DATA 0.999999500E+00, 0.100000000E+01  
3760 DATA 0.917359000E-05, 1.000000000E+00  
3770 DATA 0.999999499E+00, 1.000000000E+00  
3780 DATA 0.987654000E-04, 0.999999995E+00  
3790 DATA 0.999999495E+00, 1.000000000E+00  
3800 DATA 0.999999000E-03, 0.999999500E+00  
3810 DATA 0.999999000E+00, 1.000000000E+00  
3820 DATA 0.100000000E-02, 0.999999500E+00  
3830 DATA 0.999998999E+00, 1.000000000E+00  
3840 DATA 0.100001000E-02, 0.999999500E+00  
3850 DATA 0.999998999E+00, 1.000000000E+00  
3860 DATA 0.345678000E-02, 0.999994025E+00  
3870 DATA 0.999993525E+00, 0.999994526E+00  
3880 DATA 0.975319000E-02, 0.999952438E+00  
3890 DATA 0.999951937E+00, 0.999952939E+00  
3900 DATA 0.765432000E-01, 0.997071999E+00

3910 DATA 0.997071491E+00, 0.997072507E+00  
3920 DATA 0.987654000E-01, 0.995126661E+00  
3930 DATA 0.995126151E+00, 0.995127172E+00  
3940 DATA 0.765432000E+00, 0.721083114E+00  
3950 DATA 0.721081921E+00, 0.721084307E+00  
3960 DATA 0.987654000E+00, 0.550649666E+00  
3970 DATA 0.550648331E+00, 0.550651001E+00  
3980 DATA 0.100000000E+01, 0.540302306E+00  
3990 DATA 0.540293391E+00, 0.540311221E+00  
4000 DATA 0.120000000E+01, 0.362357754E+00  
4010 DATA 0.362347934E+00, 0.362367575E+00  
4020 DATA 0.157080000E+01,-0.367320510E-05  
4030 DATA -0.136732552E-04, 0.632679990E-05  
4060 DATA 0.230000000E+01,-0.666276021E+00  
4070 DATA -0.666283979E+00,-0.666268064E+00  
4080 DATA 0.314159000E+01,-1.000000000E+00  
4090 DATA -1.000000000E+00,-0.999999499E+00  
4100 DATA 0.390000000E+01,-0.725932304E+00  
4110 DATA -0.725939682E+00,-0.725924926E+00  
4120 DATA 0.471239000E+01, 0.101961531E-05  
4130 DATA -0.898038969E-05, 0.110196654E-04  
4140 DATA 0.590000000E+01, 0.927478431E+00  
4150 DATA 0.927474191E+00, 0.927482670E+00  
4160 DATA 0.200000000E+02, 0.408082062E+00  
4170 DATA 0.407990265E+00, 0.408173855E+00  
4180 DATA 0.500000000E+02, 0.964966028E+00  
4190 DATA 0.964939286E+00, 0.964992762E+00  
4200 DATA 0.895354000E+02,-0.937269088E-05  
4210 DATA -0.109373191E-03, 0.906273590E-04  
4220 DATA 0.990000000E+02, 0.398208804E-01  
4230 DATA 0.397209095E-01, 0.399208509E-01  
4240 DATA 0.879646000E+02, 1.000000000E+00  
4250 DATA 0.999999494E+00, 1.000000000E+00  
4260 DATA 0.994314000E+03, 0.748611695E-04  
4270 DATA -0.925139199E-03, 0.107486597E-02  
4280 DATA 0.999100000E+03, 0.997297435E+00  
4290 DATA 0.997222966E+00, 0.997370907E+00  
4300 DATA 0.986460000E+03, 0.999999996E+00  
4310 DATA 0.999998902E+00, 1.000000000E+00  
4320 DATA 0.942949000E+04,-0.349749758E-03  
4330 DATA -0.103496150E-01, 0.965010545E-02  
4340 DATA 0.942592000E+04, 0.415740724E+00  
4350 DATA 0.406624756E+00, 0.424815118E+00  
4360 DATA 0.942792000E+04,-0.999999900E+00  
4370 DATA -1.000000000E+00,-0.999944935E+00  
4380 DATA 0.942682000E+05,-0.400578571E-04  
4390 DATA -0.998733243E-01, 0.997936089E-01  
4400 DATA 0.987644000E+05, 0.548820645E+00  
4410 DATA 0.462623568E+00, 0.629534088E+00  
4420 DATA 0.987654000E+05, 0.999948424E+00  
4430 DATA 0.993938411E+00, 1.000000000E+00  
9000 DATA 999,0,0,0  
9400 END

PROGRAM FILE 120: ACCURACY OF COS FUNCTION.  
ANSI STANDARD 7.6, 8.4

SECTION 120.1: ACCURACY OF COS FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE COS FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST 2 \* THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME
-98765.4	.999948	.99995	6.19489E-4	OK
-98764.4	.548821	.548536	3.41634E-3	OK
-94268.2	-4.00579E-5	0	4.01247E-4	OK
-9427.92	-1.	-1.	5.41126E-4	OK
-9425.92	.415741	.415732	9.45742E-4	OK
-9429.49	-3.49750E-4	-3.83495E-4	3.37459E-3	OK
-986.46	1.	1	2.68456E-2	OK
-999	.99965	.99965	2.48722E-3	OK
-994.314	7.48612E-5	7.19053E-5	2.95581E-3	OK
-87.9646	1	1	0	OK - EXACT
-99	3.98209E-2	3.98213E-2	4.06642E-3	OK
-89.5354	-9.37269E-6	-1.04862E-5	1.11350E-2	OK
-50	.964966	.964966	1.11452E-3	OK
-20	.408082	.408082	1.21746E-3	OK
-5.9	.927478	.927478	1.75747E-3	OK
-4.71239	1.01962E-6	9.83081E-7	3.65343E-3	OK
-3.14159	-1	-1.	2.89855E-2	OK
-2.3	-.666276	-.666276	9.36768E-4	OK
-1.5708	-3.67321E-6	-3.68059E-6	7.38169E-4	OK
-1.2	.362358	.362358	7.58869E-4	OK
-1	.540302	.540302	8.35771E-4	OK
-.987654	.55065	.55065	5.57103E-3	OK
-.765432	.721083	.721083	6.26959E-3	OK
-9.87654E-2	.995127	.995127	1.45985E-2	OK
-7.65432E-2	.997072	.997072	1.48148E-2	OK
-9.75319E-3	.999952	.999952	2.98507E-2	OK
-3.45678E-3	.999994	.999994	1.49254E-2	OK
-1.00001E-3	1.	1.	0	OK - EXACT
-.001	1.	1.	1.45985E-2	OK

-9.99999E-4	1.	1.	1.48148E-2	OK
-9.87654E-5	1.	1.	0	OK - EXACT
-9.17359E-6	1	1	0	OK - EXACT
-9.00000E-10	1	1	0	OK - EXACT
-9.00000E-20	1	1	0	OK - EXACT
-9.00000E-30	1	1	0	OK - EXACT
-9.00000E-38	1	1	0	OK - EXACT
0	1	1	RANGE ZERO	OK - EXACT
1.00000E-38	1	1	0	OK - EXACT
9.00000E-30	1	1	0	OK - EXACT
9.00000E-20	1	1	0	OK - EXACT
9.00000E-10	1	1	0	OK - EXACT
9.17359E-6	1	1	0	OK - EXACT
9.87654E-5	1.	1.	2.89855E-2	OK
9.99999E-4	1.	1.	1.48148E-2	OK
.001	1.	1.	1.45985E-2	OK
1.00001E-3	1.	1.	1.45985E-2	OK
3.45678E-3	.999994	.999994	2.98507E-2	OK
9.75319E-3	.999952	.999952	1.49254E-2	OK
7.65432E-2	.997072	.997072	1.48148E-2	OK
9.87654E-2	.995127	.995127	0	OK - EXACT
.765432	.721083	.721083	1.25392E-2	OK
.987654	.55065	.55065	5.57103E-3	OK
1	.540302	.540302	0	OK - EXACT
1.2	.362358	.362358	3.79435E-4	OK
1.5708	-3.67321E-6	-3.69826E-6	2.50517E-3	OK
2.3	-.666276	-.666276	0	OK - EXACT
3.14159	-1	-1	0	OK - EXACT
3.9	-.725932	-.725932	4.03836E-3	OK
4.71239	1.01962E-6	9.36268E-7	8.33475E-3	OK
5.9	.927478	.927478	5.27241E-3	OK
20	.408082	.408082	6.49311E-4	OK
50	.964966	.964966	4.45807E-3	OK
89.5354	-9.37269E-6	-9.73718E-6	3.64491E-3	OK
99	3.98209E-2	3.98205E-2	3.41896E-3	OK
87.9646	1	1	0	OK - EXACT
994.314	7.48612E-5	7.19053E-5	2.95581E-3	OK
999.1	.997297	.997298	9.06527E-4	OK
986.46	1.	1	2.68456E-2	OK
9429.49	-3.49750E-4	-3.83495E-4	3.37459E-3	OK
9425.92	.415741	.415732	9.45742E-4	OK
9427.92	-1.	-1.	5.41126E-4	OK
94268.2	-4.00579E-5	1.53398E-3	1.57666E-2	OK
98764.4	.548821	.547894	1.11027E-2	OK
98765.4	.999948	.999958	3.04336E-3	OK

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 120

```
*****
*****  
10 PRINT "PROGRAM FILE 121: ACCURACY OF EXP FUNCTION."  
20 PRINT "      ANSI STANDARD 7.6, 8.4"  
30 PRINT  
40 PRINT "SECTION 121.1: ACCURACY OF EXP FUNCTION."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"  
70 PRINT "EXP FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"  
80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"  
90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"  
100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"  
110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"  
120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"  
130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"  
140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"  
150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."  
160 PRINT  
170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"  
180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."  
190 PRINT  
200 PRINT "          BEGIN TEST"  
210 PRINT  
220 PRINT "ARGUMENT","TRUE","COMPUTED","ERROR","OUTCOME"  
230 PRINT " ","VALUE","VALUE","MEASURE"  
240 PRINT  
250 LET F=0  
260 READ A,T,L,H  
270 IF A=999 THEN 2000  
280 LET C=EXP(A)  
290 IF C=T THEN 333  
300 IF C<L THEN 444  
310 IF C>H THEN 444  
320 LET O$=" OK "  
330 GOTO 500  
333 LET O$=" OK - EXACT"  
340 GOTO 500  
444 LET O$="FAIL"  
445 LET F=F+1  
450 GOTO 500  
500 LET R=H-L  
510 IF R=0 THEN 700  
520 LET E=ABS((C-T)/R)  
525 LET E=E+E  
530 PRINT A,T,C,E,O$  
540 GOTO 260  
700 PRINT A,T,C,"RANGE ZERO",O$  
710 GOTO 260  
2000 PRINT
```

```
2010 IF F=0 THEN 2100
2020 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT " END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 121"
3000 DATA -0.874981000E+02, 0.100013354E-37
3010 DATA 0.100002853E-37, 0.100023857E-37
3020 DATA -0.859876000E+02, 0.452959954E-37
3030 DATA 0.452914160E-37, 0.453005753E-37
3040 DATA -0.750000000E+02, 0.267863696E-32
3050 DATA 0.267836411E-32, 0.267890984E-32
3060 DATA -0.444444000E+02, 0.498933114E-19
3070 DATA 0.498882722E-19, 0.498983510E-19
3080 DATA -0.234567000E+02, 0.649957001E-10
3090 DATA 0.649891508E-10, 0.650022501E-10
3100 DATA -0.777777000E+01, 0.418945382E-03
3110 DATA 0.418940692E-03, 0.418950072E-03
3120 DATA -0.222222000E+01, 0.108368264E+00
3130 DATA 0.108366680E+00, 0.108369848E+00
3140 DATA -0.100000000E+01, 0.367879441E+00
3150 DATA 0.367875262E+00, 0.367883620E+00
3160 DATA -0.999999000E+00, 0.367879809E+00
3170 DATA 0.367878941E+00, 0.367880677E+00
3180 DATA -0.456789000E+00, 0.633313955E+00
3190 DATA 0.633312821E+00, 0.633315089E+00
3200 DATA -0.987654000E-01, 0.905955220E+00
3210 DATA 0.905954629E+00, 0.905955811E+00
3220 DATA -0.987654000E-02, 0.990172073E+00
3230 DATA 0.990171562E+00, 0.990172583E+00
3240 DATA -0.987654000E-03, 0.999012834E+00
3250 DATA 0.999012332E+00, 0.999013335E+00
3260 DATA -0.987654000E-04, 0.999901239E+00
3270 DATA 0.999900739E+00, 0.999901740E+00
3280 DATA -0.987654000E-05, 0.999990124E+00
3290 DATA 0.999989623E+00, 0.999990624E+00
3300 DATA -0.987654000E-06, 0.999999012E+00
3310 DATA 0.999998512E+00, 0.999999513E+00
3320 DATA -0.987654000E-07, 0.999999901E+00
3330 DATA 0.999999401E+00, 1.000000402E+00
3340 DATA -0.987654000E-08, 0.999999990E+00
3350 DATA 0.999999490E+00, 1.000000491E+00
3360 DATA -0.987654000E-09, 0.999999999E+00
3370 DATA 0.999999499E+00, 1.000000500E+00
3380 DATA -0.987654000E-14, 1.000000000E+00
3390 DATA 0.999999499E+00, 1.000000500E+00
3400 DATA -0.987654000E-19, 1.000000000E+00
3410 DATA 0.999999499E+00, 1.000000500E+00
3420 DATA -0.987654000E-29, 1.000000000E+00
3430 DATA 0.999999499E+00, 1.000000500E+00
3442 DATA -0.987654000E-37, 1.000000000E+00
3444 DATA 0.999999499E+00, 1.000000500E+00
3445 DATA 0,1,1,1
```

```
3446 DATA 0.987654000E-37, 0.100000000E+01
3448 DATA .0999995000E+01, 0.100000500E+01
3449 DATA 0.987654000E-29, 0.100000000E+01
3450 DATA .0999995000E+01, 0.100000500E+01
3460 DATA 0.987654000E-19, 0.100000000E+01
3470 DATA .0999995000E+01, 0.100000500E+01
3480 DATA 0.987654000E-14, 0.100000000E+01
3490 DATA .0999995000E+01, 0.100000501E+01
3500 DATA 0.987654000E-09, 0.100000000E+01
3510 DATA .0999995000E+01, 0.100000501E+01
3520 DATA 0.987654000E-08, 0.100000001E+01
3530 DATA .0999995009E+01, 0.100000501E+01
3540 DATA 0.987654000E-07, 0.100000010E+01
3550 DATA .0999995098E+01, 0.100000510E+01
3560 DATA 0.987654000E-06, 0.100000099E+01
3570 DATA .0999995987E+01, 0.100000599E+01
3580 DATA 0.987654000E-05, 0.100000988E+01
3590 DATA 0.100000487E+01, 0.100001488E+01
3600 DATA 0.987654000E-04, 0.100009877E+01
3610 DATA 0.100009377E+01, 0.100010378E+01
3620 DATA 0.987654000E-03, 0.100098814E+01
3630 DATA 0.100098314E+01, 0.100099315E+01
3640 DATA 0.987654000E-02, 0.100992547E+01
3650 DATA 0.100992046E+01, 0.100993049E+01
3660 DATA 0.987654000E-01, 0.110380732E+01
3670 DATA 0.110380220E+01, 0.110381243E+01
3680 DATA 0.456789000E+00, 0.157899568E+01
3690 DATA 0.157898910E+01, 0.157900227E+01
3700 DATA 0.999999000E+00, 0.271827911E+01
3710 DATA 0.271827139E+01, 0.271828683E+01
3720 DATA 0.100000000E+01, 0.271828183E+01
3730 DATA 0.271824964E+01, 0.271831402E+01
3740 DATA 0.222222000E+01, 0.922779385E+01
3750 DATA 0.922769656E+01, 0.922789113E+01
3760 DATA 0.777777000E+01, 0.238694599E+04
3770 DATA 0.238691712E+04, 0.238697487E+04
3780 DATA 0.234567000E+02, 0.153856332E+11
3790 DATA 0.153840446E+11, 0.153872219E+11
3800 DATA 0.444444000E+02, 0.200427667E+20
3810 DATA 0.200407125E+20, 0.200448211E+20
3820 DATA 0.750000000E+02, 0.373324200E+33
3830 DATA 0.373286369E+33, 0.373362034E+33
3840 DATA 0.859876000E+02, 0.220770068E+38
3850 DATA 0.220747492E+38, 0.220792647E+38
3860 DATA 0.874981000E+02, 0.999866475E+38
3870 DATA 0.999765993E+38, 0.999966967E+38
9000 DATA 999,0,0,0
9400 END
```

## SECTION 121.1: ACCURACY OF EXP FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE EXP FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST 2 \* THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME
-87.4981	1.00013E-38	1.00013E-38	0	OK - EXACT
-85.9876	4.52960E-38	4.52960E-38	0	OK - EXACT
-75	2.67864E-33	2.67864E-33	0	OK
-44.4444	4.98933E-20	4.98933E-20	2.88542E-3	OK
-23.4567	6.49957E-11	6.49957E-11	7.94544E-4	OK
-7.77777	4.18945E-4	4.18945E-4	1.31936E-2	OK
-2.22222	.108368	.108368	4.71559E-3	OK
-1	.367879	.367879	0	OK - EXACT
-.999999	.36788	.36788	.012931	OK
-.456789	.633314	.633314	0	OK - EXACT
-.87654E-2	.905955	.905955	0	OK - EXACT
-.87654E-3	.990172	.990172	0	OK - EXACT
-.87654E-4	.999013	.999013	0	OK - EXACT
-.87654E-5	.999901	.999901	2.96296E-2	OK
-.87654E-6	.99999	.99999	1.49254E-2	OK
-.87654E-7	.999999	.999999	1.49254E-2	OK
-.87654E-8	1.	1.	0	OK - EXACT
-.87654E-9	1.	1.	0	OK - EXACT
-.87654E-10	1.	1.	1.48148E-2	OK
-.87654E-15	1	1.	1.48148E-2	OK
-.87654E-20	1	1.	1.48148E-2	OK
-.87654E-30	1	1.	1.48148E-2	OK
-.87654E-38	1	1.	1.48148E-2	OK
0	1	1	RANGE ZERO	OK - EXACT
9.87654E-38	1	1	0	OK - EXACT
9.87654E-30	1	1	0	OK - EXACT
9.87654E-20	1	1	0	OK - EXACT
9.87654E-15	1	1	0	OK - EXACT
9.87654E-10	1	1	0	OK - EXACT
9.87654E-9	1	1	0	OK - EXACT
9.87654E-8	1.	1.	0	OK - EXACT
9.87654E-7	1.	1.	1.80045E-2	OK
9.87654E-6	1.00001	1.00001	1.49031E-2	OK
9.87654E-5	1.0001	1.0001	1.49254E-2	OK

9.87654E-4	1.00099	1.00099	5.96125E-3	OK
9.87654E-3	1.00993	1.00993	1.48810E-2	OK
9.87654E-2	1.10381	1.10381	2.91971E-3	OK
.456789	1.579	1.579	1.13766E-2	OK
.999999	2.71828	2.71828	0	OK - EXACT
1	2.71828	2.71828	1.85099E-3	OK
2.22222	9.22779	9.22779	1.22549E-3	OK
7.77777	2386.95	2386.95	8.46561E-3	OK
23.4567	1.53856E+10	1.53856E+10	8.05802E-4	OK
44.4444	2.00428E+19	2.00428E+19	3.21113E-3	OK
75	3.73324E+32	3.73324E+32	2.55607E-4	OK
85.9876	2.20770E+37	2.20770E+37	3.36937E-3	OK
87.4981	9.99866E+37	9.99867E+37	2.64917E-3	OK

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 121

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 122: EXCEPTION - OVERFLOW ON VALUE OF"
15 PRINT "      EXP FUNCTION."
20 PRINT "      ANSI STANDARD 8.5"
30 PRINT
40 PRINT "SECTION 122.1: EXCEPTION - OVERFLOW ON VALUE OF"
50 PRINT "      EXP FUNCTION."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES AN EXCEPTION WHEN THE VALUE RETURNED"
90 PRINT "BY THE EXP FUNCTION EXCEEDS MACHINE INFINITY."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST SUPPLY MACHINE INFINITY AND"
170 PRINT "          CONTINUE EXECUTION."
180 PRINT
190 PRINT "                  BEGIN TEST."
200 PRINT
205 PRINT
210 LET F=1.1
220 LET A=87
230 LET B=1
240 PRINT "ABOUT TO ATTEMPT EXP(";A;")"
250 LET C=EXP(A)
260 PRINT "VALUE RETURNED BY EXP = ";C
270 PRINT
280 PRINT "-----"
290 PRINT

```

```
300 IF B=C THEN 350
310 LET B=C
320 LET F=F*1.1
330 LET A=A*F
340 GOTO 240
350 PRINT "LAST TWO INVOCATIONS OF EXP SHOULD HAVE BEEN REPORTED"
360 PRINT "AS OVERFLOW, AND MACHINE INFINITY SUPPLIED; IF SO,"
370 PRINT "*** TEST PASSED *** OTHERWISE *** TEST FAILED ***"
380 PRINT
390 PRINT " END TEST."
400 PRINT
410 PRINT "END PROGRAM 122"
420 END
```

---

PROGRAM FILE 122: EXCEPTION - OVERFLOW ON VALUE OF  
EXP FUNCTION.  
ANSI STANDARD 8.5

SECTION 122.1: EXCEPTION - OVERFLOW ON VALUE OF  
EXP FUNCTION.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES AN EXCEPTION WHEN THE VALUE RETURNED  
BY THE EXP FUNCTION EXCEEDS MACHINE INFINITY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST SUPPLY MACHINE INFINITY AND  
CONTINUE EXECUTION.

BEGIN TEST.

---

ABOUT TO ATTEMPT EXP( 87 )
VALUE RETURNED BY EXP = 6.07603E+37

---

ABOUT TO ATTEMPT EXP( 105.27 )

% OVERFLOW IN EXP IN LINE 250
VALUE RETURNED BY EXP = 1.70141E+38

---

ABOUT TO ATTEMPT EXP( 140.114 )

% OVERFLOW IN EXP IN LINE 250
VALUE RETURNED BY EXP = 1.70141E+38

LAST TWO INVOCATIONS OF EXP SHOULD HAVE BEEN REPORTED AS OVERFLOW, AND MACHINE INFINITY SUPPLIED; IF SO,  
 \*\*\* TEST PASSED \*\*\* OTHERWISE \*\*\* TEST FAILED \*\*\*

END TEST.

END PROGRAM 122

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

```

10 PRINT "PROGRAM FILE 123: EXCEPTION - UNDERFLOW ON VALUE OF"
20 PRINT "          EXP FUNCTION."
30 PRINT "      ANSI STANDARD 8.4, 8.6"
40 PRINT
50 PRINT "SECTION 123.1: EXCEPTION - UNDERFLOW ON VALUE OF"
60 PRINT "          EXP FUNCTION."
70 PRINT
80 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
90 PRINT "RECOGNIZES AN EXCEPTION WHEN THE VALUE RETURNED"
100 PRINT "BY THE EXP FUNCTION IS LESS THAN MACHINE INFINITESIMAL"
110 PRINT "AND TO ENSURE THAT ZERO IS SUPPLIED ON UNDERFLOW."
120 PRINT
130 PRINT "TO PASS THIS TEST:"
140 PRINT
150 PRINT "      1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE"
160 PRINT "          DISPLAYED (NOT MANDATORY), AND"
170 PRINT
180 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"
190 PRINT "          OF THE FUNCTION AND EXECUTION CONTINUES."
220 PRINT
230 PRINT
240 PRINT "                      BEGIN TEST."
250 PRINT
260 PRINT
270 LET F=1.1
280 LET A=-87
290 PRINT "ABOUT TO ATTEMPT EXP(";A;")"
300 LET C=EXP(A)
310 PRINT "VALUE RETURNED BY EXP = ";C
320 PRINT
330 PRINT "-----"
340 PRINT
350 IF 0=C THEN 390
360 LET F=F*1.1
370 LET A=A*F
380 GOTO 290
390 PRINT "LAST INVOCATION OF EXP SHOULD HAVE BEEN REPORTED AS"
400 PRINT "UNDERFLOW IN ORDER TO COMPLY WITH ANSI RECOMMENDATION;"
410 PRINT
420 PRINT "***** TEST PASSES, SINCE ZERO WAS SUPPLIED. *****"
430 PRINT

```

```
440 PRINT "
450 PRINT
460 PRINT "END PROGRAM 123"
470 END
```

---

PROGRAM FILE 123: EXCEPTION - UNDERFLOW ON VALUE OF  
EXP FUNCTION.  
ANSI STANDARD 8.4, 8.6

SECTION 123.1: EXCEPTION - UNDERFLOW ON VALUE OF  
EXP FUNCTION.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES AN EXCEPTION WHEN THE VALUE RETURNED  
BY THE EXP FUNCTION IS LESS THAN MACHINE INFINITESIMAL  
AND TO ENSURE THAT ZERO IS SUPPLIED ON UNDERFLOW.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE  
DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE  
OF THE FUNCTION AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO ATTEMPT EXP(-87 )
VALUE RETURNED BY EXP = 1.64581E-38

---

ABOUT TO ATTEMPT EXP(-105.27 )

% UNDERFLOW IN EXP IN LINE 300
VALUE RETURNED BY EXP = 0

---

LAST INVOCATION OF EXP SHOULD HAVE BEEN REPORTED AS  
UNDERFLOW IN ORDER TO COMPLY WITH ANSI RECOMMENDATION;

\*\*\*\*\* TEST PASSES, SINCE ZERO WAS SUPPLIED. \*\*\*\*\*

END TEST

END PROGRAM 123

```
*****
*****  
10 PRINT "PROGRAM FILE 124: ACCURACY OF LOG FUNCTION."  
20 PRINT "    ANSI STANDARD 7.6, 8.4"  
30 PRINT  
40 PRINT "SECTION 124.1: ACCURACY OF LOG FUNCTION."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"  
70 PRINT "LOG FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"  
80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"  
90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"  
100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"  
110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"  
120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"  
130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"  
140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"  
150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."  
160 PRINT  
170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"  
180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."  
190 PRINT  
200 PRINT "          BEGIN TEST"  
210 PRINT  
220 PRINT "ARGUMENT", "TRUE", "COMPUTED", "ERROR", "OUTCOME"  
230 PRINT " ", "VALUE", "MEASURE"  
240 PRINT  
250 LET F=0  
260 READ A,T,L,H  
270 IF A=999 THEN 2000  
280 LET C=LOG(A)  
290 IF C=T THEN 333  
300 IF C<L THEN 444  
310 IF C>H THEN 444  
320 LET O$=" OK "  
330 GOTO 500  
333 LET O$=" OK - EXACT"  
340 GOTO 500  
444 LET O$="FAIL"  
445 LET F=F+1  
450 GOTO 500  
500 LET R=H-L  
510 IF R=0 THEN 700  
520 LET E=ABS((C-T)/R)  
525 LET E=E+E  
530 PRINT A,T,C,E,O$  
540 GOTO 260  
700 PRINT A,T,C,"RANGE ZERO",O$  
710 GOTO 260  
2000 PRINT
```

```
2010 IF F=0 THEN 2100
2020 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT " END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 124"
3000 DATA 0.10000000E-37,-0.874982335E+02
3010 DATA -0.874982936E+02,-0.874981735E+02
3020 DATA 0.100001000E-37,-0.874982235E+02
3030 DATA -0.874982836E+02,-0.874981635E+02
3040 DATA 0.987654000E-37,-0.852080713E+02
3050 DATA -0.852081223E+02,-0.852080202E+02
3060 DATA 0.898989000E-36,-0.829995478E+02
3070 DATA -0.829995990E+02,-0.829994967E+02
3080 DATA 0.787878000E-35,-0.808288903E+02
3090 DATA -0.808289416E+02,-0.808288390E+02
3100 DATA 0.676767000E-34,-0.786783214E+02
3110 DATA -0.786783729E+02,-0.786782699E+02
3120 DATA 0.565656000E-29,-0.673447369E+02
3130 DATA -0.673447887E+02,-0.673446850E+02
3140 DATA 0.454545000E-24,-0.560505006E+02
3150 DATA -0.560505528E+02,-0.560504483E+02
3160 DATA 0.343434000E-19,-0.448178771E+02
3170 DATA -0.448179301E+02,-0.448178241E+02
3180 DATA 0.232323000E-14,-0.336958179E+02
3190 DATA -0.336958723E+02,-0.336957636E+02
3200 DATA 0.121212000E-09,-0.228334800E+02
3210 DATA -0.228335383E+02,-0.228334217E+02
3220 DATA 0.909090000E-06,-0.139108217E+02
3230 DATA -0.139108729E+02,-0.139107706E+02
3240 DATA 0.888888000E-04,-0.932812441E+01
3250 DATA -0.932813054E+01,-0.932811828E+01
3260 DATA 0.777777000E-03,-0.715907071E+01
3270 DATA -0.715907700E+01,-0.715906442E+01
3280 DATA 0.666666000E-02,-0.501063629E+01
3290 DATA -0.501064280E+01,-0.501062979E+01
3300 DATA 0.555555000E-01,-0.289037276E+01
3310 DATA -0.289037956E+01,-0.289036595E+01
3320 DATA 0.123456000E+00,-0.209187046E+01
3330 DATA -0.209188357E+01,-0.209185736E+01
3340 DATA 0.298347000E+00,-0.120949804E+01
3350 DATA -0.120950640E+01,-0.120948968E+01
3360 DATA 0.535353000E+00,-0.624828937E+00
3370 DATA -0.624831305E+00,-0.624826568E+00
3380 DATA 0.893481000E+00,-0.112630209E+00
3390 DATA -0.112631829E+00,-0.112628590E+00
3400 DATA 0.999000000E+00,-0.100050033E-02
3410 DATA -0.100150634E-02,-0.999498833E-03
3420 DATA 0.999500000E+00,-0.500125042E-03
3430 DATA -0.501126043E-03,-0.499124041E-03
3440 DATA 0.999900000E+00,-0.100005000E-03
3450 DATA -0.101005601E-03,-0.990048508E-04
3460 DATA 0.999950000E+00,-0.500012500E-04
```

3470 DATA -0.510013506E-04,-0.490011505E-04  
3480 DATA 0.999990000E+00,-0.100000500E-04  
3490 DATA -0.110001106E-04,-0.900003550E-05  
3500 DATA 0.999995000E+00,-0.500001250E-05  
3510 DATA -0.600002300E-05,-0.400000299E-05  
3520 DATA 0.999998000E+00,-0.200000200E-05  
3530 DATA -0.300000950E-05,-0.0999995499E-05  
3540 DATA 0.999999000E+00,-0.100000050E-05  
3550 DATA -0.200000700E-05, 0.000000500E+00  
3560 DATA 0.100000000E+01, 0.000000000E+00  
3570 DATA -0.100001001E-04, 0.999995501E-05  
3580 DATA 0.100001000E+01, 0.999995000E-05  
3590 DATA 0.000000000E+00, 0.199998501E-04  
3600 DATA 0.100002000E+01, 0.199998000E-04  
3610 DATA 0.999994500E-05, 0.299996001E-04  
3620 DATA 0.100005000E+01, 0.499987500E-04  
3630 DATA 0.399991500E-04, 0.599982501E-04  
3640 DATA 0.100010000E+01, 0.999950003E-04  
3650 DATA 0.899959002E-04, 0.109994451E-03  
3660 DATA 0.100030000E+01, 0.299955009E-03  
3670 DATA 0.289957458E-03, 0.309952460E-03  
3680 DATA 0.100100000E+01, 0.999500333E-03  
3690 DATA 0.989509773E-03, 0.100949530E-02  
3700 DATA 0.100300000E+01, 0.299550898E-02  
3710 DATA 0.298553384E-02, 0.300548402E-02  
3720 DATA 0.101000000E+01, 0.995033085E-02  
3730 DATA 0.994042481E-02, 0.996023680E-02  
3740 DATA 0.103000000E+01, 0.295588022E-01  
3750 DATA 0.295490434E-01, 0.295685610E-01  
3760 DATA 0.111111000E+01, 0.105359516E+00  
3770 DATA 0.105350015E+00, 0.105369016E+00  
3780 DATA 0.154321000E+01, 0.433864663E+00  
3790 DATA 0.433857682E+00, 0.433871643E+00  
3800 DATA 0.200000000E+01, 0.693147181E+00  
3810 DATA 0.693141680E+00, 0.693152681E+00  
3820 DATA 0.232323000E+01, 0.842958459E+00  
3830 DATA 0.842953654E+00, 0.842963264E+00  
3840 DATA 0.271828000E+01, 0.999999327E+00  
3850 DATA 0.999995148E+00, 0.100000801E+01  
3860 DATA 0.400000000E+01, 0.138629436E+01  
3870 DATA 0.138628686E+01, 0.138630187E+01  
3880 DATA 0.714286000E+01, 0.196611326E+01  
3890 DATA 0.196610685E+01, 0.196611966E+01  
3900 DATA 0.800000000E+01, 0.207944154E+01  
3910 DATA 0.207943529E+01, 0.207944780E+01  
3920 DATA 0.536789000E+02, 0.398302000E+01  
3930 DATA 0.398301313E+01, 0.398302687E+01  
3940 DATA 0.482284000E+03, 0.617853315E+01  
3950 DATA 0.617852607E+01, 0.617854023E+01  
3960 DATA 0.962112000E+04, 0.917171596E+01  
3970 DATA 0.917170992E+01, 0.917172201E+01  
3980 DATA 0.863214000E+05, 0.113658328E+02  
3990 DATA 0.113657816E+02, 0.113658840E+02  
4000 DATA 0.777777000E+06, 0.135641951E+02  
4010 DATA 0.135641438E+02, 0.135642465E+02

```

4020 DATA 0.987654000E+08, 0.184082579E+02
4030 DATA 0.184082068E+02, 0.184083090E+02
4040 DATA 0.898989000E+11, 0.252219515E+02
4050 DATA 0.252219004E+02, 0.252220027E+02
4060 DATA 0.797979000E+16, 0.366156885E+02
4070 DATA 0.366156372E+02, 0.366157398E+02
4080 DATA 0.696969000E+21, 0.479932726E+02
4090 DATA 0.479932211E+02, 0.479933241E+02
4100 DATA 0.595959000E+26, 0.593496290E+02
4110 DATA 0.593495773E+02, 0.593496807E+02
4120 DATA 0.494949000E+31, 0.706768373E+02
4130 DATA 0.706767853E+02, 0.706768894E+02
4140 DATA 0.393939000E+36, 0.819615041E+02
4150 DATA 0.819614516E+02, 0.819615567E+02
4160 DATA 0.987654000E+37, 0.851832256E+02
4170 DATA 0.851831745E+02, 0.851832767E+02
4180 DATA 0.987654000E+38, 0.874858107E+02
4190 DATA 0.874857596E+02, 0.874858618E+02
4200 DATA 0.100000000E+39, 0.874982335E+02
4210 DATA 0.874981735E+02, 0.874982936E+02
9000 DATA 999,0,0,0
9400 END

```

---

PROGRAM FILE 124: ACCURACY OF LOG FUNCTION.

ANSI STANDARD 7.6, 8.4

#### SECTION 124.1: ACCURACY OF LOG FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE LOG FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST 2 \* THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME
----------	---------------	-------------------	------------------	---------

1.00000E-38	-87.4982	-87.4982	3.17460E-2	OK
1.00001E-38	-87.4982	-87.4982	0	OK - EXACT
9.87654E-38	-85.2081	-85.2081	0	OK - EXACT
8.98989E-37	-82.9995	-82.9995	0	OK - EXACT
7.87878E-36	-80.8289	-80.8289	0	OK - EXACT
6.76767E-35	-78.6783	-78.6783	1.86916E-2	OK
5.65656E-30	-67.3447	-67.3447	3.70370E-2	OK
4.54545E-25	-56.0505	-56.0505	2.75229E-2	OK
3.43434E-20	-44.8179	-44.8179	1.80180E-2	OK
2.32323E-15	-33.6958	-33.6958	2.63158E-2	OK
1.21212E-10	-22.8335	-22.8335	8.16327E-3	OK
9.09090E-7	-13.9108	-13.9108	1.86698E-2	OK
8.88888E-5	-9.32812	-9.32812	1.92308E-2	OK
7.77777E-4	-7.15907	-7.15907	.028436	OK
6.66666E-3	-5.01064	-5.01064	3.63636E-2	OK
5.55555E-2	-2.89037	-2.89037	2.19298E-2	OK
.123456	-2.09187	-2.09187	6.82594E-3	OK
.298347	-1.2095	-1.2095	5.32387E-3	OK
.535353	-.624829	-.624829	1.25589E-2	OK
.893481	-.11263	-.11263	2.19146E-2	OK
.999	-1.00050E-3	-1.00052E-3	1.73682E-2	OK
.9995	-5.00125E-4	-5.00152E-4	2.66397E-2	OK
.9999	-1.00005E-4	-1.00036E-4	3.10112E-2	OK
.99995	-5.00013E-5	-5.00425E-5	4.12878E-2	OK
.99999	-1.00001E-5	-1.00317E-5	3.16968E-2	OK
.999995	-5.00001E-6	-5.02749E-6	.027481	OK
.999998	-2.00000E-6	-2.03217E-6	3.21691E-2	OK
.999999	-1.00000E-6	-1.03287E-6	2.62954E-2	OK
1	0	0	0	OK - EXACT
1.00001	9.99995E-6	1.00033E-5	3.39392E-4	OK
1.00002	1.99998E-5	2.00144E-5	1.46345E-3	OK
1.00005	4.99987E-5	5.00116E-5	1.28058E-3	OK
1.0001	9.99950E-5	1.00013E-4	1.77837E-3	OK
1.0003	2.99955E-4	2.99976E-4	2.13785E-3	OK
1.001	9.99500E-4	9.99516E-4	1.55236E-3	OK
1.003	2.99551E-3	2.99552E-3	1.49384E-3	OK
1.01	9.95033E-3	9.95033E-3	9.40159E-5	OK
1.03	2.95588E-2	2.95588E-2	2.14725E-3	OK
1.11111	.10536	.10536	5.88206E-4	OK
1.54321	.433865	.433865	1.06724E-3	OK
2	.693147	.693147	2.70636E-3	OK
2.32323	.842958	.842958	3.10078E-3	OK
2.71828	.999999	.999999	3.47222E-3	OK
4	1.38629	1.38629	9.93049E-3	OK
7.14286	1.96611	1.96611	2.32829E-3	OK
8	2.07944	2.07944	9.47867E-3	OK
53.6789	3.98302	3.98302	4.36681E-3	OK
482.284	6.17853	6.17853	8.40336E-3	OK
9621.12	9.17172	9.17172	1.94175E-2	OK
86321.4	11.3658	11.3658	1.38889E-2	OK
777777	13.5642	13.5642	2.31481E-3	OK
98765400	18.4083	18.4083	1.38889E-2	OK
8.98989E+10	25.222	25.222	4.68384E-3	OK
7.97979E+15	36.6157	36.6157	9.34579E-3	OK
6.96969E+20	47.9933	47.9933	0	OK - EXACT

5.95959E+25	59.3496	59.3496	9.30233E-3	OK
4.94949E+30	70.6768	70.6768	0	OK - EXACT
3.93939E+35	81.9615	81.9615	1.81818E-2	OK
9.87654E+36	85.1832	85.1832	0	OK - EXACT
9.87654E+37	87.4858	87.4858	1.86916E-2	OK
1.00000E+38	87.4982	87.4982	0	OK - EXACT

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 124

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 125: EXCEPTION - LOG OF ZERO ARGUMENT."
20 PRINT "      ANSI STANDARD 8.5"
30 PRINT
40 PRINT "SECTION 125.1: EXCEPTION - LOG OF ZERO ARGUMENT."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES A FATAL EXCEPTION WHEN A ZERO ARGUMENT"
90 PRINT "IS PASSED TO THE LOG FUNCTION."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
140 PRINT "          DISPLAYED, AND"
150 PRINT
160 PRINT "      2) THE PROGRAM MUST TERMINATE UPON USE OF THE ZERO"
170 PRINT "          ARGUMENT."
180 PRINT
190 PRINT "                                BEGIN TEST."
200 PRINT
210 PRINT "FATAL EXCEPTION SHOULD OCCUR NOW:"
220 PRINT
230 LET A=(0)
240 LET B=LOG(A)
255 PRINT "*** TEST FAILED ***"
257 PRINT
260 PRINT "ARGUMENT WAS SET TO 0; IT NOW EQUALS ";A
270 PRINT "RESULT OF LOG IS ";B
280 PRINT
330 PRINT
340 PRINT "                                END TEST"
350 PRINT
360 PRINT "END PROGRAM 125"
370 END

```

PROGRAM FILE 125: EXCEPTION - LOG OF ZERO ARGUMENT.  
ANSI STANDARD 8.5

## SECTION 125.1: EXCEPTION - LOG OF ZERO ARGUMENT.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A ZERO ARGUMENT  
IS PASSED TO THE LOG FUNCTION.

**TO PASS THIS TEST:**

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
  - 2) THE PROGRAM MUST TERMINATE UPON USE OF THE ZERO ARGUMENT.

BEGIN TEST

FATAL EXCEPTION SHOULD OCCUR NOW:

? LOG OF ZERO IN LINE 240

```

260 PRINT "ARGUMENT WAS SET TO -3; IT NOW EQUALS ";A
270 PRINT "RESULT OF LOG IS ";B
280 PRINT
330 PRINT
340 PRINT "                                     END TEST"
350 PRINT
360 PRINT "END PROGRAM 126"
370 END

```

---

PROGRAM FILE 126: EXCEPTION - LOG OF NEGATIVE ARGUMENT.  
ANSI STANDARD 8.5

#### SECTION 126.1: EXCEPTION - LOG OF NEGATIVE ARGUMENT.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES A FATAL EXCEPTION WHEN A NEGATIVE ARGUMENT  
IS PASSED TO THE LOG FUNCTION.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) THE PROGRAM MUST TERMINATE UPON USE OF THE NEGATIVE  
ARGUMENT.

BEGIN TEST.

FATAL EXCEPTION SHOULD OCCUR NOW:

? LOG OF NEGATIVE NUMBER IN LINE 240

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 127: ACCURACY OF SIN FUNCTION."
20 PRINT "      ANSI STANDARD 7.6, 8.4"
30 PRINT
40 PRINT "SECTION 127.1: ACCURACY OF SIN FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"
70 PRINT "SIN FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"
80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"
90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"
100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"
110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"
120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"
130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"
140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"
150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."
160 PRINT

```

```
170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."
190 PRINT
200 PRINT "          BEGIN TEST"
210 PRINT
220 PRINT "ARGUMENT","TRUE","COMPUTED","ERROR","OUTCOME"
230 PRINT " ","VALUE","VALUE","MEASURE"
240 PRINT
250 LET F=0
260 READ A,T,L,H
270 IF A=999 THEN 2000
280 LET C=SIN(A)
290 IF C=T THEN 333
300 IF C<L THEN 444
310 IF C>H THEN 444
320 LET O$=" OK "
330 GOTO 500
333 LET O$=" OK - EXACT"
340 GOTO 500
444 LET O$="FAIL"
445 LET F=F+1
450 GOTO 500
500 LET R=H-L
510 IF R=0 THEN 700
520 LET E=ABS((C-T)/R)
525 LET E=E+E
530 PRINT A,T,C,E,O$
540 GOTO 260
700 PRINT A,T,C,"RANGE ZERO",O$
710 GOTO 260
2000 PRINT
2010 IF F=0 THEN 2100
2020 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT "          END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 127"
3000 DATA -0.987654000E+05,-0.101562695E-01
3010 DATA -0.109934299E+00, 0.897227872E-01
3020 DATA -0.987644000E+05, 0.835940129E+00
3030 DATA -0.776972770E+00, 0.886555051E+00
3040 DATA -0.942682000E+05,-0.999999999E+00
3050 DATA -1.000000000E+00,-0.994999665E+00
3060 DATA -0.942792000E+04, 0.446577015E-03
3070 DATA -0.955328265E-02, 0.104464371E-01
3080 DATA -0.942592000E+04,-0.909483178E+00
3090 DATA -0.913595542E+00,-0.905279866E+00
3100 DATA -0.942949000E+04, 0.999999939E+00
3110 DATA 0.999945941E+00, 1.000000000E+00
3120 DATA -0.986460000E+03, 0.932271951E-04
3130 DATA -0.906773181E-03, 0.109323198E-02
3140 DATA -0.999000000E+03, 0.264607527E-01
3150 DATA 0.254610398E-01, 0.274604392E-01
```

3160 DATA -0.994314000E+03,-0.999999997E+00  
3170 DATA -1.000000000E+00,-0.999998922E+00  
3180 DATA -0.879646000E+02,-0.569948578E-05  
3190 DATA -0.105699986E-03, 0.943005641E-04  
3200 DATA -0.990000000E+02, 0.999206834E+00  
3210 DATA 0.999202347E+00, 0.999211312E+00  
3220 DATA -0.895354000E+02,-1.000000000E+00  
3230 DATA -1.000000000E+00,-0.999999494E+00  
3240 DATA -0.500000000E+02, 0.262374854E+00  
3250 DATA 0.262277855E+00, 0.262471849E+00  
3260 DATA -0.200000000E+02,-0.912945251E+00  
3270 DATA -0.912986555E+00,-0.912903937E+00  
3280 DATA -0.590000000E+01, 0.373876665E+00  
3290 DATA 0.373866890E+00, 0.373886440E+00  
3300 DATA -0.471239000E+01, 1.000000000E+00  
3310 DATA 0.999999499E+00, 1.000000000E+00  
3320 DATA -0.314159000E+01,-0.265358979E-05  
3330 DATA -0.126536398E-04, 0.734641521E-05  
3340 DATA -0.230000000E+01,-0.745705212E+00  
3350 DATA -0.745712375E+00,-0.745698049E+00  
3360 DATA -0.157080000E+01,-1.000000000E+00  
3370 DATA -1.000000000E+00,-0.999999499E+00  
3380 DATA -0.120000000E+01,-0.932039086E+00  
3390 DATA -0.932043210E+00,-0.932034962E+00  
3400 DATA -0.100000000E+01,-0.841470985E+00  
3410 DATA -0.841476888E+00,-0.841465081E+00  
3420 DATA -0.987654000E+00,-0.834736453E+00  
3430 DATA -0.834737504E+00,-0.834735402E+00  
3440 DATA -0.765432000E+00,-0.692848571E+00  
3450 DATA -0.692849793E+00,-0.692847350E+00  
3460 DATA -0.987654000E-01,-0.986049087E-01  
3470 DATA -0.986050583E-01,-0.986047592E-01  
3480 DATA -0.765432000E-01,-0.764684792E-01  
3490 DATA -0.764686290E-01,-0.764683295E-01  
3500 DATA -0.975319000E-02,-0.975303537E-02  
3510 DATA -0.975305038E-02,-0.975302037E-02  
3520 DATA -0.345678000E-02,-0.345677312E-02  
3530 DATA -0.345678812E-02,-0.345675811E-02  
3540 DATA -0.100001000E-02,-0.100000983E-02  
3550 DATA -0.100002484E-02,-0.999999333E-03  
3560 DATA -0.100000000E-02,-0.999999833E-03  
3570 DATA -0.100001484E-02,-0.999989333E-03  
3580 DATA -0.999999000E-03,-0.999998833E-03  
3590 DATA -1.000000334E-03,-0.999997333E-03  
3600 DATA -0.987654000E-04,-0.987653998E-04  
3610 DATA -0.987655499E-04,-0.987652498E-04  
3620 DATA -0.917359000E-05,-0.917359000E-05  
3630 DATA -0.917360500E-05,-0.917357499E-05  
3640 DATA -0.900000000E-06,-0.900000000E-06  
3650 DATA -0.900001500E-06,-0.899998499E-06  
3660 DATA -0.900000000E-09,-0.900000000E-09  
3670 DATA -0.900001500E-09,-0.899998500E-09  
3680 DATA -0.900000000E-19,-0.900000000E-19  
3690 DATA -0.900001500E-19,-0.899998500E-19  
3700 DATA -0.900000000E-29,-0.900000000E-29

3710 DATA -0.900001500E-29,-0.899998500E-29  
3720 DATA -0.900000000E-37,-0.900000000E-37  
3730 DATA -0.900001500E-37,-0.899998500E-37  
3740 DATA 0,0,0,0  
3760 DATA 0.200000000E-37, 0.200000000E-37  
3770 DATA 0.199998500E-37, 0.200001500E-37  
3780 DATA 0.900000000E-29, 0.900000000E-29  
3790 DATA 0.899998500E-29, 0.900001500E-29  
3800 DATA 0.900000000E-19, 0.900000000E-19  
3810 DATA 0.899998500E-19, 0.900001500E-19  
3820 DATA 0.900000000E-09, 0.900000000E-09  
3830 DATA 0.899998500E-09, 0.900001500E-09  
3840 DATA 0.900000000E-06, 0.900000000E-06  
3850 DATA 0.899998499E-06, 0.900001500E-06  
3860 DATA 0.917359000E-05, 0.917359000E-05  
3870 DATA 0.917357499E-05, 0.917360500E-05  
3880 DATA 0.987654000E-04, 0.987653998E-04  
3890 DATA 0.987652498E-04, 0.987655499E-04  
3900 DATA 0.999999000E-03, 0.999998833E-03  
3910 DATA 0.999997333E-03, 1.000000334E-03  
3920 DATA 0.100000000E-02, 0.999999833E-03  
3930 DATA 0.999989333E-03, 0.100001484E-02  
3940 DATA 0.100001000E-02, 0.100000983E-02  
3950 DATA 0.999999333E-03, 0.100002484E-02  
3960 DATA 0.345678000E-02, 0.345677312E-02  
3970 DATA 0.345675811E-02, 0.345678812E-02  
3980 DATA 0.975319000E-02, 0.975303537E-02  
3990 DATA 0.975302037E-02, 0.975305038E-02  
4000 DATA 0.765432000E-01, 0.764684792E-01  
4010 DATA 0.764683295E-01, 0.764686290E-01  
4020 DATA 0.987654000E-01, 0.986049087E-01  
4030 DATA 0.986047592E-01, 0.986050583E-01  
4040 DATA 0.765432000E+00, 0.692848571E+00  
4050 DATA 0.692847350E+00, 0.692849793E+00  
4060 DATA 0.987654000E+00, 0.834736453E+00  
4070 DATA 0.834735402E+00, 0.834737504E+00  
4080 DATA 0.100000000E+01, 0.841470985E+00  
4090 DATA 0.841465081E+00, 0.841476888E+00  
4100 DATA 0.120000000E+01, 0.932039086E+00  
4110 DATA 0.932034962E+00, 0.932043210E+00  
4120 DATA 0.157080000E+01, 1.000000000E+00  
4130 DATA 0.999999499E+00, 1.000000000E+00  
4140 DATA 0.230000000E+01, 0.745705212E+00  
4150 DATA 0.745698049E+00, 0.745712375E+00  
4160 DATA 0.314159000E+01, 0.265358979E-05  
4170 DATA -0.734641521E-05, 0.126536398E-04  
4180 DATA 0.390000000E+01,-0.687766159E+00  
4190 DATA -0.687773919E+00,-0.687758399E+00  
4200 DATA 0.471239000E+01,-1.000000000E+00  
4210 DATA -1.000000000E+00,-0.999999499E+00  
4220 DATA 0.590000000E+01,-0.373876665E+00  
4230 DATA -0.373886440E+00,-0.373866890E+00  
4240 DATA 0.200000000E+02, 0.912945251E+00  
4250 DATA 0.912903937E+00, 0.912986555E+00  
4260 DATA 0.500000000E+02,-0.262374854E+00

```

4270 DATA -0.262471849E+00,-0.262277855E+00
4280 DATA 0.895354000E+02, 1.000000000E+00
4290 DATA 0.999999494E+00, 1.000000000E+00
4300 DATA 0.990000000E+02,-0.999206834E+00
4310 DATA -0.999211312E+00,-0.999202347E+00
4320 DATA 0.879646000E+02, 0.569948578E-05
4330 DATA -0.943005641E-04, 0.105699986E-03
4360 DATA 0.994314000E+03, 0.999999997E+00
4370 DATA 0.999998922E+00, 1.000000000E+00
4380 DATA 0.999100000E+03, 0.734699011E-01
4390 DATA 0.724725170E-01, 0.744672117E-01
4400 DATA 0.986460000E+03,-0.932271951E-04
4410 DATA -0.109323198E-02, 0.906773181E-03
4420 DATA 0.942949000E+04,-0.999999939E+00
4430 DATA -1.000000000E+00,-0.999945941E+00
4440 DATA 0.942592000E+04, 0.909483178E+00
4450 DATA 0.905279866E+00, 0.913595542E+00
4460 DATA 0.942792000E+04,-0.446577015E-03
4470 DATA -0.104464371E-01, 0.955328265E-02
4480 DATA 0.942682000E+05, 0.999999999E+00
4490 DATA 0.994999665E+00, 1.000000000E+00
4500 DATA 0.987644000E+05,-0.835940129E+00
4510 DATA -0.886555051E+00,-0.776972770E+00
4520 DATA 0.987654000E+05, 0.101562695E-01
4530 DATA -0.897227872E-01, 0.109934299E+00
9000 DATA 999,0,0,0
9400 END

```

---

PROGRAM FILE 127: ACCURACY OF SIN FUNCTION.  
ANSI STANDARD 7.6, 8.4

#### SECTION 127.1: ACCURACY OF SIN FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE SIN FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST 2 \* THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME
----------	---------------	-------------------	------------------	---------

-98765.4	-1.01563E-2	-9.97071E-3	1.85878E-3	OK
-98764.4	.83594	.836127	3.41517E-3	OK
-94268.2	-1.	-1.	1.13241E-4	OK
-9427.92	4.46577E-4	4.79369E-4	3.27924E-3	OK
-9425.92	-.909483	-.909487	9.47934E-4	OK
-9429.49	1.	1.	0	OK - EXACT
-986.46	9.32272E-5	9.58738E-5	2.64660E-3	OK
-999	2.64608E-2	2.64581E-2	2.67207E-3	OK
-994.314	-1.	-1	2.73973E-2	OK
-87.9646	-5.69949E-6	-5.99211E-6	2.92626E-3	OK
-99	.999207	.999207	4.98339E-3	OK
-89.5354	-1	-1	0	OK - EXACT
-50	.262375	.262375	3.84069E-4	OK
-20	-.912945	-.912945	7.21436E-4	OK
-5.9	.373877	.373877	4.19207E-3	OK
-4.71239	1	1	0	OK - EXACT
-3.14159	-2.65359E-6	-2.69177E-6	3.81795E-3	OK
-2.3	-.745705	-.745705	1.04004E-3	OK
-1.5708	-1	-1	0	OK - EXACT
-1.2	-.932039	-.932039	5.42005E-3	OK
-1	-.841471	-.841471	1.26342E-3	OK
-.987654	-.834736	-.834736	0	OK - EXACT
-.765432	-.692849	-.692849	6.09756E-3	OK
-9.87654E-2	-9.86049E-2	-9.86049E-2	0	OK - EXACT
-7.65432E-2	-7.64685E-2	-7.64685E-2	6.21118E-3	OK
-9.75319E-3	-9.75304E-3	-9.75304E-3	7.75194E-3	OK
-3.45678E-3	-3.45677E-3	-3.45677E-3	1.94175E-3	OK
-1.00001E-3	-1.00001E-3	-1.00001E-3	1.48571E-2	OK
-.001	-1.00000E-3	-.001	1.37065E-2	OK
-9.99999E-4	-9.99999E-4	-9.99999E-4	.117647	OK
-9.87654E-5	-9.87654E-5	-9.87654E-5	1.21212E-2	OK
-9.17359E-6	-9.17359E-6	-9.17359E-6	0	OK - EXACT
-9.00000E-7	-9.00000E-7	-9.00000E-7	0	OK - EXACT
-9.00000E-10	-9.00000E-10	-9.00000E-10	0	OK - EXACT
-9.00000E-20	-9.00000E-20	-9.00000E-20	0	OK - EXACT
-9.00000E-30	-9.00000E-30	-9.00000E-30	0	OK - EXACT
-9.00000E-38	-9.00000E-38	-9.00000E-38	0	OK - EXACT
0	0	0	RANGE ZERO	OK - EXACT
2.00000E-38	2.00000E-38	2.00000E-38	0	OK - EXACT
9.00000E-30	9.00000E-30	9.00000E-30	0	OK - EXACT
9.00000E-20	9.00000E-20	9.00000E-20	0	OK - EXACT
9.00000E-10	9.00000E-10	9.00000E-10	0	OK - EXACT
9.00000E-7	9.00000E-7	9.00000E-7	0	OK - EXACT
9.17359E-6	9.17359E-6	9.17359E-6	0	OK - EXACT
9.87654E-5	9.87654E-5	9.87654E-5	1.21212E-2	OK
9.99999E-4	9.99999E-4	9.99999E-4	.117647	OK
.001	1.00000E-3	.001	1.37065E-2	OK
1.00001E-3	1.00001E-3	1.00001E-3	1.48571E-2	OK
3.45678E-3	3.45677E-3	3.45677E-3	1.94175E-3	OK
9.75319E-3	9.75304E-3	9.75304E-3	7.75194E-3	OK
7.65432E-2	7.64685E-2	7.64685E-2	6.21118E-3	OK
9.87654E-2	9.86049E-2	9.86049E-2	0	OK - EXACT
.765432	.692849	.692849	6.09756E-3	OK
.987654	.834736	.834736	0	OK - EXACT
1	.841471	.841471	1.26342E-3	OK

1.2	.932039	.932039	5.42005E-3	OK
1.5708	1	1	0	OK - EXACT
2.3	.745705	.745705	1.04004E-3	OK
3.14159	2.65359E-6	2.69177E-6	3.81795E-3	OK
3.9	-.687766	-.687766	9.60154E-4	OK
4.71239	-1	-1	0	OK - EXACT
5.9	-.373877	-.373877	4.19207E-3	OK
20	.912945	.912945	7.21436E-4	OK
50	-.262375	-.262375	3.84069E-4	OK
89.5354	1	1	0	OK - EXACT
99	-.999207	-.999207	4.98339E-3	OK
87.9646	5.69949E-6	5.99211E-6	2.92626E-3	OK
994.314	1.	1	2.73973E-2	OK
999.1	7.34699E-2	7.34689E-2	9.54344E-4	OK
986.46	-9.32272E-5	-9.58738E-5	2.64660E-3	OK
9429.49	-1.	-1.	0	OK - EXACT
9425.92	.909483	.909487	9.47934E-4	OK
9427.92	-4.46577E-4	-4.79369E-4	3.27924E-3	OK
94268.2	1.	1.	1.13241E-4	OK
98764.4	-.83594	-.836127	3.41517E-3	OK
98765.4	1.01563E-2	9.97071E-3	1.85878E-3	OK

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 127

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 128: ACCURACY OF TAN FUNCTION."
20 PRINT "    ANSI STANDARD 7.6, 8.4"
30 PRINT
40 PRINT "SECTION 128.1: ACCURACY OF TAN FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE"
70 PRINT "TAN FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN,"
80 PRINT "ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY"
90 PRINT "THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT"
100 PRINT "PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN"
110 PRINT "'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE"
120 PRINT "IS JUST 2 * THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE;"
130 PRINT "THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY"
140 PRINT "HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES"
150 PRINT "AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3."
160 PRINT
170 PRINT "THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD"
180 PRINT "DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS."
190 PRINT
200 PRINT "                BEGIN TEST"
210 PRINT

```

```
220 PRINT "ARGUMENT","TRUE","COMPUTED","ERROR","OUTCOME"
230 PRINT " ","VALUE","VALUE","MEASURE"
240 PRINT
250 LET F=0
260 READ A,T,L,H
270 IF A=999 THEN 2000
280 LET C=TAN(A)
285 IF L>H THEN 1000
290 IF C=T THEN 333
300 IF C<L THEN 444
310 IF C>H THEN 444
320 LET O$=" OK "
330 GOTO 500
333 LET O$=" OK - EXACT"
340 GOTO 500
444 LET O$="FAIL"
445 LET F=F+1
450 GOTO 500
500 LET R=H-L
510 IF R=0 THEN 700
520 LET E=ABS((C-T)/R)
525 LET E=E+E
530 PRINT A,T,C,E,O$
540 GOTO 260
700 PRINT A,T,C,"RANGE ZERO",O$
710 GOTO 260
1000 IF C=T THEN 1333
1010 IF C>L THEN 1444
1020 IF C<H THEN 1444
1030 LET O$="FAIL"
1040 LET F=F+1
1050 GOTO 1500
1333 LET O$=" OK - EXACT"
1344 GOTO 1500
1444 LET O$=" OK "
1445 GOTO 1500
1500 PRINT A,T,C,"RANGE SPLIT",O$
1510 GOTO 260
2000 PRINT
2010 IF F=0 THEN 2100
2020 PRINT "*** INFORMATIVE TEST FAILED IN ";F;" CASE(S) ***"
2030 GOTO 2110
2100 PRINT "*** INFORMATIVE TEST PASSED ***"
2110 PRINT
2120 PRINT " END TEST"
2130 PRINT
2140 PRINT "END PROGRAM 128"
3000 DATA -0.987789000E+05,-0.138017476E+01
3010 DATA -0.171848904E+01,-0.112416124E+01
3020 DATA -0.987654000E+04, 0.724991905E+00
3030 DATA 0.709844580E+00, 0.740360468E+00
3040 DATA -0.987654000E+03,-0.252646135E+01
3050 DATA -0.253386806E+01,-0.251909194E+01
3060 DATA -0.987654000E+02,-0.506921050E+01
3070 DATA -0.507188654E+01,-0.506653715E+01
```

3080 DATA -0.987654000E+01,-0.485230117E+00  
3090 DATA -0.485242972E+00,-0.485217262E+00  
3100 DATA -0.555555000E+01, 0.890667992E+00  
3110 DATA 0.890649559E+00, 0.890686426E+00  
3120 DATA -0.471239000E+01, 0.980762048E+06  
3130 DATA 0.907472194E+05,-0.111353304E+06  
3140 DATA -0.456789000E+01,-0.687223055E+01  
3150 DATA -0.687271786E+01,-0.687174330E+01  
3160 DATA -0.400000000E+01,-0.115782128E+01  
3170 DATA -0.115784969E+01,-0.115779287E+01  
3180 DATA -0.314159000E+01, 0.265358979E-05  
3190 DATA -0.734641521E-05, 0.126536398E-04  
3200 DATA -0.287654000E+01, 0.271439087E+00  
3210 DATA 0.271427849E+00, 0.271450324E+00  
3220 DATA -0.200000000E+01, 0.218503986E+01  
3230 DATA 0.218497712E+01, 0.218510261E+01  
3240 DATA -0.161616000E+01, 0.220289470E+02  
3250 DATA 0.220240353E+02, 0.220338608E+02  
3280 DATA -0.157081000E+01, 0.731357419E+05  
3290 DATA 0.422418008E+05, 0.272242309E+06  
3300 DATA -0.157080000E+01, 0.272241808E+06  
3310 DATA 0.731356919E+05,-0.158057413E+06  
3320 DATA -0.157078000E+01,-0.612490085E+05  
3330 DATA -0.158058414E+06,-0.379840648E+05  
3340 DATA -0.123456000E+01,-0.286116592E+01  
3350 DATA -0.286126279E+01,-0.286106906E+01  
3360 DATA -0.100000000E+01,-0.155740772E+01  
3370 DATA -0.155744699E+01,-0.155736847E+01  
3380 DATA -0.876540000E+00,-0.120117664E+01  
3390 DATA -0.120118408E+01,-0.120116919E+01  
3400 DATA -0.232323000E+00,-0.236595054E+00  
3410 DATA -0.236596611E+00,-0.236593498E+00  
3420 DATA -0.767676000E-01,-0.769187603E-01  
3430 DATA -0.769189109E-01,-0.769186096E-01  
3460 DATA -0.234567000E-01,-0.234610030E-01  
3470 DATA -0.234611531E-01,-0.234608529E-01  
3480 DATA -0.123456000E-01,-0.123462273E-01  
3490 DATA -0.123463773E-01,-0.123460772E-01  
3520 DATA -0.987654000E-02,-0.987686115E-02  
3530 DATA -0.987687616E-02,-0.987684615E-02  
3540 DATA -0.345678000E-02,-0.345679377E-02  
3550 DATA -0.345680877E-02,-0.345677876E-02  
3560 DATA -0.987654000E-03,-0.987654321E-03  
3570 DATA -0.987655822E-03,-0.987652821E-03  
3580 DATA -0.345678000E-03,-0.345678014E-03  
3590 DATA -0.345679514E-03,-0.345676513E-03  
3720 DATA -0.900000000E-04,-0.900000002E-04  
3730 DATA -0.900001503E-04,-0.899998502E-04  
3740 DATA -0.900000000E-06,-0.900000000E-06  
3750 DATA -0.900001501E-06,-0.899998500E-06  
3760 DATA -0.900000000E-09,-0.900000000E-09  
3770 DATA -0.900001500E-09,-0.899998500E-09  
3780 DATA -0.900000000E-19,-0.900000000E-19  
3790 DATA -0.900001500E-19,-0.899998500E-19  
3800 DATA -0.900000000E-29,-0.900000000E-29

3810 DATA -0.900001500E-29,-0.899998500E-29  
3812 DATA -0.900000000E-37,-0.900000000E-37  
3814 DATA -0.900001500E-37,-0.899998500E-37  
3815 DATA 0,0,0,0  
3816 DATA 0.900000000E-37, 0.900000000E-37  
3818 DATA 0.899998500E-37, 0.900001500E-37  
3820 DATA 0.900000000E-29, 0.900000000E-29  
3830 DATA 0.899998500E-29, 0.900001500E-29  
3840 DATA 0.900000000E-19, 0.900000000E-19  
3850 DATA 0.899998500E-19, 0.900001500E-19  
3860 DATA 0.900000000E-09, 0.900000000E-09  
3870 DATA 0.899998500E-09, 0.900001500E-09  
3880 DATA 0.900000000E-06, 0.900000000E-06  
3890 DATA 0.899998500E-06, 0.900001501E-06  
3900 DATA 0.900000000E-04, 0.900000002E-04  
3910 DATA 0.899998502E-04, 0.900001503E-04  
3920 DATA 0.345678000E-03, 0.345678014E-03  
3930 DATA 0.345676513E-03, 0.345679514E-03  
3940 DATA 0.987654000E-03, 0.987654321E-03  
3950 DATA 0.987652821E-03, 0.987655822E-03  
3960 DATA 0.345678000E-02, 0.345679377E-02  
3970 DATA 0.345677876E-02, 0.345680877E-02  
3980 DATA 0.987654000E-02, 0.987686115E-02  
3990 DATA 0.987684615E-02, 0.987687616E-02  
4000 DATA 0.123456000E-01, 0.123462273E-01  
4010 DATA 0.123460772E-01, 0.123463773E-01  
4020 DATA 0.234567000E-01, 0.234610030E-01  
4030 DATA 0.234608529E-01, 0.234611531E-01  
4040 DATA 0.767676000E-01, 0.769187603E-01  
4050 DATA 0.769186096E-01, 0.769189109E-01  
4060 DATA 0.232323000E+00, 0.236595054E+00  
4070 DATA 0.236593498E+00, 0.236596611E+00  
4080 DATA 0.876543000E+00, 0.120118397E+01  
4090 DATA 0.120117652E+01, 0.120119141E+01  
4100 DATA 0.100000000E+01, 0.155740772E+01  
4110 DATA 0.155736847E+01, 0.155744699E+01  
4120 DATA 0.123456000E+01, 0.286116592E+01  
4130 DATA 0.286106906E+01, 0.286126279E+01  
4140 DATA 0.157078000E+01, 0.612490085E+05  
4150 DATA 0.379840648E+05, 0.158058414E+06  
4160 DATA 0.157080000E+01,-0.272241808E+06  
4170 DATA 0.158057413E+06,-0.731356919E+05  
4180 DATA 0.157081000E+01,-0.731357419E+05  
4190 DATA -0.272242309E+06,-0.422418008E+05  
4200 DATA 0.161616000E+01,-0.220289470E+02  
4210 DATA -0.220338608E+02,-0.220240353E+02  
4220 DATA 0.200000000E+01,-0.218503986E+01  
4230 DATA -0.218510261E+01,-0.218497712E+01  
4240 DATA 0.287654000E+01,-0.271439087E+00  
4250 DATA -0.271450324E+00,-0.271427849E+00  
4260 DATA 0.314159000E+01,-0.265358979E-05  
4270 DATA -0.126536398E-04, 0.734641521E-05  
4280 DATA 0.400000000E+01, 0.115782128E+01  
4290 DATA 0.115779287E+01, 0.115784969E+01  
4300 DATA 0.456789000E+01, 0.687223055E+01

```

4310 DATA 0.687174330E+01, 0.687271786E+01
4320 DATA 0.471239000E+01,-0.980762049E+06
4330 DATA 0.111353304E+06,-0.907472194E+05
4340 DATA 0.555555000E+01,-0.890667992E+00
4350 DATA -0.890686426E+00,-0.890649559E+00
4360 DATA 0.987654000E+01, 0.485230117E+00
4370 DATA 0.485217262E+00, 0.485242972E+00
4380 DATA 0.987654000E+02, 0.506921050E+01
4390 DATA 0.506653715E+01, 0.507188654E+01
4400 DATA 0.987654000E+03, 0.252646135E+01
4410 DATA 0.251909194E+01, 0.253386806E+01
4420 DATA 0.987654000E+04,-0.724991905E+00
4430 DATA -0.740360468E+00,-0.709844580E+00
4440 DATA 0.987789000E+05, 0.138017476E+01
4450 DATA 0.112416124E+01, 0.171848904E+01
9000 DATA 999,0,0,0
9400 END

```

---

#### PROGRAM FILE 128: ACCURACY OF TAN FUNCTION.

ANSI STANDARD 7.6, 8.4

#### SECTION 128.1: ACCURACY OF TAN FUNCTION.

THIS PROGRAM TESTS VALUES RETURNED BY INVOCATIONS OF THE TAN FUNCTION FOR ACCURACY. THE INVOCATION MUST RETURN, ACCURATE TO SIX DIGITS, SOME VALUE ACTUALLY TAKEN ON BY THE FUNCTION WITHIN A DOMAIN BOUNDED BY THE ACTUAL ARGUMENT PLUS OR MINUS ONE IN THE SIXTH DIGIT. ALSO, AN 'ERROR MEASURE' IS COMPUTED AND REPORTED. THIS MEASURE IS JUST 2 \* THE ABSOLUTE ERROR OVER THE SIZE OF THE RANGE; THUS, A VALUE JUST BARELY PASSING OR FAILING WILL USUALLY HAVE A MEASURE OF ABOUT 1. A VALUE 3 TIMES AS INACCURATE AS EXPECTED WILL HAVE A MEASURE OF 3.

THIS TEST IS INFORMATIVE ONLY, SINCE THE ANSI STANDARD DOES NOT MANDATE ANY ACCURACY FOR SUPPLIED-FUNCTIONS.

#### BEGIN TEST

ARGUMENT	TRUE VALUE	COMPUTED VALUE	ERROR MEASURE	OUTCOME
-98778.9	-1.38017	-1.37905	3.78712E-3	OK
-9876.54	.724992	.724991	6.10386E-5	OK
-987.654	-2.52646	-2.52653	9.64897E-3	OK
-98.7654	-5.06921	-5.06922	2.16161E-3	OK
-9.87654	-.48523	-.48523	8.69439E-4	OK
-5.55555	.890668	.890668	2.42473E-3	OK
-4.71239	980762.	1.01721E+6	RANGE SPLIT	OK
-4.56789	-6.87223	-6.87223	1.46798E-3	OK
-4	-1.15782	-1.15782	5.24659E-3	OK
-3.14159	2.65359E-6	2.69177E-6	3.81795E-3	OK
-2.87654	.271439	.271439	1.98873E-3	OK

-2	2.18504	2.18504	0	OK	- EXACT
-1.61616	22.0289	22.029	3.44593E-3	OK	
-1.57081	73135.7	73155.5	1.72078E-4	OK	
-1.5708	272242.	272120.	RANGE SPLIT	OK	
-1.57078	-61249.	-61251.4	3.93230E-5	OK	
-1.23456	-2.86117	-2.86117	9.23219E-4	OK	
-1	-1.55741	-1.55741	1.89789E-3	OK	
-.87654	-1.20118	-1.20118	3.99600E-3	OK	
-.232323	-.236595	-.236595	1.19403E-3	OK	
-7.67676E-2	-7.69188E-2	-7.69188E-2	0	OK	- EXACT
-2.34567E-2	-2.34610E-2	-2.34610E-2	0	OK	- EXACT
-1.23456E-2	-1.23462E-2	-1.23462E-2	1.55219E-3	OK	
-9.87654E-3	-9.87686E-3	-9.87686E-3	7.75194E-3	OK	
-3.45678E-3	-3.45679E-3	-3.45679E-3	3.88350E-3	OK	
-9.87654E-4	-9.87654E-4	-9.87654E-4	0	OK	- EXACT
-3.45678E-4	-3.45678E-4	-3.45678E-4	7.27273E-3	OK	
-.00009	-.00009	-.00009	0	OK	- EXACT
-9.00000E-7	-9.00000E-7	-9.00000E-7	0	OK	- EXACT
-9.00000E-10	-9.00000E-10	-9.00000E-10	0	OK	- EXACT
-9.00000E-20	-9.00000E-20	-9.00000E-20	0	OK	- EXACT
-9.00000E-37	-9.00000E-37	-9.00000E-37	0	OK	- EXACT
-9.00000E-38	-9.00000E-38	-9.00000E-38	0	OK	- EXACT
0	0	0	RANGE ZERO	OK	- EXACT
9.00000E-38	9.00000E-38	9.00000E-38	0	OK	- EXACT
9.00000E-37	9.00000E-37	9.00000E-37	0	OK	- EXACT
9.00000E-20	9.00000E-20	9.00000E-20	0	OK	- EXACT
9.00000E-10	9.00000E-10	9.00000E-10	0	OK	- EXACT
9.00000E-7	9.00000E-7	9.00000E-7	0	OK	- EXACT
.00009	.00009	.00009	0	OK	- EXACT
3.45678E-4	3.45678E-4	3.45678E-4	7.27273E-3	OK	
9.87654E-4	9.87654E-4	9.87654E-4	0	OK	- EXACT
3.45678E-3	3.45679E-3	3.45679E-3	3.88350E-3	OK	
9.87654E-3	9.87686E-3	9.87686E-3	7.75194E-3	OK	
1.23456E-2	1.23462E-2	1.23462E-2	1.55219E-3	OK	
2.34567E-2	2.34610E-2	2.34610E-2	0	OK	- EXACT
7.67676E-2	7.69188E-2	7.69188E-2	0	OK	- EXACT
.232323	.236595	.236595	1.19403E-3	OK	
.876543	1.20118	1.20118	.01	OK	
1	1.55741	1.55741	1.89789E-3	OK	
1.23456	2.86117	2.86117	9.23219E-4	OK	
1.57078	61249.	61251.4	3.93230E-5	OK	
1.5708	-272242.	-272120.	RANGE SPLIT	OK	
1.57081	-73135.7	-73155.5	1.72078E-4	OK	
1.61616	-22.0289	-22.029	3.44593E-3	OK	
2	-2.18504	-2.18504	0	OK	- EXACT
2.87654	-.271439	-.271439	1.98873E-3	OK	
3.14159	-2.65359E-6	-2.69177E-6	3.81795E-3	OK	
4	1.15782	1.15782	5.24659E-3	OK	
4.56789	6.87223	6.87223	1.46798E-3	OK	
4.71239	-980762.	-1.01721E+6	RANGE SPLIT	OK	
5.55555	-.890668	-.890668	2.42473E-3	OK	
9.87654	.48523	.48523	8.69439E-4	OK	
98.7654	5.06921	5.06922	2.16161E-3	OK	

987.654	2.52646	2.52653	9.64897E-3	OK
9876.54	-.724992	-.724991	6.10386E-5	OK
98778.9	1.38017	1.37905	3.78712E-3	OK

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 128

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 129: EXCEPTION - OVERFLOW ON VALUE OF"
15 PRINT "          TAN FUNCTION."
20 PRINT "      ANSI STANDARD 8.5"
30 PRINT
40 PRINT "SECTION 129.1: EXCEPTION - OVERFLOW ON VALUE OF"
50 PRINT "          TAN FUNCTION."
60 PRINT
70 PRINT "THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR"
80 PRINT "RECOGNIZES AN EXCEPTION IF THE VALUE RETURNED BY THE"
90 PRINT "TAN FUNCTION EXCEEDS MACHINE INFINITY."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) OVERFLOW MUST NOT OCCUR, OR"
140 PRINT
150 PRINT "      2) WHEN OVERFLOW OCCURS, THE PROCESSOR MUST DISPLAY A"
160 PRINT "          MESSAGE IDENTIFYING THE EXCEPTION, SUPPLY MACHINE"
170 PRINT "          INFINITY AND CONTINUE EXECUTION."
180 PRINT
190 PRINT
200 PRINT "                                BEGIN TEST."
210 PRINT
220 PRINT "THE PROGRAM ATTEMPTS TO FORCE CONVERGENCE OF THE ARGUMENT"
230 PRINT "TO PI/2, SO AS TO CAUSE OVERFLOW. IF OVERFLOW DOES NOT"
240 PRINT "OCCUR, IT IS LIKELY THAT THIS SYSTEM HAS NO INTERNAL"
250 PRINT "NUMERIC REPRESENTATION FOR WHICH THE TANGENT EXCEEDS"
260 PRINT "MACHINE INFINITY."
270 PRINT
280 LET A=1.56
290 LET B=1.58
300 PRINT "ARGUMENT","VALUE OF TAN"
310 PRINT
320 LET M=(A+B)/2
330 LET R=TAN(M)
340 PRINT M,R
350 IF M<=A THEN 420
360 IF M>=B THEN 420
370 IF R<0 THEN 400
380 LET A=M

```

```
390 GOTO 320
400 LET B=M
410 GOTO 320
420 PRINT
430 PRINT "ARGUMENTS HAVE CONVERGED. IF OVERFLOW HAS OCCURRED,"
440 PRINT "PROCESSOR MUST HAVE REPORTED EXCEPTION AND SUPPLIED"
450 PRINT "MACHINE INFINITY, OTHERWISE,"
455 PRINT "*** TEST FAILED ***"
460 PRINT
470 PRINT " END TEST"
480 PRINT
490 PRINT "END PROGRAM 129"
500 END
```

---

PROGRAM FILE 129: EXCEPTION - OVERFLOW ON VALUE OF  
TAN FUNCTION.

ANSI STANDARD 8.5

SECTION 129.1: EXCEPTION - OVERFLOW ON VALUE OF  
TAN FUNCTION.

THE OBJECT OF THIS TEST IS TO SEE WHETHER THE PROCESSOR  
RECOGNIZES AN EXCEPTION IF THE VALUE RETURNED BY THE  
TAN FUNCTION EXCEEDS MACHINE INFINITY.

TO PASS THIS TEST:

- 1) OVERFLOW MUST NOT OCCUR, OR
- 2) WHEN OVERFLOW OCCURS, THE PROCESSOR MUST DISPLAY A  
MESSAGE IDENTIFYING THE EXCEPTION, SUPPLY MACHINE  
INFINITY AND CONTINUE EXECUTION.

BEGIN TEST.

THE PROGRAM ATTEMPTS TO FORCE CONVERGENCE OF THE ARGUMENT  
TO PI/2, SO AS TO CAUSE OVERFLOW. IF OVERFLOW DOES NOT  
OCCUR, IT IS LIKELY THAT THIS SYSTEM HAS NO INTERNAL  
NUMERIC REPRESENTATION FOR WHICH THE TANGENT EXCEEDS  
MACHINE INFINITY.

ARGUMENT            VALUE OF TAN

1.57	1255.78
1.575	-237.886
1.5725	-586.964
1.57125	-2204.14
1.57063	5837.25
1.57094	-7082.7
1.57078	66391.3
1.57086	-15858.5
1.57082	-41640.2

1.5708	-223680.
1.57079	189039.
1.5708	2.51311E+6
1.5708	-491067.
1.5708	-1.22065E+6
1.5708	-4.74698E+6
1.5708	1.06807E+7
1.5708	-1.42409E+7
1.5708	85445660
1.5708	-42722830
1.5708	-42722830

% TAN OF PI/2 OR COTAN OF ZERO IN LINE 330

1.5708	1.70141E+38
1.5708	-42722830

ARGUMENTS HAVE CONVERGED. IF OVERFLOW HAS OCCURRED,  
 PROCESSOR MUST HAVE REPORTED EXCEPTION AND SUPPLIED  
 MACHINE INFINITY, OTHERWISE,  
 \*\*\* TEST FAILED \*\*\*

END TEST

END PROGRAM 129

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

```

10 PRINT "PROGRAM FILE 130: RND FUNCTION WITHOUT RANDOMIZE STATEMENT."
20 PRINT "      ANSI STANDARD 8.2, 8.4"
30 PRINT
40 PRINT "SECTION 130.1: RND FUNCTION WITHOUT RANDOMIZE STATEMENT."
50 PRINT
60 PRINT "THE FOLLOWING 20 NUMBERS SHOULD BE THE IMPLEMENTATION-"
70 PRINT "DEFINED SEQUENCE OF PSEUDO-RANDOM NUMBERS FOR THE"
80 PRINT "RND FUNCTION."
90 PRINT
100 PRINT "           BEGIN TEST"
110 PRINT
120 PRINT "POSITION","VALUE"
130 FOR I=1 TO 20
140 PRINT I,RND
150 NEXT I
160 PRINT
170 PRINT "TO DETERMINE WHETHER THIS TEST PASSES OR FAILS, THE USER"
180 PRINT "MUST RUN THE TEST THREE TIMES, NOTING THE SEQUENCE OF"
190 PRINT "NUMBERS PRODUCED EACH TIME. IF ALL THREE ARE IDENTICAL, "
200 PRINT "**** TEST PASSED ****"
210 PRINT
220 PRINT "           END TEST"

```

```
230 PRINT
240 PRINT "END PROGRAM 130"
250 END
```

---

PROGRAM FILE 130: RND FUNCTION WITHOUT RANDOMIZE STATEMENT.  
ANSI STANDARD 8.2, 8.4

SECTION 130.1: RND FUNCTION WITHOUT RANDOMIZE STATEMENT.

THE FOLLOWING 20 NUMBERS SHOULD BE THE IMPLEMENTATION-  
DEFINED SEQUENCE OF PSEUDO-RANDOM NUMBERS FOR THE  
RND FUNCTION.

BEGIN TEST

POSITION	VALUE
1	.217873
2	.696209
3	.29751
4	.963794
5	.463246
6	.767746
7	.181667
8	.159454
9	6.52568E-2
10	.495683
11	.644913
12	.927201
13	.67735
14	.804367
15	.992458
16	2.75721E-2
17	.322263
18	.731568
19	.704922
20	.12663

TO DETERMINE WHETHER THIS TEST PASSES OR FAILS, THE USER  
MUST RUN THE TEST THREE TIMES, NOTING THE SEQUENCE OF  
NUMBERS PRODUCED EACH TIME. IF ALL THREE ARE IDENTICAL,  
\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 130

---

---

---

10 PRINT "PROGRAM FILE 131: RND FUNCTION WITH THE RANDOMIZE STATEMENT."
20 PRINT " ANSI STANDARD 8.2, 8.4, 17.2, 17.4"
30 PRINT

```
50 PRINT "SECTION 131.1: RND FUNCTION WITH THE RANDOMIZE STATEMENT."
60 PRINT
70 PRINT "THIS TEST DETERMINES WHETHER THE RANDOMIZE STATEMENT"
80 PRINT "CAUSES A DIFFERENT SEQUENCE OF NUMBERS TO BE PRODUCED"
90 PRINT "BY THE RND FUNCTION FOR EACH PROGRAM EXECUTION."
100 PRINT
110 PRINT "               BEGIN TEST"
120 RANDOMIZE
130 PRINT
140 PRINT "POSITION","VALUE"
150 FOR I=1 TO 20
160 PRINT I,RND
170 NEXT I
180 PRINT
190 PRINT "TO DETERMINE WHETHER THIS TEST PASSES, THE USER MUST RUN"
200 PRINT "THIS PROGRAM THREE TIMES, NOTING THE SEQUENCE OF NUMBERS"
210 PRINT "PRODUCED EACH TIME. IF ALL THREE ARE DIFFERENT, THEN"
220 PRINT "*** TEST PASSED ***"
230 PRINT
240 PRINT "               END TEST"
250 PRINT
260 PRINT "END PROGRAM 131"
270 END
```

---

PROGRAM FILE 131: RND FUNCTION WITH THE RANDOMIZE STATEMENT.

ANSI STANDARD 8.2, 8.4, 17.2, 17.4

SECTION 131.1: RND FUNCTION WITH THE RANDOMIZE STATEMENT.

THIS TEST DETERMINES WHETHER THE RANDOMIZE STATEMENT  
CAUSES A DIFFERENT SEQUENCE OF NUMBERS TO BE PRODUCED  
BY THE RND FUNCTION FOR EACH PROGRAM EXECUTION.

BEGIN TEST

POSITION	VALUE
1	.651055
2	.969684
3	.638486
4	.704767
5	.415805
6	.143102
7	.355822
8	.385489
9	.781589
10	.924575
11	.771627
12	.528591
13	.280397
14	.157116
15	.31018
16	.144383
17	5.20237E-2

```

18      .685707
19      .424779
20      .20914

```

TO DETERMINE WHETHER THIS TEST PASSES, THE USER MUST RUN THIS PROGRAM THREE TIMES, NOTING THE SEQUENCE OF NUMBERS PRODUCED EACH TIME. IF ALL THREE ARE DIFFERENT, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 131

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 132: AVERAGE OF RANDOM NUMBERS APPROXIMATES 0.5"
15 PRINT "      AND 0 <= RND < 1."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 132.1: AVERAGE OF RANDOM NUMBERS APPROXIMATES 0.5"
50 PRINT "      AND 0 <= RND < 1."
60 PRINT
70 PRINT "THIS PROGRAM TESTS WHETHER THE AVERAGE OF A SERIES OF RANDOM"
80 PRINT "NUMBERS IS SIGNIFICANTLY FAR FROM THE IDEAL OF 0.5. IF SO,"
90 PRINT "THIS RESULT INDICATES NON-UNIFORM DISTRIBUTION AND THE TEST"
100 PRINT "FAILS. ALSO, IF ANY OF THE NUMBERS IS OUTSIDE THE ALLOWABLE"
110 PRINT "RANGE, AN IMMEDIATE FAILURE IS REPORTED."
120 PRINT
130 PRINT "          BEGIN TEST"
140 PRINT
150 REM N=8754 BASED ON SIGNIFICANCE LEVEL OF .05 AND 0.1 CHANCE OF
160 REM NOT FINDING A DIFFERENCE OF AT LEAST .01 FROM STANDARD
170 REM AVERAGE OF 0.5, AND STANDARD DEVIATION OF 1/(2*SQR(3))
180 LET N=8754
190 LET S1=0
200 LET S2=0
210 FOR I=1 TO N
220   LET X=RND
230   IF X<0 THEN 490
240   IF X>=1 THEN 510
250   LET S1=S1+X
260   LET S2=S2+(X*X)
270 NEXT I
280 LET S = SQR((N*S2) - (S1*S1)) / N
290 LET X1=S1/N
300 PRINT , "AVERAGE", "STD. DEV."
310 PRINT "THEORETICAL:", .5, 1/(2*SQR(3))
320 PRINT "ACTUAL:", X1, S
330 LET A1=ABS(X1-0.5)
340 REM 1.96 BASED ON SIGNIFICANCE LEVEL OF .05 AND INFINITE D.F.
350 LET A2=(1.96*S)/SQR(N)
360 PRINT

```

```
370 PRINT "ALLOWABLE DEVIATION: ";A2
380 PRINT "ACTUAL DEVIATION:      ";A1
390 PRINT
400 IF A1>A2 THEN 430
410 PRINT "*** TEST PASSED ***"
420 GOTO 440
430 PRINT "*** TEST FAILED ***"
440 PRINT
450 PRINT "                  END TEST"
460 PRINT
470 PRINT "END PROGRAM 132"
480 STOP
490 PRINT "RND#";I;" < 0: ";X
500 GOTO 430
510 PRINT "RND#";I;" >= 1: ";X
520 GOTO 430
530 END
```

---

PROGRAM FILE 132: AVERAGE OF RANDOM NUMBERS APPROXIMATES 0.5  
AND 0 <= RND < 1.  
ANSI STANDARD 8.4

SECTION 132.1: AVERAGE OF RANDOM NUMBERS APPROXIMATES 0.5  
AND 0 <= RND < 1.

THIS PROGRAM TESTS WHETHER THE AVERAGE OF A SERIES OF RANDOM NUMBERS IS SIGNIFICANTLY FAR FROM THE IDEAL OF 0.5. IF SO, THIS RESULT INDICATES NON-UNIFORM DISTRIBUTION AND THE TEST FAILS. ALSO, IF ANY OF THE NUMBERS IS OUTSIDE THE ALLOWABLE RANGE, AN IMMEDIATE FAILURE IS REPORTED.

BEGIN TEST

	AVERAGE	STD. DEV.
THEORETICAL:	.5	.288675
ACTUAL:	.494921	.288053

ALLOWABLE DEVIATION: 6.03429E-3  
ACTUAL DEVIATION: 5.07560E-3

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 132

---

---

---

```

10 PRINT "PROGRAM FILE 133: CHI-SQUARE UNIFORMITY TEST FOR"
15 PRINT "          RND FUNCTION."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 133.1: CHI-SQUARE UNIFORMITY TEST FOR"
50 PRINT "          RND FUNCTION."
55 PRINT
60 PRINT "THIS PROGRAM RUNS A SERIES OF 60 EXPERIMENTS OF 1050 SAMPLES"
70 PRINT "TO DETERMINE IF THE RANDOM NUMBERS ARE EQUALLY DISTRIBUTED"
80 PRINT "AMONG 21 EQUAL-SIZE INTERVALS BETWEEN 0 AND 1. CHI-SQUARE"
90 PRINT "IS COMPUTED AND DISPLAYED FOR EACH OF THESE EXPERIMENTS."
100 PRINT "EACH CHI-SQUARE RESULT IS CATEGORIZED ACCORDING TO THE"
110 PRINT "PERCENTILE RANGE INTO WHICH ITS VALUE FALLS:"
120 PRINT
130 PRINT "CATEGORY      % RANGE           CATEGORY      % RANGE"
140 PRINT "    1          0-1               6          50-75"
150 PRINT "    2          1-5               7          75-90"
160 PRINT "    3          5-10              8          90-95"
170 PRINT "    4          10-25             9          95-99"
180 PRINT "    5          25-50             10         99-100"
190 PRINT
200 PRINT "THESE CATEGORICAL RESULTS ARE THEN EVALUATED AGAIN USING"
210 PRINT "CHI-SQUARE (THE FIRST AND LAST THREE CATEGORIES ARE EACH"
220 PRINT "CONSOLIDATED INTO ONE), AND IF THE DISTRIBUTION AMONG"
230 PRINT "CATEGORIES IS SUFFICIENTLY IMPROBABLE (TOP OR BOTTOM 5%),"
240 PRINT "THE TEST FAILS."
250 PRINT
260 PRINT "                  BEGIN TEST"
270 PRINT
280 REM SET #TRIALS PER EXPERIMENT
290 LET P1=1050
300 DIM Y(21)
310 PRINT "EXPERIMENT","CHI-SQUARE","CATEGORY"
320 PRINT
330 REM PERFORM 60 EXPERIMENTS OF 1050 SAMPLES EACH
340 REM OF PSEUDO-RANDOM NUMBERS THEN COMPUTE
350 REM AND TEST CHI-SQ STATISTIC FOR EACH EXPERIMENT.
360 REM BELOW ARE CHI-SQ VALUES FOR 20 DEGREES OF FREEDOM FOR
370 REM PERCENTAGES 1,5,10,25,50,75,90,95,99, AND 100
380 LET L(1)=8.26040
390 LET L(2)=10.8508
400 LET L(3)=12.4426
410 LET L(4)=15.4518
420 LET L(5)=19.3374
430 LET L(6)=23.8277
440 LET L(7)=28.4120
450 LET L(8)=31.4104
460 LET L(9)=37.5662
470 LET L(10)=1E38
480 REM ZERO OUT CATEGORY COUNTS
490 FOR C=1 TO 10
500 LET Z(C)=0
510 NEXT C

```

```
520 REM DO 60 EXPERIMENTS, SO EXPECTED NUMBER FOR EACH GLOBAL
530 REM CATEGORY IS >= 5
540 FOR E=1 TO 60
550 REM INITIALIZE OBSERVED COUNTS AT 0 FOR A PARTITION OF THE
560 REM UNIT INTERVAL INTO 21 CELLS
570 FOR I = 1 TO 21
580 LET Y(I) = 0
590 NEXT I
600 REM OBTAIN FREQUENCY COUNTS FOR 1050 RANDOM NUMBERS
610 FOR I=1 TO P1
620 LET X = RND
630 LET R = INT(21*X) + 1
640 LET Y(R) = Y(R) + 1
650 NEXT I
660 REM COMPUTE CHI-SQ STATISTIC FOR EACH EXPERIMENT
670 LET S = 0
680 FOR I = 1 TO 21
690 LET S = S + (21*Y(I)*Y(I))
700 NEXT I
710 LET V1=(S/P1) - P1
720 REM TEST CHI-SQ STATISTIC FOR EACH EXPERIMENT
730 FOR C=1 TO 10
740 IF V1<L(C) THEN 760
750 NEXT C
760 LET Z(C)=Z(C)+1
770 REM PRINT EVALUATION
780 PRINT E,V1,C
790 NEXT E
800 PRINT
810 PRINT
820 PRINT
830 PRINT "SUMMARY EVALUATION STATISTICS"
840 PRINT
850 PRINT , "ACTUAL", "EXPECTED"
860 PRINT "CATEGORY", "COUNT", "COUNT"
870 LET T5=60
880 LET T(1)=.01*T5
890 LET T(2)=.04*T5
900 LET T(3)=.05*T5
910 LET T(4)=.15*T5
920 LET T(5)=.25*T5
930 LET T(6)=.25*T5
940 LET T(7)=.15*T5
950 LET T(8)=.05*T5
960 LET T(9)=.04*T5
970 LET T(10)=.01*T5
980 FOR C=1 TO 10
990 PRINT C,Z(C),T(C)
1000 NEXT C
1010 PRINT
```

```

1020 LET Y(1)=Z(1)+Z(2)+Z(3)
1030 LET Y(2)=Z(4)
1040 LET Y(3)=Z(5)
1050 LET Y(4)=Z(6)
1060 LET Y(5)=Z(7)
1070 LET Y(6)=Z(8)+Z(9)+Z(10)
1080 REM DO CHI-SQ ON CHI-SQ STATS FROM PRIMARY DATA - EXPECT
1090 REM DISTRIBUTION OF:
1100 REM   6 = 60*(.01+.04+.05)
1110 REM   9 = 60*.15
1120 REM  15 = 60*.25
1130 REM  15 = 60*.25
1140 REM   9 = 60*.15
1150 REM   6 = 60*(.05+.04+.01)
1160 LET F(1)=.1
1170 LET F(2)=.15
1180 LET F(3)=.25
1190 LET F(4)=.25
1200 LET F(5)=.15
1210 LET F(6)=.1
1220 LET S=0
1230 FOR I=1 TO 6
1240 LET W=(Y(I)*Y(I))/F(I)
1250 LET S=S+W
1260 NEXT I
1270 LET V1=(S/60)-60
1280 REM FOR 5 D.F., CHI-SQ CUTOFFS FOR EXTREMES OF 5% ARE 1.145476
1290 REM AND 11.0705
1300 PRINT
1310 PRINT "WILL CONSOLIDATE CATEGORIES 1-3 AND 8-10, SO THAT"
1320 PRINT "EXPECTED VALUE IN EACH IS >= 5, AS REQUIRED BY"
1330 PRINT "CHI-SQUARE TEST."
1340 PRINT
1350 PRINT "FOR FIVE D.F., SHOULD HAVE 1.145476 <= CHI-SQ <= 11.0705"
1360 PRINT "(5% TAIL AT EACH END.)"
1370 PRINT
1380 PRINT "ACTUAL CHI-SQ = ";V1
1390 PRINT
1400 IF V1<1.145476 THEN 1440
1410 IF V1>11.0705 THEN 1440
1420 PRINT " *** TEST PASSED *** "
1430 GOTO 1450
1440 PRINT " *** TEST FAILED *** "
1450 PRINT
1460 PRINT "                   END TEST"
1470 PRINT
1480 PRINT "END PROGRAM 133"
1490 END

```

---

PROGRAM FILE 133: CHI-SQUARE UNIFORMITY TEST FOR  
RND FUNCTION.  
ANSI STANDARD 8.4

SECTION 133.1: CHI-SQUARE UNIFORMITY TEST FOR  
RND FUNCTION.

THIS PROGRAM RUNS A SERIES OF 60 EXPERIMENTS OF 1050 SAMPLES TO DETERMINE IF THE RANDOM NUMBERS ARE EQUALLY DISTRIBUTED AMONG 21 EQUAL-SIZE INTERVALS BETWEEN 0 AND 1. CHI-SQUARE IS COMPUTED AND DISPLAYED FOR EACH OF THESE EXPERIMENTS. EACH CHI-SQUARE RESULT IS CATEGORIZED ACCORDING TO THE PERCENTILE RANGE INTO WHICH ITS VALUE FALLS:

CATEGORY	% RANGE	CATEGORY	% RANGE
1	0-1	6	50-75
2	1-5	7	75-90
3	5-10	8	90-95
4	10-25	9	95-99
5	25-50	10	99-100

THESE CATEGORICAL RESULTS ARE THEN EVALUATED AGAIN USING CHI-SQUARE (THE FIRST AND LAST THREE CATEGORIES ARE EACH CONSOLIDATED INTO ONE), AND IF THE DISTRIBUTION AMONG CATEGORIES IS SUFFICIENTLY IMPROBABLE (TOP OR BOTTOM 5%), THE TEST FAILS.

BEGIN TEST

EXPERIMENT	CHI-SQUARE	CATEGORY
1	13.4	4
2	14.44	4
3	10.2	2
4	22.36	6
5	17.76	5
6	19.52	6
7	20.08	6
8	19.48	6
9	16.2	5
10	20.88	6
11	26.04	7
12	19.08	5
13	14.6	4
14	29.56	8
15	21.36	6
16	32.92	9
17	12.36	3
18	21.2	6
19	14.92	4
20	10.92	3
21	15.4	4
22	16.16	5
23	11.16	3
24	23.24	6
25	15.64	5
26	16.56	5
27	13.76	4
28	17.36	5

29	16.92	5
30	17.52	5
31	15.12	4
32	26.76	7
33	20.16	6
34	9.36	2
35	6.39999	1
36	25.04	7
37	21.08	6
38	26.12	7
39	19.08	5
40	17.36	5
41	13.2	4
42	10.64	2
43	25.2	7
44	29.12	8
45	9.60001	2
46	13.08	4
47	15.84	5
48	15.04	4
49	14.48	4
50	8.28	2
51	20.64	6
52	23.32	6
53	29.12	8
54	14.92	4
55	22.96	6
56	16.88	5
57	17.8	5
58	9.60001	2
59	16.52	5
60	16.68	5

## SUMMARY EVALUATION STATISTICS

CATEGORY	ACTUAL	EXPECTED
	COUNT	COUNT
1	1	.6
2	6	2.4
3	3	3.
4	12	9
5	16	15
6	13	15
7	5	9
8	3	3.
9	1	2.4
10	0	.6

WILL CONSOLIDATE CATEGORIES 1-3 AND 8-10, SO THAT  
 EXPECTED VALUE IN EACH IS  $\geq 5$ , AS REQUIRED BY  
 CHI-SQUARE TEST.

FOR FIVE D.F., SHOULD HAVE 1.145476 <= CHI-SQ <= 11.0705  
(5% TAIL AT EACH END.)

ACTUAL CHI-SQ = 6.44444

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 133

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 134: KOMOLGOROV-SMIRNOV UNIFORMITY TEST FOR"
15 PRINT "          RND FUNCTION."
20 PRINT "          ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 134.1: KOMOLGOROV-SMIRNOV UNIFORMITY TEST FOR"
50 PRINT "          RND FUNCTION."
60 PRINT
70 PRINT "THIS PROGRAM RUNS 30 TESTS OF 1000 SAMPLES EACH TO CHECK"
80 PRINT "UNIFORMITY OF DISTRIBUTION OF RANDOM NUMBERS. KOMOLGOROV-"
90 PRINT "SMIRNOV (K-S) STATISTICS (K+ AND K-) ARE GENERATED FOR"
100 PRINT "EACH TEST. THESE TWO NUMBERS ARE A MEASURE OF THE GREATEST"
110 PRINT "DEVIATION, IN THE POSITIVE AND NEGATIVE DIRECTION, OF THE"
120 PRINT "ACCUMULATION OF THE 1000 NUMBERS FROM THE IDEAL DISTRIBUTION"
130 PRINT "OF F(X)=X, 0<=X<1."
140 PRINT
150 PRINT "THESE TWO SETS (K+ AND K-) OF 30 STATISTICS THEMSELVES HAVE"
160 PRINT "AN EXPECTED DISTRIBUTION OF F(X) = 1 - EXP(-2*X*X). THE"
170 PRINT "K-S TEST IS RUN AGAIN ON EACH OF THESE TWO SETS, AND IF"
180 PRINT "THE DEVIATION IS TOO GREAT (TOP OR BOTTOM 1%) FOR ANY ONE "
190 PRINT "OF THE RESULTING STATISTICS, THEN THE TEST FAILS."
200 PRINT
210 PRINT "          BEGIN TEST"
220 PRINT
230 DIM P(1200),Q(30),R(30)
240 REM SET #TRIALS PER EXPERIMENT
250 LET N=1000
260 REM SET #SLOTS USED IN ARRAY
270 LET N8=INT(1.1*N)
280 REM SET #EXPERIMENTS
290 LET N9=30
300 PRINT
310 REM BEGIN TEST LOOP
320 PRINT "TEST","K+","K-"
330 FOR L=1 TO N9
340 REM MARK SLOTS AS EMPTY
350 FOR I=1 TO N8
360 LET P(I)=3
370 NEXT I

```

```
380 REM FILL SLOTS WITH RANDOM VALUES, APPROXIMATELY IN ORDER
390 FOR I=1 TO N
400 LET X=RND
410 LET X1=INT(N8*X)
420 IF P(X1)<>3 THEN 460
430 LET P(X1)=X
440 GOTO 590
450 REM IF PRIMARY SLOT FILLED, SEARCH FOR NEAREST EMPTY SLOT
460 FOR J=1 TO N8
470 IF X1-J<1 THEN 510
480 IF P(X1-J)<>3 THEN 510
490 LET P(X1-J)=X
500 GOTO 590
510 IF X1+J>N8 THEN 550
520 IF P(X1+J)<>3 THEN 550
530 LET P(X1+J)=X
540 GOTO 590
550 NEXT J
560 PRINT "RED ALERT - NO SLOT FOUND",I,X
570 PRINT "FATAL ERROR IN PROGRAM ALGORITHM"
580 STOP
590 NEXT I
600 REM FIND GROUPS SEPARATED BY EMPTY SLOTS
605 REM ASSERT ALL ELEMENTS OF ONE GROUP < ALL ELEMENTS OF NEXT GROUP
607 REM THEREFORE, NEED SORT ONLY WITHIN GROUPS
610 LET I=1
620 FOR I8=I TO N8
630 IF P(I8)<>3 THEN 660
640 NEXT I8
650 GOTO 930
660 FOR I9=I8 TO N8
670 IF P(I9)=3 THEN 700
680 NEXT I9
690 LET I9=N8+1
700 IF I8>=I9-1 THEN 720
710 GOSUB 750
720 LET I=I9
730 GOTO 620
740 REM BUBBLE-SORT A GROUP
750 LET I5=I8
760 LET I6=I9-2
770 LET I7=1
780 LET A3=0
790 FOR J1=I5 TO I6 STEP I7
800 IF P(J1)<=P(J1+1) THEN 850
810 LET A3=9
820 LET W=P(J1)
830 LET P(J1)=P(J1+1)
840 LET P(J1+1)=W
850 NEXT J1
860 REM REVERSE DIRECTION OF SCAN
870 LET W=I5
880 LET I5=I6
```

```
890 LET I6=W
900 LET I7=0-I7
910 IF A3<>0 THEN 780
920 RETURN
930 REM GROUPS SORTED, DO COMPRESS
940 LET S1=0
950 FOR R1=1 TO N
960 LET S1=S1+1
970 IF P(S1)=3 THEN 960
980 LET P(R1)=P(S1)
990 NEXT R1
1000 LET M1=-1E38
1010 LET M2=-1E38
1020 REM FIND K+ AND K- ON PRIMARY DATA
1030 FOR I=1 TO N
1040 LET N1=(I/N)-P(I)
1050 LET N2=P(I)-((I-1)/N)
1060 IF N1<M1 THEN 1080
1070 LET M1=N1
1080 IF N2<M2 THEN 1100
1090 LET M2=N2
1100 NEXT I
1110 LET N5=SQR(N)
1120 LET M1=M1*N5
1130 LET M2=M2*N5
1140 LET Q(L)=M1
1150 LET R(L)=M2
1160 PRINT L,M1,M2
1170 NEXT L
1180 LET F=0
1190 PRINT
1200 PRINT "SUMMARY K-S STATISTICS FOR THE K+ VALUES"
1210 GOSUB 1430
1220 PRINT
1230 PRINT "SUMMARY K-S STATISTICS FOR THE K- VALUES"
1240 REM SET UP Q-ARRAY FOR SUBROUTINE
1250 FOR L=1 TO N9
1260 LET Q(L)=R(L)
1270 NEXT L
1280 GOSUB 1430
1290 PRINT
1300 REM BASED ON 30 TRIALS
1310 PRINT "ALL RESULTS SHOULD FALL BETWEEN .04354 AND 1.4801"
1320 PRINT
1330 IF F=0 THEN 1360
1340 PRINT "*** TEST FAILED ***"
1350 GOTO 1370
1360 PRINT "*** TEST PASSED ***"
1370 PRINT
1380 PRINT
1390 PRINT "END TEST"
```

```

1400 PRINT
1410 PRINT "END PROGRAM 134"
1420 STOP
1430 REM FIND K+ AND K- ON KS-STATS FROM PRIMARY DATA
1440 FOR I=1 TO N9-1
1450 FOR J=I+1 TO N9
1460 IF Q(I)<=Q(J) THEN 1500
1470 LET W=Q(I)
1480 LET Q(I)=Q(J)
1490 LET Q(J)=W
1500 NEXT J
1510 NEXT I
1520 LET M1=-1E38
1530 LET M2=-1E38
1550 FOR I=1 TO N9
1560 LET F1=1-EXP(-2*Q(I)*Q(I))
1570 LET N1=(I/N9) - F1
1580 LET N2=F1 - ((I-1)/N9)
1590 IF N1<M1 THEN 1610
1600 LET M1=N1
1610 IF N2<M2 THEN 1630
1620 LET M2=N2
1630 NEXT I
1640 LET N5=SQR(N9)
1650 LET M1=M1*N5
1660 LET M2=M2*N5
1670 PRINT "K+ = ";M1; " K- = ";M2
1680 PRINT
1690 REM VALUES BASED ON 30 TRIALS
1700 IF M1<.04354 THEN 1750
1710 IF M1>1.4801 THEN 1750
1720 IF M2<.04354 THEN 1750
1730 IF M2>1.4801 THEN 1750
1740 GOTO 1760
1750 LET F=F+1
1760 RETURN
1770 END

```

---

PROGRAM FILE 134: KOMOLGOROV-SMIRNOV UNIFORMITY TEST FOR  
RND FUNCTION.  
ANSI STANDARD 8.4

SECTION 134.1: KOMOLGOROV-SMIRNOV UNIFORMITY TEST FOR  
RND FUNCTION.

THIS PROGRAM RUNS 30 TESTS OF 1000 SAMPLES EACH TO CHECK  
UNIFORMITY OF DISTRIBUTION OF RANDOM NUMBERS. KOMOLGOROV-  
SMIRNOV (K-S) STATISTICS (K+ AND K-) ARE GENERATED FOR  
EACH TEST. THESE TWO NUMBERS ARE A MEASURE OF THE GREATEST  
DEVIATION, IN THE POSITIVE AND NEGATIVE DIRECTION, OF THE  
ACCUMULATION OF THE 1000 NUMBERS FROM THE IDEAL DISTRIBUTION  
OF  $F(X)=X$ ,  $0 \leq X < 1$ .

THESE TWO SETS (K+ AND K-) OF 30 STATISTICS THEMSELVES HAVE AN EXPECTED DISTRIBUTION OF  $F(X) = 1 - \text{EXP}(-2*X*X)$ . THE K-S TEST IS RUN AGAIN ON EACH OF THESE TWO SETS, AND IF THE DEVIATION IS TOO GREAT (TOP OR BOTTOM 1%) FOR ANY ONE OF THE RESULTING STATISTICS, THEN THE TEST FAILS.

BEGIN TEST

TEST	K+	K-
1	.189545	.503452
2	.467003	.540414
3	.714768	.372598
4	1.23148	7.02273E-2
5	.999851	.152188
6	.530693	.415809
7	.815667	.441467
8	.496637	.502196
9	.592283	.335427
10	.704123	.382455
11	1.09768	.186692
12	.268254	1.1519
13	.326314	.62775
14	.525816	.530669
15	.131413	1.31235
16	.593111	.498825
17	.780273	1.29285
18	.546175	.33704
19	.867775	.256057
20	.593898	.47154
21	.469585	.458471
22	.474387	.504202
23	.276197	.550667
24	1.19788	.106563
25	.220549	.803927
26	.681239	.477968
27	.358542	.985839
28	.556958	.701455
29	.500869	1.08348
30	.482619	.777734

SUMMARY K-S STATISTICS FOR THE K+ VALUES  
 $K+ = .879439$        $K- = .658183$

SUMMARY K-S STATISTICS FOR THE K- VALUES  
 $K+ = 1.34342$        $K- = .206827$

ALL RESULTS SHOULD FALL BETWEEN .04354 AND 1.4801

\*\*\* TEST PASSED \*\*\*

END TEST

END PROGRAM 134

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 135: SERIAL TEST FOR RANDOMNESS."  
20 PRINT "    ANSI STANDARD 8.4"  
30 PRINT  
40 PRINT "SECTION 135.1: SERIAL TEST FOR RANDOMNESS."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS WHETHER PAIRS OF ADJACENT NUMBERS FROM"  
70 PRINT "THE RND FUNCTION ARE EQUALLY DISTRIBUTED AMONG CATEGORIES"  
80 PRINT "BASED ON THEIR VALUES. THE TEST ASSIGNS A PAIR TO ONE OF"  
90 PRINT "121 CATEGORIES BASED ON WHICH 11TH OF THE SEGMENT FROM 0"  
100 PRINT "TO 1 EACH MEMBER OF THE PAIR FALLS WITHIN. CHI-SQUARE IS"  
110 PRINT "COMPUTED AND DISPLAYED AS THE RESULT."  
120 PRINT  
130 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"  
140 PRINT  
150 PRINT "                BEGIN TEST"  
160 DIM C(121)  
170 REM SET PARTITION OF 0-1 SEGMENT  
180 LET D=11  
190 REM FOR PAIRS OF NUMBERS, ARE #PARTITIONS^2 CATEGORIES  
200 LET D2=D*D  
210 FOR I=1 TO D2  
220 LET C(I)=0  
230 NEXT I  
240 REM EXPECT 20 IN EACH CATEGORY  
250 LET N=20*D2  
260 FOR N1=1 TO N  
270 LET X=RND  
280 LET Y=RND  
290 LET I=(D*INT(D*X))+INT(D*Y)+1  
300 LET C(I)=C(I)+1  
310 NEXT N1  
320 LET S=0  
330 PRINT  
340 PRINT "1ST OF PAIR","2ND OF PAIR","ACTUAL COUNT"  
350 PRINT "(EXPECTED COUNT = 20)"  
360 PRINT  
370 FOR I=1 TO D2  
380 LET I9=I-1  
390 LET I1=INT(I9/D)  
400 LET I2=I9 - (D*I1) + 1  
410 LET I1=I1+1  
420 PRINT I1,I2,C(I)  
430 LET S=S + (D2*C(I)*C(I))  
440 NEXT I  
450 LET V=(S/N) - N  
460 PRINT  
470 PRINT "FOR 120 D.F., SHOULD HAVE 95.70 <= CHI-SQ <= 146.57"  
480 PRINT "(5% TAIL AT EACH END)."  
490 PRINT
```

```

500 PRINT "ACTUAL CHI-SQ = ";V
510 PRINT
520 IF V<95.70 THEN 560
530 IF V>146.57 THEN 560
540 PRINT "*** INFORMATIVE TEST PASSED ***"
550 GOTO 570
560 PRINT "*** INFORMATIVE TEST FAILED ***"
570 REM DONE
580 PRINT
590 PRINT " END TEST"
600 PRINT
610 PRINT "END PROGRAM 135"
620 END

```

---

PROGRAM FILE 135: SERIAL TEST FOR RANDOMNESS.  
ANSI STANDARD 8.4

#### SECTION 135.1: SERIAL TEST FOR RANDOMNESS.

THIS PROGRAM TESTS WHETHER PAIRS OF ADJACENT NUMBERS FROM THE RND FUNCTION ARE EQUALLY DISTRIBUTED AMONG CATEGORIES BASED ON THEIR VALUES. THE TEST ASSIGNS A PAIR TO ONE OF 121 CATEGORIES BASED ON WHICH 11TH OF THE SEGMENT FROM 0 TO 1 EACH MEMBER OF THE PAIR FALLS WITHIN. CHI-SQUARE IS COMPUTED AND DISPLAYED AS THE RESULT.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

1ST OF PAIR	2ND OF PAIR	ACTUAL COUNT
(EXPECTED COUNT = 20)		

1	1	22
1	2	25
1	3	18
1	4	23
1	5	15
1	6	24
1	7	16
1	8	19
1	9	18
1	10	25
1	11	29
2	1	26
2	2	22
2	3	22
2	4	18
2	5	20
2	6	18
2	7	13
2	8	16
2	9	27

2	10	20
2	11	17
3	1	25
3	2	23
3	3	20
3	4	19
3	5	16
3	6	13
3	7	20
3	8	19
3	9	23
3	10	17
3	11	17
4	1	24
4	2	23
4	3	13
4	4	21
4	5	29
4	6	21
4	7	19
4	8	25
4	9	18
4	10	18
4	11	17
5	1	26
5	2	19
5	3	16
5	4	15
5	5	27
5	6	21
5	7	27
5	8	15
5	9	17
5	10	28
5	11	20
6	1	21
6	2	27
6	3	21
6	4	25
6	5	24
6	6	16
6	7	24
6	8	22
6	9	27
6	10	16
6	11	18
7	1	11
7	2	15
7	3	24
7	4	21
7	5	17
7	6	22
7	7	23
7	8	17
7	9	22

7	10	17
7	11	22
8	1	17
8	2	23
8	3	15
8	4	27
8	5	25
8	6	22
8	7	17
8	8	14
8	9	25
8	10	20
8	11	33
9	1	19
9	2	15
9	3	23
9	4	20
9	5	15
9	6	23
9	7	13
9	8	16
9	9	19
9	10	17
9	11	20
10	1	16
10	2	23
10	3	20
10	4	20
10	5	15
10	6	16
10	7	16
10	8	24
10	9	13
10	10	21
10	11	17
11	1	18
11	2	23
11	3	23
11	4	21
11	5	13
11	6	15
11	7	20
11	8	17
11	9	25
11	10	12
11	11	18

FOR 120 D.F., SHOULD HAVE  $95.70 \leq \text{CHI-SQ} \leq 146.57$   
(5% TAIL AT EACH END).

ACTUAL CHI-SQ = 110.8

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 135

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 136: GAP TEST FOR RND FUNCTION."  
20 PRINT "    ANSI STANDARD 8.4"  
30 PRINT  
40 PRINT "SECTION 136.1: GAP TEST FOR RND FUNCTION."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS THAT THE GAPS BETWEEN NUMBERS WITHIN"  
70 PRINT "A SPECIFIED INTERVAL ARE DISTRIBUTED AS EXPECTED FOR"  
80 PRINT "UNIFORMLY DISTRIBUTED RANDOM NUMBERS. CHI-SQUARE IS"  
90 PRINT "COMPUTED AND DISPLAYED AS THE RESULT."  
100 PRINT  
110 PRINT "***** THIS TEST IS INFORMATIVE ONLY. *****"  
120 PRINT  
130 PRINT "                BEGIN TEST"  
140 PRINT  
150 DIM C(121)  
160 REM SET BOUNDS FOR GAP TEST  
170 REM CONSOLIDATE ALL GAPS >=10  
180 LET T=10  
190 REM SET LOW END OF HIT INTERVAL  
200 LET A1=.03  
210 REM SET HIGH END OF HIT INTERVAL  
220 LET A2=.13  
230 REM SET SIZE OF HIT INTERVAL  
240 LET A3=A2-A1  
250 REM PROB. THAT GAP >= T  
260 LET P1=(1-A3)^T  
270 REM PROB. THAT GAP = T-1  
280 LET P2=A3*((1-A3)^(T-1))  
290 IF P1<P2 THEN 330  
300 LET P1=P2  
310 REM P1 IS PROBABILITY OF LEAST PROBABLE CATEGORY  
320 REM WANT EXPECTED NUMBER >= 10  
330 LET N=INT(10/P1)+1  
340 PRINT  
350 PRINT "WILL RUN ";N;" TRIALS, SO THAT EACH POSSIBILITY HAS"  
360 PRINT "EXPECTED VALUE >= 10."  
370 FOR I=0 TO T  
380 LET C(I)=0  
390 NEXT I  
400 PRINT  
410 PRINT  
420 PRINT "GAP SIZE","ACTUAL COUNT","EXPECTED COUNT"  
430 PRINT
```

```

440 FOR N1=1 TO N
450 LET R=0
460 LET X=RND
470 IF X<A1 THEN 500
480 IF X<A2 THEN 530
490 REM MISSED - KEEP TRYING
500 LET R=R+1
510 GOTO 460
520 REM HIT DESIGNATED INTERVAL
530 IF R<=T THEN 550
540 LET R=T
550 LET C(R)=C(R)+1
560 NEXT N1
570 FOR I=0 TO T
580 LET P9=(1-A3)^I
590 IF I=T THEN 630
600 LET P9=P9*A3
610 PRINT I,C(I),P9*N
620 GOTO 650
630 REM P9=PROB OF GAP OF SIZE I
640 PRINT ">=";I,C(I),P9*N
650 LET S=S+(C(I)*C(I)/P9)
660 NEXT I
670 LET V=(S/N)-N
680 PRINT
690 PRINT "FOR 10 D.F., SHOULD HAVE 3.9403 <= CHI-SQ <= 18.307"
700 PRINT "(5% TAIL AT EACH END.)"
710 PRINT
720 PRINT "ACTUAL CHI-SQ = ";V
730 PRINT
740 IF V<3.9403 THEN 780
750 IF V>18.307 THEN 780
760 PRINT "*** INFORMATIVE TEST PASSED ***"
770 GOTO 790
780 PRINT "*** INFORMATIVE TEST FAILED ***"
790 PRINT
800 PRINT "END TEST"
810 PRINT
820 PRINT "END PROGRAM 136"
830 END

```

---

PROGRAM FILE 136: GAP TEST FOR RND FUNCTION.  
ANSI STANDARD 8.4

#### SECTION 136.1: GAP TEST FOR RND FUNCTION.

THIS PROGRAM TESTS THAT THE GAPS BETWEEN NUMBERS WITHIN A SPECIFIED INTERVAL ARE DISTRIBUTED AS EXPECTED FOR UNIFORMLY DISTRIBUTED RANDOM NUMBERS. CHI-SQUARE IS COMPUTED AND DISPLAYED AS THE RESULT.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY. \*\*\*\*\*

BEGIN TEST

WILL RUN 259 TRIALS, SO THAT EACH POSSIBILITY HAS EXPECTED VALUE  $\geq 10$ .

GAP SIZE	ACTUAL COUNT	EXPECTED COUNT
----------	--------------	----------------

0	29	25.9
1	25	23.31
2	20	20.979
3	25	18.8811
4	9	16.993
5	4	15.2937
6	14	13.7643
7	9	12.3879
8	13	11.1491
9	8	10.0342
$\geq 10$	103	90.3077

FOR 10 D.F., SHOULD HAVE  $3.9403 \leq \text{CHI-SQ} \leq 18.307$   
(5% TAIL AT EACH END.)

ACTUAL CHI-SQ = 18.0558

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 136

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 137: POKER TEST FOR RND FUNCTION."
20 PRINT "    ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 137.1: POKER TEST FOR RND FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM CATEGORIZES FIXED-SIZE GROUPS OF RANDOM NUMBERS"
70 PRINT "BASED ON HOW MANY DIFFERENT TYPES OF NUMBER ARE IN EACH"
80 PRINT "GROUP. A TYPE IS DETERMINED BY WHICH FRACTION OF THE 0 - 1"
90 PRINT "SEGMENT THE VALUE OF THE NUMBER FALLS WITHIN. CHI-SQUARE"
100 PRINT "IS THEN COMPUTED AND DISPLAYED AS THE RESULT TO SEE IF"
110 PRINT "IT IS COMPATIBLE WITH THE EXPECTED DISTRIBUTION FOR UNIFORM"
120 PRINT "RANDOM NUMBERS."
130 PRINT
140 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"
150 PRINT
160 PRINT "                BEGIN TEST"
170 PRINT

```

```
180 DIM B(20)
190 REM SET SIZE OF GROUPS
200 LET K=5
210 REM SET #TYPES
220 LET D=7
230 IF D>=K THEN 260
240 PRINT "ALGORITHM INVALID FOR D<K. PROGRAM TERMINATING."
250 GOTO 820
260 LET N=INT(5*(D^(K-1)))
270 PRINT "RUNNING ";N;" TRIALS, SO THAT EACH POSSIBILITY HAS"
280 PRINT "EXPECTED VALUE >= 5, AS NEEDED FOR CHI-SQUARE TEST."
290 PRINT
300 FOR I=1 TO K
310 LET C(I)=0
320 NEXT I
330 FOR N1=1 TO N
340 LET R=0
350 REM MARK B-ARRAY AS NO TYPES HIT SO FAR
360 FOR I=1 TO D
370 LET B(I)=0
380 NEXT I
390 FOR I=1 TO K
400 LET J=INT(D*RND)+1
410 IF B(J)=1 THEN 450
420 REM GOT A NEW TYPE, MARK AND COUNT
430 LET R=R+1
440 LET B(J)=1
450 NEXT I
460 LET C(R)=C(R)+1
470 NEXT N1
480 LET S=0
490 REM SET TOTAL #POSSIBILITIES
500 LET D1=D^K
510 PRINT "#TYPES", "ACTUAL COUNT", "EXPECTED COUNT"
520 FOR I=1 TO K
530 LET F=1
540 FOR J=(D-I)+1 TO D
550 LET F=F*J
560 NEXT J
570 LET R=I
580 REM GET STIRLING NUMBER FOR K,R
590 REM RESULT IN A9
600 GOSUB 840
610 REM F=PROB OF I DIFFERENT TYPES
620 LET F=(F*A9)/D1
630 PRINT I,C(I),F*N
640 LET S=S+(C(I)*C(I))/F
650 NEXT I
660 LET V=(S/N) - N
670 REM CHI-SQ FOR 4 D.F.
680 PRINT
690 PRINT "FOR FOUR D.F., SHOULD HAVE .710721 <= CHI-SQ <= 9.48773"
700 PRINT "(5% TAIL AT EACH END.)"
710 PRINT
```

```
720 PRINT "ACTUAL CHI-SQ = ";V
730 PRINT
740 IF V<.710721 THEN 780
750 IF V>9.48773 THEN 780
760 PRINT "*** INFORMATIVE TEST PASSED ***"
770 GOTO 790
780 PRINT "*** INFORMATIVE TEST FAILED ***"
790 PRINT
800 PRINT "END TEST"
810 PRINT
820 PRINT "END PROGRAM 137"
830 STOP
840 REM GIVEN K,R, WHAT IS STIRLING NUMBER?
850 REM ANSWER RETURNED IN A9
860 IF R>K THEN 910
870 IF R=K THEN 930
880 IF R=1 THEN 930
890 IF R=0 THEN 910
900 GOTO 950
910 LET A9=0
920 GOTO 1130
930 LET A9=1
940 GOTO 1130
950 DIM A(20)
960 REM THIS LIMITS RANGE OF ROUTINE TO R<=20 (K UNCONSTRAINED)
970 FOR I8=1 TO 20
980 LET A(I8)=1
990 NEXT I8
1000 LET K9=K-R
1010 FOR K1=3 TO K
1020 LET M1=R
1030 IF R<K1-1 THEN 1050
1040 LET M1=K1-1
1050 LET M2=2
1060 IF 2>K1-K9 THEN 1080
1070 LET M2=K1-K9
1080 FOR R1=M1 TO M2 STEP -1
1090 LET A(R1)=R1*A(R1)+A(R1-1)
1100 NEXT R1
1110 NEXT K1
1120 LET A9=A(R)
1130 RETURN
1140 END
```

---

PROGRAM FILE 137: POKER TEST FOR RND FUNCTION.  
ANSI STANDARD 8.4

SECTION 137.1: POKER TEST FOR RND FUNCTION.

THIS PROGRAM CATEGORIZES FIXED-SIZE GROUPS OF RANDOM NUMBERS BASED ON HOW MANY DIFFERENT TYPES OF NUMBER ARE IN EACH GROUP. A TYPE IS DETERMINED BY WHICH FRACTION OF THE 0 - 1 SEGMENT THE VALUE OF THE NUMBER FALLS WITHIN. CHI-SQUARE IS THEN COMPUTED AND DISPLAYED AS THE RESULT TO SEE IF IT IS COMPATIBLE WITH THE EXPECTED DISTRIBUTION FOR UNIFORM RANDOM NUMBERS.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

RUNNING 12005 TRIALS, SO THAT EACH POSSIBILITY HAS EXPECTED VALUE  $\geq 5$ , AS NEEDED FOR CHI-SQUARE TEST.

#TYPES	ACTUAL COUNT	EXPECTED COUNT
1	6	5
2	455	450.
3	3796	3750
4	5967	6000
5	1781	1800.

FOR FOUR D.F., SHOULD HAVE  $.710721 \leq \text{CHI-SQ} \leq 9.48773$   
(5% TAIL AT EACH END.)

ACTUAL CHI-SQ = 1.2019

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 137

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 138: COUPON COLLECTOR TEST OF RND FUNCTION."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 138.1: COUPON COLLECTOR TEST OF RND FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM TESTS HOW MANY RANDOM NUMBERS MUST BE GENERATED"
70 PRINT "(UP TO A MAXIMUM) TO GET AT LEAST ONE OF EACH TYPE. A TYPE IS"
80 PRINT "DETERMINED BY WHICH FRACTION OF THE 0 - 1 SEGMENT THE VALUE"
90 PRINT "OF THE NUMBER FALLS WITHIN. CHI-SQUARE IS COMPUTED AND"
110 PRINT "DISPLAYED TO SEE WHETHER IT IS COMPATIBLE WITH THE"
120 PRINT "EXPECTED DISTRIBUTION FOR UNIFORM RANDOM NUMBERS."
130 PRINT
140 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"
150 PRINT

```

```
160 PRINT " BEGIN TEST"
170 DIM C(30)
180 DIM O(20)
190 REM SET #TRIALS
200 LET N=2000
210 REM SET #TYPES
220 LET D=5
230 REM SET MAX LENGTH
240 LET T=25
250 REM C-ARRAY COUNTS HOW MANY SEARCHES OF EACH LENGTH WERE DONE
260 FOR I=1 TO T
270 LET C(I)=0
280 NEXT I
290 FOR N1=1 TO N
300 REM R COUNTS LENGTH OF SEARCH
310 LET R=0
320 REM Q COUNTS NUMBER OF DISTINCT TYPES SO FAR
330 LET Q=0
340 REM O-ARRAY KEEPS TRACK OF WHICH TYPES HAVE BEEN HIT SO FAR
350 FOR I=0 TO D-1
360 LET O(I)=0
370 NEXT I
380 LET R=R+1
390 LET X=INT(RND*D)
400 IF O(X)=1 THEN 380
410 LET O(X)=1
420 LET Q=Q+1
430 IF Q<D THEN 380
440 IF R<=T THEN 460
450 LET R=T
460 LET C(R)=C(R)+1
470 NEXT N1
480 LET D1=1
490 FOR I=1 TO D
500 LET D1=D1*I
510 NEXT I
520 LET S=0
530 PRINT
540 PRINT "SEARCH LENGTH","ACTUAL COUNT","EXPECTED COUNT"
550 PRINT
560 FOR I=D TO T-1
570 LET K9=I-1
580 LET R9=D-1
590 REM GET STIRLING NUMBER FOR K9,R9 - RETURN RESULT IN A9
600 GOSUB 890
610 REM P=PROB OF SEARCH LENGTH I
620 LET P=(D1*A9)/(D^I)
630 PRINT I,C(I),N*P
640 LET S=S+(C(I)*C(I))/P
650 NEXT I
```

```
660 LET K9=T-1
670 LET R9=D
680 REM STIRLING NUMBER
690 GOSUB 890
700 LET P=1-((D1*A9)/(D^(T-1)))
710 PRINT ">=";T,C(T),N*P
720 LET S=S+(C(T)*C(T))/P
730 LET V=(S/N) - N
735 PRINT
740 PRINT "FOR 20 D.F., SHOULD HAVE 10.8508 <= CHI-SQ <= 31.4104"
750 PRINT "(5% TAIL AT EACH END.)"
760 PRINT
770 PRINT "ACTUAL CHI-SQ = ";V
780 PRINT
790 IF V<10.8508 THEN 830
800 IF V>31.4104 THEN 830
810 PRINT "*** INFORMATIVE TEST PASSED ***"
820 GOTO 840
830 PRINT "*** INFORMATIVE TEST FAILED ***"
840 PRINT
850 PRINT "END TEST"
860 PRINT
870 PRINT "END PROGRAM 138"
880 STOP
890 REM GIVEN K9,R9, COMPUTE STIRLING NUMBER AND RETURN IN A9
900 IF R9>K9 THEN 950
910 IF R9=K9 THEN 970
920 IF R9=1 THEN 970
930 IF R9=0 THEN 950
940 GOTO 990
950 LET A9=0
960 GOTO 1170
970 LET A9=1
980 GOTO 1170
990 DIM A(20)
1000 REM THIS LIMITS RANGE OF ROUTINE TO R9<=20 (K9 UNCONSTRAINED)
1010 FOR I8=1 TO 20
1020 LET A(I8)=1
1030 NEXT I8
1040 LET K8=K9-R9
1050 FOR K1=3 TO K9
1060 LET M1=R9
1070 IF R9<K1-1 THEN 1090
1080 LET M1=K1-1
1090 LET M2=2
1100 IF 2>K1-K8 THEN 1120
1110 LET M2=K1-K8
1120 FOR R1=M1 TO M2 STEP -1
1130 LET A(R1)=R1*A(R1)+A(R1-1)
1140 NEXT R1
1150 NEXT K1
1160 LET A9=A(R9)
1170 RETURN
1180 END
```

---

PROGRAM FILE 138: COUPON COLLECTOR TEST OF RND FUNCTION.  
ANSI STANDARD 8.4

SECTION 138.1: COUPON COLLECTOR TEST OF RND FUNCTION.

THIS PROGRAM TESTS HOW MANY RANDOM NUMBERS MUST BE GENERATED (UP TO A MAXIMUM) TO GET AT LEAST ONE OF EACH TYPE. A TYPE IS DETERMINED BY WHICH FRACTION OF THE 0 - 1 SEGMENT THE VALUE OF THE NUMBER FALLS WITHIN. CHI-SQUARE IS COMPUTED AND DISPLAYED TO SEE WHETHER IT IS COMPATIBLE WITH THE EXPECTED DISTRIBUTION FOR UNIFORM RANDOM NUMBERS.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

SEARCH LENGTH ACTUAL COUNT EXPECTED COUNT

5	72	76.8
6	161	153.6
7	192	199.68
8	239	215.04
9	209	209.019
10	204	190.956
11	157	167.633
12	133	143.278
13	109	120.226
14	100	99.5831
15	85	81.724
16	73	66.6202
17	56	54.0433
18	47	43.6839
19	39	35.2171
20	25	28.3359
21	22	22.7661
22	19	18.2713
23	15	14.6521
24	10	11.7427
>= 25	33	47.1288

FOR 20 D.F., SHOULD HAVE  $10.8508 \leq \text{CHI-SQ} \leq 31.4104$   
(5% TAIL AT EACH END.)

ACTUAL CHI-SQ = 13.3958

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 138

```
*****
*****  
10 PRINT "PROGRAM FILE 139: PERMUTATION TEST FOR THE RND FUNCTION."  
20 PRINT "      ANSI STANDARD 8.4"  
30 PRINT  
40 PRINT "SECTION 139.1: PERMUTATION TEST FOR THE RND FUNCTION."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS TO SEE THAT WITHIN FIXED SIZE GROUPS"  
70 PRINT "OF RANDOM NUMBERS, ALL ORDERS OF THE VALUES ARE EQUALLY"  
80 PRINT "PROBABLE. CHI-SQUARE IS COMPUTED AND DISPLAYED TO SEE"  
90 PRINT "WHETHER THE RESULTS ARE COMPATIBLE WITH THE EXPECTED"  
100 PRINT "EXPECTED DISTRIBUTION FOR UNIFORM RANDOM NUMBERS."  
110 PRINT  
120 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"  
130 PRINT  
140 PRINT "                      BEGIN TEST"  
150 PRINT  
160 DIM C(25)  
170 REM SET SIZE OF GROUP:  
180 LET T=4  
190 LET T1=1  
200 REM T1=T!  
210 FOR I=2 TO T  
220 LET T1=T1*I  
230 NEXT I  
240 REM SET #TRIALS  
250 LET N=10*T1  
260 FOR I=0 TO T1  
270 LET C(I)=0  
280 NEXT I  
290 FOR N1=1 TO N  
300 LET A=0  
310 LET K=1  
320 FOR I=1 TO T  
330 LET B(I)=RND  
340 NEXT I  
350 FOR R=T TO 1 STEP -1  
360 LET M1=-1  
370 REM FIND MAX OF REMAINDER OF THIS GROUP  
380 FOR I=1 TO R  
390 IF B(I)<M1 THEN 420  
400 LET M1=B(I)  
410 LET S=I  
420 NEXT I  
430 LET A=A+((S-1)*K)  
440 LET K=K*R  
450 LET B(S)=B(R)  
460 NEXT R
```

```
470 REM VARIABLE A IS AN INTEGER, 0<=A<=T1, UNIQUELY SPECIFYING
480 REM THE ORDERING OF THIS GROUP.
490 LET C(A)=C(A)+1
500 NEXT N1
510 LET S=0
520 PRINT
530 PRINT "ORDER ID","ACTUAL COUNT"
540 PRINT "(EXPECTED COUNT = 10 FOR ALL CASES.)"
550 PRINT
560 FOR I=0 TO T1-1
570 LET S=S+(C(I)*C(I)*T1)
580 PRINT I,C(I)
590 NEXT I
600 LET V=(S/N) - N
605 PRINT
610 PRINT "FOR 23 D.F., SHOULD HAVE 13.0905 <= CHI-SQ <= 35.1725"
620 PRINT "(5% TAIL AT EACH END.)"
630 PRINT
640 PRINT "ACTUAL CHI-SQ = ";V
650 PRINT
660 IF V<13.0905 THEN 700
670 IF V>35.1725 THEN 700
680 PRINT "*** INFORMATIVE TEST PASSED ***"
690 GOTO 710
700 PRINT "*** INFORMATIVE TEST FAILED ***"
710 PRINT
720 PRINT "END TEST"
730 PRINT
740 PRINT "END PROGRAM 139"
750 END
```

---

PROGRAM FILE 139: PERMUTATION TEST FOR THE RND FUNCTION.  
ANSI STANDARD 8.4

SECTION 139.1: PERMUTATION TEST FOR THE RND FUNCTION.

THIS PROGRAM TESTS TO SEE THAT WITHIN FIXED SIZE GROUPS  
OF RANDOM NUMBERS, ALL ORDERS OF THE VALUES ARE EQUALLY  
PROBABLE. CHI-SQUARE IS COMPUTED AND DISPLAYED TO SEE  
WHETHER THE RESULTS ARE COMPATIBLE WITH THE EXPECTED  
DISTRIBUTION FOR UNIFORM RANDOM NUMBERS.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

ORDER ID ACTUAL COUNT
(EXPECTED COUNT = 10 FOR ALL CASES.)

```
0      10
1      10
2      7
3      12
4      8
5      13
6      15
7      15
8      9
9      7
10     10
11     9
12     8
13     10
14     13
15     10
16     15
17     7
18     5
19     8
20     4
21     17
22     8
23     10
```

FOR 23 D.F., SHOULD HAVE 13.0905 <= CHI-SQ <= 35.1725  
(5% TAIL AT EACH END.)

ACTUAL CHI-SQ = 25.2

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 139

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 140: RUNS TEST FOR THE RND FUNCTION."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 140.1: RUNS TEST FOR THE RND FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM COUNTS HOW MANY RUNS OF A GIVEN LENGTH OCCUR"
70 PRINT "WITHIN A SERIES OF RANDOM NUMBERS. A RUN IS A SUBSEQUENCE"
80 PRINT "OF THE NUMBERS WITHIN WHICH THE VALUES ARE IN ASCENDING"
90 PRINT "ORDER. THE RESULTS ARE COMPARED WITH THE EXPECTED OUTCOME"
100 PRINT "FOR UNIFORM RANDOM NUMBERS, AND CHI-SQUARE IS COMPUTED"
110 PRINT "AND DISPLAYED. ALL RUNS OF LENGTH >= 6 ARE CONSOLIDATED"
120 PRINT "INTO ONE CATEGORY."
130 PRINT
140 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"
150 PRINT
```

```
160 PRINT " BEGIN TEST"
170 PRINT
180 LET T=6
190 LET N=5000
200 REM ZERO OUT TYPE COUNTS FOR RUNS
210 FOR I=1 TO T
220 LET C(I)=0
230 NEXT I
240 LET Y=RND
250 REM GATHER STATS ON N TRIALS
260 LET N1=1
270 REM COUNT LENGTH OF NEW RUN
280 LET K=1
290 LET X=RND
300 LET N1=N1+1
310 IF N1>N THEN 390
320 REM TO TEST RUNS DOWN, SIMPLY CHANGE "<" TO ">" IN NEXT LINE
330 IF X<Y THEN 390
340 REM RUN CONTINUES
350 LET K=K+1
360 LET Y=X
370 GOTO 290
380 REM END OF THIS RUN
390 LET Y=X
400 IF K<=T THEN 420
410 LET K=T
420 LET C(K)=C(K)+1
430 IF N1<=N THEN 280
440 PRINT "RUN LENGTH","ACTUAL COUNT","EXPECTED COUNT"
450 REM INITIALIZE FOR CALCULATION
460 REM CHI-SQUARE NOT COMPUTED SIMILARLY TO OTHER TESTS
470 FOR I=1 TO T
480 REM READ B-ARRAY VALUES
490 READ F
500 LET W(I)=C(I) - N*F
510 IF I<T THEN 530
520 PRINT ">=";
530 PRINT I,C(I),N*F
540 NEXT I
550 LET S=0
560 REM COMPUTE MATRIX SUM
570 FOR I=1 TO T
580 FOR J=1 TO T
590 REM READ A-MATRIX VALUES
600 READ F
610 LET S=S+(W(I)*W(J)*F)
620 NEXT J
630 NEXT I
640 LET V=S/N
650 PRINT
660 PRINT "FOR 6 D.F., SHOULD HAVE 1.63539 <= CHI-SQ <= 12.5916"
670 PRINT "(5% TAIL ON EACH END.)"
680 PRINT
```

```

690 PRINT "ACTUAL CHI-SQ = ";V
700 REM V SHOULD BE DISTRIBUTED AS CHI-SQ WITH 6 D.F.
710 PRINT
720 IF V<1.63539 THEN 760
730 IF V>12.5916 THEN 760
740 PRINT "*** INFORMATIVE TEST PASSED ***"
750 GOTO 770
760 PRINT "*** INFORMATIVE TEST FAILED ***"
770 PRINT
780 PRINT " END TEST"
790 PRINT
800 PRINT "END PROGRAM 140"
810 REM DATA FOR B-ARRAY:
820 DATA .166667,.208333,9.16667E-2,2.63889E-2,5.75397E-3,1.19048E-3
830 REM DATA FOR (CONCEPTUAL) A-MATRIX:
840 DATA 4529.4, 9044.9, 13568, 18091, 22615, 27892
850 DATA 9044.9, 18097, 27139, 36187, 45234, 55789
860 DATA 13568, 27139, 40721, 54281, 67852, 83685
870 DATA 18091, 36187, 54281, 72414, 90470, 111580
880 DATA 22615, 45234, 67852, 90470, 113262,139476
890 DATA 27892, 55789, 83685, 111580,139476,172860
900 END

```

---

PROGRAM FILE 140: RUNS TEST FOR THE RND FUNCTION.

ANSI STANDARD 8.4

SECTION 140.1: RUNS TEST FOR THE RND FUNCTION.

THIS PROGRAM COUNTS HOW MANY RUNS OF A GIVEN LENGTH OCCUR WITHIN A SERIES OF RANDOM NUMBERS. A RUN IS A SUBSEQUENCE OF THE NUMBERS WITHIN WHICH THE VALUES ARE IN ASCENDING ORDER. THE RESULTS ARE COMPARED WITH THE EXPECTED OUTCOME FOR UNIFORM RANDOM NUMBERS, AND CHI-SQUARE IS COMPUTED AND DISPLAYED. ALL RUNS OF LENGTH  $\geq 6$  ARE CONSOLIDATED INTO ONE CATEGORY.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

RUN LENGTH	ACTUAL COUNT	EXPECTED COUNT
1	836	833.335
2	998	1041.66
3	474	458.334
4	138	131.945
5	34	28.7699
$\geq 6$	4	5.9524

FOR 6 D.F., SHOULD HAVE  $1.63539 \leq \text{CHI-SQ} \leq 12.5916$   
(5% TAIL ON EACH END.)

ACTUAL CHI-SQ = 4.57648

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 140

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 141: MAXIMUM OF GROUP TEST OF RND FUNCTION."
20 PRINT "      ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 141.1: MAXIMUM OF GROUP TEST OF RND FUNCTION."
50 PRINT
60 PRINT "THIS PROGRAM FINDS THE MAXIMUM RANDOM NUMBER AMONG FIXED-"
70 PRINT "SIZE GROUPS AND DETERMINES WHETHER THE DISTRIBUTION OF THE"
80 PRINT "MAXIMA IS SUFFICIENTLY CLOSE TO THE EXPECTED DISTRIBUTION"
90 PRINT "FOR UNIFORM RANDOM NUMBERS OF F(X) = X^T, WHERE T IS THE"
100 PRINT "SIZE OF THE GROUPS. THE KOMOLGOROV-SMIRNOV TEST IS USED"
110 PRINT "TO MEASURE HOW WELL THE RESULTS MATCH THE EXPECTATION."
120 PRINT
130 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"
140 PRINT
150 PRINT "                  BEGIN TEST"
160 PRINT
170 DIM P(1000)
180 REM SET SIZE OF GROUP
190 LET T=3
200 REM SET NUMBER OF TRIALS
210 LET N=1000
220 FOR I=1 TO N
230 REM MARK AS EMPTY
240 LET P(I)=3
250 NEXT I
260 FOR I=1 TO N
270 LET M=0
280 FOR J=1 TO T
290 LET X=RND
300 IF X<M THEN 320
310 LET M=X
320 NEXT J
330 REM M IS MAX OF GROUP
340 REM X1 IS BEST GUESS FOR LOCATION OF M, BASED ON
350 REM EXPECTED DISTRIBUTION OF F(X) = M^T
360 LET X1=INT((M^T)*N) + 1
370 IF P(X1)<>3 THEN 420
380 REM PRIMARY SLOT EMPTY - FILL IN WITH M
390 LET P(X1)=M
400 GOTO 550
410 REM IF PRIMARY SLOT FILLED, SEARCH FOR NEAREST EMPTY SLOT
420 FOR J=1 TO N
430 IF X1-J<1 THEN 470
440 IF P(X1-J)<>3 THEN 470
450 LET P(X1-J)=M
```

```
460 GOTO 550
470 IF X1+J>N THEN 510
480 IF P(X1+J)<>3 THEN 510
490 LET P(X1+J)=M
500 GOTO 550
510 NEXT J
520 PRINT "RED ALERT - NO SLOT FOUND",I,M
530 PRINT "ERROR IN SORT ALGORITHM - PROGRAM TERMINATES."
540 GOTO 1050
550 NEXT I
560 REM BUBBLE-SORT TABLE VALUES
570 LET B1=1
580 LET B3=1
590 REM A3=0 IMPLIES NO SWAPPING DONE YET
600 LET A3=0
610 FOR I=B1 TO N-B1 STEP B3
620 IF P(I)<=P(I+1) THEN 670
630 LET W=P(I)
640 LET P(I)=P(I+1)
650 LET P(I+1)=W
660 LET A3=3
670 NEXT I
680 REM B1 AND B3 CAUSE THE BUBBLE-SORT TO ALTERNATELY SCAN THE ARRAY
690 REM FORWARDS AND BACKWARDS
700 LET B1=N-B1
710 LET B3=-B3
720 IF A3<>0 THEN 600
730 LET M1=-1E38
740 LET M2=-1E38
750 REM FIND K+ AND K- ON PRIMARY DATA
760 FOR I=1 TO N
770 LET P1=P(I)^T
780 LET N1=(I/N) - P1
790 LET N2=P1 - ((I-1)/N)
800 IF N1<M1 THEN 820
810 LET M1=N1
820 IF N2<M2 THEN 840
830 LET M2=N2
840 NEXT I
850 LET N5=SQR(N)
860 LET M1=M1*N5
870 LET M2=M2*N5
880 PRINT
890 REM EXPECTED DISTRIBUTION FOR K+ AND K- WHEN N IS LARGE.
900 LET P1=1 - (EXP(-2*M1*M1))
910 LET P2=1 - (EXP(-2*M2*M2))
920 PRINT "    K+ = ";M1,"    PERCENTILE FOR K+ = ";P1
930 PRINT "    K- = ";M2,"    PERCENTILE FOR K- = ";P2
940 PRINT
950 PRINT "PERCENTILES SHOULD BE BETWEEN .05 AND .95"
960 PRINT
```

```

970 IF P1<.05 THEN 1030
980 IF P1>.95 THEN 1030
990 IF P2<.05 THEN 1030
1000 IF P2>.95 THEN 1030
1010 PRINT "*** INFORMATIVE TEST PASSED ***"
1020 GOTO 1040
1030 PRINT "*** INFORMATIVE TEST FAILED ***"
1040 PRINT
1050 PRINT " END TEST"
1060 PRINT
1070 PRINT "END PROGRAM 141"
1080 END

```

---

PROGRAM FILE 141: MAXIMUM OF GROUP TEST OF RND FUNCTION.  
ANSI STANDARD 8.4

#### SECTION 141.1: MAXIMUM OF GROUP TEST OF RND FUNCTION.

THIS PROGRAM FINDS THE MAXIMUM RANDOM NUMBER AMONG FIXED-SIZE GROUPS AND DETERMINES WHETHER THE DISTRIBUTION OF THE MAXIMA IS SUFFICIENTLY CLOSE TO THE EXPECTED DISTRIBUTION FOR UNIFORM RANDOM NUMBERS OF  $F(X) = X^T$ , WHERE T IS THE SIZE OF THE GROUPS. THE KOMOLGOROV-SMIRNOV TEST IS USED TO MEASURE HOW WELL THE RESULTS MATCH THE EXPECTATION.

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

K+ = .315179	PERCENTILE FOR K+ = .180184
K- = .675644	PERCENTILE FOR K- = .598678

PERCENTILES SHOULD BE BETWEEN .05 AND .95

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 141

\*\*\*\*\*
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 142: SERIAL CORRELATION TEST OF RND FUNCTION."
20 PRINT " ANSI STANDARD 8.4"
30 PRINT
40 PRINT "SECTION 142.1: SERIAL CORRELATION TEST OF RND FUNCTION."
50 PRINT

```

```

60 PRINT "THIS PROGRAM COMPUTES THE SERIAL CORRELATION COEFFICIENT"
70 PRINT "BETWEEN PAIRS OF ADJACENT RANDOM NUMBERS. THE RESULT IS "
80 PRINT "COMPARED WITH THE EXPECTED VALUE FOR UNIFORM INDEPENDENT"
90 PRINT "RANDOM NUMBERS. (BECAUSE THE PAIR I AND I+1 IS NOT"
100 PRINT "COMPLETELY INDEPENDENT OF I+1, I+2, THE EXPECTED VALUE"
110 PRINT "OF THE COEFFICIENT IS NOT EXACTLY ZERO.)"
120 PRINT
130 PRINT "***** THIS TEST IS INFORMATIVE ONLY *****"
140 PRINT
150 PRINT "                                BEGIN TEST"
160 PRINT
170 LET N=1000
180 LET Y=RND
190 LET S1=Y
200 LET S2=Y*Y
210 LET S3=0
220 LET Z=Y
230 REM IN LOOP, X IS CURRENT RND, Y IS PREVIOUS RND.
240 FOR N1=1 TO N
250 LET X=RND
260 LET S1=S1+X
270 LET S2=S2+(X*X)
280 LET S3=S3+(X*Y)
290 LET Y=X
300 NEXT N1
310 LET S3=S3+(X*Z)
320 LET S4=S1*S1
330 LET C=((N*S3) - S4) / ((N*S2) - S4)
340 LET U=1/(1-N)
350 LET D=SQR((N*(N-3)) / (N+1)) / (N-1)
360 LET D=D+D
370 PRINT "WITH ABOUT 95% PROBABILITY, WE EXPECT CORRELATION TO"
380 PRINT "EQUAL ";U;" + OR - ";D
390 PRINT
400 PRINT "THAT IS, WE EXPECT ";U-D%;" <= CORRELATION <= ";U+D
410 PRINT
420 PRINT "ACTUAL CORRELATION = ";C
430 LET E1=ABS(C-U)
440 PRINT
450 IF E1<=D THEN 480
460 PRINT "*** INFORMATIVE TEST FAILED ***"
470 GOTO 490
480 PRINT "*** INFORMATIVE TEST PASSED ***"
490 PRINT
500 PRINT "                                END TEST"
510 PRINT
520 PRINT "END PROGRAM 142"
530 END

```

## SECTION 142.1: SERIAL CORRELATION TEST OF RND FUNCTION.

THIS PROGRAM COMPUTES THE SERIAL CORRELATION COEFFICIENT BETWEEN PAIRS OF ADJACENT RANDOM NUMBERS. THE RESULT IS COMPARED WITH THE EXPECTED VALUE FOR UNIFORM INDEPENDENT RANDOM NUMBERS. (BECAUSE THE PAIR I AND I+1 IS NOT COMPLETELY INDEPENDENT OF I+1, I+2, THE EXPECTED VALUE OF THE COEFFICIENT IS NOT EXACTLY ZERO.)

\*\*\*\*\* THIS TEST IS INFORMATIVE ONLY \*\*\*\*\*

BEGIN TEST

WITH ABOUT 95% PROBABILITY, WE EXPECT CORRELATION TO EQUAL -1.00100E-3 + OR - 6.31822E-2

THAT IS, WE EXPECT -6.41832E-2 <= CORRELATION <= 6.21812E-2

ACTUAL CORRELATION = -2.07958E-2

\*\*\* INFORMATIVE TEST PASSED \*\*\*

END TEST

END PROGRAM 142

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 143: ERROR - TWO ARGUMENTS IN LIST FOR"
15 PRINT "      SIN FUNCTION."
20 PRINT "      ANSI STANDARD 7.2, 7.4, 8.2, 8.4"
30 PRINT
40 PRINT "SECTION 143.1: ERROR - TWO ARGUMENTS IN LIST FOR"
45 PRINT "      SIN FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE SIN FUNCTION IS"
70 PRINT "INVOKED WITH TWO ARGUMENTS, INSTEAD OF ONE AS SPECIFIED"
80 PRINT "IN THE STANDARD."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT          BEGIN TEST"
230 PRINT

```

```
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF SIN."
250 LET A=SIN(1,1)
260 PRINT "PROCESSOR HAS EVALUATED SIN(1,1) = ";A
270 PRINT
280 PRINT "           END TEST"
290 PRINT
300 PRINT "END PROGRAM 143"
310 END
```

---

? , WAS SEEN WHERE ) WAS EXPECTED IN LINE 250

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 144: ERROR - TWO ARGUMENTS IN LIST FOR"
15 PRINT "           ATN FUNCTION."
20 PRINT "     ANSI STANDARD 7.2, 7.4, 8.2, 8.4"
30 PRINT
40 PRINT "SECTION 144.1: ERROR - TWO ARGUMENTS IN LIST FOR"
45 PRINT "           ATN FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE ATN FUNCTION IS"
70 PRINT "INVOKED WITH TWO ARGUMENTS, INSTEAD OF ONE AS SPECIFIED"
80 PRINT "IN THE STANDARD."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "           ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "           BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "           BEGIN TEST"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF ATN."
250 LET A=ATN(1,1)
260 PRINT "PROCESSOR HAS EVALUATED ATN(1,1) = ";A
270 PRINT
280 PRINT "           END TEST"
290 PRINT
300 PRINT "END PROGRAM 144"
310 END
```

---

? , WAS SEEN WHERE ) WAS EXPECTED IN LINE 250

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 145: ERROR - TWO ARGUMENTS IN LIST FOR"  
15 PRINT "      RND FUNCTION."  
20 PRINT "      ANSI STANDARD 7.2, 7.4, 8.2, 8.4"  
30 PRINT  
40 PRINT "SECTION 145.1: ERROR - TWO ARGUMENTS IN LIST FOR"  
45 PRINT "      RND FUNCTION."  
50 PRINT  
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE RND FUNCTION IS"  
70 PRINT "INVOKED WITH TWO ARGUMENTS, INSTEAD OF NONE AS SPECIFIED"  
80 PRINT "IN THE STANDARD."  
90 PRINT  
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
120 PRINT  
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
150 PRINT "        BY THE PROCESSOR, OR"  
160 PRINT  
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
180 PRINT  
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
200 PRINT "FOR DETAILED CRITERIA."  
210 PRINT  
220 PRINT "      BEGIN TEST"  
230 PRINT  
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF RND."  
250 LET A=RND(1,1)  
260 PRINT "PROCESSOR HAS EVALUATED RND(1,1) = ";A  
270 PRINT  
280 PRINT "      END TEST"  
290 PRINT  
300 PRINT "END PROGRAM 145"  
310 END
```

---

? , WAS SEEN WHERE ) WAS EXPECTED IN LINE 250

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 146: ERROR - ONE ARGUMENT IN LIST FOR"  
15 PRINT "      RND FUNCTION."  
20 PRINT "      ANSI STANDARD 7.2, 7.4, 8.2, 8.4"  
30 PRINT  
40 PRINT "SECTION 146.1: ERROR - ONE ARGUMENT IN LIST FOR"  
45 PRINT "      RND FUNCTION."  
50 PRINT
```

```

60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE RND FUNCTION IS"
70 PRINT "INVOOKED WITH ONE ARGUMENT, INSTEAD OF NONE AS SPECIFIED"
80 PRINT "IN THE STANDARD."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                BEGIN TEST"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF RND."
250 LET A=RND(0)
260 PRINT "PROCESSOR HAS EVALUATED RND(0) = ";A
270 PRINT
280 PRINT "                END TEST"
290 PRINT
300 PRINT "END PROGRAM 146"
310 END

```

---

? ( WAS SEEN WHERE OPERATOR WAS EXPECTED IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 147: ERROR - NULL ARGUMENT-LIST FOR"
15 PRINT "        INT FUNCTION."
20 PRINT "        ANSI STANDARD 7.2, 7.4, 8.2, 8.4"
30 PRINT
40 PRINT "SECTION 147.1: ERROR - NULL ARGUMENT-LIST FOR"
45 PRINT "        INT FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE INT FUNCTION IS"
70 PRINT "INVOOKED WITH NO ARGUMENTS, INSTEAD OF ONE AS SPECIFIED"
80 PRINT "IN THE STANDARD."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT

```

```
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "          BEGIN TEST"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF INT."
250 LET A=INT()
260 PRINT "PROCESSOR HAS EVALUATED INT() = ";A
270 PRINT
280 PRINT "          END TEST"
290 PRINT
300 PRINT "END PROGRAM 147"
310 END
```

---

? ILLEGAL FORMULA IN LINE 250

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 148: ERROR - MISSING ARGUMENT LIST FOR"
15 PRINT "          TAN FUNCTION."
20 PRINT "          ANSI STANDARD 7.2, 7.4, 8.2, 8.4"
30 PRINT
40 PRINT "SECTION 148.1: ERROR - MISSING ARGUMENT LIST FOR"
45 PRINT "          TAN FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE TAN FUNCTION IS"
70 PRINT "INVOKED WITHOUT AN ARGUMENT-LIST."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "          BEGIN TEST"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF TAN."
250 LET A=TAN
260 PRINT "PROCESSOR HAS EVALUATED TAN = ";A
270 PRINT
280 PRINT "          END TEST"
```

```
290 PRINT
300 PRINT "END PROGRAM 148"
310 END
```

---

```
? INCORRECT NUMBER OF ARGUMENTS IN LINE 250
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 149: ERROR - NULL ARGUMENT-LIST FOR"
15 PRINT "      RND FUNCTION."
20 PRINT "      ANSI STANDARD 7.2, 7.4, 8.2, 8.4"
30 PRINT
40 PRINT "SECTION 149.1: ERROR - NULL ARGUMENT-LIST FOR"
45 PRINT "      RND FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE RND FUNCTION IS"
70 PRINT "INVOKED WITH A NULL ARGUMENT-LIST."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "      BEGIN TEST"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF RND."
250 LET A=RND()
260 PRINT "PROCESSOR HAS EVALUATED RND() = ";A
270 PRINT
280 PRINT "      END TEST"
290 PRINT
300 PRINT "END PROGRAM 149"
310 END
```

---

```
? ( WAS SEEN WHERE OPERATOR WAS EXPECTED IN LINE 250
```

```
*****
*****
*****
```

10 PRINT "PROGRAM FILE 150: ERROR - USING A STRING AS AN ARGUMENT"  
 20 PRINT " FOR AN IMPLEMENTATION-SUPPLIED FUNCTION."  
 30 PRINT " ANSI STANDARD 7.2, 7.4, 8.2, 8.4"  
 40 PRINT  
 50 PRINT "SECTION 150.1: ERROR - USING A STRING AS AN ARGUMENT"  
 60 PRINT " FOR AN IMPLEMENTATION-SUPPLIED FUNCTION."  
 70 PRINT  
 80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE ATN FUNCTION IS"  
 90 PRINT "INVOKED WITH A CHARACTER STRING ARGUMENT, INSTEAD OF A"  
 100 PRINT "NUMERIC EXPRESSION."  
 110 PRINT  
 120 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
 130 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
 140 PRINT  
 150 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
 160 PRINT " ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
 170 PRINT " BY THE PROCESSOR, OR"  
 180 PRINT  
 190 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
 200 PRINT  
 210 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
 220 PRINT "FOR DETAILED CRITERIA."  
 230 PRINT  
 240 PRINT " BEGIN TEST"  
 260 PRINT  
 280 PRINT "SETTING X = 5"  
 290 PRINT "SETTING X\$ = 'ABC'"  
 300 PRINT  
 310 LET X=5  
 320 LET X\$="ABC"  
 330 PRINT "ABOUT TO ATTEMPT INVOCATION OF ATN(X\$)."  
 340 LET A=ATN(X\$)  
 350 PRINT "PROCESSOR HAS EVALUATED ATN(X\$) = ";A  
 360 PRINT  
 370 PRINT " END TEST"  
 380 PRINT  
 390 PRINT "END PROGRAM 150"  
 400 END

---

? ILLEGAL FORMULA IN LINE 340

```
*****
*****
```

10 PRINT "PROGRAM FILE 151: USER-DEFINED FUNCTIONS."  
 20 PRINT " ANSI STANDARD 16.2, 16.4, 7.2, 7.4"  
 30 PRINT

```
40 PRINT "THIS PROGRAM TESTS VARIOUS KINDS OF USER-DEFINED FUNCTIONS."
50 PRINT "NUMERIC EXPRESSIONS ARE USED IN TWO NEW CONTEXTS: BOTH AS"
55 PRINT "THE FUNCTION DEFINITION, AND ALSO AS ARGUMENTS IN THE"
60 PRINT "FUNCTION INVOCATIONS."
65 PRINT
70 PRINT "SECTION 151.1: THE EXPRESSION IN THE DEFINITION USING ONLY"
80 PRINT "          THE PARAMETER OF THE PARAMETER-LIST."
90 PRINT
100 PRINT "                                BEGIN TEST."
110 PRINT
120 DEF FNA(X)=X*X
130 DEF FNB(Y)=Y*Y*Y
140 LET A=5
150 LET B=4
160 LET C=3
170 LET E=0
180 LET X=FNA(12)
190 LET Y=FNA(A)
200 LET Z=FNA(2*A^3+4*B+C)
210 LET W=FNA(FNB(2)+B^3)
220 LET F=1
230 IF X=144 THEN 250
240 GOSUB 350
250 LET F=2
260 IF Y=25 THEN 280
270 GOSUB 350
280 LET F=3
285 LET C9=Z-72361
290 IF ABS(C9) < .1 THEN 310
300 GOSUB 350
310 LET F=4
315 LET C9=W-5184
320 IF ABS(C9) < .1 THEN 490
330 GOSUB 350
340 GOTO 490
350 REM SUBROUTINE TO PRINT ERROR MESSAGE
355 LET E=1
357 LET B$=""
360 ON F GOTO 370,390,410,430
370 LET A$="A CONSTANT"
380 GOTO 440
390 LET A$="A VARIABLE"
400 GOTO 440
410 LET A$="CONSTANTS AND"
415 LET B$=" VARIABLES"
420 GOTO 440
430 LET A$="ANOTHER FUNCTION"
440 PRINT "FUNCTION FAILED TO EXECUTE PROPERLY WHEN THE EXPRESSION"
450 PRINT "IN THE ARGUMENT-LIST FOR THE FUNCTION REFERENCE USED"
460 PRINT A$;B$;"."
470 PRINT
480 RETURN
```

```

490 IF E=0 THEN 520
500 PRINT "*** TEST FAILED. NOTE THE ABOVE REASONS ***"
510 GOTO 530
520 PRINT "*** TEST PASSED ***"
530 PRINT
540 PRINT " END TEST."
550 PRINT
560 PRINT "SECTION 151.2: THE EXPRESSION IN THE DEFINITION USING THE"
570 PRINT " PARAMETER OF THE PARAMETER-LIST AND OTHER VARIABLES."
590 PRINT
600 PRINT " BEGIN TEST."
610 PRINT
620 DEF FNC(A)=A*A-B+C
630 LET B=-15
640 LET C=-35
650 LET X=FNC(13)
660 IF X=149 THEN 690
670 PRINT "*** TEST FAILED ***"
680 GOTO 700
690 PRINT "*** TEST PASSED ***"
700 PRINT
710 PRINT " END TEST."
720 PRINT
730 PRINT "SECTION 151.3: THE EXPRESSION IN THE DEFINITION USING THE"
740 PRINT " PARAMETER OF THE PARAMETER-LIST AND CONSTANTS."
760 PRINT
770 PRINT " BEGIN TEST."
780 PRINT
790 DEF FND(Y)=Y^4-16
800 LET A=4
810 LET B=2
820 LET Y=FND(A^3-3*A^2*B+3*A*B^2-B^3)
825 LET C9=Y-4080
830 IF ABS(C9) < .1 THEN 860
840 PRINT "*** TEST FAILED ***"
850 GOTO 870
860 PRINT "*** TEST PASSED ***"
870 PRINT
880 PRINT " END TEST."
890 PRINT
900 PRINT "SECTION 151.4: THE EXPRESSION IN THE DEFINITION USING THE"
910 PRINT " PARAMETER OF THE PARAMETER-LIST AND A REFERENCE"
920 PRINT " TO ANOTHER FUNCTION."
930 PRINT
940 PRINT " BEGIN TEST."
950 PRINT
960 DEF FNE(Z)=Z^2-4
970 DEF FNF(Z)=Z*X+FNE(A^2+B)
980 LET A=9
990 LET B=19
1000 LET X=3
1010 LET Y=2
1020 LET Z=FNF(X^3-3*X^2*Y+3*X*Y^2-Y^3)
1025 LET C9=Z-9999
1030 IF ABS(C9) < .1 THEN 1060

```

```
1040 PRINT "*** TEST FAILED ***"
1050 GOTO 1070
1060 PRINT "*** TEST PASSED ***"
1070 PRINT
1080 PRINT "                                END TEST."
1090 PRINT
1100 PRINT "SECTION 151.5: THE EXPRESSION IN THE DEFINITION HAS A"
1110 PRINT "      CONSTANT VALUE."
1120 PRINT
1130 PRINT "                                BEGIN TEST."
1140 PRINT
1150 DEF FNM=123
1160 DEF FNN(D)=3*37
1170 LET A=FNM
1180 LET B=FNN(FNM+A)
1190 IF A<>123 THEN 1300
1200 IF B<>111 THEN 1300
1210 PRINT "*** TEST PASSED ***"
1220 GOTO 1310
1300 PRINT "*** TEST FAILED ***"
1310 PRINT
1320 PRINT "                                END TEST."
1330 PRINT
1500 PRINT "SECTION 151.6: THE PARAMETER OF A USER-DEFINED"
1510 PRINT "      FUNCTION IS LOCAL TO THE EXPRESSION OF THE"
1520 PRINT "      DEFINITION AND OTHER VARIABLES ARE GLOBAL."
1530 PRINT
1540 PRINT "                                BEGIN TEST."
1550 PRINT
1560 LET X=1
1570 LET Y=2
1580 LET Z=3
1590 LET P1=-100
1600 LET Q1=-100
1610 DEF FNG(P1)=3+P1+X
1620 DEF FNH(P1)=7+X+Y+P1+FNG(30.E-1+P1)+FNG(P1)+FNG(Y)
1630 LET X=4
1640 LET Y=5
1650 LET Z=6
1660 LET P1=7
1670 LET Q1=8
1680 DEF FNI(Q1)=Z+Q1*10.
1690 LET A=FNH(Q1+3)+FNG(10)+FNI(Q1+4)
1700 IF X<>4 THEN 1790
1710 IF Y<>5 THEN 1790
1720 IF Z<>6 THEN 1790
1730 IF P1<>7 THEN 1790
1740 IF Q1<>8 THEN 1790
1750 LET C=221-A
1760 IF ABS(C)>.01 THEN 1790
1770 PRINT "*** TEST PASSED ***"
1780 GOTO 1800
1790 PRINT "*** TEST FAILED ***"
1800 PRINT
1810 PRINT "                                END TEST."
```

```

1820 PRINT
1830 PRINT "SECTION 151.7: FULL GENERAL USE OF USER-DEFINED FUNCTIONS"
1840 PRINT "           IN NUMERIC EXPRESSIONS."
1850 PRINT
1860 PRINT "                                BEGIN TEST."
1870 PRINT
1880 DEF FNT=3
1890 DEF FNU(D)=100
1900 LET X=4
1910 LET Y=5
1920 LET Z=6
1930 LET Q1=8
1940 LET A=FNI(FNT)
1950 LET A=A/(FNG(FNH(Q1+FNT)/(FNI(Q1+4)-FNU(1E38))+FNG(FNG(-10)))-2)
1960 LET C9=A-3
1970 IF ABS(C9) < .01 THEN 2000
1980 PRINT "*** TEST FAILED ***"
1990 GOTO 2010
2000 PRINT "*** TEST PASSED ***"
2010 PRINT
2020 PRINT "                                END TEST."
2030 PRINT
2040 PRINT "END PROGRAM 151."
2050 END

```

---

PROGRAM FILE 151: USER-DEFINED FUNCTIONS.

ANSI STANDARD 16.2, 16.4, 7.2, 7.4

THIS PROGRAM TESTS VARIOUS KINDS OF USER-DEFINED FUNCTIONS.  
 NUMERIC EXPRESSIONS ARE USED IN TWO NEW CONTEXTS: BOTH AS  
 THE FUNCTION DEFINITION, AND ALSO AS ARGUMENTS IN THE  
 FUNCTION INVOCATIONS.

SECTION 151.1: THE EXPRESSION IN THE DEFINITION USING ONLY  
 THE PARAMETER OF THE PARAMETER-LIST.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 151.2: THE EXPRESSION IN THE DEFINITION USING THE  
 PARAMETER OF THE PARAMETER-LIST AND OTHER VARIABLES.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 151.3: THE EXPRESSION IN THE DEFINITION USING THE PARAMETER OF THE PARAMETER-LIST AND CONSTANTS.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 151.4: THE EXPRESSION IN THE DEFINITION USING THE PARAMETER OF THE PARAMETER-LIST AND A REFERENCE TO ANOTHER FUNCTION.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 151.5: THE EXPRESSION IN THE DEFINITION HAS A CONSTANT VALUE.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 151.6: THE PARAMETER OF A USER-DEFINED FUNCTION IS LOCAL TO THE EXPRESSION OF THE DEFINITION AND OTHER VARIABLES ARE GLOBAL.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 151.7: FULL GENERAL USE OF USER-DEFINED FUNCTIONS IN NUMERIC EXPRESSIONS.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 151.

```
*****  
10 PRINT "PROGRAM FILE 152: VALID NAMES FOR USER-DEFINED FUNCTIONS."  
20 PRINT "      ANSI STANDARD 16.2"  
30 PRINT  
40 PRINT "SECTION 152.1: VALID NAMES FOR USER-DEFINED FUNCTIONS."  
60 PRINT  
70 PRINT "FUNCTION NAMES ARE IN THE FORM: FN FOLLOWED BY A LETTER."  
80 PRINT  
90 PRINT "                                BEGIN TEST."  
100 PRINT  
110 DEF FNA(X)=X/7-7  
120 DEF FNB(X)=X/8-8  
130 DEF FNC(X)=X/9-9  
140 DEF FND(X)=X/10-10  
150 DEF FNE(X)=X/11-11  
160 DEF FNF(X)=X/12-12  
170 DEF FNG(X)=X/13-13  
180 DEF FNH(X)=X/14-14  
190 DEF FNI(X)=X/15-15  
200 DEF FNJ(X)=X/16-16  
210 DEF FNK(X)=X/17-17  
220 DEF FNL(X)=X/18-18  
230 DEF FNM(X)=X/19-19  
240 DEF FNN(X)=X/20-20  
250 DEF FNO(X)=X/21-21  
260 DEF FNP(X)=X/22-22  
270 DEF FNQ(X)=X/23-23  
280 DEF FNR(X)=X/24-24  
290 DEF FNS(X)=X/25-25  
300 DEF FNT(X)=X/26-26  
310 DEF FNU(X)=X/27-27  
320 DEF FNV(X)=X/28-28  
330 DEF FNW(X)=X/29-29  
340 DEF FNX(X)=X/30-30  
350 DEF FNY(X)=X/31-31  
360 DEF FNZ(X)=X/32-32  
370 DIM A(26)  
380 LET A(1)=FNA(77)  
390 LET A(2)=FNB(88)  
400 LET A(3)=FNC(99)  
410 LET A(4)=FND(110)  
420 LET A(5)=FNE(121)  
430 LET A(6)=FNF(132)  
440 LET A(7)=FNG(143)  
450 LET A(8)=FNH(154)  
460 LET A(9)=FNI(165)  
470 LET A(10)=FNJ(176)  
480 LET A(11)=FNK(187)  
490 LET A(12)=FNL(198)  
500 LET A(13)=FNM(209)  
510 LET A(14)=FNN(220)  
520 LET A(15)=FNO(231)
```

```
530 LET A(16)=FNP(242)
540 LET A(17)=FNQ(253)
550 LET A(18)=FNR(264)
560 LET A(19)=FNS(275)
570 LET A(20)=FNT(286)
580 LET A(21)=FNU(297)
590 LET A(22)=FNV(308)
600 LET A(23)=FNW(319)
610 LET A(24)=FNX(330)
620 LET A(25)=FNY(341)
630 LET A(26)=FNZ(352)
640 LET S=0
650 FOR I=1 TO 26
660 LET S=S+A(I)
670 NEXT I
680 IF S=-221 THEN 710
690 PRINT "*** TEST FAILED ***"
700 GOTO 720
710 PRINT "*** TEST PASSED ***"
720 PRINT
730 PRINT "END TEST."
740 PRINT
750 PRINT "END PROGRAM 152."
760 END
```

---

PROGRAM FILE 152: VALID NAMES FOR USER-DEFINED FUNCTIONS.  
ANSI STANDARD 16.2

SECTION 152.1: VALID NAMES FOR USER-DEFINED FUNCTIONS.

FUNCTION NAMES ARE IN THE FORM: FN FOLLOWED BY A LETTER.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 152.

---

---

---

```
10 PRINT "PROGRAM FILE 153: ERROR - SUPERFLUOUS ARGUMENT-LIST FOR "
15 PRINT "      USER-DEFINED FUNCTION."
20 PRINT "      ANSI STANDARD 16.4"
30 PRINT
40 PRINT "SECTION 153.1: ERROR - SUPERFLUOUS ARGUMENT-LIST FOR "
45 PRINT "      USER-DEFINED FUNCTION."
50 PRINT
```

```

60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"
70 PRINT "INVOKED WITH AN ARGUMENT-LIST, THOUGH NONE IS REQUIRED"
80 PRINT "IN THE DEF STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                BEGIN TEST"
225 DEF FNP=3.14159
228 PRINT
229 PRINT "DEFINITION OF FNP=3.14159"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF FNP."
250 LET A=FNP(0)
260 PRINT "PROCESSOR HAS EVALUATED FNP(0) = ";A
270 PRINT
280 PRINT "                END TEST"
290 PRINT
300 PRINT "END PROGRAM 153"
310 END

```

---

? INCORRECT NUMBER OF ARGUMENTS IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 154: ERROR - MISSING ARGUMENT-LIST FOR "
15 PRINT "        USER-DEFINED FUNCTION."
20 PRINT "        ANSI STANDARD 16.4"
30 PRINT
40 PRINT "SECTION 154.1: ERROR - MISSING ARGUMENT-LIST FOR "
45 PRINT "        USER-DEFINED FUNCTION."
50 PRINT
60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"
70 PRINT "INVOKED WITHOUT AN ARGUMENT-LIST, THOUGH ONE IS REQUIRED"
80 PRINT "IN THE DEF STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT

```

```

130 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "      BY THE PROCESSOR, OR"
160 PRINT
170 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "          BEGIN TEST"
225 DEF FND(R)=(R*180)/3.14159
228 PRINT
229 PRINT "DEFINITION OF FND(R)=(R*180)/3.14159"
230 PRINT
240 PRINT "ABOUT TO ATTEMPT INVOCATION OF FND."
250 LET A=FND
260 PRINT "PROCESSOR HAS EVALUATED FND = ";A
270 PRINT
280 PRINT "          END TEST"
290 PRINT
300 PRINT "END PROGRAM 154"
310 END

```

---

? INCORRECT NUMBER OF ARGUMENTS IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 155: ERROR - NULL ARGUMENT-LIST FOR"
20 PRINT "      USER-DEFINED FUNCTION."
30 PRINT "      ANSI STANDARD 7.2, 7.4, 16.2, 16.4"
40 PRINT
50 PRINT "SECTION 155.1: ERROR - NULL ARGUMENT-LIST FOR"
60 PRINT "      USER-DEFINED FUNCTION."
70 PRINT
80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"
90 PRINT "INVOKED WITH A NULL ARGUMENT-LIST."
100 PRINT
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
130 PRINT
140 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
150 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
160 PRINT "      BY THE PROCESSOR, OR"
170 PRINT
180 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
190 PRINT
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
210 PRINT "FOR DETAILED CRITERIA."
220 PRINT
230 PRINT "          BEGIN TEST"
240 PRINT

```

```
250 DEF FNP=3.14159
260 PRINT "DEFINITION OF FNP=3.14159"
270 PRINT
280 PRINT "ABOUT TO ATTEMPT INVOCATION OF FNP."
290 LET A=FNP()
300 PRINT "PROCESSOR HAS EVALUATED FNP() = ";A
310 PRINT
320 PRINT "           END TEST"
330 PRINT
340 PRINT "END PROGRAM 155"
350 END
```

---

? ILLEGAL FORMULA IN LINE 290

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 156: ERROR - EXCESS ARGUMENT IN LIST FOR"
20 PRINT "           USER-DEFINED FUNCTION."
30 PRINT "           ANSI STANDARD 16.4"
40 PRINT
50 PRINT "SECTION 156.1: ERROR - EXCESS ARGUMENT IN LIST FOR"
60 PRINT "           USER-DEFINED FUNCTION."
70 PRINT
80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A FUNCTION DEFINED"
90 PRINT "WITH ONE PARAMETER IS INVOKED WITH TWO ARGUMENTS."
100 PRINT
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
130 PRINT
140 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
150 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
160 PRINT "        BY THE PROCESSOR, OR"
170 PRINT
180 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
190 PRINT
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
210 PRINT "FOR DETAILED CRITERIA."
220 PRINT
230 PRINT "           BEGIN TEST."
240 PRINT
250 DEF FNA(X)=X+10
260 PRINT "DEFINITION OF FNA(X)=X+10"
270 PRINT
280 PRINT "ABOUT TO ATTEMPT INVOCATION OF FNA."
290 LET A=FNA(5,6)
300 PRINT "PROCESSOR HAS EVALUATED FNA(5,6) = ";A
310 PRINT
320 PRINT "           END TEST."
```

```
340 PRINT
350 PRINT "END PROGRAM 156."
360 END
```

---

```
? INCORRECT NUMBER OF ARGUMENTS IN LINE 290
```

```
*****  
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 157: ERROR - USER-DEFINED FUNCTION WITH TWO"
20 PRINT "          PARAMETERS."
30 PRINT "      ANSI STANDARD 16.2, 16.4, 7.2, 7.4"
40 PRINT
50 PRINT "SECTION 157.1: ERROR - USER-DEFINED FUNCTION WITH TWO"
60 PRINT "          PARAMETERS."
70 PRINT
80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A FUNCTION IS"
90 PRINT "DEFINED WITH TWO PARAMETERS AND INVOKED WITH TWO"
100 PRINT "ARGUMENTS."
110 PRINT
120 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
130 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
140 PRINT
150 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
160 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
170 PRINT "          BY THE PROCESSOR, OR"
180 PRINT
190 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
200 PRINT
210 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
220 PRINT "FOR DETAILED CRITERIA."
230 PRINT
240 PRINT "                                BEGIN TEST."
250 PRINT
260 DEF FNA(X,Y)=X+Y
270 PRINT "DEFINITION OF FNA(X,Y)=X+Y"
280 PRINT "SETTING X=1 AND Y=10"
290 PRINT
300 LET X=1
310 LET Y=10
320 PRINT "ABOUT TO ATTEMPT INVOCATION OF FNA(100,1000)."
330 LET A=FNA(100,1000)
340 PRINT "PROCESSOR HAS EVALUATED FNA(100,1000) = ";A
350 PRINT
360 PRINT "                                END TEST."
370 PRINT
380 PRINT "END PROGRAM 157."
390 END
```

---

```
? , WAS SEEN WHERE ) WAS EXPECTED IN LINE 260  
? , WAS SEEN WHERE ) WAS EXPECTED IN LINE 330
```

```
*****  
*****  
*****  
*****  
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 158: ERROR - USING A STRING AS AN ARGUMENT FOR"  
20 PRINT "          A USER-DEFINED FUNCTION."  
30 PRINT "      ANSI STANDARD 7.2, 7.4, 16.2, 16.4"  
40 PRINT  
50 PRINT "SECTION 158.1: ERROR - USING A STRING AS AN ARGUMENT FOR"  
60 PRINT "          A USER-DEFINED FUNCTION."  
70 PRINT  
80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"  
90 PRINT "INVOKED WITH A CHARACTER STRING ARGUMENT, INSTEAD OF A"  
100 PRINT "NUMERIC EXPRESSION."  
110 PRINT  
120 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
130 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
140 PRINT  
150 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
160 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
170 PRINT "          BY THE PROCESSOR, OR"  
180 PRINT  
190 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
200 PRINT  
210 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
220 PRINT "FOR DETAILED CRITERIA."  
230 PRINT  
240 PRINT "          BEGIN TEST"  
250 DEF FND(R)=R + 3  
260 PRINT  
270 PRINT "DEFINITION OF FND(R)=R + 3"  
280 PRINT "SETTING X = 5"  
290 PRINT "SETTING X$ = 'ABC'"  
300 PRINT  
310 LET X=5  
320 LET X$="ABC"  
330 PRINT "ABOUT TO ATTEMPT INVOCATION OF FND(X$)."  
340 LET A=FND(X$)  
350 PRINT "PROCESSOR HAS EVALUATED FND(X$) = ";A  
360 PRINT  
370 PRINT "          END TEST"  
380 PRINT  
390 PRINT "END PROGRAM 158"  
400 END
```

---

```
? ILLEGAL FORMULA IN LINE 340
```

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 159: ERROR - USING A STRING AS AN ARGUMENT AND"  
20 PRINT "      PARAMETER FOR A USER-DEFINED FUNCTION."  
30 PRINT "      ANSI STANDARD 7.2, 7.4, 16.2, 16.4"  
40 PRINT  
50 PRINT "SECTION 159.1: ERROR - USING A STRING AS AN ARGUMENT AND"  
60 PRINT "      PARAMETER FOR A USER-DEFINED FUNCTION."  
70 PRINT  
80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"  
90 PRINT "INVOKED WITH A CHARACTER STRING ARGUMENT AND PARAMETER,"  
100 PRINT "INSTEAD OF A NUMERIC EXPRESSION AND VARIABLE"  
110 PRINT  
120 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
130 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
140 PRINT  
150 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
160 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
170 PRINT "      BY THE PROCESSOR, OR"  
180 PRINT  
190 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
200 PRINT  
210 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
220 PRINT "FOR DETAILED CRITERIA."  
230 PRINT  
240 PRINT "      BEGIN TEST"  
250 DEF FND(R$)=R + 3  
260 PRINT  
270 PRINT "DEFINITION OF FND(R$)=R + 3"  
280 PRINT "SETTING R = 5"  
290 PRINT "SETTING R$ = 'ABC'"  
300 PRINT  
310 LET R=5  
320 LET R$="ABC"  
330 PRINT "ABOUT TO ATTEMPT INVOCATION OF FND(R$)."  
340 LET A=FND(R$)  
350 PRINT "PROCESSOR HAS EVALUATED FND(R$) = ";A  
360 PRINT  
370 PRINT "      END TEST"  
380 PRINT  
390 PRINT "END PROGRAM 159"  
400 END
```

? \$ WAS SEEN WHERE ) WAS EXPECTED IN LINE 250  
? ILLEGAL FORMULA IN LINE 340

```
*****
*****  

10 PRINT "PROGRAM FILE 160: ERROR - FUNCTION DEFINED MORE THAN ONCE."  

20 PRINT "      ANSI STANDARD 16.4"  

30 PRINT  

40 PRINT "SECTION 160.1: ERROR - FUNCTION DEFINED MORE THAN ONCE."  

50 PRINT  

60 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"  

70 PRINT "DEFINED MORE THAN ONCE IN THE SAME PROGRAM."  

80 PRINT  

90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  

100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  

110 PRINT  

120 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  

130 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  

140 PRINT "          BY THE PROCESSOR, OR"  

150 PRINT  

160 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  

170 PRINT  

180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  

190 PRINT "FOR DETAILED CRITERIA."  

200 PRINT  

210 PRINT "          BEGIN TEST"  

220 DEF FND(R)=R+10  

230 PRINT  

240 PRINT "1ST DEFINITION OF FND(R)=R+10"  

250 PRINT "2ND DEFINITION OF FND(R)=R+100"  

260 PRINT  

270 PRINT "ABOUT TO ATTEMPT INVOCATION OF FND(5)."  

280 LET A=FND(5)  

290 PRINT "PROCESSOR HAS EVALUATED FND(5) = ";A  

300 PRINT  

310 PRINT "          END TEST"  

320 PRINT  

330 PRINT "END PROGRAM 160"  

340 DEF FND(R)=R+100  

350 END
```

---

? FUNCTION DEFINED TWICE IN LINE 340

```
*****
*****  

10 PRINT "PROGRAM FILE 161: ERROR - REFERENCING A FUNCTION INSIDE"  

20 PRINT "      ITS OWN DEFINITION."  

30 PRINT "      ANSI STANDARD 16.4"  

40 PRINT  

50 PRINT "SECTION 161.1: ERROR - REFERENCING A FUNCTION INSIDE"  

60 PRINT "      ITS OWN DEFINITION."  

70 PRINT
```

```

80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN THE DEFINITION OF A"
90 PRINT "FUNCTION REFERS TO THAT SAME FUNCTION."
100 PRINT
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
130 PRINT
140 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
150 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
160 PRINT "        BY THE PROCESSOR, OR"
170 PRINT
180 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
190 PRINT
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
210 PRINT "FOR DETAILED CRITERIA."
220 PRINT
230 PRINT "                                BEGIN TEST."
240 PRINT
250 DEF FNA(X)=X/FNA(X-1)
260 PRINT "DEFINITION OF FNA(X) = X / FNA(X-1)"
270 PRINT
280 PRINT "ABOUT TO ATTEMPT INVOCATION OF FNA."
290 LET Z=FNA(5)
300 PRINT "PROCESSOR HAS EVALUATED FNA(5) = ";Z
310 PRINT
320 PRINT "                                END TEST."
330 PRINT
340 PRINT "END PROGRAM 161"
350 END

```

---

? SUBROUTINE OR FUNCTION CALLS ITSELF IN LINE 250

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 162: ERROR - REFERENCE TO FUNCTION PRECEDES"
20 PRINT "        ITS DEFINITION."
30 PRINT "        ANSI STANDARD 16.4"
40 PRINT
50 PRINT "SECTION 162.1: ERROR - REFERENCE TO FUNCTION PRECEDES"
60 PRINT "        ITS DEFINITION."
70 PRINT
80 PRINT "THIS SECTION TESTS WHAT HAPPENS WHEN A USER FUNCTION IS"
90 PRINT "INVOKED IN A LOWER-NUMBERED LINE THAN THE DEF STATEMENT"
100 PRINT "FOR THAT FUNCTION."
110 PRINT
120 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
130 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
140 PRINT
150 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCOMPANIED BY DOCUMENTATION"
160 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
170 PRINT "        BY THE PROCESSOR, OR"
180 PRINT

```

```
190 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
200 PRINT
210 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
220 PRINT "FOR DETAILED CRITERIA."
230 PRINT
240 PRINT "           BEGIN TEST"
250 PRINT
260 PRINT "(LATER) DEFINITION OF FND(R)=R+10"
270 PRINT
280 PRINT "ABOUT TO ATTEMPT INVOCATION OF FND(5)."
290 LET A=FND(5)
300 PRINT "PROCESSOR HAS EVALUATED FND(5) = ";A
310 PRINT
320 DEF FND(R)=R+10
330 PRINT "           END TEST"
340 PRINT
350 PRINT "END PROGRAM 162"
360 END
```

---

? REFERENCE TO FUNCTION NOT PRECEDED BY DEF IN LINE 290

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 163: ERROR - REFERENCE TO AN UNDEFINED FUNCTION."
20 PRINT "    ANSI STANDARD 16.4"
30 PRINT
40 PRINT "SECTION 163.1: ERROR - REFERENCE TO AN UNDEFINED FUNCTION."
50 PRINT
60 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
70 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
80 PRINT
90 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
100 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
110 PRINT "        BY THE PROCESSOR, OR"
120 PRINT
130 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
140 PRINT
150 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
160 PRINT "FOR DETAILED CRITERIA."
170 PRINT
180 PRINT "           BEGIN TEST."
190 PRINT
200 PRINT "ABOUT TO ATTEMPT INVOCATION OF FNA, WHICH IS UNDEFINED."
210 LET A=FNA(1)
220 PRINT "PROCESSOR HAS EVALUATED FNA(1) = ";A
230 PRINT
240 PRINT "           END TEST."
```

```
250 PRINT
260 PRINT "END PROGRAM 163"
270 END
```

---

```
? REFERENCE TO FUNCTION NOT PRECEDED BY DEF IN LINE 210
```

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 164: GENERAL USE OF NUMERIC EXPRESSIONS"
20 PRINT "           IN LET-STATEMENT."
30 PRINT "      ANSI STANDARD 6.2, 6.4, 7.2, 7.4, 8.2, 8.4, "
40 PRINT "           16.2, 16.4"
50 PRINT
60 PRINT "THIS PROGRAM EXERCISES THE FULL RANGE OF USES AND VARIETIES"
70 PRINT "OF NUMERIC EXPRESSIONS IN THE LET-STATEMENT. EXPRESSIONS ARE"
80 PRINT "COMPOSED OF ALL TYPES OF PRIMARIES AND ARE USED AS ARGUMENTS"
90 PRINT "OF FUNCTIONS AND AS SUBSCRIPTS."
100 PRINT
110 LET P=3.14159265
120 PRINT "SECTION 164.1: NUMERIC EXPRESSIONS CONTAINING"
130 PRINT "           SUBSCRIPTED VARIABLES AND FUNCTION REFERENCES."
150 PRINT
160 PRINT "THIS SECTION TESTS WHETHER SUBSCRIPTED VARIABLES AND"
170 PRINT "REFERENCES TO BOTH IMPLEMENTATION-SUPPLIED AND"
180 PRINT "USER-DEFINED FUNCTIONS CAN BE USED AS PRIMARIES IN"
190 PRINT "NUMERIC EXPRESSIONS."
200 GOSUB 7000
270 LET X=10*RND+3/RND-TAN(-.348E-2)
280 LET A=SIN(X)*SIN(X)+COS(X)^(-(-1.3+.5-1.2))
290 LET Y=1
300 GOSUB 9000
310 DIM F(20)
320 REM SET UP FIBONACCI SERIES IN F
330 LET F(1)=1
340 LET F(2)=1
350 FOR I=3 TO 20
360 LET J=I-1
370 LET K=I-2
380 LET F(I)=F(J)+F(K)
390 NEXT I
400 LET A=F(20)/F(19)/((1+SQR(5))/2)
410 LET Y=1
420 GOSUB 9000
430 FOR I=1 TO 7
440 FOR J=1 TO 7
450 LET Q(I,J)=I/J
460 NEXT J
470 NEXT I
```

```
480 LET X=P/10
490 LET A=-F(19)/F(20)/(-SIN(X))/Q(2,7)
500 LET Y=7
510 GOSUB 9000
520 REM DEF COMMON LOG
530 DEF FNC(X)=LOG(X)/LOG(10)
540 LET X=P/6
550 LET A=SIN(X)*FNC(1E36)/((F(20)+F(20))/F(19)-1)^FNC(100)*Q(5,6)
560 LET Y=3
570 GOSUB 9000
600 GOSUB 8000
660 PRINT "SECTION 164.2: NUMERIC EXPRESSIONS USED AS FUNCTION"
670 PRINT "          ARGUMENTS AND ARRAY SUBSCRIPTS."
680 PRINT
690 PRINT "THIS SECTION TESTS WHETHER NUMERIC EXPRESSIONS CAN BE"
700 PRINT "USED AS ARGUMENTS TO BOTH IMPLEMENTATION-SUPPLIED AND"
710 PRINT "USER-DEFINED FUNCTIONS AND ALSO AS SUBSCRIPTS."
750 GOSUB 7000
790 REM DEF ARCSIN OF X IN DEGREES
800 DEF FNA(X)=ATN(1/(1/X^2-1)^.5)/P*180
810 LET A=FNA(SQR(3/4))/2/2/3/5
820 LET Y=1
830 GOSUB 9000
840 LET A=Q(1+FNA(Q(F(4),ABS(-6)))/6,INT(SQR(F(+1+2^3)))))
850 LET Y=1.2
860 GOSUB 9000
870 LET X=RND
880 LET D=P/180
890 LET F(F(F(6))*Q(2,7)-TAN(P/4))=TAN(ATN(SIN(D*FNA(LOG(EXP(X))))))
900 LET A=F(5)
910 LET Y=X
920 GOSUB 9000
930 LET F(5)=5
1000 GOSUB 8000
1010 PRINT "SECTION 164.3: EXPRESSIONS USED AS SUBSCRIPTS ARE"
1015 PRINT "          ROUNDED TO NEAREST INTEGER."
1020 PRINT
1030 PRINT "THIS SECTION ENSURES THAT WHEN EXPRESSIONS ARE USED AS"
1040 PRINT "SUBSCRIPTS, THEY ARE ROUNDED TO THE NEAREST INTEGER."
1060 GOSUB 7000
1080 DIM V(5,11)
1090 FOR I=0 TO 5
1100 FOR J=0 TO 11
1110 LET V(I,J)=12*I+J
1120 NEXT J
1130 NEXT I
1160 LET A=V(FNC(316)-.1,FNC(ABS(-316))+10E-2)
1170 LET Y=27
1180 GOSUB 9000
1190 LET A=V(Q(3,7),Q(4,7))
1200 LET Y=1
1210 GOSUB 9000
1220 LET A=V(FNC(1E-3)+2.51,13*7/9+1)
1230 LET Y=11
1240 GOSUB 9000
```

```
1250 LET A=1+V(LOG(1.0)-.4,.6-EXP(0))
1260 LET Y=1
1270 GOSUB 9000
1280 LET A=V(F(5)+.49,F(7)-Q(5,3))
1290 LET Y=71
1300 GOSUB 9000
1700 GOSUB 8000
6000 PRINT "END PROGRAM 164"
6010 STOP
7000 REM SUBROUTINE TO BEGIN SECTION
7010 PRINT
7020 PRINT "               BEGIN TEST."
7030 PRINT
7040 PRINT "CASE #","SHOULD BE","ACTUAL","OUTCOME"
7050 PRINT
7060 LET E1=0
7070 LET C1=0
7080 RETURN
8000 REM SUBROUTINE TO END SECTION
8010 PRINT
8020 IF E1=0 THEN 8050
8030 PRINT "*** TEST FAILED IN ";E1;" CASE(S) ***"
8040 GOTO 8060
8050 PRINT "*** TEST PASSED ***"
8060 PRINT
8070 PRINT "               END TEST."
8080 PRINT
8090 RETURN
9000 REM HANDLE REPORT
9010 REM THIS SUBROUTINE ADOPTS A VERY LOOSE CRITERION OF CORRECTNESS
9020 REM (RELATIVE ERROR < .001). ITS PURPOSE ISN'T TO MEASURE ACCURACY
9030 REM BUT ONLY TO ASSURE THAT THE SEMANTICS OF THE EXPRESSION
9040 REM HAVE BEEN CORRECTLY IMPLEMENTED.
9050 LET C1=C1+1
9060 PRINT C1,Y,A,
9070 LET P$="FAIL"
9080 LET M=0.001
9090 LET T=(A-Y)/Y
9100 IF T>M THEN 9150
9110 LET T=-T
9120 IF T>M THEN 9150
9130 LET P$=" OK "
9140 GOTO 9160
9150 LET E1=E1+1
9160 PRINT P$
9170 RETURN
9180 END
```

---

PROGRAM FILE 164: GENERAL USE OF NUMERIC EXPRESSIONS  
IN LET-STATEMENT.

ANSI STANDARD 6.2, 6.4, 7.2, 7.4, 8.2, 8.4,  
16.2, 16.4

THIS PROGRAM EXERCISES THE FULL RANGE OF USES AND VARIETIES OF NUMERIC EXPRESSIONS IN THE LET-STATEMENT. EXPRESSIONS ARE COMPOSED OF ALL TYPES OF PRIMARIES AND ARE USED AS ARGUMENTS OF FUNCTIONS AND AS SUBSCRIPTS.

SECTION 164.1: NUMERIC EXPRESSIONS CONTAINING  
SUBSCRIPTED VARIABLES AND FUNCTION REFERENCES.

THIS SECTION TESTS WHETHER SUBSCRIPTED VARIABLES AND REFERENCES TO BOTH IMPLEMENTATION-SUPPLIED AND USER-DEFINED FUNCTIONS CAN BE USED AS PRIMARIES IN NUMERIC EXPRESSIONS.

BEGIN TEST.

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	1	1.	OK
2	1	1.	OK
3	7	7.	OK
4	3	3.	OK

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 164.2: NUMERIC EXPRESSIONS USED AS FUNCTION ARGUMENTS AND ARRAY SUBSCRIPTS.

THIS SECTION TESTS WHETHER NUMERIC EXPRESSIONS CAN BE USED AS ARGUMENTS TO BOTH IMPLEMENTATION-SUPPLIED AND USER-DEFINED FUNCTIONS AND ALSO AS SUBSCRIPTS.

BEGIN TEST.

CASE #	SHOULD BE	ACTUAL	OUTCOME
1	1	1.	OK
2	1.2	1.2	OK
3	0.29751	0.29751	OK

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 164.3: EXPRESSIONS USED AS SUBSCRIPTS ARE ROUNDED TO NEAREST INTEGER.

THIS SECTION ENSURES THAT WHEN EXPRESSIONS ARE USED AS SUBSCRIPTS, THEY ARE ROUNDED TO THE NEAREST INTEGER.

BEGIN TEST.

CASE #	SHOULD BE	ACTUAL	OUTCOME
--------	-----------	--------	---------

1	27	27	OK
2	1	1	OK
3	11	11	OK
4	1	1	OK
5	71	71	OK

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 164

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 165: COMPOUND EXPRESSIONS AND PRINT."
20 PRINT "      ANSI STANDARD 7.2, 7.4, 12.2, 12.4"
30 PRINT
40 PRINT "THIS PROGRAM TESTS THE USE OF NUMERIC EXPRESSIONS IN THE"
50 PRINT "CONTEXT OF THE PRINT STATEMENT."
60 PRINT
70 PRINT "SECTION 165.1: NUMERIC EXPRESSIONS AS PRINT ITEMS."
80 PRINT
90 PRINT "                                BEGIN TEST."
100 PRINT
110 DEF FNA(X)=X^2+1
120 LET A1=0.5
130 LET B1=-.25
140 LET C1=16.0
150 LET D1=-4.0
152 FOR I=1 TO 7
153 FOR J=1 TO 7
155 LET Q(I,J)=I/J
156 NEXT J
157 NEXT I
160 PRINT "EXPECTED","CALCULATED"
170 PRINT " VALUES", " VALUES"
180 PRINT
190 PRINT -.25 ,3*A1+7*B1
200 PRINT 6.5 ,ABS(A1+1.0-(C1+D1)+0.5*8.0)
210 PRINT 16.4794255,SIN(A1^2+ABS(B1))+C1
220 PRINT 1.54192554,FNA(B1+A1)+SIN(A1)
225 PRINT 5.24288566E-22,EXP(-Q(7,1)/Q(1,7))
230 PRINT
240 PRINT "*** TEST PASSED IF THE EXPECTED VALUES AND THE CALCULATED"
250 PRINT "      VALUES ARE NUMERICALLY EQUAL OR APPROXIMATELY SO ***"
260 PRINT
270 PRINT "                                END TEST."
```

```

280 PRINT
290 PRINT "SECTION 165.2: NUMERIC EXPRESSIONS AS ARGUMENTS"
295 PRINT "          TO TAB-CALLS."
300 PRINT
310 PRINT "                                BEGIN TEST."
320 PRINT
330 DEF FNB(X)=X^3-8
340 LET A1=2.75
350 LET B1=1.5
360 LET C1=3.1
370 PRINT "0000000001111111122222222333333334444444445";
380 PRINT "55555555566666666777"
390 PRINT "1234567890123456789012345678901234567890";
400 PRINT "1234567890123456789012"
410 PRINT TAB(2*C1-A1); "A"; TAB(2*ABS(-B1)+A1-Q(1,5)); "B";
420 PRINT TAB(FNB(A1+B1)-.15); "C"
430 PRINT
440 PRINT "*** TEST PASSED IF THE CHARACTERS 'A', 'B' AND 'C' ARE"
450 PRINT "      PRINTED IN COLUMNS 3, 6 AND 69 RESPECTIVELY ***"
460 PRINT
470 PRINT "                                END TEST."
480 PRINT
490 PRINT "END PROGRAM 165"
500 END

```

---

PROGRAM FILE 165: COMPOUND EXPRESSIONS AND PRINT.  
ANSI STANDARD 7.2, 7.4, 12.2, 12.4

THIS PROGRAM TESTS THE USE OF NUMERIC EXPRESSIONS IN THE CONTEXT OF THE PRINT STATEMENT.

SECTION 165.1: NUMERIC EXPRESSIONS AS PRINT ITEMS.

BEGIN TEST.

EXPECTED VALUES	CALCULATED VALUES
-0.25	-0.25
6.5	6.5
16.4794	16.4794
1.54193	1.54193
5.24289E-22	5.24289E-22

\*\*\* TEST PASSED IF THE EXPECTED VALUES AND THE CALCULATED VALUES ARE NUMERICALLY EQUAL OR APPROXIMATELY SO \*\*\*

END TEST.

SECTION 165.2: NUMERIC EXPRESSIONS AS ARGUMENTS  
TO TAB-CALLS.

BEGIN TEST.

0000000001111111122222222333333334444444455555555666666666777  
12345678901234567890123456789012345678901234567890123456789012  
A B C

\*\*\* TEST PASSED IF THE CHARACTERS 'A', 'B' AND 'C' ARE  
PRINTED IN COLUMNS 3, 6 AND 69 RESPECTIVELY \*\*\*

END TEST.

END PROGRAM 165

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 166: COMPOUND EXPRESSIONS USED WITH CONTROL"
20 PRINT "      STATEMENTS AND FOR-STATEMENTS."
30 PRINT "      ANSI STANDARD 7.2, 7.4, 10.2, 10.4, 11.2, 11.4"
40 PRINT
50 PRINT "THIS PROGRAM TESTS THE USE OF NUMERIC EXPRESSIONS WITHIN"
60 PRINT "THE CONTEXT OF THE CONTROL STATEMENTS IF-THEN AND ON-GOTO,"
70 PRINT "AND ALSO WITHIN THE FOR-STATEMENT."
80 PRINT
90 PRINT "SECTION 166.1: NUMERIC EXPRESSIONS AND"
100 PRINT "      THE IF-THEN-STATEMENT."
110 PRINT
120 PRINT "                  BEGIN TEST."
130 PRINT
140 DEF FNA(X)=X^2+2*X+1
150 DEF FNC(X1)=LOG(X1)/LOG(10)
160 LET P1=3.14159265
170 LET R(0)=0
180 LET R(1)=1
190 FOR I=2 TO 10
200 LET R(I)=R(I-1)+R(I-2)
210 NEXT I
220 LET A1=3.5
230 LET B1=1.625
240 LET C1=.815
250 LET D1=-4.5
260 LET F=0
270 IF A1+B1<5.13 THEN 300
280 LET K$=" 1ST "
290 GOSUB 520
300 IF -.99>A1-ABS(D1) THEN 330
310 LET K$=" 2ND "
320 GOSUB 520
330 IF INT(SQR(ABS(D1)))=2 THEN 360
340 LET K$=" 3RD "
350 GOSUB 520

```

```
360 IF FNA(A1)+5<=ABS(D1)+A1^3 THEN 390
370 LET K$=" 4TH "
380 GOSUB 520
390 IF EXP(A1)+D1>=(A1-.36)^2+D1 THEN 420
400 LET K$=" 5TH "
410 GOSUB 520
420 IF TAN(C1)+A1<>ATN(C1)+A1 THEN 450
430 LET K$=" 6TH "
440 GOSUB 520
450 IF -FNC(SQR(10^(SIN(P1/3)^2*R(6)))) < (R(5)-R(9))/10-.09 THEN 480
460 LET K$=" 7TH "
470 GOSUB 520
480 IF -FNC(SQR(10^(SIN(P1/3)^2*R(6)))) > (R(5)-R(9))/10-.11 THEN 510
490 LET K$=" 8TH "
500 GOSUB 520
510 GOTO 580
520 REM SUBROUTINE TO HANDLE ERRORS
530 PRINT "EVALUATION OF THE RELATION BETWEEN THE";K$;"PAIR OF"
540 PRINT "EXPRESSIONS FAILED."
550 LET F=F+1
560 PRINT
570 RETURN
580 IF F=0 THEN 610
590 PRINT "**** TEST FAILED IN ";F;" CASE(S) ****"
600 GOTO 620
610 PRINT "**** TEST PASSED ****"
620 PRINT
630 PRINT " END TEST."
640 PRINT
660 PRINT "SECTION 166.2: NUMERIC EXPRESSIONS AND"
670 PRINT " THE ON-GOTO-STATEMENT."
680 PRINT
690 PRINT " BEGIN TEST."
700 PRINT
710 LET F=0
720 REM ON-GOTO SHOULD BRANCH TO 1ST, 4TH, AND 3RD BRANCH POINT FOR
730 REM X = 1, 3, AND 5 RESPECTIVELY.
740 FOR X=1 TO 5 STEP 2
750 ON -(X*20*X+(R(8)-5*R(9))*X+SQR(11449))/40 GOTO 870,910,830,790
760 PRINT "**** TEST FAILURE: ON GOTO FELL THROUGH FOR X = ";X
770 LET F=F+1
780 GOTO 980
790 REM 4TH ROUTINE
800 IF X=3 THEN 980
810 LET K$="4TH"
820 GOTO 940
830 REM 3RD ROUTINE
840 IF X=5 THEN 980
850 LET K$="3RD"
860 GOTO 940
870 REM 1ST ROUTINE
880 IF X=1 THEN 980
890 LET K$="1ST"
900 GOTO 940
910 REM 2ND ROUTINE
```

```

920 LET K$="2ND"
930 GOTO 940
940 PRINT "*** TEST FAILURE: ON-GOTO BRANCHED TO ";K$;" ROUTINE"
950 PRINT " FOR X = ";X
960 PRINT
970 LET F=F+1
980 NEXT X
990 IF F=0 THEN 1020
1000 PRINT "*** TEST FAILED IN ";F;" CASE(S) ***"
1010 GOTO 1030
1020 PRINT "*** TEST PASSED ***"
1030 PRINT
1040 PRINT " END TEST."
1050 PRINT
1060 PRINT "SECTION 166.3: NUMERIC EXPRESSIONS AND"
1070 PRINT " THE FOR-NEXT-STATEMENT."
1080 PRINT
1090 PRINT " BEGIN TEST."
1100 PRINT
1110 REM CORRECT EXECUTION PARAMETERS:
1120 REM CTL. VAR.      FROM      TO      STEP      NO. LOOPS
1130 REM   I0            3         6         1          4
1140 REM   I1            10        4        -1          7
1150 REM   I2           228       1000      100         8
1160 REM   I3            2        -20       -2         12
1200 LET N1=3
1210 LET M1=6
1220 LET N(M1)=10
1230 LET M(M1)=4
1240 LET O=-1
1250 LET K=0
1260 LET J0=3
1270 FOR I0=N1 TO M1
1280 LET J1=10
1290 FOR I1=N(M1) TO M(M1) STEP O
1300 LET J2=228
1310 FOR I2=M1^2+2*M1*M(M1)^2 TO (N(M1)^3-1)+ABS(O) STEP 20*N(M1)/2
1320 LET J3=2
1330 FOR I3=ABS(M(M1)-M1) TO M(M1)^2-M1^2 STEP 2*SGN(M(M1)^2-M1^2)
1340 IF I0<>J0 THEN 1520
1350 IF I1<>J1 THEN 1520
1360 IF I2<>J2 THEN 1520
1370 IF I3<>J3 THEN 1520
1380 LET K=K+1
1390 LET M1 = M1 + 1
1400 LET J3=J3-2
1410 NEXT I3
1420 LET M1=6
1430 LET J2=J2+100
1440 NEXT I2
1450 LET J1=J1-1
1460 NEXT I1
1470 LET J0=J0+1
1480 NEXT I0
1490 IF K<>2688 THEN 1600

```

```
1500 PRINT "*** TEST PASSED ***"
1510 GOTO 1640
1520 PRINT "*** TEST FAILED ***"
1530 PRINT
1540 PRINT "CTL. VAR.", "SHOULD BE", "ACTUAL"
1550 PRINT "I0", J0, I0
1560 PRINT "I1", J1, I1
1570 PRINT "I2", J2, I2
1580 PRINT "I3", J3, I3
1590 GOTO 1640
1600 PRINT "*** TEST FAILED ***"
1610 PRINT
1620 PRINT "INNERMOST LOOP SHOULD HAVE BEEN EXECUTED 2688 TIMES."
1630 PRINT "ACTUAL COUNT = "; K
1640 PRINT
1650 PRINT "                                END TEST."
1660 PRINT
1670 PRINT "END PROGRAM 166."
1680 END
```

---

PROGRAM FILE 166: COMPOUND EXPRESSIONS USED WITH CONTROL  
STATEMENTS AND FOR-STATEMENTS.  
ANSI STANDARD 7.2, 7.4, 10.2, 10.4, 11.2, 11.4

THIS PROGRAM TESTS THE USE OF NUMERIC EXPRESSIONS WITHIN  
THE CONTEXT OF THE CONTROL STATEMENTS IF-THEN AND ON-GOTO,  
AND ALSO WITHIN THE FOR-STATEMENT.

SECTION 166.1: NUMERIC EXPRESSIONS AND  
THE IF-THEN-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 166.2: NUMERIC EXPRESSIONS AND  
THE ON-GOTO-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 166.3: NUMERIC EXPRESSIONS AND  
THE FOR-NEXT-STATEMENT.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 166.

```
*****  
*****  
*****  
10 PRINT "PROGRAM FILE 167: EXCEPTION - EVALUATION OF NUMERIC"  
20 PRINT "      EXPRESSIONS ACTING AS FUNCTION ARGUMENTS."  
40 PRINT "      ANSI STANDARD 7.5, 8.4, 16.4"  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS THAT THE RECOVERY PROCEDURES FOR"  
70 PRINT "EXCEPTIONS OCCURRING DURING THE EVALUATION OF NUMERIC"  
80 PRINT "EXPRESSIONS WORK PROPERLY IN CONTEXTS OTHER THAN SIMPLE"  
90 PRINT "ASSIGNMENT OF THE EXPRESSION TO A VARIABLE. SPECIFICALLY,"  
100 PRINT "EXPRESSIONS ARE USED AS ARGUMENTS TO FUNCTIONS."  
120 PRINT  
130 PRINT "SECTION 167.1: DIVISION BY ZERO IN AN ARGUMENT TO A"  
140 PRINT "      USER-DEFINED FUNCTION."  
150 PRINT  
160 PRINT "TO PASS THIS TEST:"  
170 PRINT  
180 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
190 PRINT "          DISPLAYED, AND"  
200 PRINT  
210 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"  
220 PRINT "          OF THE ARGUMENT AND EXECUTION CONTINUES."  
230 PRINT  
240 PRINT "      BEGIN TEST."  
250 PRINT  
260 DEF FNC(X)=LOG(X)/LOG(10)  
270 LET A=5  
280 LET B=0  
290 PRINT "ABOUT TO EVALUATE FNC(5/0), WHERE FNC = COMMON LOG."  
300 PRINT "EXCEPTION SHOULD BE REPORTED NOW:"  
310 PRINT  
320 LET C=FNC(A/B)  
330 PRINT  
340 PRINT "RESULT OF FNC(5/0) = ";C  
350 PRINT  
360 IF C >= 37.99 THEN 400  
370 PRINT "*** TEST FAILED: RESULT < 38 ***"  
380 GOTO 440  
400 PRINT "IF EXCEPTION REPORTED, AND"  
410 PRINT "      RESULT = COMMON LOG OF MACHINE INFINITY, THEN"  
420 PRINT "*** TEST PASSED ***"  
440 PRINT  
450 PRINT "      END TEST."
```

```

460 PRINT
1110 PRINT "SECTION 167.2: ZERO RAISED TO A NEGATIVE POWER IN AN"
1120 PRINT "           ARGUMENT TO AN IMPLEMENTATION-SUPPLIED FUNCTION."
1130 PRINT
1140 PRINT "TO PASS THIS TEST:"
1150 PRINT
1160 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
1170 PRINT "           DISPLAYED, AND"
1180 PRINT
1190 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
1200 PRINT "           OF THE ARGUMENT AND EXECUTION CONTINUES."
1210 PRINT
1220 PRINT "           BEGIN TEST."
1230 PRINT
1250 LET A=-5
1260 LET B=0
1270 PRINT "ABOUT TO EVALUATE ATN(0^(-5))."
1280 PRINT "EXCEPTION SHOULD BE REPORTED NOW:"
1290 PRINT
1300 LET C=ATN(B^A)
1310 PRINT
1320 PRINT "RESULT OF ATN(0^(-5)) = ";C
1330 PRINT
1340 IF ABS(C-1.5708) <= .001 THEN 1370
1350 PRINT "*** TEST FAILED: RESULT <> PI/2 ***"
1360 GOTO 1400
1370 PRINT "IF EXCEPTION REPORTED, THEN"
1390 PRINT "*** TEST PASSED ***"
1400 PRINT
1410 PRINT "           END TEST."
1420 PRINT
1440 PRINT "END PROGRAM 167"
1450 END

```

---

PROGRAM FILE 167: EXCEPTION - EVALUATION OF NUMERIC  
EXPRESSIONS ACTING AS FUNCTION ARGUMENTS.  
ANSI STANDARD 7.5, 8.4, 16.4

THIS PROGRAM TESTS THAT THE RECOVERY PROCEDURES FOR  
EXCEPTIONS OCCURRING DURING THE EVALUATION OF NUMERIC  
EXPRESSIONS WORK PROPERLY IN CONTEXTS OTHER THAN SIMPLE  
ASSIGNMENT OF THE EXPRESSION TO A VARIABLE. SPECIFICALLY,  
EXPRESSIONS ARE USED AS ARGUMENTS TO FUNCTIONS.

SECTION 167.1: DIVISION BY ZERO IN AN ARGUMENT TO A  
USER-DEFINED FUNCTION.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND

- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE ARGUMENT AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO EVALUATE FNC(5/0), WHERE FNC = COMMON LOG.  
EXCEPTION SHOULD BE REPORTED NOW:

% DIVISION BY ZERO IN LINE 320

RESULT OF FNC(5/0) = 38.2308

IF EXCEPTION REPORTED, AND  
RESULT = COMMON LOG OF MACHINE INFINITY, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 167.2: ZERO RAISED TO A NEGATIVE POWER IN AN ARGUMENT TO AN IMPLEMENTATION-SUPPLIED FUNCTION.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE ARGUMENT AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO EVALUATE ATN(0^(-5)).  
EXCEPTION SHOULD BE REPORTED NOW:

% ZERO TO A NEGATIVE POWER IN LINE 1300

RESULT OF ATN(0^(-5)) = 1.5708

IF EXCEPTION REPORTED, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 167

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

10 PRINT "PROGRAM FILE 168: EXCEPTION - OVERFLOW IN THE SUBSCRIPT"  
20 PRINT " OF AN ARRAY."  
40 PRINT " ANSI STANDARD 6.4, 6.5, 7.5"  
50 PRINT

```
60 PRINT "THIS PROGRAM TESTS THAT THE RECOVERY PROCEDURES FOR"
70 PRINT "EXCEPTIONS OCCURRING DURING THE EVALUATION OF NUMERIC"
80 PRINT "EXPRESSIONS WORK PROPERLY IN CONTEXTS OTHER THAN SIMPLE"
90 PRINT "ASSIGNMENT OF THE EXPRESSION TO A VARIABLE. SPECIFICALLY,"
100 PRINT "AN EXPRESSION IS USED AS A SUBSCRIPT OF AN ARRAY."
110 PRINT
120 PRINT "SECTION 168.1: EXCEPTION - OVERFLOW IN THE SUBSCRIPT"
125 PRINT "      OF AN ARRAY."
130 PRINT
140 PRINT "TO PASS THIS TEST:"
150 PRINT
160 PRINT "      1) A MESSAGE IDENTIFYING THE OVERFLOW EXCEPTION MUST"
170 PRINT "          BE DISPLAYED."
180 PRINT
190 PRINT "      2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE"
200 PRINT "          OF THE SUBSCRIPT AND EXECUTION CONTINUES."
210 PRINT
220 PRINT "      3) WHEREUPON AN EXCEPTION MUST BE REPORTED"
230 PRINT "          FOR SUBSCRIPT OUT OF RANGE, AND"
240 PRINT
250 PRINT "      4) EXECUTION MUST TERMINATE."
260 PRINT
270 PRINT "                  BEGIN TEST."
280 PRINT
290 FOR I=0 TO 10
300 LET Z(I)=I+100
310 NEXT I
320 LET A=9999
330 PRINT "ABOUT TO EVALUATE Z(9999^9999), WHERE Z(I) = I + 100."
340 PRINT
390 LET C=Z(A^A)
400 PRINT
410 PRINT "RESULT OF Z(9999^9999) = ";C
420 PRINT
430 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
440 PRINT
450 PRINT "                  END TEST."
460 PRINT
470 PRINT "END PROGRAM 168"
480 END
```

---

PROGRAM FILE 168: EXCEPTION - OVERFLOW IN THE SUBSCRIPT  
OF AN ARRAY.

ANSI STANDARD 6.4, 6.5, 7.5

THIS PROGRAM TESTS THAT THE RECOVERY PROCEDURES FOR  
EXCEPTIONS OCCURRING DURING THE EVALUATION OF NUMERIC  
EXPRESSIONS WORK PROPERLY IN CONTEXTS OTHER THAN SIMPLE  
ASSIGNMENT OF THE EXPRESSION TO A VARIABLE. SPECIFICALLY,  
AN EXPRESSION IS USED AS A SUBSCRIPT OF AN ARRAY.

SECTION 168.1: EXCEPTION - OVERFLOW IN THE SUBSCRIPT OF AN ARRAY.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE OVERFLOW EXCEPTION MUST BE DISPLAYED,
- 2) POSITIVE INFINITY MUST BE SUPPLIED AS THE VALUE OF THE SUBSCRIPT AND EXECUTION CONTINUES,
- 3) WHEREUPON AN EXCEPTION MUST BE REPORTED FOR SUBSCRIPT OUT OF RANGE, AND
- 4) EXECUTION MUST TERMINATE.

BEGIN TEST.

ABOUT TO EVALUATE Z(9999^9999), WHERE Z(I) = I + 100.

% OVERFLOW IN LINE 390

? DIMENSION ERROR IN LINE 390

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 169: EXCEPTION - NUMERIC UNDERFLOW IN THE"  
20 PRINT "          EVALUATION OF NUMERIC EXPRESSIONS ACTING AS "  
30 PRINT "          ARGUMENTS AND SUBSCRIPTS."  
40 PRINT "          ANSI STANDARD 6.4, 7.4, 7.6, 8.4"  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS THAT THE HANDLING OF"  
70 PRINT "UNDERFLOW OCCURRING DURING THE EVALUATION OF NUMERIC"  
80 PRINT "EXPRESSIONS WORKS PROPERLY IN CONTEXTS OTHER THAN SIMPLE"  
90 PRINT "ASSIGNMENT OF THE EXPRESSION TO A VARIABLE. SPECIFICALLY,"  
100 PRINT "EXPRESSIONS ARE USED AS ARGUMENTS TO FUNCTIONS AND"  
110 PRINT "SUBSCRIPTS OF ARRAYS."  
120 PRINT  
130 PRINT "SECTION 169.1: NUMERIC UNDERFLOW IN AN ARGUMENT TO A"  
140 PRINT "          SUPPLIED FUNCTION."  
150 PRINT  
160 PRINT "TO PASS THIS TEST:"  
170 PRINT  
180 PRINT "      1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE"  
190 PRINT "          DISPLAYED (NOT MANDATORY), AND"  
200 PRINT  
210 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"  
220 PRINT "          OF THE ARGUMENT AND EXECUTION CONTINUES."  
230 PRINT  
240 PRINT "          BEGIN TEST."  
250 PRINT
```

```
270 LET A=5000
280 LET B=1E-20
290 PRINT "ABOUT TO EVALUATE EXP(1E-20 ^ 5000)."
310 PRINT
320 LET C=EXP(B^A)
330 PRINT
340 PRINT "RESULT OF EXP(1E-20 ^ 5000) = ";C
350 PRINT
360 IF ABS(C-1) <= .001 THEN 400
370 PRINT "*** TEST FAILED: RESULT <> 1 ***"
380 GOTO 440
400 PRINT "*** TEST PASSED ***"
440 PRINT
450 PRINT "                               END TEST."
460 PRINT
1110 PRINT "SECTION 169.2: NUMERIC UNDERFLOW IN AN ARRAY SUBSCRIPT."
1130 PRINT
1160 PRINT "TO PASS THIS TEST:"
1170 PRINT
1180 PRINT "      1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE"
1190 PRINT "          DISPLAYED (NOT MANDATORY), AND"
1200 PRINT
1210 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE"
1220 PRINT "          OF THE SUBSCRIPT AND EXECUTION CONTINUES."
1230 PRINT
1240 PRINT "                               BEGIN TEST."
1250 PRINT
1270 LET A=5001
1280 LET B=-1E-20
1282 FOR I=0 TO 10
1284 LET Z(I)=100+I
1286 NEXT I
1290 PRINT "ABOUT TO EVALUATE Z(-1E-20 ^ 5001), WHERE Z(I) = I + 100."
1310 PRINT
1320 LET C=Z(B^A)
1330 PRINT
1340 PRINT "RESULT OF Z(-1E-20 ^ 5001) = ";C
1350 PRINT
1360 IF ABS(C-100) <= .01 THEN 1400
1370 PRINT "*** TEST FAILED: RESULT <> 100 ***"
1380 GOTO 1440
1400 PRINT "*** TEST PASSED ***"
1440 PRINT
1450 PRINT "                               END TEST."
1460 PRINT
3440 PRINT "END PROGRAM 169"
3450 END
```

---

PROGRAM FILE 169: EXCEPTION - NUMERIC UNDERFLOW IN THE  
EVALUATION OF NUMERIC EXPRESSIONS ACTING AS  
ARGUMENTS AND SUBSCRIPTS.  
ANSI STANDARD 6.4, 7.4, 7.6, 8.4

THIS PROGRAM TESTS THAT THE HANDLING OF UNDERFLOW OCCURRING DURING THE EVALUATION OF NUMERIC EXPRESSIONS WORKS PROPERLY IN CONTEXTS OTHER THAN SIMPLE ASSIGNMENT OF THE EXPRESSION TO A VARIABLE. SPECIFICALLY, EXPRESSIONS ARE USED AS ARGUMENTS TO FUNCTIONS AND SUBSCRIPTS OF ARRAYS.

SECTION 169.1: NUMERIC UNDERFLOW IN AN ARGUMENT TO A SUPPLIED FUNCTION.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE ARGUMENT AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO EVALUATE EXP(1E-20 ^ 5000).

% UNDERFLOW IN LINE 320

RESULT OF EXP(1E-20 ^ 5000) = 1

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 169.2: NUMERIC UNDERFLOW IN AN ARRAY SUBSCRIPT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING AN EXCEPTION MAY BE DISPLAYED (NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE SUBSCRIPT AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO EVALUATE Z(-1E-20 ^ 5001), WHERE Z(I) = I + 100.

% UNDERFLOW IN LINE 1320

RESULT OF Z(-1E-20 ^ 5001) = 100

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 169

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 170: EXCEPTION - NEGATIVE QUANTITY RAISED TO A"  
20 PRINT "      NON-INTEGRAL POWER IN A SUBSCRIPT."  
30 PRINT "      ANSI STANDARD 7.5, 6.2"  
40 PRINT  
50 PRINT "SECTION 170.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO A"  
60 PRINT "      NON-INTEGRAL POWER IN A SUBSCRIPT."  
70 PRINT  
80 PRINT "TO PASS THIS TEST:"  
90 PRINT  
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
110 PRINT "          DISPLAYED, AND"  
120 PRINT  
130 PRINT "      2) EXECUTION MUST TERMINATE"  
140 PRINT  
150 PRINT "          BEGIN TEST"  
160 PRINT  
170 FOR I=0 TO 10  
180 LET Z(I)=I+100  
190 NEXT I  
200 PRINT "ABOUT TO EVALUATE Z((-2) ^ 3.00001),"  
210 PRINT "WHERE Z(I) = I + 100"  
220 PRINT  
230 LET B=3.00001  
240 LET A=-2  
250 LET C=Z(A^B)  
260 PRINT  
270 PRINT "RESULT OF Z((-2 ^ 3.00001) = ";C  
280 PRINT  
290 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"  
300 PRINT  
310 PRINT "END TEST"  
320 PRINT  
330 PRINT "END PROGRAM 170"  
340 END
```

---

PROGRAM FILE 170: EXCEPTION - NEGATIVE QUANTITY RAISED TO A  
 NON-INTEGRAL POWER IN A SUBSCRIPT.  
 ANSI STANDARD 7.5, 6.2

SECTION 170.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO A  
 NON-INTEGRAL POWER IN A SUBSCRIPT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
 DISPLAYED, AND

2) EXECUTION MUST TERMINATE

BEGIN TEST

ABOUT TO EVALUATE Z((-2) ^ 3.00001),  
WHERE Z(I) = I + 100

? NEGATIVE VALUE RAISED TO NON-INTEGRAL POWER IN LINE 290

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 171: EXCEPTION - LOG OF A NEGATIVE QUANTITY"
20 PRINT "           IN AN ARGUMENT."
30 PRINT "       ANSI STANDARD 8.5, 16.2"
40 PRINT
50 PRINT "SECTION 171.1: EXCEPTION - LOG OF A NEGATIVE QUANTITY"
60 PRINT "           IN AN ARGUMENT."
70 PRINT
80 PRINT "TO PASS THIS TEST:"
90 PRINT
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
110 PRINT "          DISPLAYED, AND"
120 PRINT
130 PRINT "      2) EXECUTION MUST TERMINATE"
140 PRINT
150 PRINT "           BEGIN TEST"
160 PRINT
170 DEF FNT(X)=X/3
180 PRINT "ABOUT TO EVALUATE FNT(LOG(-2)),"
190 PRINT "WHERE FNT(X) = X/3"
200 PRINT
260 LET A=-2
270 LET C=FNT(LOG(A))
280 PRINT
290 PRINT "RESULT OF FNT(LOG(-2)) = ";C
300 PRINT
310 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
320 PRINT
330 PRINT "           END TEST"
340 PRINT
350 PRINT "END PROGRAM 171"
360 END
```

---

PROGRAM FILE 171: EXCEPTION - LOG OF A NEGATIVE QUANTITY  
IN AN ARGUMENT.

ANSI STANDARD 8.5, 16.2

SECTION 171.1: EXCEPTION - LOG OF A NEGATIVE QUANTITY  
IN AN ARGUMENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE

BEGIN TEST

ABOUT TO EVALUATE FNT(LOG(-2)),  
WHERE FNT(X) = X/3

? LOG OF NEGATIVE NUMBER IN LINE 270

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 172: EXCEPTION - SQR OF NEGATIVE QUANTITY "
20 PRINT "           IN PRINT-ITEM."
30 PRINT "           ANSI STANDARD 8.5, 12.2"
40 PRINT
50 PRINT "SECTION 172.1: EXCEPTION - SQR OF NEGATIVE QUANTITY "
55 PRINT "           IN PRINT-ITEM."
60 PRINT
70 PRINT "TO PASS THIS TEST:"
80 PRINT
90 PRINT "   1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
100 PRINT "      DISPLAYED, AND"
110 PRINT
120 PRINT "   2) EXECUTION MUST TERMINATE"
130 PRINT
140 PRINT "           BEGIN TEST."
150 PRINT
160 PRINT "ABOUT TO EXECUTE:"
170 PRINT "   PRINT SQR (-2)"
180 LET A=-2
200 PRINT SQR(A)
230 PRINT
240 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
250 PRINT
260 PRINT "           END TEST."
270 PRINT
280 PRINT "END PROGRAM 172."
290 END
```

---

PROGRAM FILE 172: EXCEPTION - SQR OF NEGATIVE QUANTITY  
IN PRINT-ITEM.  
ANSI STANDARD 8.5, 12.2

SECTION 172.1: EXCEPTION - SQR OF NEGATIVE QUANTITY  
IN PRINT-ITEM.

**TO PASS THIS TEST:**

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE DISPLAYED, AND
  - 2) EXECUTION MUST TERMINATE

BEGIN TEST.

**ABOUT TO EXECUTE:**

PRINT SQR (-2)

? SQR OF NEGATIVE NUMBER IN LINE 200

A large grid of black asterisks on a white background, arranged in approximately 20 rows and 20 columns.

```
10 PRINT "PROGRAM FILE 173: EXCEPTION - NEGATIVE QUANTITY RAISED TO"
20 PRINT "          A NON-INTEGRAL POWER IN TAB-ITEM."
30 PRINT "          ANSI STANDARD 7.5, 12.2"
40 PRINT
50 PRINT "SECTION 173.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO"
55 PRINT "          A NON-INTEGRAL POWER IN TAB-ITEM."
60 PRINT
70 PRINT "TO PASS THIS TEST:"
80 PRINT
90 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
100 PRINT "          DISPLAYED, AND"
110 PRINT
120 PRINT "      2) EXECUTION MUST TERMINATE"
130 PRINT
140 PRINT "          BEGIN TEST."
150 PRINT
160 PRINT "ABOUT TO EXECUTE:"
170 PRINT "      PRINT A$;TAB((-3) ^ 1.99999);B$"
190 PRINT "      (WHERE A$ = '111' AND B$ = '222')"
195 PRINT
200 LET A$="111"
210 LET B$="222"
220 LET C=-3
225 LET D=1.99999
230 PRINT A$;TAB(C^D);B$
240 PRINT
250 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
260 PRINT
270 PRINT "          END TEST."
```

```
280 PRINT
290 PRINT "END PROGRAM 173."
300 END
```

---

PROGRAM FILE 173: EXCEPTION - NEGATIVE QUANTITY RAISED TO  
A NON-INTEGRAL POWER IN TAB-ITEM.

ANSI STANDARD 7.5, 12.2

SECTION 173.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO  
A NON-INTEGRAL POWER IN TAB-ITEM.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE

BEGIN TEST.

ABOUT TO EXECUTE:

```
PRINT A$;TAB((-3) ^ 1.99999);B$  
(WHERE A$ = '111' AND B$ = '222')
```

111

? NEGATIVE VALUE RAISED TO NON-INTEGRAL POWER IN LINE 230

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 174: EXCEPTION - EVALUATION OF NUMERIC"
20 PRINT "      EXPRESSIONS IN THE PRINT STATEMENT."
30 PRINT "      ANSI STANDARD 7.5, 8.5, 12.2"
40 PRINT
50 PRINT "THIS PROGRAM TESTS THE EFFECT OF THE VARIOUS EXCEPTIONS THAT"
60 PRINT "CAN OCCUR IN NUMERIC EXPRESSIONS IN THE CONTEXT OF THE PRINT"
70 PRINT "STATEMENT."
80 PRINT
90 PRINT "SECTION 174.1: EXCEPTIONAL EXPRESSIONS AS PRINT-ITEMS."
100 PRINT
110 PRINT "THIS SECTION TESTS THE EFFECT OF PRINTING EXPRESSIONS WHICH"
120 PRINT "CAUSE NON-FATAL EXCEPTIONS."
130 PRINT
160 PRINT "TO PASS THIS TEST:"
170 PRINT
180 PRINT "      1) FOUR EXCEPTIONS MUST BE REPORTED: OVERFLOW, DIVISION"
190 PRINT "          BY ZERO, ZERO RAISED TO A NEGATIVE POWER, AND"
195 PRINT "          OVERFLOW OF EXP FUNCTION, AND"
200 PRINT
```

```
210 PRINT "      2) NEGATIVE, NEGATIVE, POSITIVE, AND POSITIVE MACHINE"
220 PRINT "           INFINITY MUST BE SUPPLIED AS THE VALUES OF THE"
230 PRINT "           EXPRESSIONS AND THEN PRINTED."
240 PRINT
242 PRINT "               BEGIN TEST."
246 PRINT
250 PRINT "ABOUT TO EXECUTE:"
260 PRINT "    PRINT (-1E-33) ^ (-3333), (-1E-33) / 0, 0 ^ (-1E-33);"
265 PRINT ", EXP(1E20)"
270 LET A=0
280 LET B=-3333
290 LET C=-1E-33
295 LET D=1E20
300 PRINT
310 PRINT C^B, C/A, A^C, EXP(D)
320 PRINT
330 PRINT "IF FOUR EXCEPTIONS REPORTED AND NEGATIVE, NEGATIVE,"
340 PRINT "    POSITIVE, AND POSITIVE INFINITY PRINTED, THEN"
350 PRINT "*** TEST PASSED ***"
360 PRINT
370 PRINT "               END TEST."
380 PRINT
390 PRINT "SECTION 174.2: EXCEPTIONAL EXPRESSIONS AS TAB-ITEMS."
400 PRINT
410 PRINT "THIS SECTION TESTS THE EFFECT OF OVERFLOW IN A TAB-ITEM."
420 PRINT
430 PRINT "TO PASS THIS TEST:"
440 PRINT
450 PRINT "    1) THE OVERFLOW EXCEPTION MUST BE REPORTED, AND"
460 PRINT
470 PRINT "    2) EXECUTION CONTINUES, WITH THE TAB EVALUATING TO"
480 PRINT "           SOME ARBITRARY PRINT POSITION."
490 PRINT
500 PRINT "               BEGIN TEST."
510 PRINT
520 PRINT "ABOUT TO EXECUTE:"
530 PRINT "    PRINT TAB(9^(9^9));X$"
540 PRINT "    (WHERE X$ = 'X')"
550 PRINT
560 PRINT "0000000001111111112222222233333333444444444";
570 PRINT "5555555555666666666777"
580 PRINT "1234567890123456789012345678901234567890123456789";
590 PRINT "01234567890123456789012"
600 LET X$="X"
610 LET D=9
620 PRINT TAB(D^(D^D));X$
630 PRINT
640 PRINT "IF OVERFLOW EXCEPTION REPORTED, AND 'X' APPEARED FOLLOWING"
650 PRINT "    NUMBERED LINES, THEN"
660 PRINT "*** TEST PASSED ***"
670 PRINT
680 PRINT "               END TEST."
```

```
690 PRINT
700 PRINT "END PROGRAM 174"
710 END
```

---

PROGRAM FILE 174: EXCEPTION - EVALUATION OF NUMERIC  
EXPRESSIONS IN THE PRINT STATEMENT.

ANSI STANDARD 7.5, 8.5, 12.2

THIS PROGRAM TESTS THE EFFECT OF THE VARIOUS EXCEPTIONS THAT  
CAN OCCUR IN NUMERIC EXPRESSIONS IN THE CONTEXT OF THE PRINT  
STATEMENT.

#### SECTION 174.1: EXCEPTIONAL EXPRESSIONS AS PRINT-ITEMS.

THIS SECTION TESTS THE EFFECT OF PRINTING EXPRESSIONS WHICH  
CAUSE NON-FATAL EXCEPTIONS.

TO PASS THIS TEST:

- 1) FOUR EXCEPTIONS MUST BE REPORTED: OVERFLOW, DIVISION  
BY ZERO, ZERO RAISED TO A NEGATIVE POWER, AND  
OVERFLOW OF EXP FUNCTION, AND
- 2) NEGATIVE, NEGATIVE, POSITIVE, AND POSITIVE MACHINE  
INFINITY MUST BE SUPPLIED AS THE VALUES OF THE  
EXPRESSIONS AND THEN PRINTED.

BEGIN TEST.

ABOUT TO EXECUTE:

```
PRINT (-1E-33) ^ (-3333), (-1E-33) / 0, 0 ^ (-1E-33), EXP(1E20)
```

% OVERFLOW IN LINE 310

-1.70141E+38

% DIVISION BY ZERO IN LINE 310

-1.70141E+38

% ZERO TO A NEGATIVE POWER IN LINE 310

1.70141E+38

% OVERFLOW IN EXP IN LINE 310

1.70141E+38

IF FOUR EXCEPTIONS REPORTED AND NEGATIVE, NEGATIVE,  
POSITIVE, AND POSITIVE INFINITY PRINTED, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

SECTION 174.2: EXCEPTIONAL EXPRESSIONS AS TAB-ITEMS.

THIS SECTION TESTS THE EFFECT OF OVERFLOW IN A TAB-ITEM.

TO PASS THIS TEST:

- 1) THE OVERFLOW EXCEPTION MUST BE REPORTED, AND
- 2) EXECUTION CONTINUES, WITH THE TAB EVALUATING TO SOME ARBITRARY PRINT POSITION.

BEGIN TEST.

ABOUT TO EXECUTE:

```
PRINT TAB(9^(9^9));X$  
(WHERE X$ = 'X')
```

```
00000000011111111122222222333333334444444455555555666666666777  
12345678901234567890123456789012345678901234567890123456789012
```

% OVERFLOW IN LINE 620

X

IF OVERFLOW EXCEPTION REPORTED, AND 'X' APPEARED FOLLOWING NUMBERED LINES, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 174

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 175: EXCEPTION - UNDERFLOW IN THE EVALUATION"  
20 PRINT "          OF NUMERIC EXPRESSIONS IN THE PRINT STATEMENT."  
30 PRINT "      ANSI STANDARD 7.4, 7.6, 8.6, 12.2"  
40 PRINT  
50 PRINT "THIS PROGRAM TESTS THE EFFECT OF UNDERFLOW IN NUMERIC"  
60 PRINT "EXPRESSIONS IN THE CONTEXT OF THE PRINT STATEMENT."  
70 PRINT  
80 PRINT "SECTION 175.1: UNDERFLOW IN PRINT-ITEMS."  
90 PRINT  
100 PRINT "THIS SECTION TESTS THE EFFECT OF PRINTING EXPRESSIONS WHICH"  
110 PRINT "CAUSE NUMERIC UNDERFLOW."  
120 PRINT  
130 PRINT "TO PASS THIS TEST:"  
140 PRINT  
150 PRINT "      1) THREE UNDERFLOW EXCEPTIONS MAY BE REPORTED"  
160 PRINT "          (NOT MANDATORY), AND"  
170 PRINT  
180 PRINT "      2) ZEROS MUST BE SUPPLIED AS THE VALUES OF THE "  
190 PRINT "          EXPRESSIONS AND THEN PRINTED."  
200 PRINT
```

```
210 PRINT "                BEGIN TEST."
220 PRINT
230 PRINT "ABOUT TO EXECUTE:"
240 PRINT "    PRINT (-1E33) ^ (-3333), (-1E-33) ^ 3333, EXP (-1E33)"
250 LET A=-1E33
260 LET B=-3333
270 PRINT
280 PRINT A^B, (1/A)^(-B), EXP(A)
290 PRINT
300 PRINT "IF THREE ZEROS PRINTED, THEN"
310 PRINT "*** TEST PASSED ***"
320 PRINT
330 PRINT "                END TEST."
340 PRINT
350 PRINT "SECTION 175.2: NUMERIC UNDERFLOW IN TAB-ITEMS."
360 PRINT
370 PRINT "THIS SECTION TESTS THE EFFECT OF UNDERFLOW IN A TAB-ITEM."
380 PRINT
390 PRINT "TO PASS THIS TEST:"
400 PRINT
410 PRINT "    1) AN UNDERFLOW EXCEPTION MAY BE REPORTED"
420 PRINT "        (NOT MANDATORY),"
430 PRINT
440 PRINT "    2) EXECUTION CONTINUES, WITH ZERO SUPPLIED AS THE"
450 PRINT "        VALUE OF THE TAB ARGUMENT,"
460 PRINT
470 PRINT "    3) A TAB EXCEPTION MUST BE REPORTED, AND A VALUE OF"
480 PRINT "        ONE SUPPLIED AS THE TAB ARGUMENT."
490 PRINT
500 PRINT "                BEGIN TEST."
510 PRINT
520 PRINT "ABOUT TO EXECUTE:"
530 PRINT "    PRINT A$;TAB(1E-33 ^ 3333);B$"
540 PRINT "        (WHERE A$ = 'AAA' AND B$ = 'BBB')"
550 PRINT
560 PRINT "00000000011111111122222222333333334444444444";
570 PRINT "555555555666666666777"
580 PRINT "1234567890123456789012345678901234567890123456789";
590 PRINT "01234567890123456789012"
600 LET A$="AAA"
610 LET B$="BBB"
620 LET E=1E-33
630 LET F=3333
640 PRINT A$;TAB(E^F);B$
650 PRINT
660 PRINT "IF TAB-ARGUMENT EXCEPTION REPORTED, AND 'AAA' APPEARED IN"
670 PRINT "    COL. 1-3, AND 'BBB' IN COL. 1-3 ON A LATER LINE, THEN"
680 PRINT "*** TEST PASSED ***"
690 PRINT
700 PRINT "                END TEST."
```

```
710 PRINT
720 PRINT "END PROGRAM 175"
730 END
```

---

PROGRAM FILE 175: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF NUMERIC EXPRESSIONS IN THE PRINT STATEMENT.  
ANSI STANDARD 7.4, 7.6, 8.6, 12.2

THIS PROGRAM TESTS THE EFFECT OF UNDERFLOW IN NUMERIC  
EXPRESSIONS IN THE CONTEXT OF THE PRINT STATEMENT.

#### SECTION 175.1: UNDERFLOW IN PRINT-ITEMS.

THIS SECTION TESTS THE EFFECT OF PRINTING EXPRESSIONS WHICH  
CAUSE NUMERIC UNDERFLOW.

TO PASS THIS TEST:

- 1) THREE UNDERFLOW EXCEPTIONS MAY BE REPORTED  
(NOT MANDATORY), AND
- 2) ZEROS MUST BE SUPPLIED AS THE VALUES OF THE  
EXPRESSIONS AND THEN PRINTED.

BEGIN TEST.

ABOUT TO EXECUTE:

```
PRINT (-1E33) ^ (-3333), (-1E-33) ^ 3333, EXP (-1E33)
```

```
% UNDERFLOW IN LINE 280
0
% UNDERFLOW IN LINE 280
0
% UNDERFLOW IN EXP IN LINE 280
0
```

IF THREE ZEROS PRINTED, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 175.2: NUMERIC UNDERFLOW IN TAB-ITEMS.

THIS SECTION TESTS THE EFFECT OF UNDERFLOW IN A TAB-ITEM.

TO PASS THIS TEST:

- 1) AN UNDERFLOW EXCEPTION MAY BE REPORTED  
(NOT MANDATORY),

- 2) EXECUTION CONTINUES, WITH ZERO SUPPLIED AS THE VALUE OF THE TAB ARGUMENT,
- 3) A TAB EXCEPTION MUST BE REPORTED, AND A VALUE OF ONE SUPPLIED AS THE TAB ARGUMENT.

BEGIN TEST.

ABOUT TO EXECUTE:

```
PRINT A$;TAB(1E-33 ^ 3333);B$  
(WHERE A$ = 'AAA' AND B$ = 'BBB')
```

```
000000000111111112222222233333333444444445555555566666666777  
12345678901234567890123456789012345678901234567890123456789012  
AAA  
% UNDERFLOW IN LINE 640  
BBB
```

IF TAB-ARGUMENT EXCEPTION REPORTED, AND 'AAA' APPEARED IN COL. 1-3, AND 'BBB' IN COL. 1-3 ON A LATER LINE, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 175

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 176: EXCEPTION - NEGATIVE QUANTITY RAISED TO"  
20 PRINT "      A NON-INTEGRAL POWER IN IF-STATEMENT."  
30 PRINT "      ANSI STANDARD 7.5, 10.2"  
40 PRINT  
50 PRINT "SECTION 176.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO"  
60 PRINT "      A NON-INTEGRAL POWER IN IF-STATEMENT."  
70 PRINT  
80 PRINT "TO PASS THIS TEST:"  
90 PRINT  
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
110 PRINT "          DISPLAYED, AND"  
120 PRINT  
130 PRINT "      2) EXECUTION MUST TERMINATE"  
140 PRINT  
150 PRINT "          BEGIN TEST."  
160 PRINT  
170 PRINT "ABOUT TO EXECUTE:"  
180 PRINT "      IF (-3) ^ 3.00001 < 0 ..."  
190 PRINT
```

```

210 LET B=3.00001
220 LET A=-3
230 IF A^B < 0 THEN 260
240 PRINT "IF-STATEMENT TESTED FALSE."
250 GOTO 270
260 PRINT "IF-STATEMENT TESTED TRUE."
270 PRINT
320 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
330 PRINT
340 PRINT "END TEST."
350 PRINT
360 PRINT "END PROGRAM 176."
370 END

```

---

PROGRAM FILE 176: EXCEPTION - NEGATIVE QUANTITY RAISED TO  
A NON-INTEGRAL POWER IN IF-STATEMENT.

ANSI STANDARD 7.5, 10.2

SECTION 176.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO  
A NON-INTEGRAL POWER IN IF-STATEMENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE

BEGIN TEST.

ABOUT TO EXECUTE:

IF (-3) ^ 3.00001 < 0 ...

? NEGATIVE VALUE RAISED TO NON-INTEGRAL POWER IN LINE 230

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 177: EXCEPTION - EVALUATION OF NUMERIC"
20 PRINT "      EXPRESSIONS IN THE IF-STATEMENT."
30 PRINT "      ANSI STANDARD 7.5, 10.2"
40 PRINT
50 PRINT "SECTION 177.1: EXCEPTION - EVALUATION OF NUMERIC"
60 PRINT "      EXPRESSIONS IN THE IF-STATEMENT."
70 PRINT
80 PRINT "THIS SECTION TESTS THE EFFECT OF COMPARING EXPRESSIONS"
90 PRINT "WHICH CAUSE NON-FATAL EXCEPTIONS."
100 PRINT
130 PRINT "TO PASS THIS TEST:"
140 PRINT

```

```

150 PRINT "      1) TWO EXCEPTIONS MUST BE REPORTED: OVERFLOW, "
160 PRINT "          AND ZERO RAISED TO A NEGATIVE POWER, AND"
170 PRINT
180 PRINT "      2) POSITIVE MACHINE INFINITY MUST BE SUPPLIED AS"
190 PRINT "          THE VALUES OF THE EXPRESSIONS, AND"
200 PRINT
210 PRINT "      3) THE IF MUST TEST EQUAL, AND EXECUTION CONTINUES."
220 PRINT
222 PRINT "                  BEGIN TEST."
225 PRINT
230 PRINT "ABOUT TO EXECUTE:"
240 PRINT "    IF (-1E-33) ^ (-4444) = 0 ^ (-1E-33)"
250 LET A=0
260 LET B=-4444
270 LET C=-1E-33
280 PRINT
290 IF C^B = A^C THEN 320
300 PRINT "*** TEST FAILED: 'IF' TESTED AS UNEQUAL ***"
310 GOTO 350
320 PRINT "IF OVERFLOW AND ZERO TO NEGATIVE POWER EXCEPTIONS"
330 PRINT " WERE REPORTED, THEN"
340 PRINT "*** TEST PASSED ***"
350 PRINT
360 PRINT "                  END TEST."
370 PRINT
380 PRINT "END PROGRAM 177"
390 END

```

---

PROGRAM FILE 177: EXCEPTION - EVALUATION OF NUMERIC  
EXPRESSIONS IN THE IF-STATEMENT.

ANSI STANDARD 7.5, 10.2

SECTION 177.1: EXCEPTION - EVALUATION OF NUMERIC  
EXPRESSIONS IN THE IF-STATEMENT.

THIS SECTION TESTS THE EFFECT OF COMPARING EXPRESSIONS  
WHICH CAUSE NON-FATAL EXCEPTIONS.

TO PASS THIS TEST:

- 1) TWO EXCEPTIONS MUST BE REPORTED: OVERFLOW,  
AND ZERO RAISED TO A NEGATIVE POWER, AND
- 2) POSITIVE MACHINE INFINITY MUST BE SUPPLIED AS  
THE VALUES OF THE EXPRESSIONS, AND
- 3) THE IF MUST TEST EQUAL, AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO EXECUTE:

IF (-1E-33) ^ (-4444) = 0 ^ (-1E-33)

% OVERFLOW IN LINE 290

% ZERO TO A NEGATIVE POWER IN LINE 290  
IF OVERFLOW AND ZERO TO NEGATIVE POWER EXCEPTIONS  
WERE REPORTED, THEN  
\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 177

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

10 PRINT "PROGRAM FILE 178: EXCEPTION - UNDERFLOW IN THE EVALUATION"  
20 PRINT " OF NUMERIC EXPRESSIONS IN THE IF-STATEMENT."  
30 PRINT " ANSI STANDARD 7.4, 7.6, 10.2"  
40 PRINT  
50 PRINT "SECTION 178.1: EXCEPTION - UNDERFLOW IN THE EVALUATION"  
60 PRINT " OF NUMERIC EXPRESSIONS IN THE IF-STATEMENT."  
70 PRINT  
80 PRINT "THIS SECTION TESTS THE EFFECT OF UNDERFLOW IN NUMERIC"  
90 PRINT "EXPRESSIONS IN THE CONTEXT OF THE IF STATEMENT."  
100 PRINT  
130 PRINT "TO PASS THIS TEST:"  
140 PRINT  
150 PRINT " 1) AN UNDERFLOW EXCEPTION MAY BE REPORTED"  
160 PRINT " (NOT MANDATORY), AND"  
170 PRINT  
180 PRINT " 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE "  
190 PRINT " EXPRESSION, AND "  
200 PRINT  
210 PRINT " 3) THE IF MUST TEST EQUAL AND EXECUTION CONTINUES."  
220 PRINT  
222 PRINT " BEGIN TEST."  
225 PRINT  
230 PRINT "ABOUT TO EXECUTE:"  
240 PRINT " IF (-1E33) ^ (-3333) = 0"  
250 LET A=-1E33  
260 LET B=-3333  
270 PRINT  
280 IF A^B = 0 THEN 310  
290 PRINT "\*\*\* TEST FAILED: 'IF' TESTED AS UNEQUAL \*\*\*"  
300 GOTO 320  
310 PRINT "\*\*\* TEST PASSED \*\*\*"  
320 PRINT  
330 PRINT " END TEST."  
340 PRINT  
350 PRINT "END PROGRAM 178"  
360 END

-----

PROGRAM FILE 178: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF NUMERIC EXPRESSIONS IN THE IF-STATEMENT.  
ANSI STANDARD 7.4, 7.6, 10.2

SECTION 178.1: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF NUMERIC EXPRESSIONS IN THE IF-STATEMENT.

THIS SECTION TESTS THE EFFECT OF UNDERFLOW IN NUMERIC  
EXPRESSIONS IN THE CONTEXT OF THE IF STATEMENT.

TO PASS THIS TEST:

- 1) AN UNDERFLOW EXCEPTION MAY BE REPORTED  
(NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE  
EXPRESSION, AND
- 3) THE IF MUST TEST EQUAL AND EXECUTION CONTINUES.

BEGIN TEST.

ABOUT TO EXECUTE:

```
IF (-1E33) ^ (-3333) = 0
```

% UNDERFLOW IN LINE 280

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 178

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 179: EXCEPTION - LOG OF ZERO IN"  
20 PRINT "          ON-GOTO-STATEMENT."  
30 PRINT "      ANSI STANDARD 8.5, 10.2"  
40 PRINT  
50 PRINT "SECTION 179.1: EXCEPTION - LOG OF ZERO IN"  
60 PRINT "          ON-GOTO-STATEMENT."  
70 PRINT  
80 PRINT "TO PASS THIS TEST:"  
90 PRINT  
100 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"  
110 PRINT "          DISPLAYED, AND"  
120 PRINT  
130 PRINT "      2) EXECUTION MUST TERMINATE"  
140 PRINT  
150 PRINT "          BEGIN TEST."  
160 PRINT
```

```
170 PRINT "ABOUT TO EXECUTE:"
180 PRINT "    ON LOG (0) GOTO ..."
190 LET A=0
210 ON LOG(A) GOTO 220,230,240,250,260
212 PRINT "CONTROL FELL THROUGH ON-GOTO."
215 GOTO 300
220 LET I=1
225 GOTO 290
230 LET I=2
235 GOTO 290
240 LET I=3
245 GOTO 290
250 LET I=4
255 GOTO 290
260 LET I=5
265 GOTO 290
290 PRINT "ON-GOTO JUMPED TO LINE-NUMBER #";I;" IN THE LIST."
300 PRINT
1270 PRINT "**** TEST FAILED: EXECUTION DID NOT TERMINATE. ****"
1280 PRINT
1290 PRINT "                END TEST."
1300 PRINT
1310 PRINT "END PROGRAM 179."
1320 END
```

---

PROGRAM FILE 179: EXCEPTION - LOG OF ZERO IN  
ON-GOTO-STATEMENT.

ANSI STANDARD 8.5, 10.2

SECTION 179.1: EXCEPTION - LOG OF ZERO IN  
ON-GOTO-STATEMENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE

BEGIN TEST.

ABOUT TO EXECUTE:

ON LOG (0) GOTO ...

% LOG OF ZERO IN LINE 210

---

---

---

---

---

```
10 PRINT "PROGRAM FILE 180: EXCEPTION - EVALUATION OF NUMERIC"
20 PRINT "      EXPRESSIONS IN THE ON-GOTO STATEMENT."
30 PRINT "      ANSI STANDARD 7.5, 10.2, 10.5"
40 PRINT
50 PRINT "SECTION 180.1: EXCEPTION - EVALUATION OF NUMERIC"
60 PRINT "      EXPRESSIONS IN THE ON-GOTO STATEMENT."
70 PRINT
80 PRINT "THIS SECTION TESTS THE EFFECT OF USING EXPRESSIONS,"
90 PRINT "WHICH CAUSE NON-FATAL EXCEPTIONS, TO CONTROL THE ON-GOTO."
100 PRINT
130 PRINT "TO PASS THIS TEST:"
140 PRINT
150 PRINT "      1) TWO EXCEPTIONS MUST BE REPORTED: DIVISION "
160 PRINT "          BY ZERO AND ON-GOTO OUT OF RANGE, AND"
170 PRINT
180 PRINT "      2) EXECUTION MUST TERMINATE."
190 PRINT
193 PRINT "                  BEGIN TEST."
196 PRINT
200 PRINT "ABOUT TO EXECUTE:"
210 PRINT "      ON 1E-33 / 0 GOTO ..."
220 LET A=0
230 LET C=1E-33
240 PRINT
250 ON C/A GOTO 280,300,320
260 LET I=0
270 GOTO 340
280 LET I=1
290 GOTO 340
300 LET I=2
310 GOTO 340
320 LET I=3
330 GOTO 340
340 PRINT
350 PRINT "      PATH TAKEN FOR CONTROL-EXPRESSION = ";I
360 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE ***"
370 PRINT
380 PRINT "                  END TEST."
390 PRINT
400 PRINT "END PROGRAM 180"
410 END
```

---

PROGRAM FILE 180: EXCEPTION - EVALUATION OF NUMERIC  
 EXPRESSIONS IN THE ON-GOTO STATEMENT.  
 ANSI STANDARD 7.5, 10.2, 10.5

SECTION 180.1: EXCEPTION - EVALUATION OF NUMERIC  
 EXPRESSIONS IN THE ON-GOTO STATEMENT.

THIS SECTION TESTS THE EFFECT OF USING EXPRESSIONS,  
WHICH CAUSE NON-FATAL EXCEPTIONS, TO CONTROL THE ON-GOTO.

TO PASS THIS TEST:

- 1) TWO EXCEPTIONS MUST BE REPORTED: DIVISION BY ZERO AND ON-GOTO OUT OF RANGE, AND
- 2) EXECUTION MUST TERMINATE.

BEGIN TEST.

ABOUT TO EXECUTE:

ON 1E-33 / 0 GOTO ...

% DIVISION BY ZERO IN LINE 250

? ON EVALUATED OUT OF RANGE IN LINE 250

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 181: EXCEPTION - UNDERFLOW IN THE EVALUATION"
20 PRINT "          OF THE EXP FUNCTION IN THE ON-GOTO STATEMENT."
30 PRINT "      ANSI STANDARD 7.4, 8.6, 10.2, 10.5"
40 PRINT
50 PRINT "SECTION 181.1: EXCEPTION - UNDERFLOW IN THE EVALUATION"
60 PRINT "          OF THE EXP FUNCTION IN THE ON-GOTO STATEMENT."
70 PRINT
130 PRINT "TO PASS THIS TEST:"
140 PRINT
150 PRINT "      1) AN UNDERFLOW EXCEPTION MAY BE REPORTED"
160 PRINT "          (NOT MANDATORY), AND"
170 PRINT
180 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE "
190 PRINT "          EXP FUNCTION,"
200 PRINT
210 PRINT "      3) AN EXCEPTION FOR ON-GOTO OUT OF RANGE MUST BE"
220 PRINT "          REPORTED, AND"
230 PRINT
240 PRINT "      4) EXECUTION MUST TERMINATE."
250 PRINT
252 PRINT "                  BEGIN TEST."
255 PRINT
260 PRINT "ABOUT TO EXECUTE:"
270 PRINT "    ON EXP (-1E11) GOTO ..."
280 PRINT
290 LET A=-1E11
300 ON EXP(A) GOTO 330,350,370
310 LET I=0
320 GOTO 390
330 LET I=1
340 GOTO 390
350 LET I=2

```

```

360 GOTO 390
370 LET I=3
380 GOTO 390
390 PRINT
400 PRINT "      PATH TAKEN FOR CONTROL-EXPRESSION = ";I
405 PRINT
410 PRINT "**** TEST FAILED: EXECUTION DID NOT TERMINATE ****"
420 PRINT
430 PRINT "                  END TEST."
440 PRINT
450 PRINT "END PROGRAM 181"
460 END

```

---

PROGRAM FILE 181: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF THE EXP FUNCTION IN THE ON-GOTO STATEMENT.

ANSI STANDARD 7.4, 8.6, 10.2, 10.5

SECTION 181.1: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF THE EXP FUNCTION IN THE ON-GOTO STATEMENT.

TO PASS THIS TEST:

- 1) AN UNDERFLOW EXCEPTION MAY BE REPORTED  
(NOT MANDATORY), AND
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE  
EXP FUNCTION,
- 3) AN EXCEPTION FOR ON-GOTO OUT OF RANGE MUST BE  
REPORTED, AND
- 4) EXECUTION MUST TERMINATE.

BEGIN TEST.

ABOUT TO EXECUTE:

ON EXP (-1E11) GOTO ...

% UNDERFLOW IN EXP IN LINE 300

? ON EVALUATED OUT OF RANGE IN LINE 300

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

10 PRINT "PROGRAM FILE 182: EXCEPTION - NEGATIVE QUANTITY RAISED TO"  
20 PRINT " A NON-INTEGRAL POWER IN FOR-STATEMENT."  
30 PRINT " ANSI STANDARD 7.5, 11.2"  
40 PRINT

```
50 PRINT "SECTION 182.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO"
55 PRINT "          A NON-INTEGRAL POWER IN FOR-STATEMENT."
60 PRINT
70 PRINT "TO PASS THIS TEST:"
80 PRINT
90 PRINT "      1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE"
100 PRINT "          DISPLAYED, AND"
110 PRINT
120 PRINT "      2) EXECUTION MUST TERMINATE"
130 PRINT
140 PRINT "          BEGIN TEST."
150 PRINT
160 PRINT "ABOUT TO EXECUTE:"
165 PRINT "    FOR I= -2 ^ 1E-33 TO 9"
170 PRINT
175 LET A=-2
180 LET B=1E-33
190 FOR I= A^B TO 9
200 LET X=I
210 GOTO 220
215 NEXT I
220 PRINT "I WAS INITIALIZED TO:";X
240 PRINT
250 PRINT "*** TEST FAILED: EXECUTION DID NOT TERMINATE. ***"
260 PRINT
270 PRINT "          END TEST."
280 PRINT
290 PRINT "END PROGRAM 182."
300 END
```

---

PROGRAM FILE 182: EXCEPTION - NEGATIVE QUANTITY RAISED TO  
A NON-INTEGRAL POWER IN FOR-STATEMENT.  
ANSI STANDARD 7.5, 11.2

SECTION 182.1: EXCEPTION - NEGATIVE QUANTITY RAISED TO  
A NON-INTEGRAL POWER IN FOR-STATEMENT.

TO PASS THIS TEST:

- 1) A MESSAGE IDENTIFYING THE EXCEPTION MUST BE  
DISPLAYED, AND
- 2) EXECUTION MUST TERMINATE

BEGIN TEST.

ABOUT TO EXECUTE:

FOR I= -2 ^ 1E-33 TO 9

? NEGATIVE VALUE RAISED TO NON-INTEGRAL POWER IN LINE 190

```
*****
*****  
10 PRINT "PROGRAM FILE 183: EXCEPTION - EVALUATION OF NUMERIC"  
20 PRINT "      EXPRESSIONS IN THE FOR-STATEMENT."  
30 PRINT "      ANSI STANDARD 7.5, 11.2"  
40 PRINT  
50 PRINT "SECTION 183.1: EXCEPTION - EVALUATION OF NUMERIC"  
60 PRINT "      EXPRESSIONS IN THE FOR-STATEMENT."  
70 PRINT  
80 PRINT "THIS SECTION TESTS THE EFFECT OF USING EXPRESSIONS,"  
90 PRINT "WHICH CAUSE NON-FATAL EXCEPTIONS, TO CONTROL A"  
100 PRINT "FOR-NEXT LOOP."  
110 PRINT  
120 PRINT "TO PASS THIS TEST:"  
130 PRINT  
140 PRINT "      1) A DIVISION BY ZERO EXCEPTION MUST BE REPORTED, AND"  
160 PRINT  
170 PRINT "      2) THE FOR-LOOP MUST EXECUTE 2 TIMES."  
180 PRINT  
190 PRINT "                  BEGIN TEST."  
250 PRINT  
260 PRINT "ABOUT TO ATTEMPT EXECUTION OF:"  
270 PRINT "      FOR I = ATN((-9) / 0) TO 0"  
280 LET J=-1.5707963  
300 LET Z=0  
310 LET C=-9  
320 PRINT  
330 PRINT "ITERATION","CONTROL-VARIABLE VALUES"  
340 PRINT "COUNT","ACTUAL","SHOULD BE"  
350 LET X=0  
360 FOR I = ATN(C/Z) TO Z  
370 LET X=X+1  
380 PRINT X,I,J  
390 IF ABS(1-(I/J)) > .01 THEN 530  
400 LET J=J+1  
410 NEXT I  
420 PRINT "EXITING LOOP."  
430 LET J=.4292037  
435 PRINT " ",I,J  
440 IF ABS(1-(I/J)) > .01 THEN 530  
450 IF X=2 THEN 490  
460 PRINT  
470 PRINT "NO. TIMES THROUGH LOOP SHOULD BE 2, ACTUALLY =" ; X  
480 GOTO 550  
490 PRINT  
500 PRINT "IF ZERO-DIVIDE REPORTED, THEN"  
510 PRINT "*** TEST PASSED ***"  
520 GOTO 570  
530 PRINT  
540 PRINT "CONTROL-VARIABLE SHOULD BE";J;" ACTUALLY =" ; I  
550 PRINT  
560 PRINT "*** TEST FAILED ***"  
570 PRINT
```

```
580 PRINT " END TEST."
590 PRINT
600 PRINT "END PROGRAM 183"
610 END
```

---

PROGRAM FILE 183: EXCEPTION - EVALUATION OF NUMERIC  
EXPRESSIONS IN THE FOR-STATEMENT.

ANSI STANDARD 7.5, 11.2

SECTION 183.1: EXCEPTION - EVALUATION OF NUMERIC  
EXPRESSIONS IN THE FOR-STATEMENT.

THIS SECTION TESTS THE EFFECT OF USING EXPRESSIONS,  
WHICH CAUSE NON-FATAL EXCEPTIONS, TO CONTROL A  
FOR-NEXT LOOP.

TO PASS THIS TEST:

- 1) A DIVISION BY ZERO EXCEPTION MUST BE REPORTED, AND
- 2) THE FOR-LOOP MUST EXECUTE 2 TIMES.

BEGIN TEST.

ABOUT TO ATTEMPT EXECUTION OF:

FOR I = ATN((-9) / 0) TO 0

ITERATION COUNT	CONTROL-VARIABLE VALUES	ACTUAL SHOULD BE
-----------------	-------------------------	------------------

% DIVISION BY ZERO IN LINE 360

1	-1.5708	-1.5708
2	-0.570796	-0.570796

EXITING LOOP.

0.429204	0.429204
----------	----------

IF ZERO-DIVIDE REPORTED, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 183

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```
10 PRINT "PROGRAM FILE 184: EXCEPTION - UNDERFLOW IN THE EVALUATION"
20 PRINT "      OF NUMERIC EXPRESSIONS IN THE FOR-STATEMENT."
30 PRINT "      ANSI STANDARD 7.4, 7.6, 11.2"
40 PRINT
```

```
50 PRINT "SECTION 184.1: EXCEPTION - UNDERFLOW IN THE EVALUATION"
60 PRINT "          OF NUMERIC EXPRESSIONS IN THE FOR-STATEMENT."
70 PRINT
80 PRINT "THIS SECTION TESTS THE EFFECT OF UNDERFLOW IN NUMERIC"
90 PRINT "EXPRESSIONS IN THE CONTEXT OF THE FOR STATEMENT."
100 PRINT
110 PRINT "TO PASS THIS TEST:"
120 PRINT
130 PRINT "      1) AN UNDERFLOW EXCEPTION MAY BE REPORTED"
140 PRINT "          (NOT MANDATORY),"
150 PRINT
160 PRINT "      2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE "
170 PRINT "          EXPRESSION, AND"
180 PRINT
190 PRINT "      3) THE FOR-LOOP MUST EXECUTE 4 TIMES."
200 PRINT
210 PRINT "                  BEGIN TEST."
220 PRINT
230 PRINT "ABOUT TO EXECUTE:"
240 PRINT "    FOR I = .1 ^ 99999 TO 1E-36 STEP 3E-37"
250 LET J=0
260 LET C=99999
270 PRINT
280 PRINT "ITERATION","CONTROL-VARIABLE VALUES"
290 PRINT "COUNT","ACTUAL","SHOULD BE"
300 LET X=0
310 FOR I = .1^C TO 1E-36 STEP 3E-37
320 LET X=X+1
330 PRINT X,I,J
340 IF J<>0 THEN 370
350 IF I=0 THEN 380
360 GOTO 510
370 IF ABS(1-(I/J)) > .01 THEN 510
380 LET J=J+3E-37
390 NEXT I
400 PRINT "EXITING LOOP."
410 LET J=12E-37
420 PRINT " ",I,J
430 IF ABS(1-(I/J)) > .01 THEN 510
440 IF X=4 THEN 480
450 PRINT
460 PRINT "NO. TIMES THROUGH LOOP SHOULD BE 4, ACTUALLY =" ;X
470 GOTO 530
480 PRINT
490 PRINT "*** TEST PASSED ***"
500 GOTO 550
510 PRINT
520 PRINT "CONTROL-VARIABLE SHOULD BE ";J;" ACTUALLY = ";I
530 PRINT
540 PRINT "*** TEST FAILED ***"
550 PRINT
560 PRINT "                  END TEST."
```

```
570 PRINT
580 PRINT "END PROGRAM 184"
590 END
```

---

PROGRAM FILE 184: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF NUMERIC EXPRESSIONS IN THE FOR-STATEMENT.  
ANSI STANDARD 7.4, 7.6, 11.2

SECTION 184.1: EXCEPTION - UNDERFLOW IN THE EVALUATION  
OF NUMERIC EXPRESSIONS IN THE FOR-STATEMENT.

THIS SECTION TESTS THE EFFECT OF UNDERFLOW IN NUMERIC  
EXPRESSIONS IN THE CONTEXT OF THE FOR STATEMENT.

TO PASS THIS TEST:

- 1) AN UNDERFLOW EXCEPTION MAY BE REPORTED  
(NOT MANDATORY),
- 2) ZERO MUST BE SUPPLIED AS THE VALUE OF THE  
EXPRESSION, AND
- 3) THE FOR-LOOP MUST EXECUTE 4 TIMES.

BEGIN TEST.

ABOUT TO EXECUTE:

```
FOR I = .1 ^ 99999 TO 1E-36 STEP 3E-37
```

ITERATION COUNT	CONTROL-VARIABLE VALUES
	ACTUAL SHOULD BE

% UNDERFLOW IN LINE 310		
1	0	0
2	3.00000E-37	3.00000E-37
3	6.00000E-37	6.00000E-37
4	9.00000E-37	9.00000E-37

EXITING LOOP.	1.20000E-36	1.20000E-36
---------------	-------------	-------------

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 184

---



---



---

```
10 PRINT "PROGRAM FILE 185: ERROR - MISSING KEYWORD LET."
20 PRINT "    ANSI STANDARD 9.2, 9.4"
30 PRINT
```

```

40 PRINT "SECTION 185.1: ERROR - MISSING KEYWORD LET."
50 PRINT
60 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
70 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
80 PRINT
90 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
100 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
110 PRINT "        BY THE PROCESSOR, OR"
120 PRINT
130 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
140 PRINT
150 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
160 PRINT "FOR DETAILED CRITERIA."
170 PRINT
180 PRINT "                BEGIN TEST."
190 PRINT
200 PRINT "ABOUT TO EXECUTE 'LET X1 = 11' AND THEN 'X1 = 12'"
230 LET X1=11
240 X1=12
280 PRINT "VALUE OF X1 = ";X1
290 PRINT
300 PRINT "                END TEST."
310 PRINT
320 PRINT "END PROGRAM 185"
330 END

```

---

? ILLEGAL KEYWORD IN LINE 240

```
*****  
*****  
*****
```

```

10 PRINT "PROGRAM FILE 186: EXTRA SPACES HAVE NO EFFECT."
20 PRINT "    ANSI STANDARD 3.4"
30 PRINT
40 PRINT "SECTION 186.1: EXTRA SPACES HAVE NO EFFECT."
50 PRINT
60 PRINT "THIS PROGRAM TESTS THAT, OTHER THAN IN QUOTED-STRINGS"
70 PRINT "AND UNQUOTED-STRINGS, AN ARBITRARY NUMBER OF SPACES MAY"
80 PRINT "APPEAR IN THE PROGRAM ANYWHERE A SINGLE SPACE IS ALLOWED"
90 PRINT "WITHOUT AFFECTING ITS EXECUTION."
100 PRINT
110 PRINT "                BEGIN TEST."
120 PRINT
130     LET      I      =      -      3      +      1      +      2
140     GO          TO      160
150 LET I=I+1
160             IF      I      =      1      THEN      180
170     LET      I      =      +      2      +      I
180 IF I=2 THEN 210
190 PRINT "*** TEST FAILED ***"

```

```
200 GOTO 220
210 PRINT "*** TEST PASSED ***"
220 PRINT
230 PRINT " END TEST."
240 PRINT
250 PRINT "END PROGRAM 186"
260 END
```

---

PROGRAM FILE 186: EXTRA SPACES HAVE NO EFFECT.  
ANSI STANDARD 3.4

SECTION 186.1: EXTRA SPACES HAVE NO EFFECT.

THIS PROGRAM TESTS THAT, OTHER THAN IN QUOTED-STRINGS  
AND UNQUOTED-STRINGS, AN ARBITRARY NUMBER OF SPACES MAY  
APPEAR IN THE PROGRAM ANYWHERE A SINGLE SPACE IS ALLOWED  
WITHOUT AFFECTING ITS EXECUTION.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 186

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 187: ERROR - SPACES AT THE BEGINNING OF A LINE."
15 PRINT " ANSI STANDARD 3.4, 4.4"
20 PRINT
30 PRINT "SECTION 187.1: ERROR - SPACES AT THE BEGINNING OF A LINE."
40 PRINT
50 PRINT " THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"
60 PRINT "ACCEPTS PROGRAMS CONTAINING LINES BEGINNING WITH SPACES."
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT
110 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
120 PRINT " ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT " BY THE PROCESSOR, OR"
140 PRINT
150 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT " BEGIN TEST."
210 PRINT
```

```
220 PRINT "THE PROCESSOR HAS EXECUTED A LINE BEGINNING WITH A SPACE."  
230 PRINT  
240 PRINT " END TEST."  
250 PRINT  
260 PRINT "END PROGRAM 187"  
270 END
```

---

? MISSING LINE NUMBER FOLLOWING LINE 210

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 188: ERROR - SPACES WITHIN LINE-NUMBERS."  
20 PRINT " ANSI STANDARD 3.4, 4.4"  
30 PRINT  
40 PRINT "SECTION 188.1: ERROR - SPACES WITHIN LINE-NUMBERS."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR ACCEPTS"  
70 PRINT "PROGRAMS CONTAINING SPACES WITHIN LINE-NUMBERS."  
80 PRINT  
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
110 PRINT  
120 PRINT " 1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
130 PRINT " ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
140 PRINT " BY THE PROCESSOR, OR"  
150 PRINT  
160 PRINT " 2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
170 PRINT  
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
190 PRINT "FOR DETAILED CRITERIA."  
200 PRINT  
210 PRINT  
220 PRINT " BEGIN TEST."  
230 PRINT  
240 PRINT "THE PROCESSOR HAS EXECUTED TWO LINES CONTAINING"  
250 PRINT "SPACES WITHIN LINE-NUMBERS."  
260 PRINT  
270 PRINT " END TEST."  
280 PRINT  
290 PRINT "END PROGRAM 188"  
300 END
```

---

? LINE NUMBER OUT OF ORDER FOLLOWING LINE 230  
? LINE NUMBER OUT OF ORDER FOLLOWING LINE 230

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 189: ERROR - SPACES WITHIN KEYWORDS."  
20 PRINT "    ANSI STANDARD 3.4"  
30 PRINT  
40 PRINT "SECTION 189.1: ERROR - SPACES WITHIN KEYWORDS."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"  
70 PRINT "ACCEPTS PROGRAMS CONTAINING SPACES WITHIN KEYWORDS."  
80 PRINT  
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
110 PRINT  
120 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
130 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
140 PRINT "        BY THE PROCESSOR, OR"  
150 PRINT  
160 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
170 PRINT  
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
190 PRINT "FOR DETAILED CRITERIA."  
200 PRINT  
210 PRINT "                                BEGIN TEST."  
220 PRINT  
230 LET X=0  
240 L E T X=10  
250 I F X=10 THEN 370  
260 IF X=10 THEN 310  
270 I F X=0 THEN 340  
280 PRINT "THE PROCESSOR DID NOT EXECUTE THE STATEMENTS"  
290 PRINT "CONTAINING SPACES WITHIN THE KEYWORDS LET AND IF."  
300 GOTO 390  
310 PRINT "THE PROCESSOR EXECUTED A STATEMENT CONTAINING A"  
320 PRINT "SPACE WITHIN THE KEYWORD LET."  
330 GOTO 390  
340 PRINT "THE PROCESSOR EXECUTED A STATEMENT CONTAINING A"  
350 PRINT "SPACE WITHIN THE KEYWORD IF."  
360 GOTO 390  
370 PRINT "THE PROCESSOR EXECUTED STATEMENTS CONTAINING SPACES"  
380 PRINT "WITHIN THE KEYWORDS LET AND IF."  
390 PRINT  
400 PRINT "                                END TEST."  
410 PRINT  
420 PRINT "END PROGRAM 189"  
430 END
```

---

```
? ILLEGAL KEYWORD IN LINE 240  
? ILLEGAL KEYWORD IN LINE 250  
? ILLEGAL KEYWORD IN LINE 270
```

```
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 190: ERROR - NO SPACES BEFORE KEYWORDS."  
20 PRINT "    ANSI STANDARD 3.4"  
30 PRINT  
40 PRINT "SECTION 190.1: ERROR - NO SPACES BEFORE KEYWORDS."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR ACCEPTS"  
70 PRINT "PROGRAMS CONTAINING KEYWORDS WHICH ARE NOT PRECEDED BY"  
80 PRINT "AT LEAST ONE SPACE."  
90 PRINT  
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
120 PRINT  
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
150 PRINT "        BY THE PROCESSOR, OR"  
160 PRINT  
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
180 PRINT  
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
200 PRINT "FOR DETAILED CRITERIA."  
210 PRINT  
220 PRINT "                                BEGIN TEST."  
230 PRINT  
240 LET X=0  
250LET X=10  
260IF X=10THEN 390  
270 IF X=10 THEN 330  
280IF X=0 THEN 360  
290 PRINT "THE PROCESSOR DID NOT EXECUTE THE STATEMENTS"  
300 PRINT "WHICH DID NOT CONTAIN A SPACE BEFORE THE KEYWORDS"  
310 PRINT "LET AND IF."  
320 GOTO 410  
330 PRINT "THE PROCESSOR EXECUTED A STATEMENT WHICH DID NOT"  
340 PRINT "CONTAIN A SPACE BEFORE THE KEYWORD LET."  
350 GOTO 410  
360 PRINT "THE PROCESSOR EXECUTED A STATEMENT WHICH DID NOT"  
370 PRINT "CONTAIN A SPACE BEFORE THE KEYWORD IF."  
380 GOTO 410  
390 PRINT "THE PROCESSOR EXECUTED STATEMENTS WHICH DID NOT"  
400 PRINT "CONTAIN A SPACE BEFORE THE KEYWORDS LET AND IF."  
410 PRINT  
420 PRINT "                                END TEST."  
430 PRINT  
440 PRINT "END PROGRAM 190"  
450 END
```

---

```
? ILLEGAL LINE NUMBER FOLLOWING LINE 240
? ILLEGAL LINE NUMBER FOLLOWING LINE 240
? ILLEGAL LINE NUMBER FOLLOWING LINE 270
```

```
*****
*****
```

```
10 PRINT "PROGRAM FILE 191: ERROR - NO SPACES AFTER KEYWORDS."
20 PRINT "    ANSI STANDARD 3.4"
30 PRINT
40 PRINT "SECTION 191.1: ERROR - NO SPACES AFTER KEYWORDS."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"
70 PRINT "ACCEPTS PROGRAMS CONTAINING KEYWORDS WHICH ARE NOT FOLLOWED"
80 PRINT "BY AT LEAST ONE SPACE."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                                BEGIN TEST."
230 PRINT
240 LET X=0
250 LET X=10
260 IF X=10 THEN 390
270 IF X=10 THEN 330
280 IF X=0 THEN 360
290 PRINT "THE PROCESSOR DID NOT EXECUTE THE STATEMENTS"
300 PRINT "WHICH DID NOT CONTAIN A SPACE AFTER THE KEYWORDS"
310 PRINT "LET AND IF."
320 GOTO 410
330 PRINT "THE PROCESSOR EXECUTED A STATEMENT WHICH DID NOT"
340 PRINT "CONTAIN A SPACE AFTER THE KEYWORD LET."
350 GOTO 410
360 PRINT "THE PROCESSOR EXECUTED A STATEMENT WHICH DID NOT"
370 PRINT "CONTAIN A SPACE AFTER THE KEYWORD IF."
380 GOTO 410
390 PRINT "THE PROCESSOR EXECUTED STATEMENTS WHICH DID NOT"
400 PRINT "CONTAIN A SPACE AFTER THE KEYWORDS LET AND IF."
410 PRINT
420 PRINT "                                END TEST."
```

```
430 PRINT
440 PRINT "END PROGRAM 191"
450 END
```

---

```
? ILLEGAL KEYWORD IN LINE 250
? ILLEGAL KEYWORD IN LINE 260
? ILLEGAL KEYWORD IN LINE 280
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 192: ERROR - PRINT-ITEM QUOTED STRINGS"
15 PRINT "          CONTAINING SINGLE QUOTE."
20 PRINT "          ANSI STANDARD 3.2, 12.2, 12.4"
30 PRINT
40 PRINT "SECTION 192.1: ERROR - PRINT-ITEM QUOTED STRINGS"
45 PRINT "          CONTAINING SINGLE QUOTE."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING A SINGLE OCCURRENCE OF THE QUOTE CHARACTER WITHIN"
80 PRINT "THE QUOTED STRING OF A PRINT-STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                      BEGIN TEST."
230 PRINT
240 PRINT "A PRINT-ITEM CONTAINING AN ASTERISK, A SINGLE "
250 PRINT "OCCURRENCE OF THE QUOTE CHARACTER AND A QUESTION MARK"
260 PRINT "APPEARS IN THE NEXT STATEMENT AFTER THIS PARAGRAPH. CHECK"
270 PRINT "THE NEXT LINE FOR THE PROCESSOR'S INTERPRETATION."
280 PRINT "          *?""
290 PRINT
300 PRINT "                      END TEST."
310 PRINT
320 PRINT "END PROGRAM 192"
330 END
```

---

```
? ILLEGAL FORMULA IN LINE 280
```

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 193: ERROR - PRINT-ITEM QUOTED STRINGS"  
15 PRINT "          CONTAINING DOUBLE QUOTES."  
20 PRINT "          ANSI STANDARD 3.2, 12.2, 12.4"  
30 PRINT  
40 PRINT "SECTION 193.1: ERROR - PRINT-ITEM QUOTED STRINGS"  
45 PRINT "          CONTAINING DOUBLE QUOTES."  
50 PRINT  
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"  
70 PRINT "ACCEPTS PROGRAMS CONTAINING TWO CONSECUTIVE OCCURRENCES"  
80 PRINT "OF THE QUOTE CHARACTER WITHIN THE QUOTED STRING OF A"  
90 PRINT "PRINT STATEMENT."  
100 PRINT  
110 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
120 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
130 PRINT  
140 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
150 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
160 PRINT "          BY THE PROCESSOR, OR"  
170 PRINT  
180 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
190 PRINT  
200 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
210 PRINT "FOR DETAILED CRITERIA."  
220 PRINT  
230 PRINT "                                BEGIN TEST."  
240 PRINT  
250 PRINT "A PRINT-ITEM CONTAINING AN ASTERISK, TWO CONSECUTIVE"  
260 PRINT "OCCURRENCES OF THE QUOTE CHARACTER AND A QUESTION MARK "  
270 PRINT "APPEARS IN THE NEXT STATEMENT AFTER THIS PARAGRAPH."  
280 PRINT "CHECK THE NEXT LINE OF OUTPUT FOR THE PROCESSOR'S"  
290 PRINT "INTERPRETATION."  
300 PRINT "          *""?!"  
310 PRINT  
320 PRINT "                                END TEST."  
330 PRINT  
340 PRINT "END PROGRAM 193"  
350 END
```

---

? ILLEGAL FORMULA IN LINE 300

```
*****  
*****  
*****  
  
10 PRINT "PROGRAM FILE 194: ERROR - ASSIGNED QUOTED STRINGS "  
15 PRINT "          CONTAINING SINGLE QUOTE."  
20 PRINT "          ANSI STANDARD 3.2, 9.2"  
30 PRINT
```

```

40 PRINT "SECTION 194.1: ERROR - ASSIGNED QUOTED STRINGS "
45 PRINT "          CONTAINING SINGLE QUOTE."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING A SINGLE OCCURRENCE OF THE QUOTE CHARACTER WITHIN"
80 PRINT "THE QUOTED STRING OF A LET-STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                                BEGIN TEST."
230 PRINT
240 PRINT "ABOUT TO ASSIGN TO A$ A QUOTED-STRING CONTAINING:"
250 PRINT "ASTERISK, QUOTE, QUESTION-MARK."
260 LET A$="*?""
270 PRINT "A$=";A$
310 PRINT
320 PRINT "                                END TEST."
330 PRINT
340 PRINT "END PROGRAM 194"
350 END

```

---

? ? WAS SEEN WHERE A LINE TERMINATOR OR APOSTROPHE WAS EXPECTED IN LINE 260

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 195: ERROR - ASSIGNED QUOTED STRING "
15 PRINT "          CONTAINING DOUBLE QUOTES."
20 PRINT "      ANSI STANDARD 3.2, 9.2"
30 PRINT
40 PRINT "SECTION 195.1: ERROR - ASSIGNED QUOTED STRING "
45 PRINT "          CONTAINING DOUBLE QUOTES."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING TWO CONSECUTIVE OCCURRENCES OF THE QUOTE "
80 PRINT "CHARACTER WITHIN THE QUOTED STRING OF A LET-STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT

```

```

130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                                BEGIN TEST."
230 PRINT
240 PRINT "ABOUT TO ASSIGN TO A$ A QUOTED-STRING CONTAINING:"
250 PRINT "ASTERISK, QUOTE, QUOTE, QUESTION-MARK."
260 LET A$="*""?""
270 PRINT "A$=";A$
310 PRINT
320 PRINT "                                END TEST."
330 PRINT
340 PRINT "END PROGRAM 195"
350 END

```

---

? " WAS SEEN WHERE A LINE TERMINATOR OR APOSTROPHE WAS EXPECTED IN LINE 260

```
*****
*****
```

```

5 PRINT "PROGRAM FILE 196: LINE-NUMBERS WITH LEADING ZEROS."
10 PRINT "    ANSI STANDARD 4.2, 4.4"
15 PRINT
20 PRINT "SECTION 196.1: LINE-NUMBERS WITH LEADING ZEROS."
25 PRINT
30 PRINT "THIS PROGRAM TESTS THAT LINE-NUMBERS WITH LEADING ZEROS"
35 PRINT "ARE ACCEPTED AND HAVE NO EFFECT ON THE SEQUENCE OF EXECUTION."
40 PRINT
42 PRINT "                                BEGIN TEST."
44 PRINT
45 LET X=0
50 IF X<>0 THEN 100
052 LET X=X+1
55 IF X<>1 THEN 100
0057 LET X=X+1
059 IF X<>2 THEN 100
60 LET X=X+1
0062 IF X<>3 THEN 100
70 PRINT "*** TEST PASSED ***"
80 GOTO 110
100 PRINT "*** TEST FAILED ***"
110 PRINT
120 PRINT "                                END TEST."

```

```
130 PRINT
140 PRINT "END PROGRAM 196"
150 END
```

---

PROGRAM FILE 196: LINE-NUMBERS WITH LEADING ZEROS.  
ANSI STANDARD 4.2, 4.4

SECTION 196.1: LINE-NUMBERS WITH LEADING ZEROS.

THIS PROGRAM TESTS THAT LINE-NUMBERS WITH LEADING ZEROS  
ARE ACCEPTED AND HAVE NO EFFECT ON THE SEQUENCE OF EXECUTION.

BEGIN TEST.

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 196

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 197: ERROR - DUPLICATE LINE-NUMBERS."
20 PRINT "    ANSI STANDARD 4.4"
30 PRINT
40 PRINT "SECTION 197.1: ERROR - DUPLICATE LINE-NUMBERS."
50 PRINT
60 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
70 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
80 PRINT
90 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
100 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
110 PRINT "        BY THE PROCESSOR, OR"
120 PRINT
130 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
140 PRINT
150 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
160 PRINT "FOR DETAILED CRITERIA."
170 PRINT
180 PRINT                                BEGIN TEST."
190 PRINT
200 LET A=1111
210 LET B=1111
220 LET A=9999
220 LET B=9999
240 IF A=1111 THEN 290
250 IF B=1111 THEN 330
260 PRINT "THE PROCESSOR EXECUTED EACH OF TWO LET STATEMENTS WITH"
270 PRINT "DUPLICATE LINE-NUMBERS."
280 GOTO 380
290 IF B=1111 THEN 360
```

```
300 PRINT "THE PROCESSOR EXECUTED THE SECOND, BUT NOT THE FIRST"
310 PRINT "OF TWO LET STATEMENTS WITH DUPLICATE LINE-NUMBERS."
320 GOTO 380
330 PRINT "THE PROCESSOR EXECUTED THE FIRST, BUT NOT THE SECOND"
340 PRINT "OF TWO LET STATEMENTS WITH DUPLICATE LINE-NUMBERS."
350 GOTO 380
360 PRINT "THE PROCESSOR EXECUTED NEITHER OF TWO LET STATEMENTS"
370 PRINT "WITH DUPLICATE LINE-NUMBERS."
380 PRINT
390 PRINT "                                END TEST."
400 PRINT
410 PRINT "END PROGRAM 197"
420 END
```

---

? LINE NUMBER OUT OF ORDER FOLLOWING LINE 220

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 198: ERROR - LINES OUT OF ORDER."
20 PRINT "      ANSI STANDARD 4.4"
30 PRINT
40 PRINT "SECTION 198.1: ERROR - LINES OUT OF ORDER."
50 PRINT
60 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
70 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
80 PRINT
90 PRINT "      1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
100 PRINT "          ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
110 PRINT "          BY THE PROCESSOR, OR"
120 PRINT
130 PRINT "      2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
140 PRINT
150 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
160 PRINT "FOR DETAILED CRITERIA."
170 PRINT
180 PRINT "                                BEGIN TEST."
190 PRINT
200 LET A=1
220 LET A=2
210 LET A=3
230 ON A GOTO 250,270,290
250 PRINT "NEITHER OF THE UNORDERED LINES WAS EXECUTED."
260 GOTO 300
270 PRINT "THE LINES WERE EXECUTED IN ORDER OF THEIR LINE-NUMBERS."
280 GOTO 300
290 PRINT "THE LINES WERE EXECUTED IN THEIR PHYSICAL SEQUENCE."
300 PRINT
320 PRINT
330 PRINT "                                END TEST."
```

```
340 PRINT
350 PRINT "END PROGRAM 198."
360 PRINT
370 END
```

? LINE NUMBER OUT OF ORDER FOLLOWING LINE 220

A grid of black asterisks arranged in a pattern of alternating horizontal rows. The first row contains 10 asterisks, the second row contains 9, the third row contains 10, and so on, creating a decorative border-like effect.

```
9780 PRINT "PROGRAM FILE 199: ERROR - FIVE-DIGIT LINE-NUMBERS."
9790 PRINT "      ANSI STANDARD 4.2"
9800 PRINT
9810 PRINT "SECTION 199.1: ERROR - FIVE-DIGIT LINE-NUMBERS."
9820 PRINT
9830 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
9840 PRINT "CONTAINING LINE-NUMBERS CONSISTING OF FIVE DIGITS."
9850 PRINT
9860 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
9870 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
9880 PRINT
9890 PRINT "  1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
9900 PRINT "      ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
9910 PRINT "      BY THE PROCESSOR, OR"
9920 PRINT
9930 PRINT "  2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
9940 PRINT
9950 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
9960 PRINT "FOR DETAILED CRITERIA."
9970 PRINT
9980 PRINT "                                BEGIN TEST."
9990 PRINT
10000 PRINT "THE PROCESSOR HAS EXECUTED STATEMENTS WITH FIVE-DIGIT"
10010 PRINT "LINE-NUMBERS."
10020 PRINT
10030 PRINT "                                END TEST."
10040 PRINT
10050 PRINT "END PROGRAM 199"
10060 END
```

```
*****  
*****  
*****  
  
0 PRINT "A STATEMENT WITH A LINE-NUMBER OF 0 HAS BEEN EXECUTED."  
10 PRINT  
20 PRINT "PROGRAM FILE 200: ERROR - LINE-NUMBER ZERO."  
30 PRINT "    ANSI STANDARD 4.4"  
40 PRINT  
50 PRINT "SECTION 200.1: ERROR - LINE-NUMBER ZERO."  
60 PRINT  
170 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."  
180 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"  
190 PRINT  
200 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"  
210 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"  
220 PRINT "        BY THE PROCESSOR, OR"  
230 PRINT  
240 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"  
250 PRINT  
260 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"  
270 PRINT "FOR DETAILED CRITERIA."  
280 PRINT  
1070 PRINT "                                BEGIN TEST."  
1080 PRINT  
1090 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"  
1100 PRINT "CONTAINING LINE-NUMBERS WITH THE VALUE OF 0. THE FIRST"  
1110 PRINT "STATEMENT IN THIS PROGRAM HAS A LINE-NUMBER OF 0. IT IS A"  
1120 PRINT "PRINT-STATEMENT WHICH PRINTS THE MESSAGE THAT A LINE-NUMBER"  
1130 PRINT "OF 0 HAS BEEN EXECUTED. CHECK THE FIRST LINE OF OUTPUT."  
1140 PRINT  
1150 PRINT "                                END TEST."  
1160 PRINT  
1290 PRINT "END PROGRAM 200"  
1300 END
```

---

? ZERO INVALID AS LINE NUMBER

```
*****  
*****  
*****  
  
PRINT "PROGRAM FILE 201: ERROR - STATEMENTS WITHOUT LINE-NUMBERS."  
PRINT "    ANSI STANDARD 4.2, 4.4"  
PRINT  
PRINT "SECTION 201.1: ERROR - STATEMENTS WITHOUT LINE-NUMBERS."  
PRINT  
PRINT "THIS PROGRAM TESTS TO SEE IF THE BASIC PROCESSOR"  
PRINT "ACCEPTS PROGRAMS CONTAINING STATEMENTS WITH NO"  
PRINT "LINE-NUMBERS."  
PRINT
```

```

PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
PRINT
PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
PRINT "        BY THE PROCESSOR, OR"
PRINT
PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
PRINT
PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
PRINT "FOR DETAILED CRITERIA."
PRINT
PRINT "                                BEGIN TEST."
PRINT
PRINT "IF THIS PROGRAM GENERATED ANY OUTPUT, THEN THE"
PRINT "PROCESSOR HAS EXECUTED STATEMENTS WITH NO LINE-NUMBERS."
PRINT
PRINT "                                END TEST."
PRINT
PRINT "END PROGRAM 201"
END

```

---

#### % MISSING LINE NUMBERS

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 202: ERROR - LINES LONGER THAN 72 CHARACTERS."
20 PRINT "    ANSI STANDARD 4.4"
30 PRINT
40 PRINT "SECTION 202.1: ERROR - LINES LONGER THAN 72 CHARACTERS."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING LINES LONGER THAN 72 CHARACTERS."
80 PRINT
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
110 PRINT
120 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
130 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
140 PRINT "        BY THE PROCESSOR, OR"
150 PRINT
160 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
170 PRINT
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
190 PRINT "FOR DETAILED CRITERIA."
200 PRINT
210 PRINT "                                BEGIN TEST."
220 PRINT
230 PRINT "THE PROCESSOR HAS EXECUTED A STATEMENT CONTAINING"; 9999      -9921;
235 PRINT "CHARACTERS."
240 PRINT

```

```
250 PRINT "END TEST."
260 PRINT
270 PRINT "END PROGRAM 202"
280 END
```

---

```
? EXCESSIVE LINE-LENGTH IN LINE 230
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 203: EFFECT OF ZONES AND MARGIN ON PRINT."
20 PRINT "      ANSI STANDARD 12.4, 12.2"
30 PRINT
40 PRINT "THIS TEST IS CONCERNED WITH THE VARIOUS RULES GOVERNING THE"
50 PRINT "INTERACTION OF ZONE-WIDTH, MARGIN-WIDTH, COLUMNAR POSITION,"
60 PRINT "AND PRINT-SEPARATORS AS DESCRIBED IN THE STANDARD."
70 PRINT
80 PRINT "PLEASE ENTER ZONE-WIDTH FOR THIS IMPLEMENTATION."
90 INPUT Z
100 IF Z=INT(Z) THEN 130
110 PRINT "MUST BE AN INTEGER."
120 GOTO 80
130 IF Z>=14 THEN 160
140 PRINT "ZONE-WIDTH MUST BE >= 14."
150 GOTO 80
160 PRINT "PLEASE ENTER MARGIN FOR THIS IMPLEMENTATION."
170 INPUT M
180 IF M=INT(M) THEN 210
190 PRINT "MUST BE AN INTEGER."
200 GOTO 160
210 IF M>=Z THEN 235
220 PRINT "MARGIN MUST BE >= ZONE-WIDTH."
230 GOTO 160
235 PRINT "PLEASE ENTER NUMBER OF PRINT ZONES ON A LINE."
240 INPUT Z9
245 IF Z9=INT(Z9) THEN 260
250 PRINT "MUST BE AN INTEGER."
255 GOTO 235
260 IF Z9 <= ((M-1)/Z) + 1 THEN 275
265 PRINT "NUMBER OF ZONES TOO LARGE FOR MARGIN AND ZONE-WIDTH."
270 GOTO 80
275 REM LENGTH OF LAST ZONE
280 LET Z8=M-(Z*(Z9-1))
290 PRINT
300 PRINT "THROUGHOUT THESE TESTS, PAIRS OF OUTPUTS WILL BE PRINTED,"
310 PRINT "THE FIRST OF WHICH IS GENERATED ACCORDING TO THE INFORMATION"
320 PRINT "JUST SUPPLIED, AND THE SECOND OF WHICH REPRESENTS THE "
330 PRINT "FEATURE BEING TESTED. THESE TWO OUTPUTS MUST BE IDENTICAL"
340 PRINT "FOR THE TEST TO PASS."
350 PRINT
```

```
360 LET C=0
370 PRINT "SECTION 203.1: COMMA ADVANCES COLUMNAR POSITION TO NEXT"
380 PRINT "          PRINT-ZONE."
390 PRINT
400 PRINT "                  BEGIN TEST."
403 PRINT
405 PRINT "COMMA ADVANCES TO THE BEGINNING OF EACH OF THE ";Z9;
407 PRINT " PRINT ZONES:"
410 GOSUB 2000
420 LET T=Z-1
430 FOR I9=1 TO Z9-1
440 PRINT "A";
450 GOSUB 2100
460 NEXT I9
470 PRINT "B"
480 FOR I=1 TO Z9-1
490 PRINT "A",
500 NEXT I
510 PRINT "B"
512 PRINT
515 PRINT "PRINT-STATEMENT CONTAINING ONLY A COMMA ADVANCES TO NEXT ";
517 PRINT "PRINT-ZONE:"
520 GOSUB 2000
530 PRINT "C";
540 LET T=Z+Z-1
550 GOSUB 2100
560 PRINT "D"
570 PRINT "C",
580 PRINT ,
590 PRINT "D"
592 PRINT
595 PRINT "IF COLUMNAR POSITION IS THE LAST POSITION IN A ZONE, "
597 PRINT "A COMMA ADVANCES ONE SPACE TO THE NEXT ZONE:"
600 GOSUB 2000
610 LET T=Z-2
620 GOSUB 2100
630 PRINT "E F"
640 GOSUB 2100
650 PRINT "E","F"
652 PRINT
655 PRINT "IF COLUMNAR POSITION IS THE FIRST POSITION IN A ZONE,"
657 PRINT "A COMMA ADVANCES ONE FULL ZONE:"
660 GOSUB 2000
670 LET T=Z-1
680 GOSUB 2100
690 PRINT "G";
700 LET T=Z
710 GOSUB 2100
720 PRINT "H"
730 LET T=Z-1
740 GOSUB 2100
750 PRINT "G","H"
752 PRINT
```

```
755 PRINT "LEADING COMMAS IN A PRINT-STATEMENT SKIP PRINT-ZONES:"  
760 GOSUB 2000  
770 LET T=Z+Z  
780 GOSUB 2100  
790 PRINT "I"  
800 PRINT ,, "I"  
810 GOSUB 3000  
820 PRINT "SECTION 203.2: INTERACTION OF TAB AND MARGIN."  
830 PRINT  
840 PRINT " BEGIN TEST."  
850 LET C=0  
860 PRINT  
862 PRINT  
865 PRINT "TABMING TO A POSITION < COLUMNAR POSITION FORCES A NEW LINE:"  
870 GOSUB 2200  
880 PRINT " A"  
890 PRINT " B"  
900 PRINT TAB(5); "A"; TAB(5); "B"  
902 PRINT  
905 PRINT "TABMING TO A POSITION >= COLUMNAR POSITION DOES NOT"  
909 PRINT "FORCE A NEW LINE:"  
910 GOSUB 2000  
920 PRINT " CD"  
930 PRINT TAB(5); "C"; TAB(6); "D"  
932 PRINT  
935 PRINT "TABMING TO A POSITION, N > MARGIN, M, CAUSES TAB TO"  
937 PRINT "N - M*INT((N-1)/M):"  
940 GOSUB 2000  
950 PRINT " E"  
960 PRINT TAB(M+M+M+3); "E"  
970 GOSUB 3000  
980 PRINT "SECTION 203.3: MARGIN OVERFLOW."  
990 PRINT  
1000 PRINT " BEGIN TEST."  
1010 LET C=0  
1012 PRINT  
1015 PRINT "IF A PRINT-ITEM DOESN'T FIT IN REMAINDER OF CURRENT LINE,"  
1017 PRINT "IT FORCES A NEW LINE BEFORE PRINTING:"  
1020 GOSUB 2200  
1030 LET T=M-2  
1040 GOSUB 2100  
1050 PRINT "A"  
1060 PRINT "BC"  
1065 GOSUB 2100  
1070 PRINT "A";  
1075 PRINT "BC"  
1076 PRINT  
1077 PRINT "FULL LENGTH OF MARGIN IS USED BEFORE FORCING NEW LINE:"  
1080 GOSUB 2000  
1090 GOSUB 2100  
1100 PRINT "DE"  
1105 PRINT "F"  
1110 GOSUB 2100
```

```
1112 PRINT "D","E";
1117 PRINT "F"
1118 PRINT
1119 PRINT "FILL RIGHTMOST POSITION OF LAST ZONE:"
1120 GOSUB 2000
1130 LET T=M-1
1140 GOSUB 2100
1150 PRINT "G"
1160 FOR I1=1 TO Z9-1
1170 PRINT ,
1180 NEXT I1
1190 LET T=Z8-1
1200 GOSUB 2100
1210 PRINT "G"
1212 PRINT
1215 PRINT "IF COLUMNAR POSITION IS IN LAST ZONE, A COMMA FORCES ";
1218 PRINT "A NEW LINE:"
1220 GOSUB 2200
1230 PRINT "H"
1240 PRINT "I"
1250 PRINT "H";
1260 FOR I1=1 TO Z9-2
1270 PRINT ,
1280 NEXT I1
1290 PRINT ,,"I"
1300 GOSUB 3000
1980 PRINT "END PROGRAM 203"
1990 STOP
2000 REM SUBROUTINE TO HEAD LINE-PAIRS
2010 LET C=C+1
2040 PRINT
2050 PRINT "THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE"
2060 PRINT "IDENTICAL, CASE #";C
2070 GOSUB 2300
2080 RETURN
2100 REM SUBROUTINE TO PRINT T SPACES
2110 FOR I=1 TO T
2120 PRINT " ";
2130 NEXT I
2140 RETURN
2200 REM SUBROUTINE TO HEAD PAIRS OF SETS OF LINES
2210 LET C=C+1
2240 PRINT
2250 PRINT "THE FIRST TWO LINES (1 & 2) FOLLOWING THE COLUMN NUMBERS"
2260 PRINT "SHOULD BE IDENTICAL TO THE NEXT TWO (3 & 4), CASE #";C
2270 GOSUB 2300
2280 RETURN
2300 REM SUBROUTINE TO PRINT COLUMN NUMBERS
2310 PRINT "00000000011111111222222223333333333";
2320 IF M<60 THEN 2420
2330 PRINT "44444444455555555555";
2340 IF M<72 THEN 2420
2350 PRINT "66666666677";
2353 IF M<80 THEN 2420
2355 PRINT "77777777";
```

```
2360 IF M<100 THEN 2420
2370 PRINT "88888888899999999999";
2380 IF M<120 THEN 2420
2390 PRINT "00000000001111111111";
2400 IF M<132 THEN 2420
2410 PRINT "22222222233";
2420 PRINT
2430 LET A$="0123456789"
2440 PRINT "123456789";A$;A$;A$;
2450 IF M<60 THEN 2550
2460 PRINT A$;A$;
2470 IF M<72 THEN 2550
2480 PRINT A$;"01";
2483 IF M<80 THEN 2550
2485 PRINT "23456789";
2490 IF M<100 THEN 2550
2500 PRINT A$;A$;
2510 IF M<120 THEN 2550
2520 PRINT A$;A$;
2530 IF M<132 THEN 2550
2540 PRINT A$;"1";
2550 PRINT
2560 RETURN
3000 REM SUBROUTINE TO END SECTION
3010 PRINT
3020 PRINT "IF ALL THE ABOVE PAIRS ARE IDENTICAL, THEN "
3030 PRINT "*** TEST PASSED ***"
3040 PRINT
3050 PRINT "END TEST."
3060 PRINT
3070 PRINT
3080 RETURN
5000 END
```

---

PROGRAM FILE 203: EFFECT OF ZONES AND MARGIN ON PRINT.

ANSI STANDARD 12.4, 12.2

THIS TEST IS CONCERNED WITH THE VARIOUS RULES GOVERNING THE INTERACTION OF ZONE-WIDTH, MARGIN-WIDTH, COLUMNAR POSITION, AND PRINT-SEPARATORS AS DESCRIBED IN THE STANDARD.

PLEASE ENTER ZONE-WIDTH FOR THIS IMPLEMENTATION.

? 14

PLEASE ENTER MARGIN FOR THIS IMPLEMENTATION.

? 72

PLEASE ENTER NUMBER OF PRINT ZONES ON A LINE.

? 5

THROUGHOUT THESE TESTS, PAIRS OF OUTPUTS WILL BE PRINTED, THE FIRST OF WHICH IS GENERATED ACCORDING TO THE INFORMATION JUST SUPPLIED, AND THE SECOND OF WHICH REPRESENTS THE FEATURE BEING TESTED. THESE TWO OUTPUTS MUST BE IDENTICAL FOR THE TEST TO PASS.

SECTION 203.1: COMMA ADVANCES COLUMNAR POSITION TO NEXT PRINT-ZONE.

BEGIN TEST.

COMMA ADVANCES TO THE BEGINNING OF EACH OF THE 5 PRINT ZONES:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE IDENTICAL, CASE # 1

000000001111111122222222333333344444444555555556666666677				
1234567890123456789012345678901234567890123456789012345678901				
A	A	A	A	B
A	A	A	A	B

PRINT-STATEMENT CONTAINING ONLY A COMMA ADVANCES TO NEXT PRINT-ZONE:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE IDENTICAL, CASE # 2

000000001111111122222222333333344444444555555556666666677	
1234567890123456789012345678901234567890123456789012345678901	
C	D
C	D

IF COLUMNAR POSITION IS THE LAST POSITION IN A ZONE, A COMMA ADVANCES ONE SPACE TO THE NEXT ZONE:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE IDENTICAL, CASE # 3

000000001111111122222222333333344444444555555556666666677	
1234567890123456789012345678901234567890123456789012345678901	
E F	
E F	

IF COLUMNAR POSITION IS THE FIRST POSITION IN A ZONE, A COMMA ADVANCES ONE FULL ZONE:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE IDENTICAL, CASE # 4

000000001111111122222222333333344444444555555556666666677	
1234567890123456789012345678901234567890123456789012345678901	
G	H
G	H

LEADING COMMAS IN A PRINT-STATEMENT SKIP PRINT-ZONES:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE  
IDENTICAL, CASE # 5

0000000001111111112222222233333333444444445555555566666666677

1234567890123456789012345678901234567890123456789012345678901

I

I

IF ALL THE ABOVE PAIRS ARE IDENTICAL, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

#### SECTION 203.2: INTERACTION OF TAB AND MARGIN.

BEGIN TEST.

TABBING TO A POSITION < COLUMNAR POSITION FORCES A NEW LINE:

THE FIRST TWO LINES (1 & 2) FOLLOWING THE COLUMN NUMBERS

SHOULD BE IDENTICAL TO THE NEXT TWO (3 & 4), CASE # 1

0000000001111111112222222233333333444444445555555566666666677

1234567890123456789012345678901234567890123456789012345678901

A

B

A

B

TABBING TO A POSITION  $\geq$  COLUMNAR POSITION DOES NOT  
FORCE A NEW LINE:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE  
IDENTICAL, CASE # 2

00000000011111111122222222333333333444444445555555566666666677

1234567890123456789012345678901234567890123456789012345678901

CD

CD

TABBING TO A POSITION, N > MARGIN, M, CAUSES TAB TO  
 $N - M * \text{INT}((N-1)/M)$ :

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE  
IDENTICAL, CASE # 3

00000000011111111122222222333333333444444445555555566666666677

1234567890123456789012345678901234567890123456789012345678901

E

E

IF ALL THE ABOVE PAIRS ARE IDENTICAL, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

## SECTION 203.3: MARGIN OVERFLOW.

BEGIN TEST.

IF A PRINT-ITEM DOESN'T FIT IN REMAINDER OF CURRENT LINE,  
IT FORCES A NEW LINE BEFORE PRINTING:

THE FIRST TWO LINES (1 & 2) FOLLOWING THE COLUMN NUMBERS

SHOULD BE IDENTICAL TO THE NEXT TWO (3 & 4), CASE # 1

0000000001111111112222222233333333444444444555555555666666666677  
1234567890123456789012345678901234567890123456789012345678901

A

BC

A

BC

FULL LENGTH OF MARGIN IS USED BEFORE FORCING NEW LINE:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE

IDENTICAL, CASE # 2

000000000111111111222222223333333344444444455555555566666666677  
1234567890123456789012345678901234567890123456789012345678901

DE

F

DE

F

FILL RIGHTMOST POSITION OF LAST ZONE:

THE TWO LINES FOLLOWING THE COLUMN NUMBERS SHOULD BE

IDENTICAL, CASE # 3

000000000111111111222222223333333344444444455555555566666666677  
1234567890123456789012345678901234567890123456789012345678901

G

G

IF COLUMNAR POSITION IS IN LAST ZONE, A COMMA FORCES A NEW LINE:

THE FIRST TWO LINES (1 & 2) FOLLOWING THE COLUMN NUMBERS

SHOULD BE IDENTICAL TO THE NEXT TWO (3 & 4), CASE # 4

000000000111111111222222223333333344444444455555555566666666677  
1234567890123456789012345678901234567890123456789012345678901

H

I

H

I

IF ALL THE ABOVE PAIRS ARE IDENTICAL, THEN

\*\*\* TEST PASSED \*\*\*

END TEST.

END PROGRAM 203

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 204: ERROR - PRINT-STATEMENTS CONTAINING"
15 PRINT "      LOWERCASE CHARACTERS."
20 PRINT "      ANSI STANDARD 3.2, 3.4, 12.2"
30 PRINT
40 PRINT "SECTION 204.1: ERROR - PRINT-STATEMENTS CONTAINING"
45 PRINT "      LOWERCASE CHARACTERS."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING LOWER CASE CHARACTERS IN A PRINT-STATEMENT."
80 PRINT
90 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
100 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
110 PRINT
120 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
130 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
140 PRINT "        BY THE PROCESSOR, OR"
150 PRINT
160 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
170 PRINT
180 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
190 PRINT "FOR DETAILED CRITERIA."
200 PRINT
210 PRINT "                                BEGIN TEST."
220 print
230 PRINT "IF A BLANK LINE IMMEDIATELY PRECEDES THIS SENTENCE, THEN"
240 PRINT "THE PROCESSOR EXECUTED A PRINT STATEMENT IN WHICH THE"
250 PRINT "WORD 'PRINT' APPEARS IN LOWERCASE LETTERS."
260 PRINT
270 PRINT
280 PRINT "this sentence is generated by a quoted-string print-item"
290 PRINT "containing lower-case characters."
320 PRINT
330 PRINT "                                END TEST."
340 PRINT
350 PRINT "END PROGRAM 204"
360 END

```

---

? ILLEGAL KEYWORD IN LINE 220  
? ILLEGAL FORMULA IN LINE 280  
? ILLEGAL FORMULA IN LINE 290

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

```

10 PRINT "PROGRAM FILE 205: ERROR - ASSIGNED STRING CONTAINING"
15 PRINT "      LOWERCASE CHARACTERS."
20 PRINT "      ANSI STANDARD 3.2, 3.4, 9.2"
30 PRINT

```

```

40 PRINT "SECTION 205.1: ERROR - ASSIGNED STRING CONTAINING"
45 PRINT "      LOWERCASE CHARACTERS."
50 PRINT
60 PRINT "THIS PROGRAM TESTS TO SEE IF THE PROCESSOR ACCEPTS PROGRAMS"
70 PRINT "CONTAINING LOWERCASE CHARACTERS IN THE QUOTED STRING OF A"
80 PRINT "LET-STATEMENT."
90 PRINT
100 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
110 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
120 PRINT
130 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
140 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
150 PRINT "        BY THE PROCESSOR, OR"
160 PRINT
170 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
180 PRINT
190 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
200 PRINT "FOR DETAILED CRITERIA."
210 PRINT
220 PRINT "                                BEGIN TEST."
230 PRINT
240 LET A$="abcdefghijklmnpqr"
250 PRINT "AN ASSIGNMENT STATEMENT HAS ATTEMPTED TO ASSOCIATE A STRING"
260 PRINT "CONSISTING OF THE FIRST 18 CHARACTERS OF THE ALPHABET, (A-R)"
270 PRINT "IN LOWERCASE, WITH A$."
290 PRINT "A$=";A$
300 PRINT
310 PRINT "                                END TEST."
320 PRINT
330 PRINT "END PROGRAM 205"
340 END

```

---

? ILLEGAL FORMULA IN LINE 240

```
*****
*****
```

```

10 PRINT "PROGRAM FILE 206: ERROR - ORDERING RELATIONS BETWEEN STRINGS."
20 PRINT "    ANSI STANDARD 3.2, 3.4, 3.6, 10.2"
30 PRINT
40 PRINT "THIS PROGRAM TESTS WHETHER, AS AN ENHANCEMENT, THE COMPARISON"
50 PRINT "OF STRINGS WITH THE '<' AND '>' RELATIONS IS ALLOWED."
60 PRINT
70 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
80 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
90 PRINT
100 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCOMPANIED BY DOCUMENTATION"
110 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
120 PRINT "        BY THE PROCESSOR, OR"
130 PRINT
140 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
150 PRINT

```

```
160 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
170 PRINT "FOR DETAILED CRITERIA."
180 PRINT
190 PRINT "SECTION 206.1: COLLATING SEQUENCE."
200 PRINT
210 PRINT "IF THIS IMPLEMENTATION ALLOWS THE ORDERING RELATIONS,"
220 PRINT "AN INFORMATIVE SORT IS DONE TO DISPLAY THE IMPLICIT"
230 PRINT "COLLATING SEQUENCE OF THE CHARACTERS (ASSUMING THE NATURAL"
240 PRINT "INTERPRETATION IS GIVEN TO THESE RELATIONS)."
250 PRINT
260 PRINT "                                BEGIN TEST."
270 PRINT
280 PRINT "COLLATING SEQUENCE OF SELECTED CHARACTERS IS:"
290 PRINT "(CHARACTERS SURROUNDED BY PERIODS)"
300 PRINT
310 REM SORT 30 CHARACTERS, 15 AT A TIME BECAUSE OF NAME SPACE LIMITS
320 LET L=15
330 READ W$
340 REM THROUGHOUT, VARIABLES A$ - O$ ARE TREATED AS A STRING ARRAY,
350 REM INDEXED BY N, 1 <= N <= 15.
360 LET A$=W$
370 REM Z$ IS HIGHEST SO FAR, H1 ITS INDEX
380 LET Z$=W$
390 LET H1=1
400 REM LOAD IN 1ST L, AND NOTE HIGHEST
410 FOR N=2 TO L
420 READ W$
430 GOSUB 4000
440 IF W$ < Z$ THEN 470
450 LET Z$=W$
460 LET H1=N
470 NEXT N
480 REM READ NEXT L CHARACTERS. FOR EACH, COMPARE TO HIGHEST IN TABLE.
490 REM IF THIS ONE < HIGHEST, REPLACE HIGHEST WITH IT, AND THEN
500 REM FIND NEW HIGHEST.
510 FOR I=1 TO L
520 READ W$
530 REM IF THIS > HIGHEST, SKIP
540 IF W$ > Z$ THEN 660
550 LET N=H1
560 GOSUB 4000
570 LET H1=1
580 LET Z$=A$
590 REM SCAN FOR NEW HIGHEST
600 FOR N=2 TO L
610 GOSUB 3000
620 IF W$<Z$ THEN 650
630 LET Z$=W$
640 LET H1=N
650 NEXT N
660 NEXT I
```

```
670 REM TABLE NOW HAS LOWEST L CHARACTERS; DO SORT ON THESE
680 GOSUB 2000
690 RESTORE
700 LET N=1
710 REM READ ALL L+L. PUT INTO TABLE ONLY THOSE > HIGHEST OF THE
720 REM LOWEST L CHARACTERS ALREADY SORTED.
730 FOR I=1 TO L+L
740 READ W$
750 IF W$ > Z$ THEN 770
760 GOTO 790
770 GOSUB 4000
780 LET N=N+1
790 NEXT I
800 REM SORT THE HIGH L CHARACTERS.
810 GOSUB 2000
812 PRINT
815 PRINT "IF DOCUMENTATION AGREES WITH ABOVE RESULTS, THEN"
818 PRINT "*** TEST PASSED ***"
820 PRINT
830 PRINT " END TEST."
840 PRINT
850 PRINT "SECTION 206.2: COLLATING MULTI-CHARACTER STRINGS."
860 PRINT
870 PRINT "THIS SECTION DISPLAYS SOME SUGGESTIVE TEST CASES TO"
880 PRINT "SHOW HOW THIS IMPLEMENTATION HANDLES COMPARISONS OF"
890 PRINT "MULTI-CHARACTER STRINGS. ALL STRINGS SHOWN ARE DELIMITED"
895 PRINT "BY APOSTROPES TO ENABLE THE USER TO DISTINGUISH BLANKS"
897 PRINT "FROM NULL."
900 PRINT
903 PRINT " BEGIN TEST."
906 PRINT
910 READ A$,B$
920 IF A$ = "999" THEN 1110
930 LET C$ = "="
940 IF A$=B$ THEN 1000
950 LET C$=<""
960 IF A$<B$ THEN 1000
970 LET C$=>""
980 IF A$>B$ THEN 1000
990 LET C$="INCOMPARABLE TO"
1000 PRINT "';A$;"' ";C$;"'";B$;"'"
1010 GOTO 910
1110 PRINT
1120 PRINT "IF DOCUMENTATION AGREES WITH ABOVE RESULTS, THEN"
1130 PRINT "*** TEST PASSED ***"
1140 PRINT
1150 PRINT " END TEST."
1160 PRINT
1900 PRINT "END PROGRAM 206"
1990 STOP
```

```
2000 REM THIS SUBROUTINE SORTS AND PRINTS THE CHARACTERS IN A$ - O$  
2010 REM ALGORITHM IS SIMPLE EXCHANGE SORT  
2020 FOR I=1 TO L-1  
2030 FOR J=I+1 TO L  
2040 LET N=I  
2050 GOSUB 3000  
2060 LET X$=W$  
2070 LET N=J  
2080 GOSUB 3000  
2090 LET Y$=W$  
2100 IF X$ < Y$ THEN 2180  
2110 REM SWAP ITH AND JTH CHARACTERS  
2120 LET N=I  
2130 LET W$=Y$  
2140 GOSUB 4000  
2150 LET N=J  
2160 LET W$=X$  
2170 GOSUB 4000  
2180 NEXT J  
2190 NEXT I  
2200 PRINT ".";  
2210 FOR N=1 TO L  
2220 GOSUB 3000  
2230 PRINT W$;"."  
2240 NEXT N  
2250 PRINT  
2260 RETURN  
3000 REM SUBROUTINE TO GET NTH CHARACTER INTO W$  
3010 IF N > 7 THEN 3030  
3020 ON N GOTO 3040,3060,3080,3100,3120,3140,3160  
3030 ON N-7 GOTO 3180,3200,3220,3240,3260,3280,3300,3320  
3040 LET W$=A$  
3050 GOTO 3340  
3060 LET W$=B$  
3070 GOTO 3340  
3080 LET W$=C$  
3090 GOTO 3340  
3100 LET W$=D$  
3110 GOTO 3340  
3120 LET W$=E$  
3130 GOTO 3340  
3140 LET W$=F$  
3150 GOTO 3340  
3160 LET W$=G$  
3170 GOTO 3340  
3180 LET W$=H$  
3190 GOTO 3340  
3200 LET W$=I$  
3210 GOTO 3340  
3220 LET W$=J$  
3230 GOTO 3340  
3240 LET W$=K$  
3250 GOTO 3340  
3260 LET W$=L$  
3270 GOTO 3340
```

```
3280 LET W$=M$  
3290 GOTO 3340  
3300 LET W$=N$  
3310 GOTO 3340  
3320 LET W$=O$  
3340 RETURN  
4000 REM SUBROUTINE TO GET W$ INTO NTH CHARACTER  
4010 IF N > 7 THEN 4030  
4020 ON N GOTO 4040,4060,4080,4100,4120,4140,4160  
4030 ON N-7 GOTO 4180,4200,4220,4240,4260,4280,4300,4320  
4040 LET A$=W$  
4050 GOTO 4340  
4060 LET B$=W$  
4070 GOTO 4340  
4080 LET C$=W$  
4090 GOTO 4340  
4100 LET D$=W$  
4110 GOTO 4340  
4120 LET E$=W$  
4130 GOTO 4340  
4140 LET F$=W$  
4150 GOTO 4340  
4160 LET G$=W$  
4170 GOTO 4340  
4180 LET H$=W$  
4190 GOTO 4340  
4200 LET I$=W$  
4210 GOTO 4340  
4220 LET J$=W$  
4230 GOTO 4340  
4240 LET K$=W$  
4250 GOTO 4340  
4260 LET L$=W$  
4270 GOTO 4340  
4280 LET M$=W$  
4290 GOTO 4340  
4300 LET N$=W$  
4310 GOTO 4340  
4320 LET O$=W$  
4340 RETURN  
5000 DATA "A","M","Z","0","1","8"  
5010 DATA "9","","!","#",$,"%"  
5020 DATA "&","!","(",")","*","+  
5030 DATA ",","-",".",":"/",":",";"  
5040 DATA "<","=",">","?","^","_"  
6000 DATA "ABC","ABC"  
6005 DATA "ABC","ABD"  
6010 DATA "ABC","XYZ"  
6020 DATA "ABC","ABC "  
6030 DATA "ABC"," ABC"  
6040 DATA "ABC "," ABC"  
6050 DATA "ABC","ABCD"  
6060 DATA "ABC"," "  
6070 DATA "ABC",""  
6080 DATA "", "
```

```
6090 DATA " ",," "
6100 DATA " ABC"," "
6110 DATA " ABC"," "
6120 DATA " ABC"," "
7000 DATA "999","999"
8000 END
```

---

```
? ILLEGAL STRING RELATION IN LINE 440
? ILLEGAL STRING RELATION IN LINE 540
? ILLEGAL STRING RELATION IN LINE 620
? ILLEGAL STRING RELATION IN LINE 750
? ILLEGAL STRING RELATION IN LINE 960
? ILLEGAL STRING RELATION IN LINE 980
? ILLEGAL STRING RELATION IN LINE 2100
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 207: ERROR - ASSIGNMENT OF A STRING TO A"
20 PRINT "           NUMERIC VARIABLE."
30 PRINT "           ANSI STANDARD 9.2"
40 PRINT
50 PRINT "SECTION 207.1: ERROR - ASSIGNMENT OF A STRING TO A"
60 PRINT "           NUMERIC VARIABLE."
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT
110 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
120 PRINT "           ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT "           BY THE PROCESSOR, OR"
140 PRINT
150 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT "           BEGIN TEST."
210 PRINT
220 PRINT "SETTING A$='123' AND X=111"
230 PRINT
240 LET A$="123"
250 LET X=111
260 PRINT "ABOUT TO EXECUTE: LET X = A$"
270 LET X=A$
280 PRINT
290 PRINT "RESULTING VALUE OF X = ";X
300 PRINT
310 PRINT "           END TEST."
```

```
320 PRINT
330 PRINT "END PROGRAM 207"
340 END
```

---

```
? MIXED STRINGS AND NUMBERS IN LINE 270
```

```
*****  
*****  
*****
```

```
10 PRINT "PROGRAM FILE 208: ERROR - ASSIGNMENT OF A NUMBER TO A"
20 PRINT "      STRING VARIABLE."
30 PRINT "      ANSI STANDARD 9.2"
40 PRINT
50 PRINT "SECTION 208.1: ERROR - ASSIGNMENT OF A NUMBER TO A"
60 PRINT "      STRING VARIABLE."
70 PRINT
80 PRINT "THIS IS A TEST FOR A NON-STANDARD FEATURE OF MINIMAL BASIC."
90 PRINT "TO PASS THIS TEST, THE PROCESSOR MUST EITHER:"
100 PRINT
110 PRINT "    1) ACCEPT THE PROGRAM AND BE ACCCOMPANIED BY DOCUMENTATION"
120 PRINT "        ACCURATELY DESCRIBING THE FEATURE'S INTERPRETATION"
130 PRINT "        BY THE PROCESSOR, OR"
140 PRINT
150 PRINT "    2) REJECT THE PROGRAM WITH AN APPROPRIATE ERROR MESSAGE"
160 PRINT
170 PRINT "SEE THE NBS MINIMAL BASIC TEST PROGRAMS USER'S MANUAL"
180 PRINT "FOR DETAILED CRITERIA."
190 PRINT
200 PRINT "                  BEGIN TEST."
210 PRINT
220 PRINT "SETTING A$='123' AND X=111"
240 LET A$="123"
250 LET X=111
260 PRINT "ABOUT TO EXECUTE: LET A$ = X"
270 LET A$=X
280 PRINT
290 PRINT "RESULTING VALUE OF A$ = ";A$
300 PRINT
310 PRINT "                  END TEST."
320 PRINT
330 PRINT "END PROGRAM 208"
340 END
```

---

```
? MIXED STRINGS AND NUMBERS IN LINE 270
```

U.S. DEPT. OF COMM.  
**BIBLIOGRAPHIC DATA  
 SHEET** (See instructions)

1. PUBLICATION OR  
 REPORT NO.  
 NBS SP 500-70/2

2. Performing Organ. Report No.

3. Publication Date

November 1980

## 4. TITLE AND SUBTITLE

Computer Science and Technology

*NBS Minimal BASIC Test Programs - Version 2 - User's Manual, Volume 2 - Source Listings and Sample Output*

## 5. AUTHOR(S)

John V. Cugini, Joan S. Bowden, Mark W. Skall

## 6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions)

**NATIONAL BUREAU OF STANDARDS**  
**DEPARTMENT OF COMMERCE**  
**WASHINGTON, D.C. 20234**

7. Contract/Grant No.

8. Type of Report &amp; Period Covered

Final

## 9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP)

Same as item 6.

## 10. SUPPLEMENTARY NOTES

*Library of Congress Catalog Card Number: 80-600163*

Document describes a computer program; SF-185, FIPS Software Summary, is attached.

## 11. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)

*This publication describes the set of programs developed by NBS for the purpose of testing conformance of implementations of the computer language BASIC to the American National Standard for Minimal BASIC, ANSI X3.60-1978. The Department of Commerce has adopted this ANSI standard as Federal Information Processing Standard 68. By submitting the programs to a candidate implementation, the user can test the various features which an implementation must support in order to conform to the standard. While some programs can determine whether or not a given feature is correctly implemented, others produce output which the user must then interpret to some degree. This manual describes how the programs should be used so as to interpret correctly the results of the tests. Such interpretation depends strongly on a solid understanding of the conformance rules laid down in the standard, and there is a brief discussion of these rules and how they relate to the test programs and to the various ways in which the language may be implemented.*

## 12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)

*Basic; language processor testing; minimal basic; programming language standards; software standards; software testing*

## 13. AVAILABILITY

Unlimited

For Official Distribution. Do Not Release to NTIS

Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D C 20402.

Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF  
PRINTED PAGES

487

15. Price

\$9.50

# NBS TECHNICAL PUBLICATIONS

## PERIODICALS

**JOURNAL OF RESEARCH**—The Journal of Research of the National Bureau of Standards reports NBS research and development in those disciplines of the physical and engineering sciences in which the Bureau is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Bureau's technical and scientific programs. As a special service to subscribers each issue contains complete citations to all recent Bureau publications in both NBS and non-NBS media. Issued six times a year. Annual subscription: domestic \$13; foreign \$16.25. Single copy, \$3 domestic; \$3.75 foreign.

**NOTE:** The Journal was formerly published in two sections: Section A "Physics and Chemistry" and Section B "Mathematical Sciences."

**DIMENSIONS/NBS**—This monthly magazine is published to inform scientists, engineers, business and industry leaders, teachers, students, and consumers of the latest advances in science and technology, with primary emphasis on work at NBS. The magazine highlights and reviews such issues as energy research, fire protection, building technology, metric conversion, pollution abatement, health and safety, and consumer product performance. In addition, it reports the results of Bureau programs in measurement standards and techniques, properties of matter and materials, engineering standards and services, instrumentation, and automatic data processing. Annual subscription: domestic \$11; foreign \$13.75.

## NONPERIODICALS

**Monographs**—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

**Handbooks**—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

**Special Publications**—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

**Applied Mathematics Series**—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

**National Standard Reference Data Series**—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NBS under the authority of the National Standard Data Act (Public Law 90-396).

**NOTE:** The principal publication outlet for the foregoing data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St., NW, Washington, DC 20005.

**Building Science Series**—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

**Technical Notes**—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

**Voluntary Product Standards**—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

**Consumer Information Series**—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

*Order the above NBS publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

*Order the following NBS publications—FIPS and NBSIR's—from the National Technical Information Services, Springfield, VA 22161.*

**Federal Information Processing Standards Publications (FIPS PUB)**—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

**NBS Interagency Reports (NBSIR)**—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services, Springfield, VA 22161, in paper copy or microfiche form.

## BIBLIOGRAPHIC SUBSCRIPTION SERVICES

The following current-awareness and literature-survey bibliographies are issued periodically by the Bureau:

**Cryogenic Data Center Current Awareness Service.** A literature survey issued biweekly. Annual subscription: domestic \$35; foreign \$45.

**Liquefied Natural Gas.** A literature survey issued quarterly. Annual subscription: \$30.

**Superconducting Devices and Materials.** A literature survey issued quarterly. Annual subscription: \$45. Please send subscription orders and remittances for the preceding bibliographic services to the National Bureau of Standards, Cryogenic Data Center (736) Boulder, CO 80303.

**U.S. DEPARTMENT OF COMMERCE**  
**National Bureau of Standards**  
Washington, D.C. 20234

OFFICIAL BUSINESS

Penalty for Private Use, \$300

POSTAGE AND FEES PAID  
U.S. DEPARTMENT OF COMMERCE  
COM-215



SPECIAL FOURTH-CLASS RATE  
BOOK

---

1298











