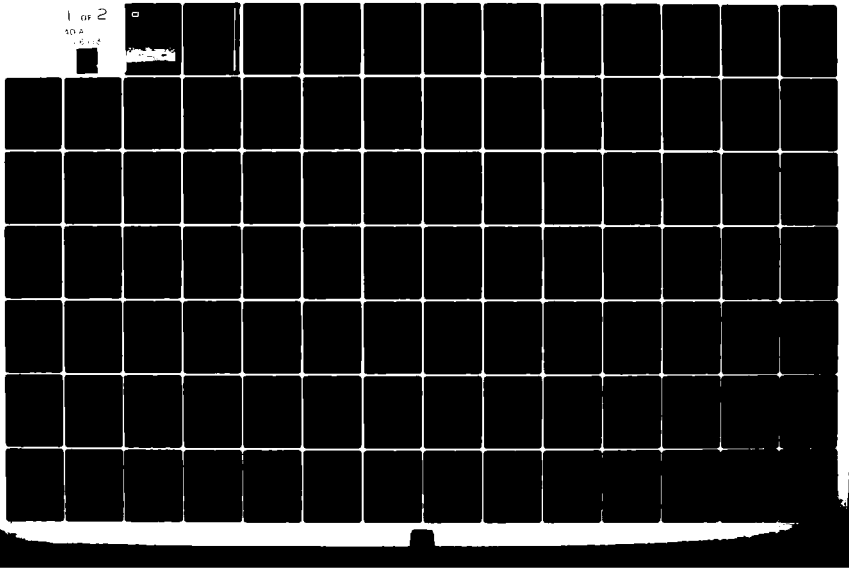
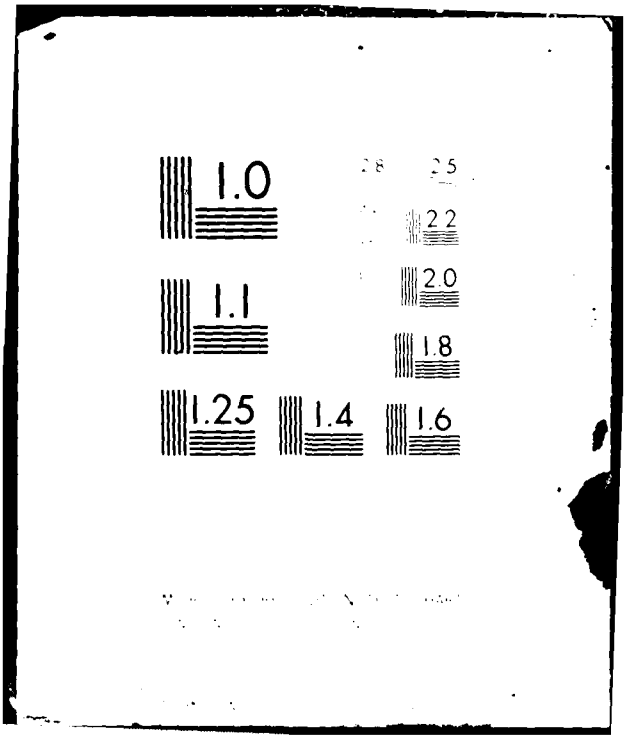
AD-A116 113 ARMY ENGINEER WATERWAYS EXPERIMENT STATION VICKSBURG MS F/G 9/2
NOTEBOOK FOR LESSONS ON THE GRAPHICS COMPATIBILITY SYSTEM (GCS)--ETC(U)
APR 82 D WARD, J M JONES, M E GEORGE
UNCLASSIFIED WES-INSTRUCTION-K-82-4 NL




1.0
1.1
1.25
1.4
1.6
2.2
2.0
1.8

INSTRUCTION REPORT K-82-4

NOTEBOOK FOR LESSONS ON THE GRAPHICS COMPATIBILITY SYSTEM (GCS)

by

Darrell Ward, James M. Jones II, Michael E. George

Automatic Data Processing Center
U. S. Army Engineer Waterways Experiment Station
P. O. Box 631, Vicksburg, Miss. 39180

April 1982

Final Report

Prepared for Office, Chief of Engineers, U. S. Army
Washington, D. C. 20314

Destroy this report when no longer needed. Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Instruction Report K-82-4	2. GOVT ACCESSION NO. AD-A116 113	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) NOTEBOOK FOR LESSONS ON THE GRAPHICS COMPATIBILITY SYSTEM (GCS)		5. TYPE OF REPORT & PERIOD COVERED Final report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Darrell Ward James M. Jones II Michael E. George		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U. S. Army Engineer Waterways Experiment Station Automatic Data Processing Center P. O. Box 631, Vicksburg, Miss. 39180		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Office, Chief of Engineers, U. S. Army Washington, D. C. 20314		12. REPORT DATE April 1982
		13. NUMBER OF PAGES 143
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

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

18. SUPPLEMENTARY NOTES

Available from National Technical Information Service, 5285 Port Royal Road, Springfield, Va. 22151.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer graphics
Computer programs
FORTRAN (Computer program language)
Programming (Electronic computers)

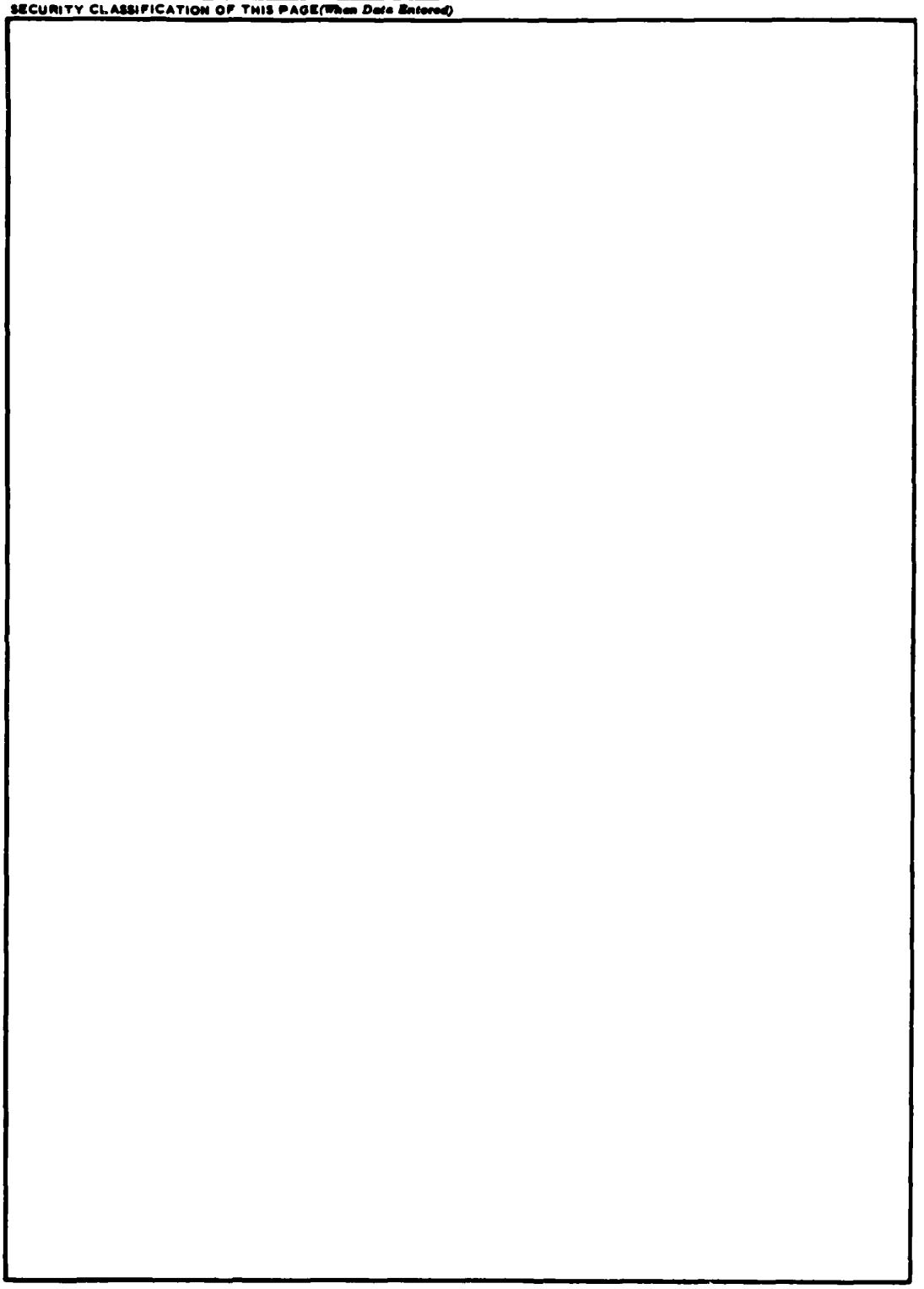
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report presents graphics examples and corresponding computer codes for each example to supplement the computer resident course on Computer-Aided Instruction (CAI) for the Graphics Compatibility System (GCS).

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)



Preface

This report presents graphics examples and corresponding computer codes for each example to supplement the computer resident course on Computer-Aided Instruction (CAI) for the Graphics Compatibility System (GCS). The work in preparing the CAI lessons and this report was performed at the U. S. Army Engineer Waterways Experiment Station (WES) as a part of a project sponsored by the Computation and Analysis Section, Office, Chief of Engineers, U. S. Army (OCE), to develop computer graphics applications for the Corps of Engineers and to maintain and support GCS.

The work in preparing the lessons was done by Dr. Darrell Ward, expert, Automatic Data Processing (ADP) Center, WES, and Mr. James M. Jones II, formerly with the Research and Development Software Group (RADSG), ADP Center, WES. Mr. Michael E. George, RADSG, made some changes to the lessons and compiled this report. The work was done under the supervision of Mr. Fred T. Tracy, Chief, RADSG, and Dr. N. Radhakrishnan, Special Technical Assistant, ADP Center, and under the general supervision of Mr. Donald L. Neumann, Chief, ADP Center.

Directors of WES during the preparation and publication of this report were COL N. P. Conover, CE, and COL T. C. Creel, CE. Technical Director was Mr. F. R. Brown.

Contents

NOTEBOOK FOR LESSONS ON THE GRAPHICS COMPATIBILITY SYSTEM (GCS)

Introduction

1. A series of lessons has been developed for teaching graphics programming with computer assistance. These lessons are intended to serve as a refresher course as well as an initial exposure to graphics programming with the Graphics Compatibility System (GCS). The lessons effectively communicate the concepts, examples, and information contained in the first 11 chapters of the GCS "Primer on Computer Graphics Programming." There are two versions of each lesson: one provides text output at an alphanumeric terminal, and the other provides both text and graphics output when used on a Tektronix 4014 graphics terminal. The 13 lessons have been implemented on the Honeywell computers at the U. S. Army Engineer Waterways Experiment Station (WES) in Vicksburg, Miss., and at Macon, Ga., and the CDC computer with Boeing Computer Services and must be executed in time-sharing.

Execution

2. To execute the lessons on the WES or Macon systems, enter the following:

```
*FORT NEW
*RUN GRAPHICS/GCSCAI,R
```

3. To execute the lessons on the Boeing system, enter the following:

```
C>OLD,GCSCAI/UN=CECELB
C>CALL,GCSCAI
```

4. Upon execution, the following is printed from the computer:

GREETINGS TO YOU TODAY. IF YOU WOULD LIKE TO SEE THE LESSON INDEX
THEN TYPE IN YES WHEN THE EQUAL SIGN IS TYPED, OTHERWISE TYPE IN
NO AND IN EITHER CASE PRESS THE RETURN KEY TO TERMINATE YOUR REPLY.
=

5. A response of YES will yield the following index of lessons:

LESSON	LESSON CONTENTS
1	IMPORTANT TECHNICAL CONCEPTS AND CONVENTIONS IN GCS
2	GCS PROGRAMMING FUNDAMENTALS
3	VIRTUAL AND DEVICE GRAPHICS IN GCS
4	ALPHANUMERIC OUTPUT WITH GCS
5	GRAPHICAL AND ALPHANUMERIC INPUT WITH GCS
6	PROBLEM 1
7	GCS UTILITY ROUTINES
8	HIGH LEVEL GRAPHICS WITH GCS
9	PROBLEM 2
10	COORDINATE SYSTEMS AND TRANSFORMATIONS
11	THREE-DIMENSIONAL GRAPHICS
12	GRAPHICAL DATA STRUCTURING PROCESSING
13	PICTURE SEGMENTATION AND NAMING

6. The following is then output:

PLEASE INPUT THE NUMBER OF THE LESSON THAT YOU WISH TO TAKE
THEN PRESS THE RETURN KEY
= (enter a number from 1 to 13)

ARE YOU USING A TEKTRONIX 4014 GRAPHICS TERMINAL (YES/NO)?
= (enter either YES or NO)

The lesson will now execute.

7. After the lesson has been completed, the following message is output:

WOULD YOU LIKE TO TAKE ANOTHER LESSON (YES/NO)?
= (enter either YES or NO)

8. If a YES answer is entered, the following is then output:

WOULD YOU LIKE TO SEE THE LESSON INDEX AGAIN (YES/NO)?
= (enter either YES or NO)

9. If the user enters NO to taking another lesson, the lesson session is terminated.

Examples

10. The following examples are the same examples that are plotted in the lessons. If the user is taking the lessons on an alphanumeric terminal, the lessons will pause, giving the user a chance to look at the example being discussed. If the user is taking the lessons using a Tektronix 4014 graphics terminal, the example will be plotted and the user given a chance to replot the example as many times as necessary to fully understand the example. In either case, the source code for each example can only be viewed by using this manual.

E X A M P L E 2.1

```
C
C THIS SAMPLE PROGRAM WILL DEMONSTRATE SIMPLE LINE-DRAWING BY
C DRAWING A SQUARE BOX IN 4 PEN MOVEMENTS.
C
C INITIALIZE GCS
C
C THIS INITIALIZATION SETS GCS TO RECTANGULAR, ABSOLUTE COORDINATES,
C SOLID LINE PEN-DRAWING MODE AND INITIAL PEN COORDINATES (0.,0.).
C
      CALL USTART
C
C NOTE THAT ALL GCS SUBROUTINES USE 'REAL' CALLING PARAMETERS.  THUS
C COORDINATES MUST BE ENTERED AS REAL NUMBERS (WITH DECIMAL POINTS).
C
C MOVE PEN TO (0.,50.) THEREBY DRAWING LINE (0.,0.) TO (0.,50.)
C
      CALL UPEN (0.,50.)
C
C MOVE PEN TO (50.,50.) THEREBY DRAWING LINE (0.,50.) TO (50.,50.)
C
      CALL UPEN (50.,50.)
C
C MOVE PEN TO (50.,0.) THEREBY DRAWING LINE (50.,50.) TO (50.,0.)
C
      CALL UPEN (50.,0.)
C
C MOVE PEN TO (0.,0.) THEREBY DRAWING LINE (50.,0.) TO (0.,0.)
C
      CALL UPEN (0.,0.)
C
C THIS COMPLETES DRAWING OF THE SQUARE.
C
C WRAP-UP.  FIRST TERMINATE GCS BY CALL UEND.  THEN STOP EXECUTION
C WITH STOP.  FINALLY END FORTRAN PROGRAM WITH END.
C
      CALL UEND
      STOP
      END
```

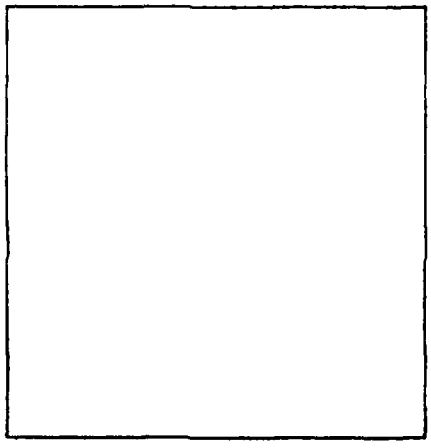


Figure 1. Example 2.1

E X A M P L E 2.2

```
C
C THIS PROGRAM DEMONSTRATES USE OF THE MOVE COMMAND TO MOVE THE PEN
C INVISIBLY WITHOUT NEED FOR A MODE CHANGE.  IT DRAWS A SQUARE
C IDENTICAL TO THE PREVIOUS ONE.
C
C INITIALIZE GCS
C
      CALL USTART
C
C DRAW A BOX AROUND THE DEFAULT DEVICE PLOTTING AREA.
C
      CALL UMOVE (0.,0.)
      CALL UPEN (100.,0.)
      CALL UPEN (100.,100.)
      CALL UPEN (0.,100.)
      CALL UPEN (0.,0.)
C
C MOVE PEN INVISIBLY TO COORDINATES (45.,45.)
C
      CALL UMOVE (45.,45.)
C
C NO CHANGE HAS BEEN MADE IN PEN STATUS SO IT IS STILL IN THE DEFAULT
C CASE OF SOLID LINES.  DRAW THE SQUARE.
C
      CALL UPEN (45.,95.)
      CALL UPEN (95.,95.)
      CALL UPEN (95.,45.)
      CALL UPEN (45.,45.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

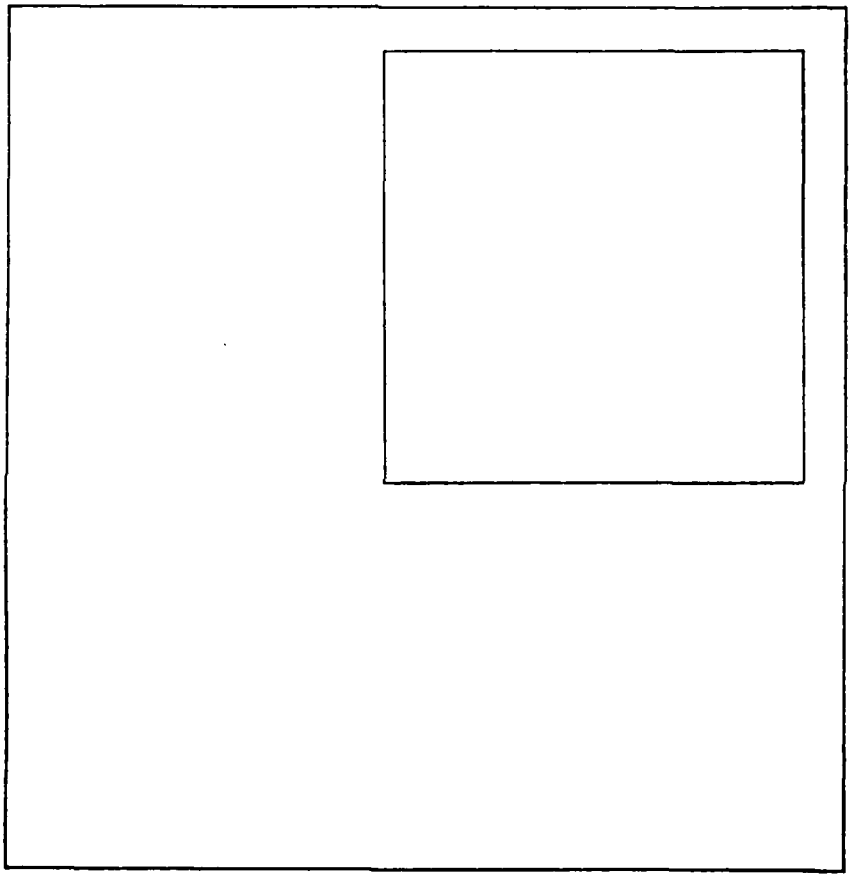


Figure 2. Example 2.2

E X A M P L E 2.3

```
C
C THIS PROGRAM DEMONSTRATES THE USE OF A MODE CHANGE TO MOVE THE PEN
C POSITION WITHOUT DRAWING A LINE.  OTHERWISE IT DRAWS A SQUARE
C IDENTICAL TO THE PREVIOUS EXAMPLE.
C
C INITIALIZATION BY USTART IS ALWAYS NECESSARY.  AMONG OTHER THINGS IT
C AUTOMATICALLY SETS PEN STATUS FOR DRAWING SOLID LINES AND INITIAL
C PEN POSITION TO COORDINATES (0.,0.)
C
      CALL USTART
C
C DRAW A BOX AROUND THE DEFAULT DEVICE PLOTTING AREA.
C
      CALL UMOVE (0.,0.)
      CALL UPEN (100.,0.)
      CALL UPEN (100.,100.)
      CALL UPEN (0.,100.)
      CALL UPEN (0.,0.)
C
C SET MODE TO 'NOLINE' AND THEN MOVE PEN TO COORDINATES (45.,45.)
C WITHOUT DRAWING A LINE.
C
      CALL USET ('NOLINE')
      CALL UPEN (45.,45.)
C
C NOW RESET PEN STATUS FOR DRAWING SOLID LINES AND DRAW A SQUARE.
C
      CALL USET ('LINE')
      CALL UPEN (45.,95.)
      CALL UPEN (95.,95.)
      CALL UPEN (95.,45.)
      CALL UPEN (45.,45.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

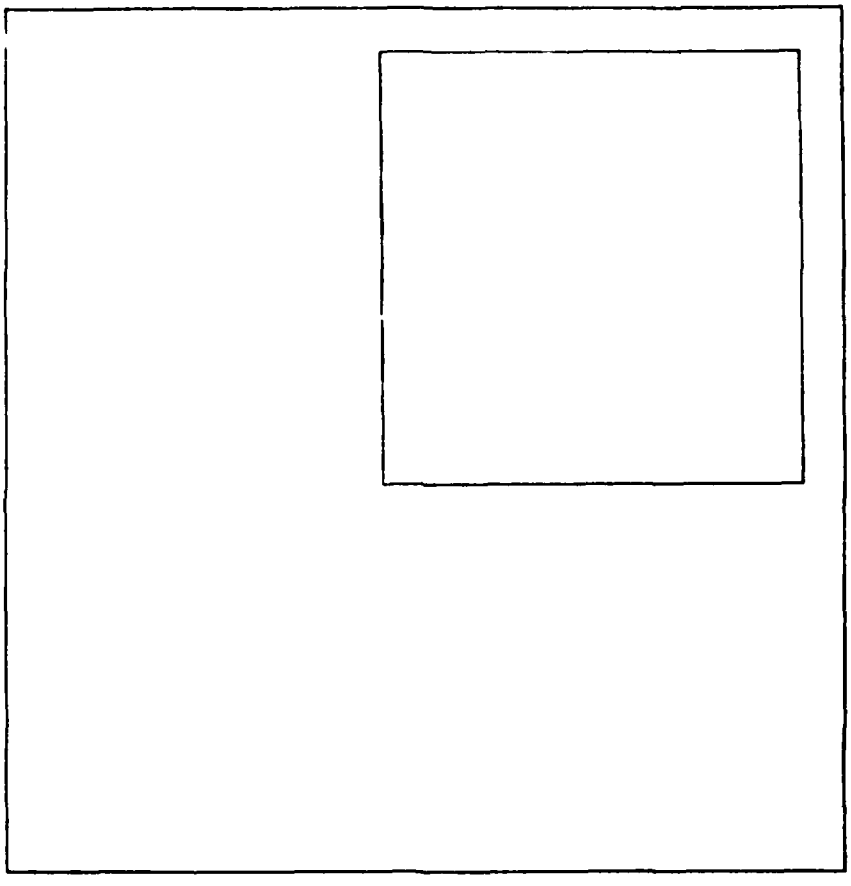


Figure 3. Example 2.3

E X A M P L E 2.4

```
C
C SAMPLE PROGRAM TO ILLUSTRATE 'ARROW','BACKARROW' AND 'DOUBLEARROW'
C LINE OPTIONS AVAILABLE THROUGH USET/UPEN.
C
C INITIALIZE GCS.
C
      CALL USTART
C
C DRAW A BOX AROUND THE DEFAULT DEVICE PLOTTING AREA.
C
      CALL UMOVE (0.,0.)
      CALL UPEN (100.,0.)
      CALL UPEN (100.,100.)
      CALL UPEN (0.,100.)
      CALL UPEN (0.,0.)
C
C MOVE TO VIRTUAL LOCATION (25.,75.), SET LINE TYPE TO 'ARROW' AND
C DRAW A LINE WHICH EXTENDS TO VIRTUAL LOCATION (75.,75.)
C
      CALL UMOVE (25.,75.)
      CALL USET ('ARROWHEAD LINE')
      CALL UPEN (75.,75.)
C
C MOVE TO VIRTUAL LOCATION (25.,50.), SET LINE TYPE TO 'BACKARROW'
C AND DRAW A LINE WHICH EXTENDS TO VIRTUAL LOCATION (75.,50.)
C
      CALL UMOVE (25.,50.)
      CALL USET ('BACK ARROWHEAD LINE')
      CALL UPEN (75.,50.)
C
C MOVE TO VIRTUAL LOCATION (25.,25.), SET LINE TYPE TO 'DOUBLEARROW'
C AND DRAW A LINE WHICH EXTENDS TO VIRTUAL LOCATION (75.,25.)
C
      CALL UMOVE (25.,25.)
      CALL USET ('DOUBLE ARROWHEAD LINE')
      CALL UPEN (75.,25.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

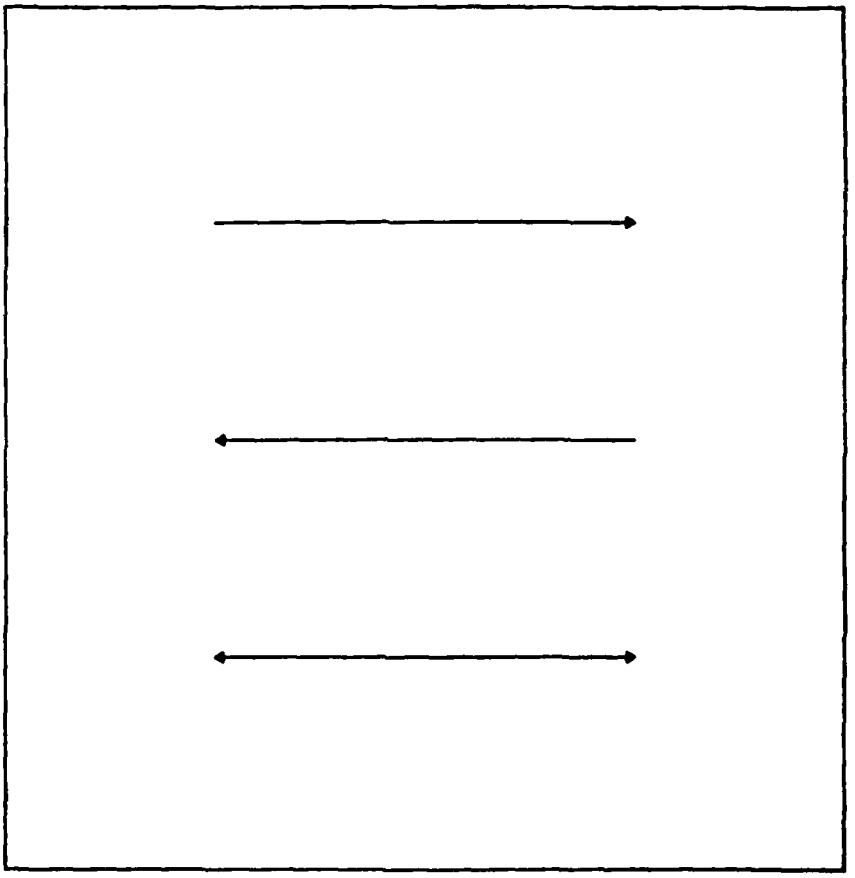


Figure 4. Example 2.4

E X A M P L E 2.5

```
C
C SAMPLE PROGRAM TO ILLUSTRATE TIC LINE GENERATION OPTIONS
C AVAILABLE THROUGH GCS.  INITIALIZE GCS, SET PEN STATUS TO
C THE 'TIC' MODE, AND THEN DRAW A LINE WHICH BEGINS AT (0.,99.)
C AND TERMINATES AT (100.,99.) USING THE DEFAULT TIC LENGTH.
C
      CALL USTART
      CALL USET ('TICLINE')
      CALL UMOVE (0.,99.)
      CALL UPEN (100.,99.)
C
C REQUEST TICS TO APPEAR AT EVERY 2.0 VIRTUAL UNITS AND DRAW
C A LINE WHICH STARTS AT (0.,80.) AND ENDS AT (100.,80.)
C
      CALL UPSET ('TICINTERVAL',2.)
      CALL UMOVE (0.,80.)
      CALL UPEN (100.,80.)
C
C REQUEST TICS TO APPEAR AT EVERY 5.0 VIRTUAL UNITS AND DRAW
C A LINE WHICH STARTS AT (0.,60.) AND ENDS AT (100.,60.)
C
      CALL UPSET ('TICINTERVAL',5.)
      CALL UMOVE (0.,60.)
      CALL UPEN (100.,60.)
C
C REQUEST TICS TO APPEAR AT EVERY 10.0 VIRTUAL UNITS AND DRAW
C A LINE WHICH STARTS AT (0.,40.) AND ENDS AT (100.,40.)
C
      CALL UPSET ('TICINTERVAL',10.)
      CALL UMOVE (0.,40.)
      CALL UPEN (100.,40.)
C
C REQUEST TICS TO APPEAR AT EVERY 20.0 VIRTUAL UNITS AND DRAW
C A LINE WHICH STARTS AT (0.,20.) AND ENDS AT (100.,20.)
C TURN OFF THE TOP PART OF THE TIC  ('TICPLUS').
C
      CALL UPSET ('TICINTERVAL',20.)
      CALL UPSET ('TICPLUS',0.)
      CALL UMOVE (0.,20.)
      CALL UPEN (100.,20.)
C
C REQUEST TICS TO APPEAR AT EVERY 50.0 VIRTUAL UNITS AND DRAW
C A LINE WHICH STARTS AT (0.,1.) AND ENDS AT (100.,1.)
C TURN OFF THE BOTTOM OF THE TIC  ('TICMINUS') AND CHANGE
C THE LENGTH OF THE TOP PART OF THE TIC  ('TICPLUS').
C
      CALL UPSET ('TICINTERVAL',50.)
      CALL UPSET ('TICPLUS',1.)
      CALL UPSET ('TICMINUS',0.)
      CALL UMOVE (0.,1.)
      CALL UPEN (100.,1.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

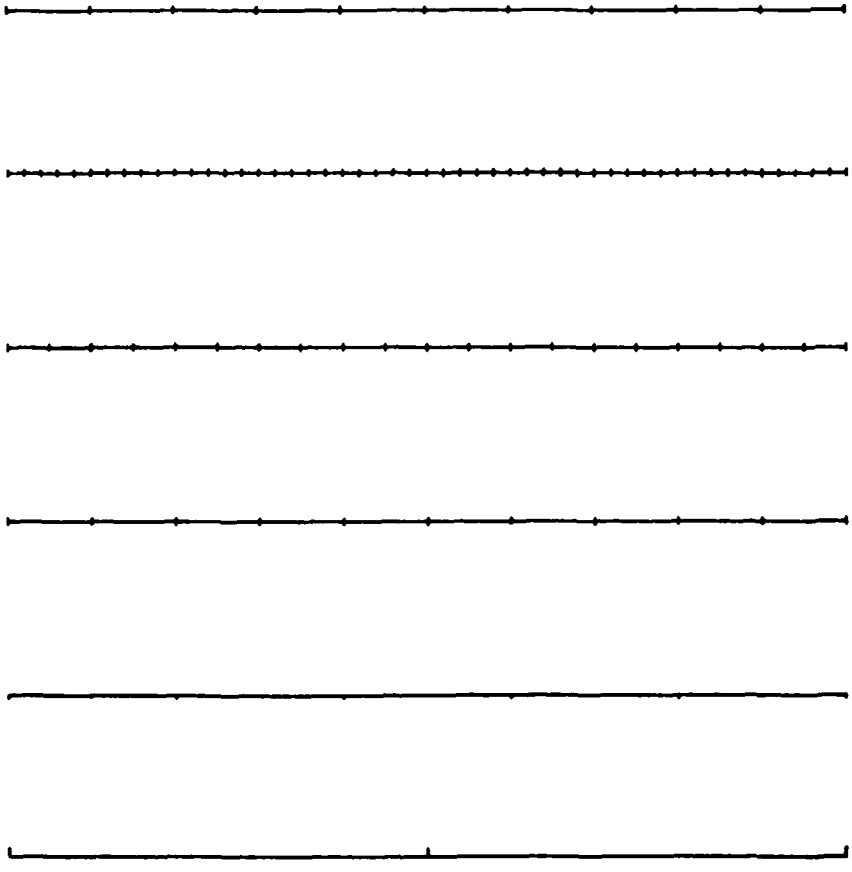


Figure 5. Example 2.5

E X A M P L E 2.6

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE DASHED LINE GENERATION
C OPTIONS AVAILABLE THROUGH GCS.  INITIALIZE GCS, SET THE
C PEN STATUS TO 'DASHLINE' MODE AND DRAW A LINE WHICH BEGINS AT
C  (0.,100.) AND TERMINATES AT (100.,100.).  THE DEFAULT
C VALUE OF DASH WILL BE USED FOR THIS CASE.
C
      CALL USTART
      CALL USET ('DASHLINE')
      CALL UMOVE (0.,100.)
      CALL UPEN (100.,100.)
C
C SET THE DASH SPECIFICATION TO 54.0 AND DRAW A LINE THAT
C STARTS AT (0.,80.) AND ENDS AT (100.,80.)
C
      CALL UPSET ('SETDASH',54.)
      CALL UMOVE (0.,80.)
      CALL UPEN (100.,80.)
C
C SET THE DASH SPECIFICATION TO 56.0 AND DRAW A LINE THAT
C STARTS AT (0.,60.) AND ENDS AT (100.,60.)
C
      CALL UPSET ('SETDASH',56.)
      CALL UMOVE (0.,60.)
      CALL UPEN (100.,60.)
C
C SET THE DASH SPECIFICATION TO 5212.0 AND DRAW A LINE THAT
C STARTS AT (0.,40.) AND ENDS AT (100.,40.)
C
      CALL UPSET ('SETDASH',5212.)
      CALL UMOVE (0.,40.)
      CALL UPEN (100.,40.)
C
C SET THE DASH SPECIFICATION TO 3.0 AND DRAW A LINE THAT
C STARTS AT (0.,20.) AND ENDS AT (100.,20.)
C NOTICE THE USE OF HARDWARE GENERATED LINES.
C
      CALL UPSET ('SETDASH',3.)
      CALL UMOVE (0.,20.)
      CALL UPEN (100.,20.)
C
C SET THE DASH SPECIFICATION TO 9.0 AND DRAW A LINE THAT
C STARTS AT (0.,0.) AND ENDS AT (100.,0.)
C NOTICE THE USE OF HARDWARE GENERATED LINES.
C
      CALL UPSET ('SETDASH',9.)
      CALL UMOVE (0.,0.)
      CALL UPEN (100.,0.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

Figure 6. Example 2.6

E X A M P L E 2.7

```
C
C SAMPLE PROGRAM THAT ILLUSTRATES THE USE OF POLAR
C PLOTTING IN RELATIVE MODE.  INITIALIZE GCS, SET
C THE COORDINATE TYPE TO 'POLAR' AND MOVE TO THE
C STARTING LOCATION.
C
      CALL USTART
      CALL USET ('POLAR COORDINATES')
      CALL UMOVE (50.*SQRT(2.),45.)
C
C ALTERNATE BETWEEN 'RELATIVE' AND 'ABSOLUTE'
C COORDINATE MODE TO DRAW A SERIES OF RADIAL LINES.
C
      DO 100 K = 1, 361, 10
      I = K - 1
      CALL USET ('RELATIVE PLOTTING MODE')
      CALL UPEN (50.,FLOAT (I))
      CALL USET ('ABSOLUTE PLOTTING MODE')
      CALL UMOVE (50.*SQRT(2.),45.)
  100 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

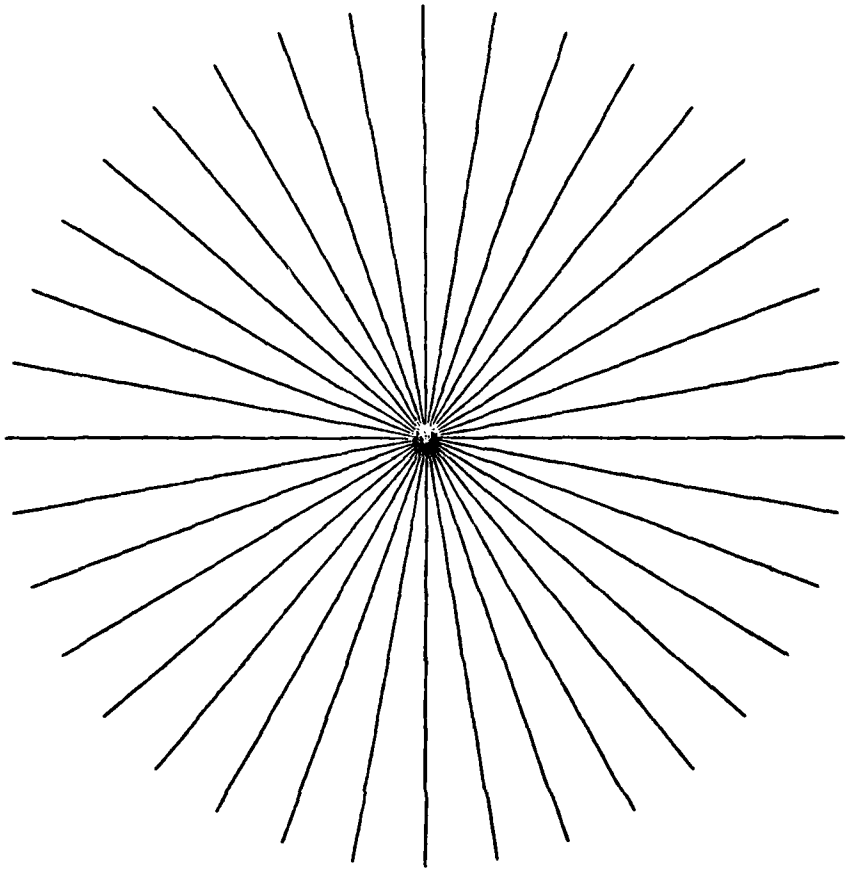


Figure 7. Example 2.7

E X A M P L E 3.1

```
C
C THIS PROGRAM GENERATES TWO VECTORS WITH ARROW LINES,
C AND THE RESULTANT VECTOR WITH A DASHED ARROW LINE.
C
C INITIALIZE GCS AND GENERATE AN OUTLINE
C
      CALL USTART
      CALL UOUTLN
C
C REDEFINE THE 'VIRTUAL' WINDOW.
C
      CALL UWINDO (-50000.,50000.,0.00001,0.00005)
C
C DRAW THE TWO VECTORS.
C
C MOVE TO THE BEGINNING POINT OF THE FIRST VECTOR AND
C SET TO ARROW MODE AND DRAW VECTOR FROM (-40000.,0.00004)
C TO (40000.,0.00004)
C
      CALL UMOVE (-40000.,0.00004)
      CALL USET ('ARROWHEAD LINE')
      CALL UPEN (40000.,0.00004)
C
C DRAW SECOND VECTOR FROM END OF FIRST TO (40000.,0.00002)
C
      CALL UPEN (40000.,0.00002)
C
C MOVE TO BEGINNING OF VECTOR SYSTEM
C
      CALL UMOVE (-40000.,0.00004)
C
C SET PEN STATUS TO DRAW A DASHED ARROW AND DRAW
C RESULTANT VECTOR
C
      CALL USET ('DARROWHEAD LINE')
      CALL UPEN (40000.,0.00002)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

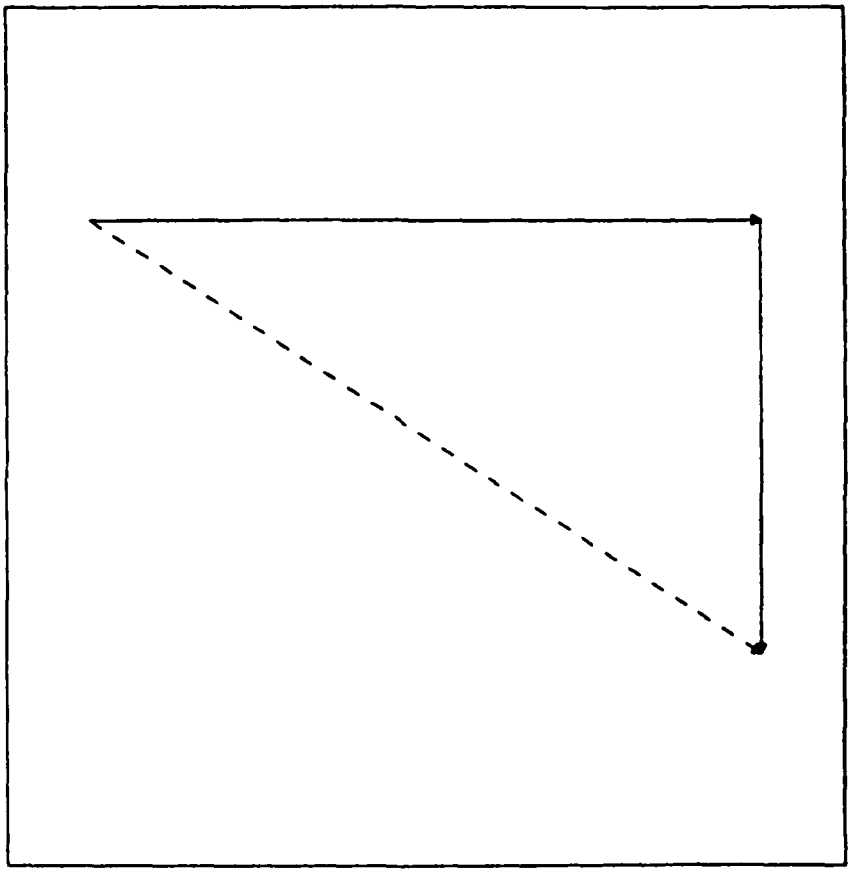


Figure 8. Example 3.1

E X A M P L E 3.2

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE ELEMENTARY SIX-LEVEL
C ZOOMING BY ADJUSTING ONLY THE VIRTUAL WINDOW BOUNDARIES.
C NOTE THAT THE PEN COMMANDS REQUIRED TO DRAW THE FIGURE
C REMAIN UNCHANGED.
C
C ENTER GCS AND SET UP LOOP TO PERMIT US TO ZOOM AWAY
C FROM FIGURE.
C
      CALL USTART
      DO 1 I = 1, 6
C
C ERASE THE SCREEN AND DEFINE THE BOUNDARIES FOR OUR NEW
C WINDOW.
C
      CALL UERASE
      BOUNDS = 50. * FLOAT(I)
      CALL UWINDO (-BOUNDS,BOUNDS,-BOUNDS,BOUNDS)
C
C OUTLINE THE DEFAULT DEVICE AREA AND DRAW THE FIGURE.
C
      CALL UOUTLN
      CALL DRWFIG
    1 CONTINUE
C
C WRAP UP ALL GRAPHICS ACTIVITY AND TERMINATE THE FORTRAN
C PROGRAM.
C
      CALL UEND
      STOP
      END
      SUBROUTINE DRWFIG
C
C SUBROUTINE USED TO GENERATE A PENTAGON WITHIN A CIRCLE,
C RING A BELL AND PAUSE.  PUSH THE RETURN KEY TO CONTINUE.
C
      CALL USET ('POLAR COORDINATES')
      DO 10 I = 1, 361, 10
      K = I - 1
      IF (K.EQ.0) CALL UMOVE (25.,FLOAT(K))
      IF (K.NE.0) CALL UPEN (25.,FLOAT (K))
   10 CONTINUE
      DO 20 I = 1, 6
      ANGLE = 18. + FLOAT(I-1) * 72.
      IF (I.EQ.1) CALL UMOVE (15.,ANGLE)
      IF (I.NE.1) CALL UPEN (15.,ANGLE)
   20 CONTINUE
      CALL UBELL
      CALL UPAUSE
      CALL UEND
      STOP
      END
```

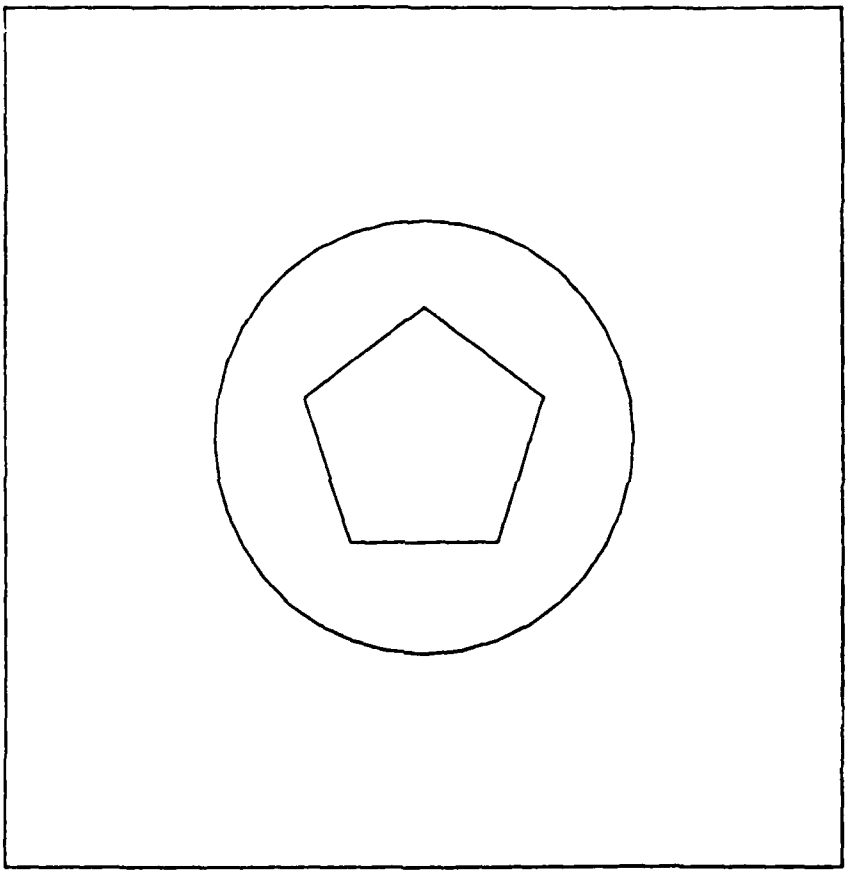


Figure 9. Example 3.2

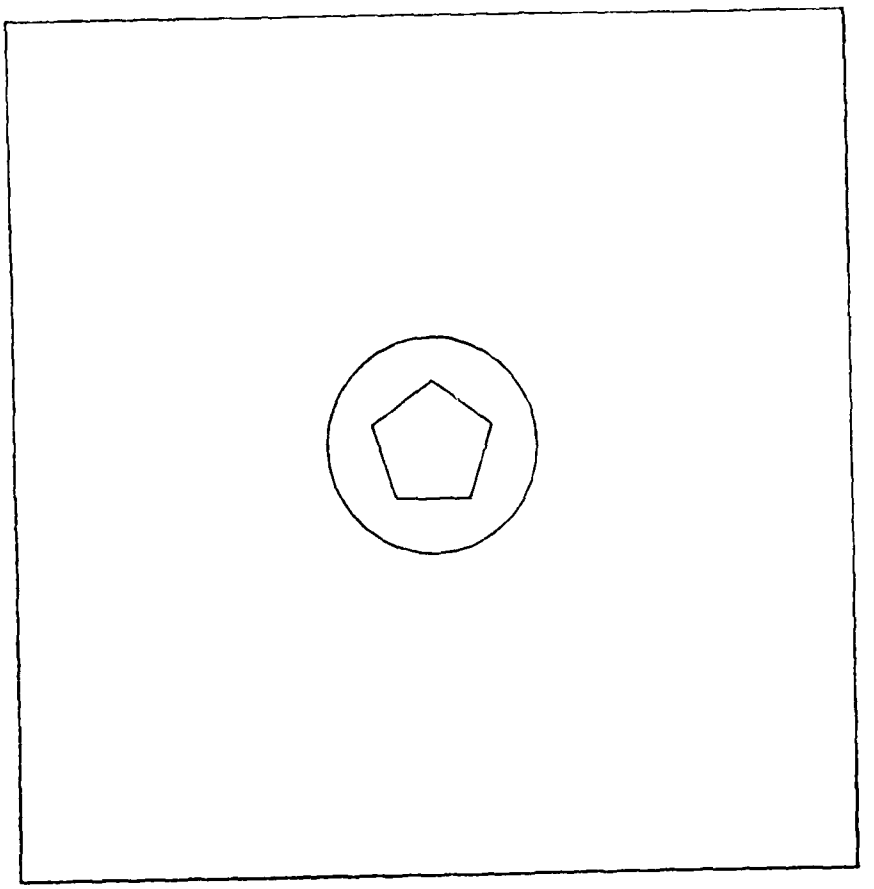


Figure 10. Example 3.2 (continued)

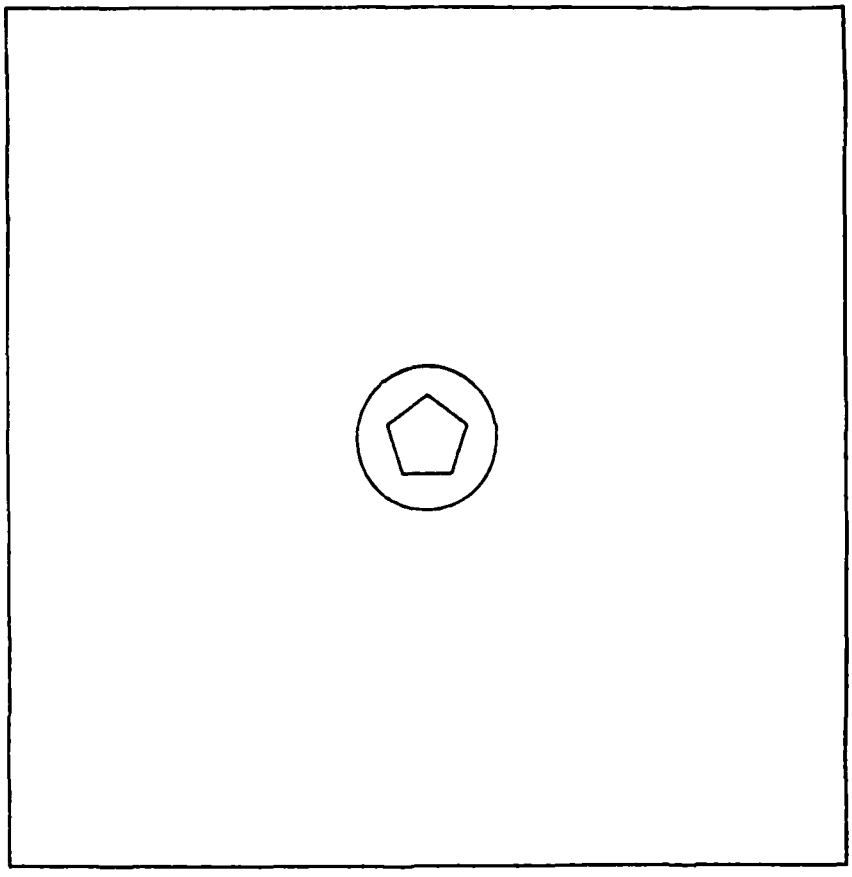


Figure 11. Example 3.2 (continued)

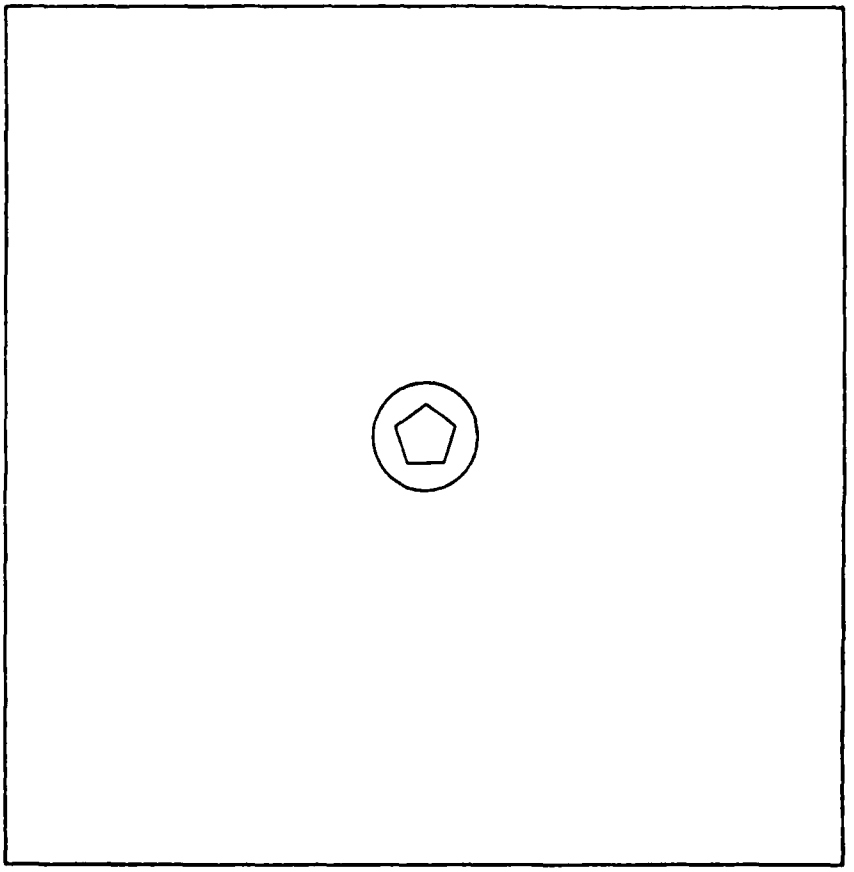


Figure 12. Example 3.2 (continued)

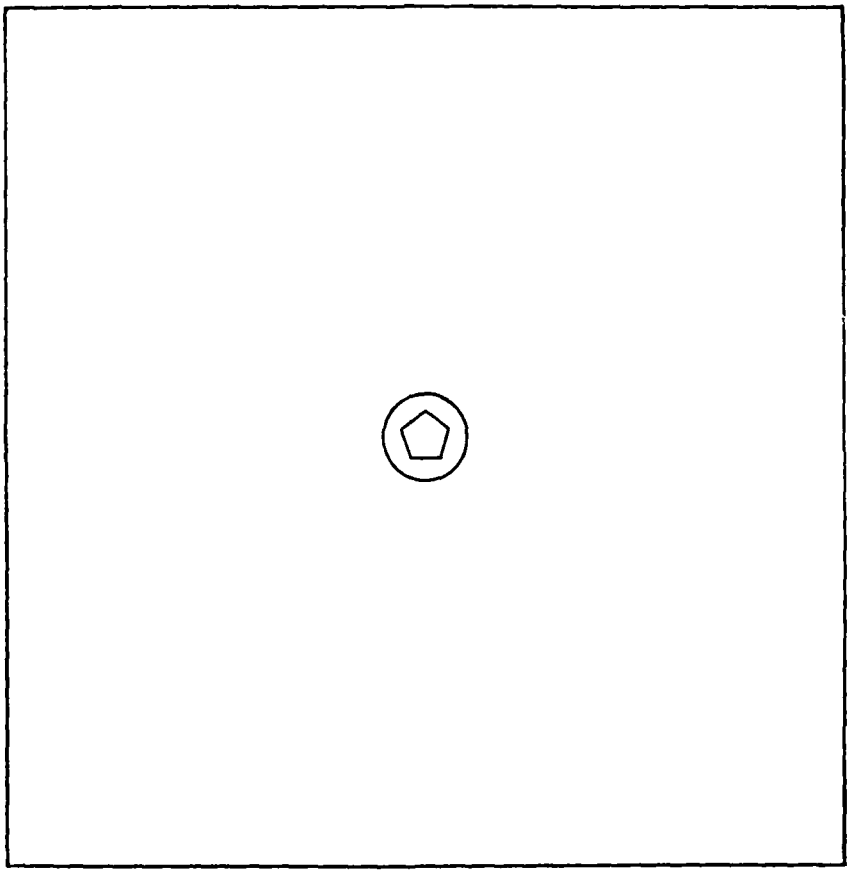


Figure 13. Example 3.2 (continued)

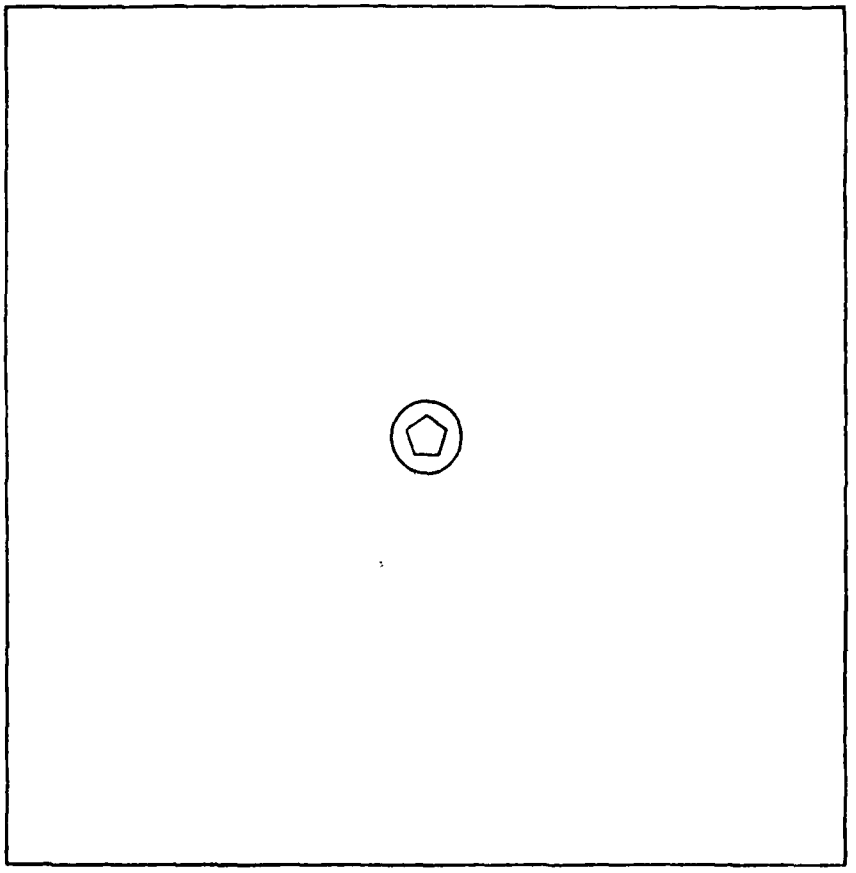


Figure 14. Example 3.2 (continued)

E X A M P L E 3.3

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE ELEMENTARY SIX-LEVEL
C ZOOMING BY ADJUSTING ONLY THE VIRTUAL WINDOW
C BOUNDARIES.  NOTE THAT THE PEN COMMANDS REQUIRED TO
C DRAW THE FIGURE REMAIN UNCHANGED.
C
C ENTER GCS AND SET UP LOOP TO PERMIT US TO ZOOM TOWARD
C THE FIGURE.
C
      CALL USTART
      DO 1 I = 1, 6
C
C ERASE THE SCREEN AND DEFINE THE BOUNDARIES FOR OUR
C NEW WINDOW.
C
      CALL UERASE
      BOUNDS = 50. - (5. * FLOAT(I-1))
      CALL UWINDO (-BOUNDS,BOUNDS,-BOUNDS,BOUNDS)
C
C OUTLINE THE DEFAULT DEVICE AREA AND DRAW THE FIGURE.
C
      CALL UOUTLN
      CALL DRWFIG
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
      SUBROUTINE DRWFIG
C
C SUBROUTINE USED TO GENERATE A PENTAGON WITHIN A CIRCLE,
C RING A BELL AND PAUSE.  PUSH THE RETURN KEY TO CONTINUE.
C
      CALL USET ('POLAR COORDINATES')
      DO 10 I = 1, 361, 10
      K = I - 1
      IF (K.EQ.0) CALL UMOVE (25.,FLOAT(K))
      IF (K.NE.0) CALL UPEN (25.,FLOAT (K))
   10 CONTINUE
      DO 20 I = 1, 6
      ANGLE = 18. + FLOAT(I-1) * 72.
      IF (I.EQ.1) CALL UMOVE (15.,ANGLE)
      IF (I.NE.1) CALL UPEN (15.,ANGLE)
   20 CONTINUE
      CALL UBELL
      CALL UPAUSE
      CALL UEND
      STOP
      END
```

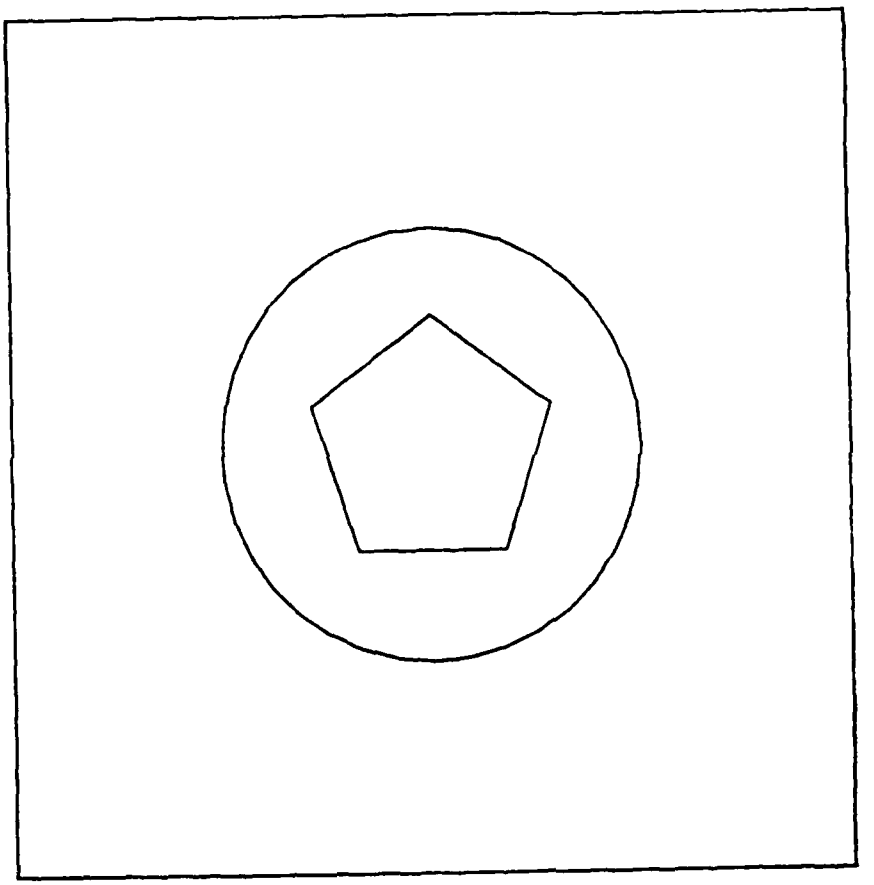


Figure 15. Example 3.3

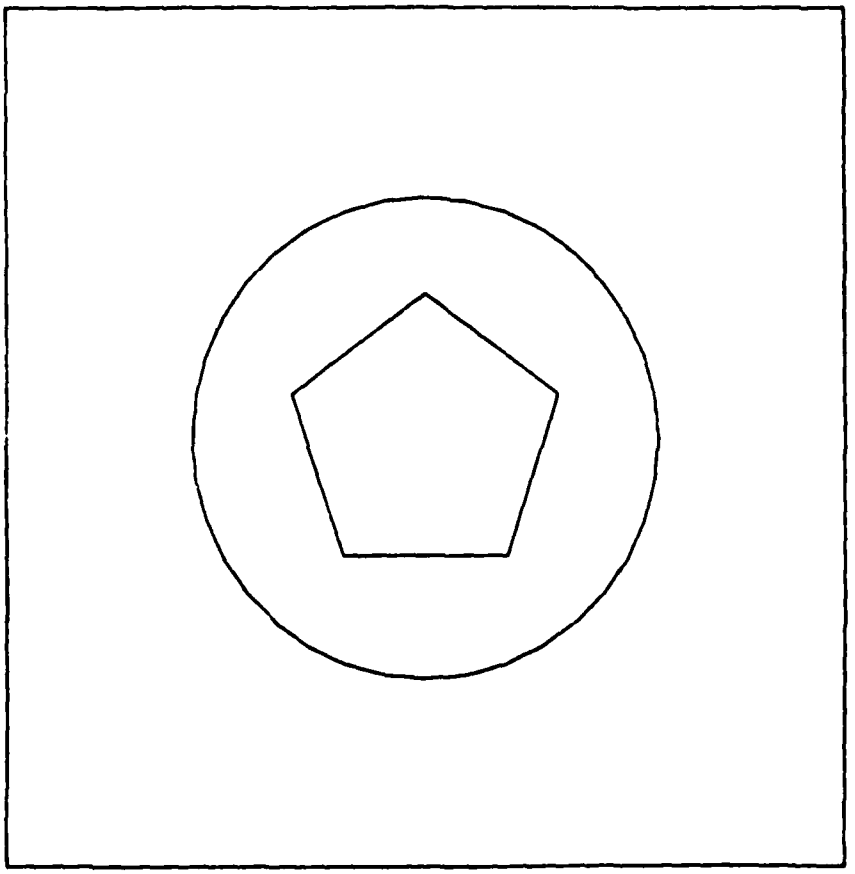


Figure 16. Example 3.3 (continued)

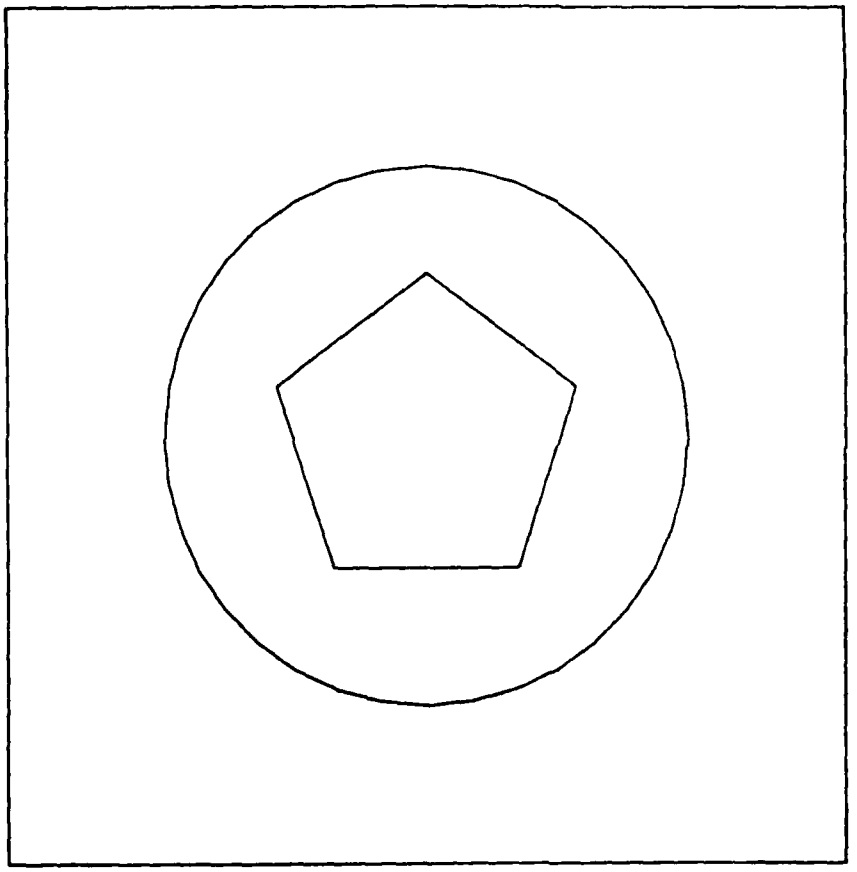


Figure 17. Example 3.3 (continued)

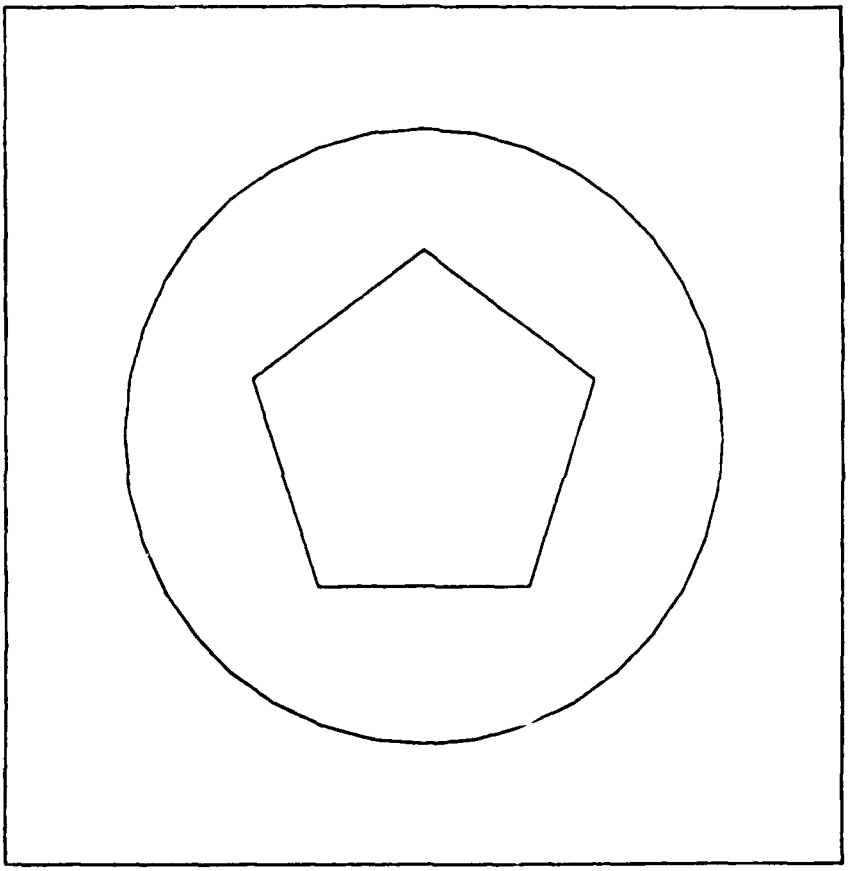


Figure 18. Example 3.3 (continued)

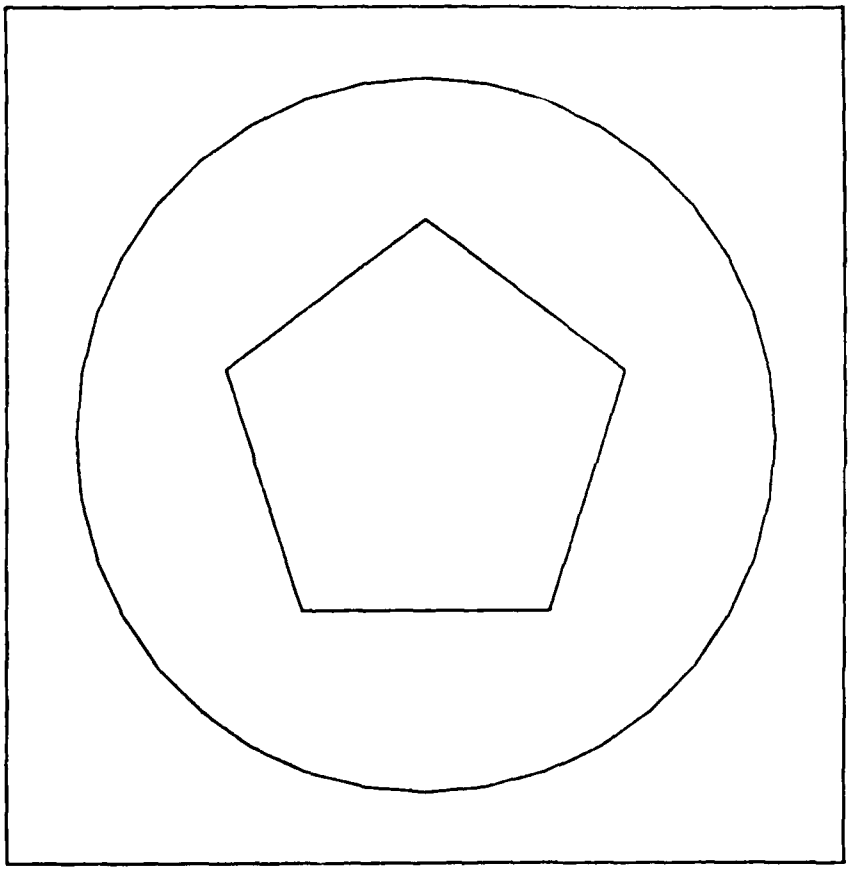


Figure 19. Example 3.3 (continued)

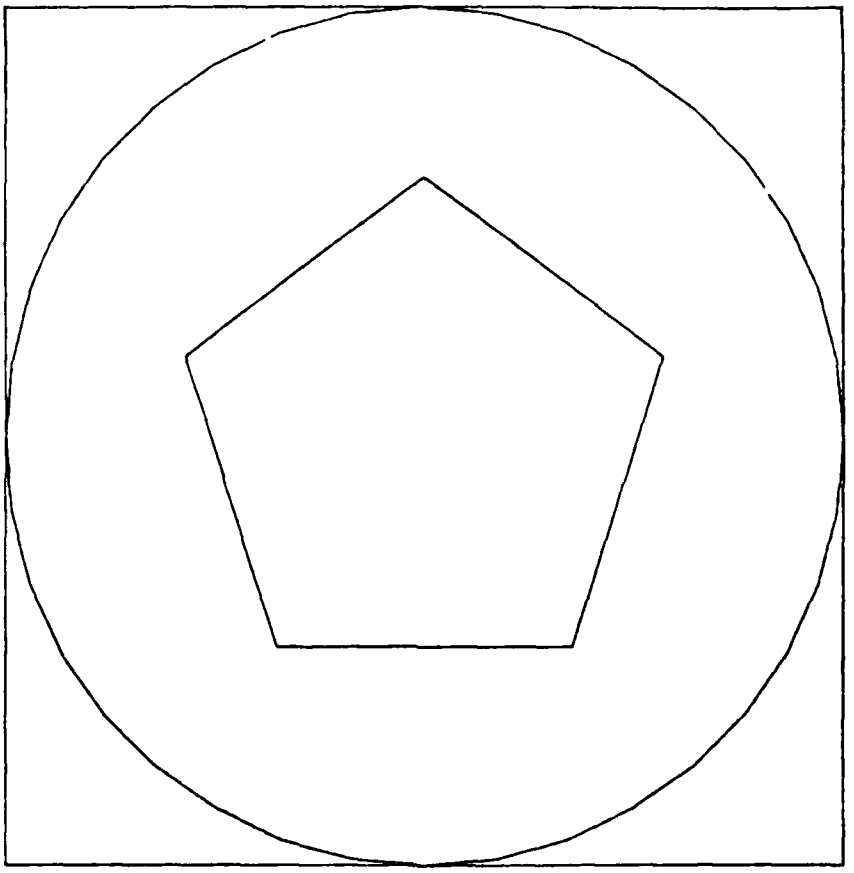


Figure 20. Example 3.3 (continued)

E X A M P L E 3.4

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE ELEMENTARY SIX-LEVEL
C ZOOMING BY ADJUSTING ONLY THE VIRTUAL WINDOW BOUNDARIES.
C NOTE THAT THE PEN COMMANDS REQUIRED TO DRAW THE FIGURE
C REMAIN UNCHANGED.  ALSO NOTE THE DISTORTION DUE TO THE
C OF NON-SQUARE WINDOWING.
C
C INITIALIZE GCS AND SET UP A LOOP TO PERMIT US TO ZOOM
C TOWARD THE FIGURE.
C
      CALL USTART
      DO 1 I = 1, 6
C
C ERASE THE SCREEN AND DEFINE THE BOUNDARIES FOR OUR
C NEW WINDOW.
C
      CALL UERASE
      XBOUND = 50. - (5.*FLOAT(I-1))
      YBOUND = 50. - (2.5*FLOAT(I-1))
      CALL UWINDO (-XBOUND,XBOUND,-YBOUND,YBOUND)
C
C OUTLINE THE DEFAULT DEVICE AREA AND DRAW THE FIGURE.
C
      CALL UOUTLN
      CALL DRWFIG
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
      SUBROUTINE DRWFIG
C
C SUBROUTINE USED TO GENERATE A PENTAGON WITHIN A CIRCLE,
C RING A BELL AND PAUSE.  PUSH THE RETURN KEY TO CONTINUE.
C
      CALL USET ('POLAR COORDINATES')
      DO 10 I = 1, 361, 10
      K = I - 1
      IF (K.EQ.0) CALL UMOVE (25.,FLOAT(K))
      IF (K.NE.0) CALL UPEN (25.,FLOAT (K))
   10 CONTINUE
      DO 20 I = 1, 6
      ANGLE = 18. + FLOAT(I-1) * 72.
      IF (I.EQ.1) CALL UMOVE (15.,ANGLE)
      IF (I.NE.1) CALL UPEN (15.,ANGLE)
   20 CONTINUE
      CALL UBELL
      CALL UPAUSE
      CALL UEND
      STOP
      END
```

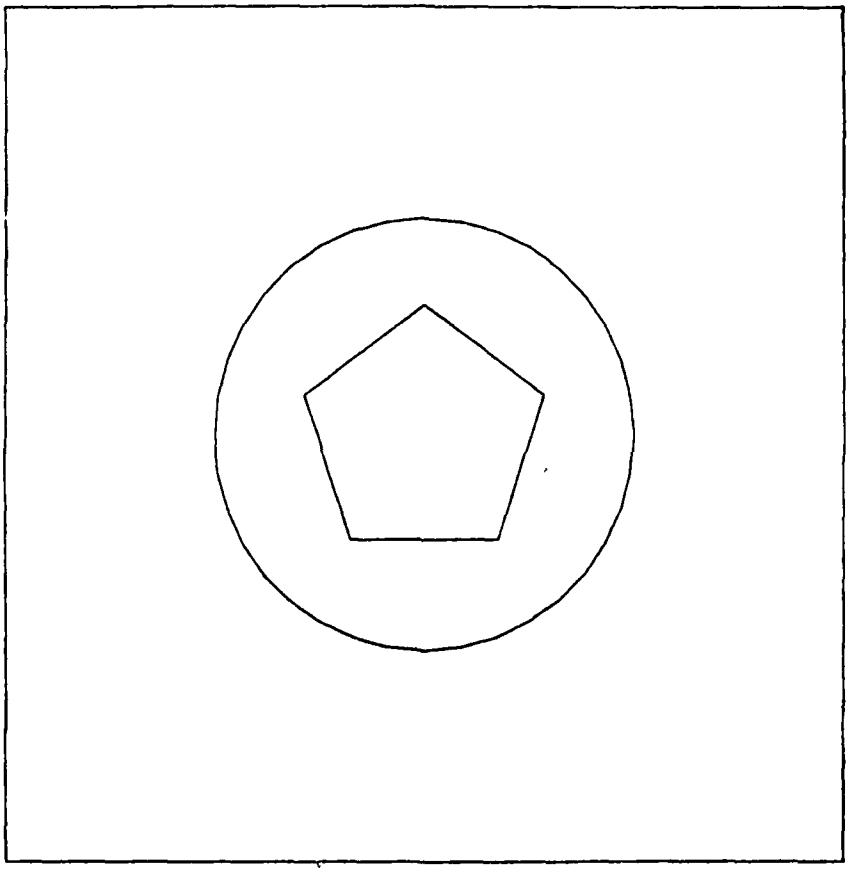


Figure 21. Example 3.4

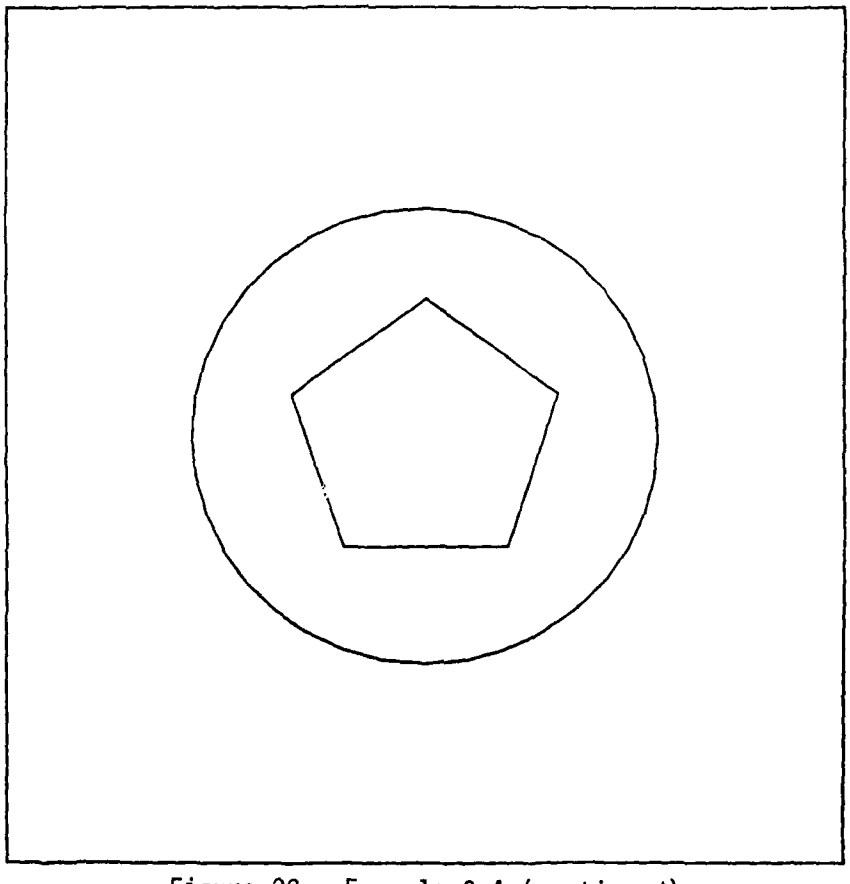


Figure 22. Example 3.4 (continued)

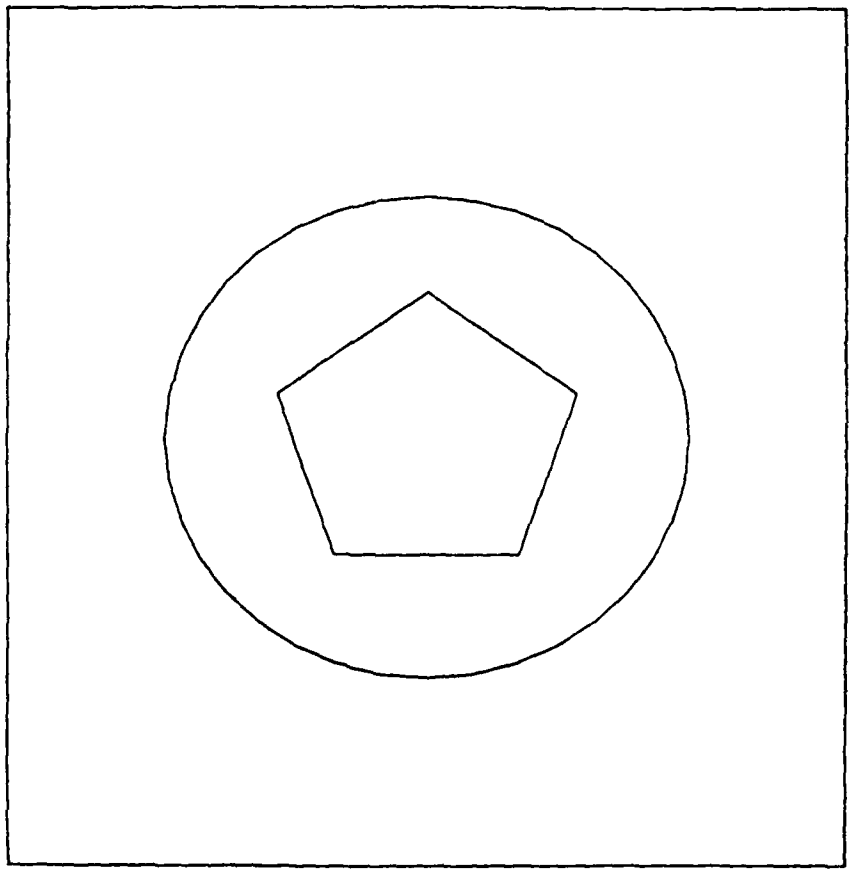


Figure 23. Example 3.4 (continued)

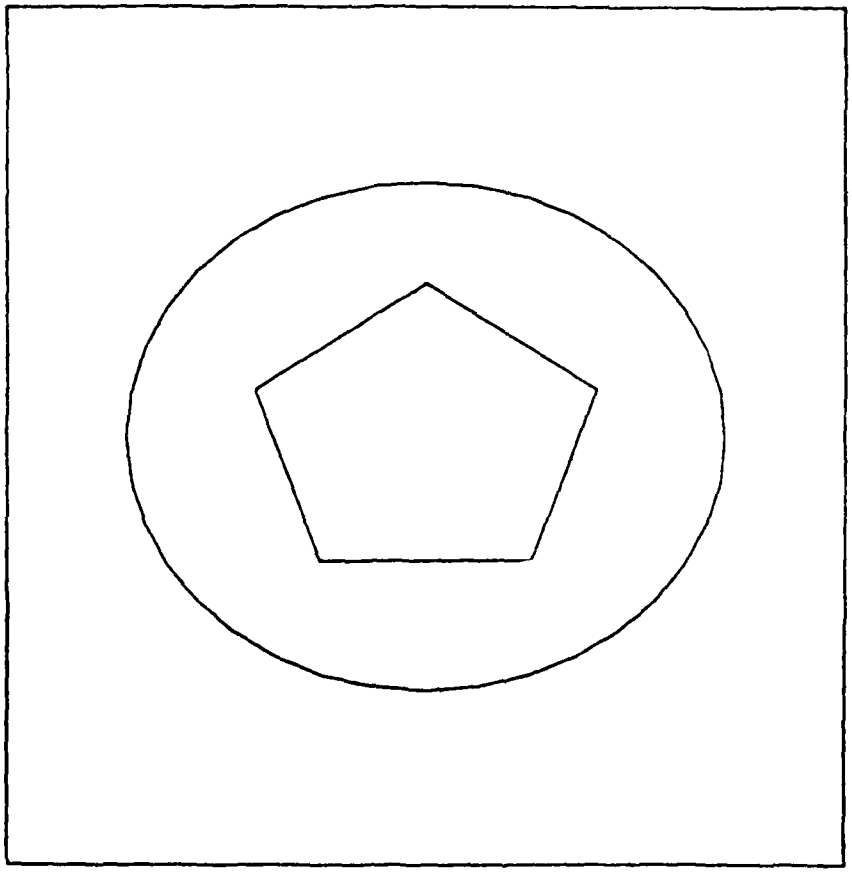


Figure 24. Example 3.4 (continued)

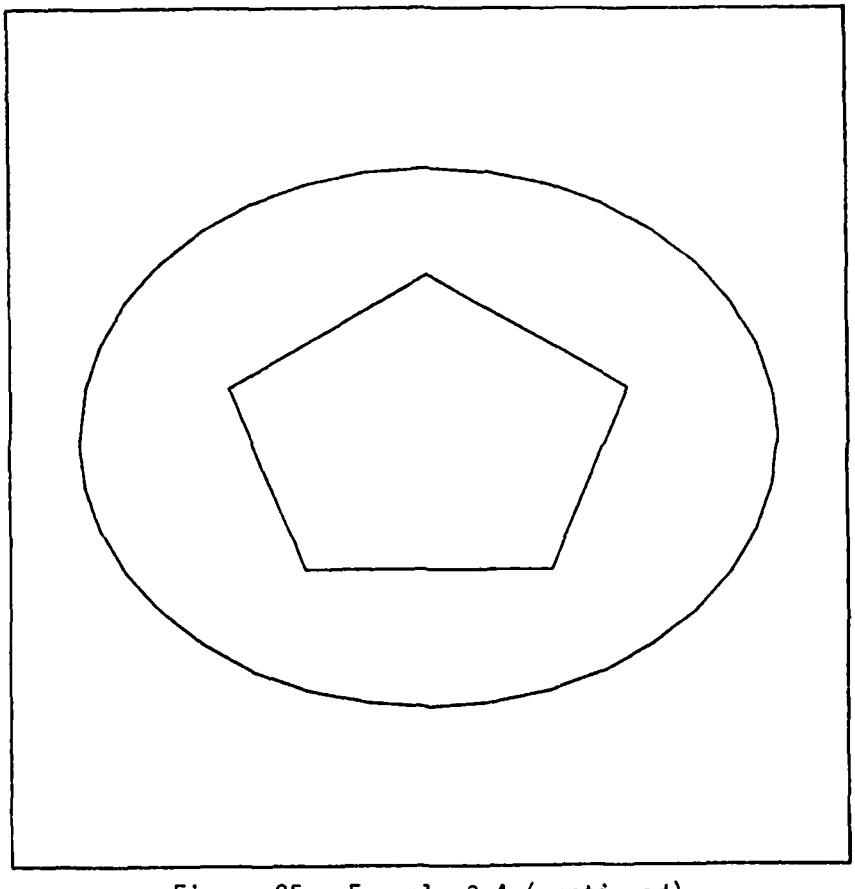


Figure 25. Example 3.4 (continued)

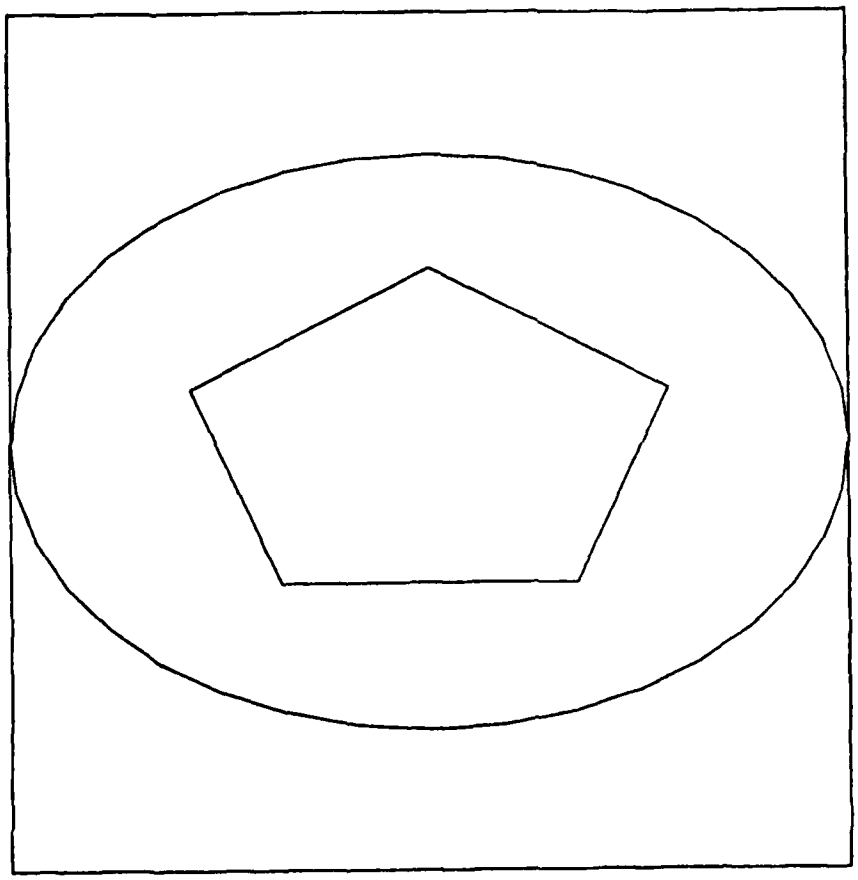


Figure 26. Example 3.4 (continued)

E X A M P L E 3.5

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE ELEMENTARY SIX-LEVEL
C ZOOMING BY ADJUSTING ONLY THE VIRTUAL WINDOW BOUNDARIES.
C NOTE THAT THE PEN COMMANDS REQUIRED TO DRAW THE FIGURE
C REMAIN UNCHANGED.  ALSO NOTE THE CLIPPING OF THE FIGURE
C AT THE WINDOW BOUNDARY.
C
C INITIALIZE GCS AND SET UP A LOOP TO PERMIT US TO ZOOM
C TOWARD THE FIGURE.
C
      CALL USTART
      DO 1 I = 1, 6
C
C ERASE THE SCREEN AND DEFINE THE BOUNDARIES FOR OUR NEW
C WINDOW.
C
      CALL UERASE
      XYMIN = -25.
      XYMAX = 25. - (7.5*FLOAT(I-1))
      CALL UWINDO (XYMIN,XYMAX,XYMIN,XYMAX)
C
C OUTLINE THE DEFAULT DEVICE AREA AND DRAW THE FIGURE.
C
      CALL UOUTLN
      CALL DRWFIG
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
      SUBROUTINE DRWFIG
C
C SUBROUTINE USED TO GENERATE A PENTAGON WITHIN A CIRCLE,
C RING A BELL AND PAUSE.  PUSH THE RETURN KEY TO CONTINUE.
C
      CALL USET ('POLAR COORDINATES')
      DO 10 I = 1, 361, 10
      K = I - 1
      IF (K.EQ.0) CALL UMOVE (25.,FLOAT(K))
      IF (K.NE.0) CALL UPEN (25.,FLOAT (K))
   10 CONTINUE
      DO 20 I = 1, 6
      ANGLE = 18. + FLOAT(I-1) * 72.
      IF (I.EQ.1) CALL UMOVE (15.,ANGLE)
      IF (I.NE.1) CALL UPEN (15.,ANGLE)
   20 CONTINUE
      CALL UBELL
      CALL UPAUSE
      CALL UEND
      STOP
      END
```

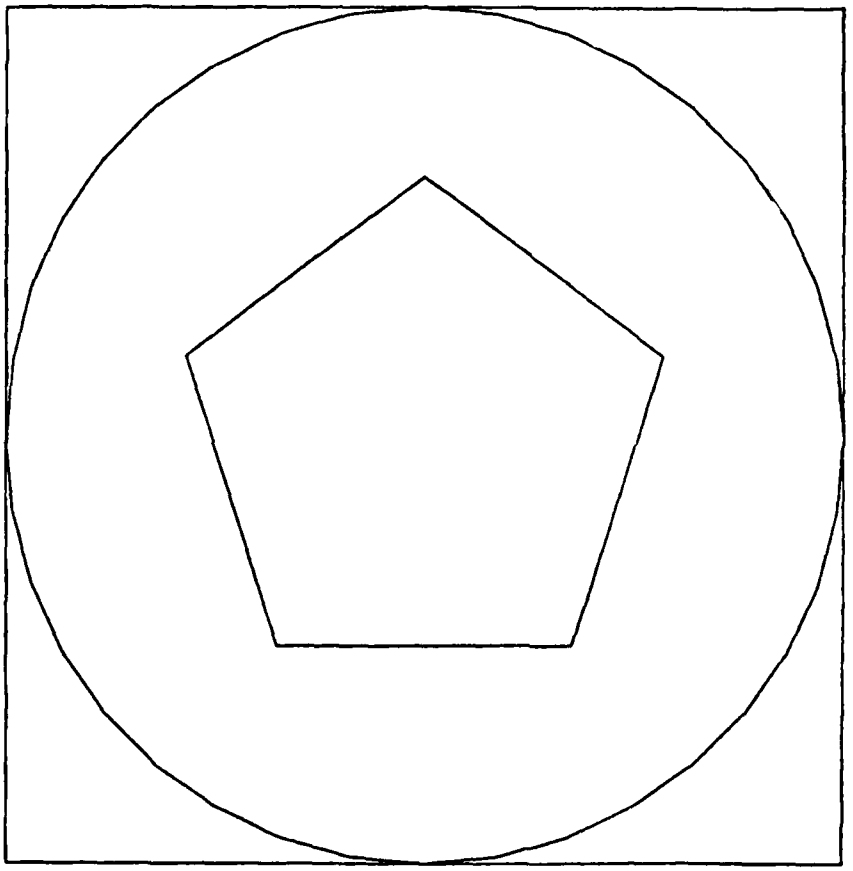


Figure 27. Example 3.5

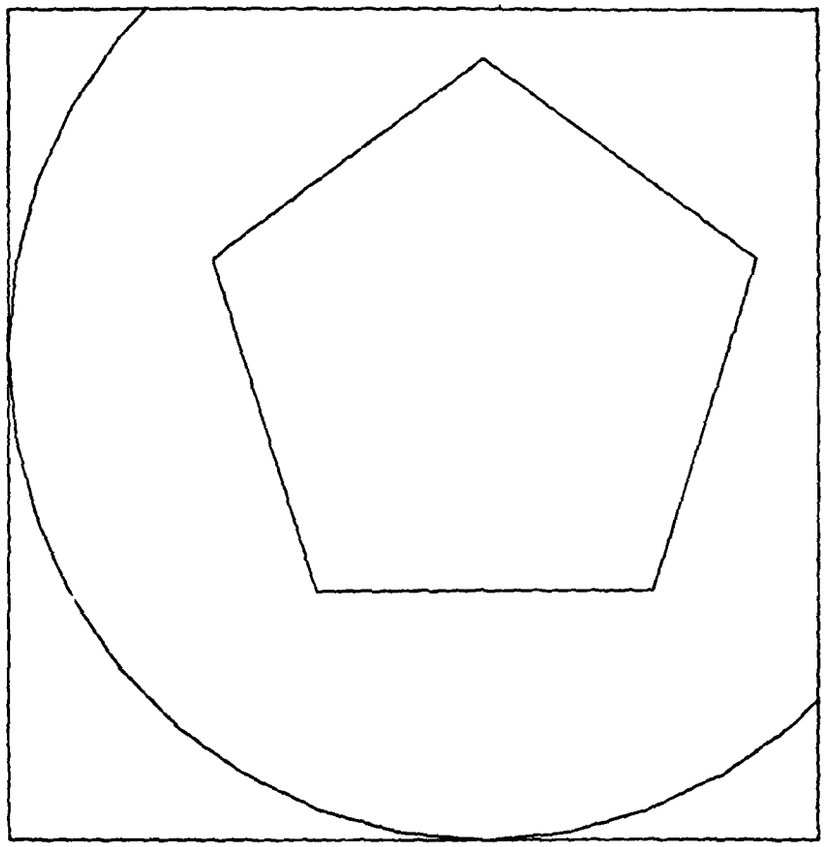


Figure 28. Example 3.5 (continued)

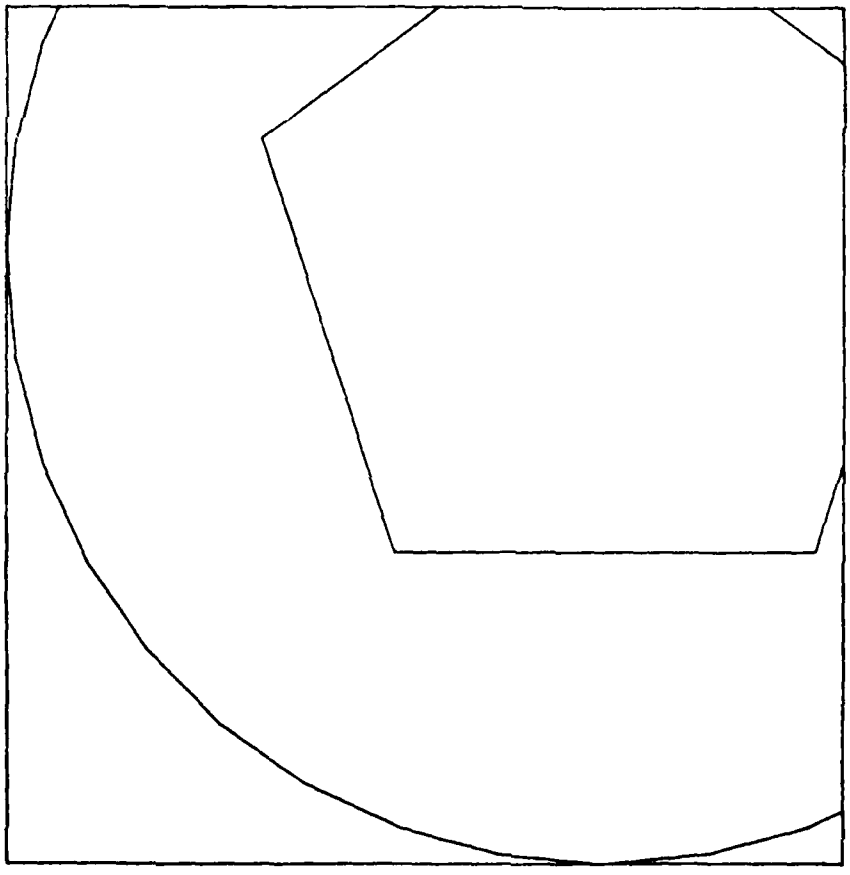


Figure 29. Example 3.5 (continued)

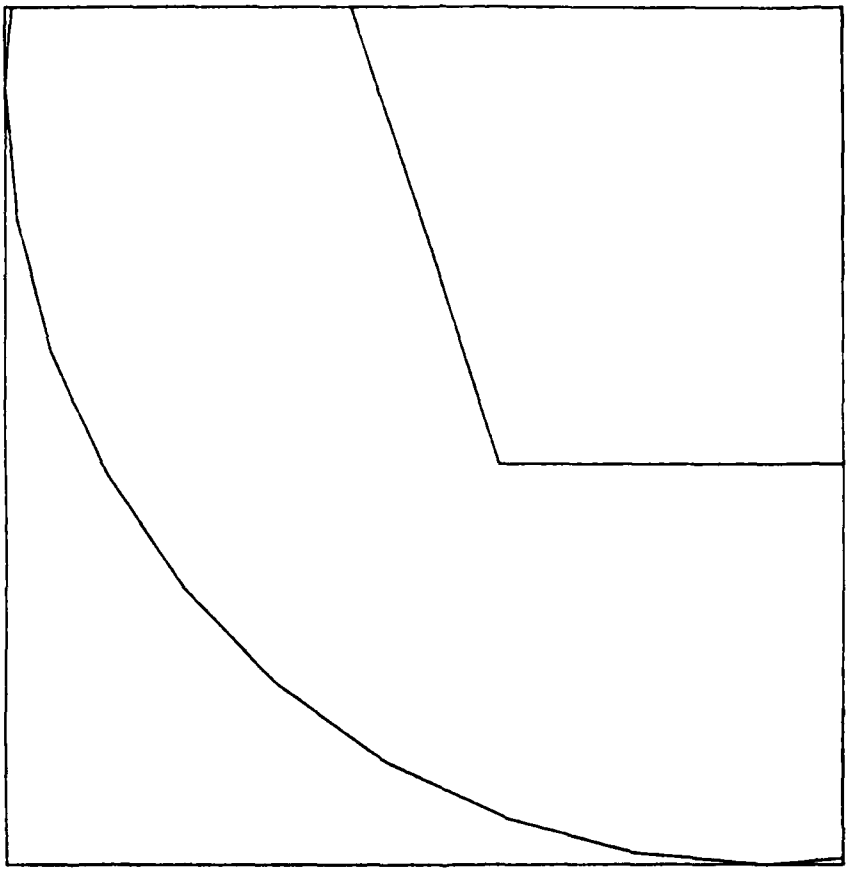


Figure 30. Example 3.5 (continued)

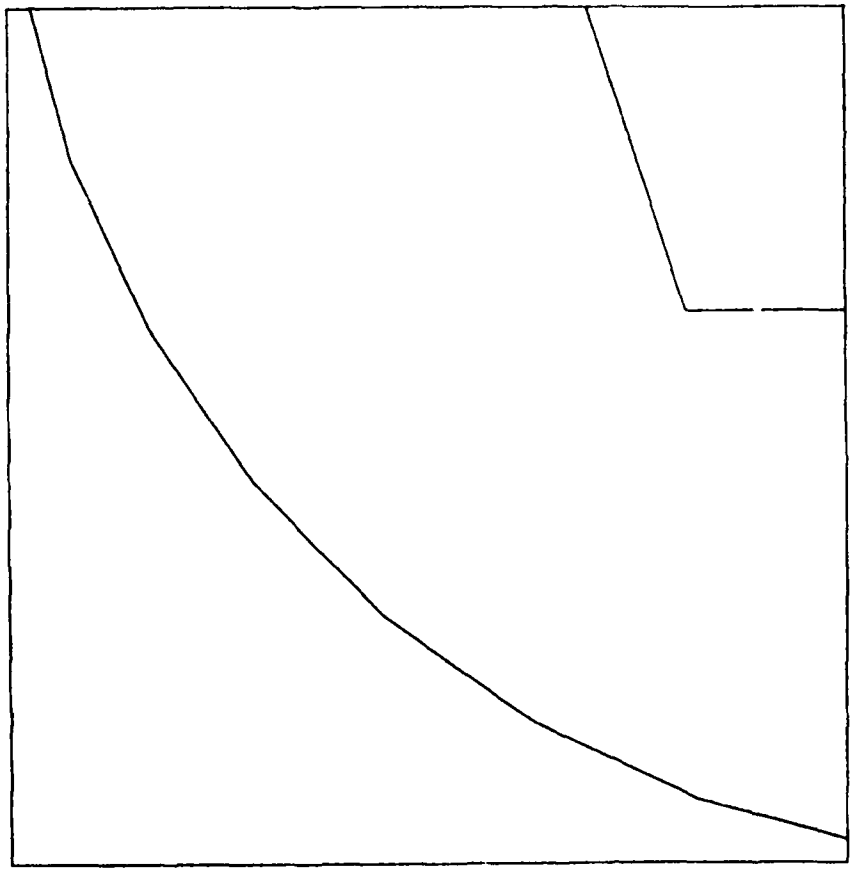


Figure 31. Example 3.5 (continued)

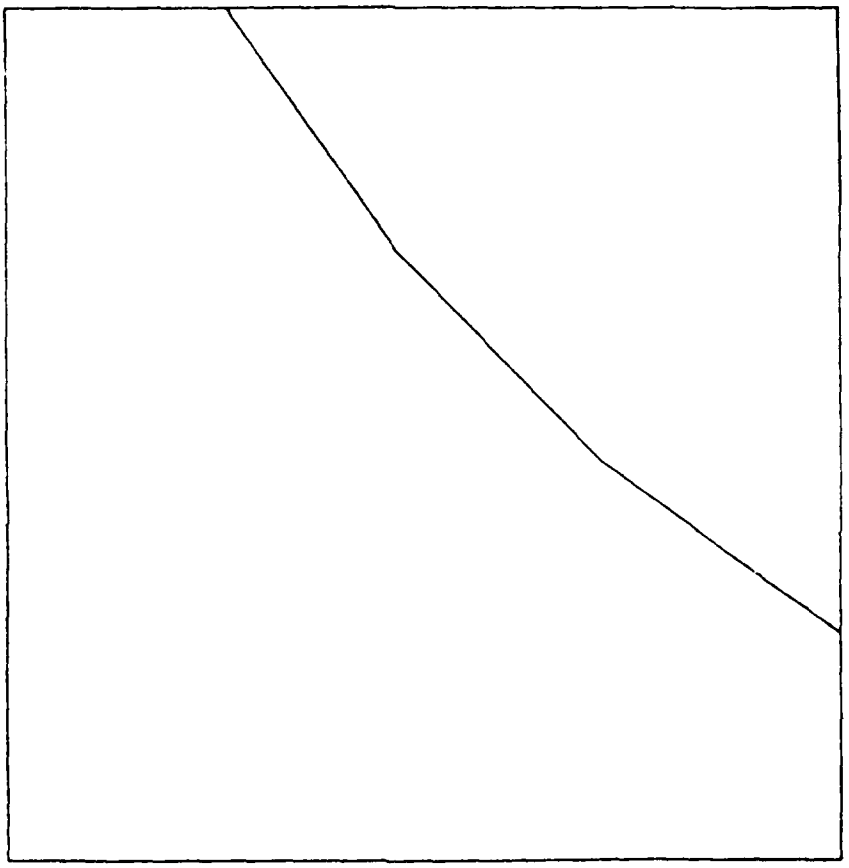


Figure 32. Example 3.5 (continued)

E X A M P L E 3.6

```
C
C SAMPLE PROGRAM TO GENERATE A SIMPLE STREET DIAGRAM
C FOR A TEKTRONIX 4010/4013 TERMINAL.  NOTE THAT ALL
C (X,Y) VALUES ARE GIVEN IN DEVICE UNITS.  THE DEFAULT
C DEVICE UNIT IS INCHES.
C
C INITIALIZATION, DEVICE MODE ENTRY AND OUTLINE
C GENERATION.
C
      CALL USTART
      CALL USET ('DEVICE UNITS')
      CALL UOUTLN
C
C GENERATION OF ROADS WITH THE DEFAULT CASE OF LINES
C IN TERMS OF INCHES.
C
      CALL UMOVE (0.3,2.7)
      CALL UPEN (7.2,2.7)
      CALL UMOVE (7.2,2.1)
      CALL UPEN (4.5,2.1)
      CALL UPEN (2.5,0.3)
      CALL UMOVE (1.7,0.3)
      CALL UPEN (3.7,2.1)
      CALL UPEN (0.3,2.1)
C
C GENERATION OF HOUSES WITH DASHED LINES IN TERMS OF
C CENTIMETERS.
C
      CALL USET ('CENTIMETERS')
      CALL USET ('DASHLINE')
      CALL UMOVE (12.5,5.0)
      CALL UPEN (15.,5.)
      CALL UPEN (15.,2.5)
      CALL UPEN (12.5,2.5)
      CALL UPEN (12.5,5.)
      CALL UMOVE (5.,7.5)
      CALL UPEN (11.3,7.5)
      CALL UPEN (11.3,10.0)
      CALL UPEN (5.,10.)
      CALL UPEN (5.,7.5)
C
C GENERATION OF DIRECTION REFERENCES WITH ARROW LINES
C IN TERMS OF PERCENTUNITS.
C
      CALL USET ('PERCENTUNITS')
      CALL USET ('ARROWHEAD LINE')
      CALL UMOVE (10.,80.)
      CALL UPEN (20.,80.)
      CALL UMOVE (15.,75.)
      CALL UPEN (15.,85.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

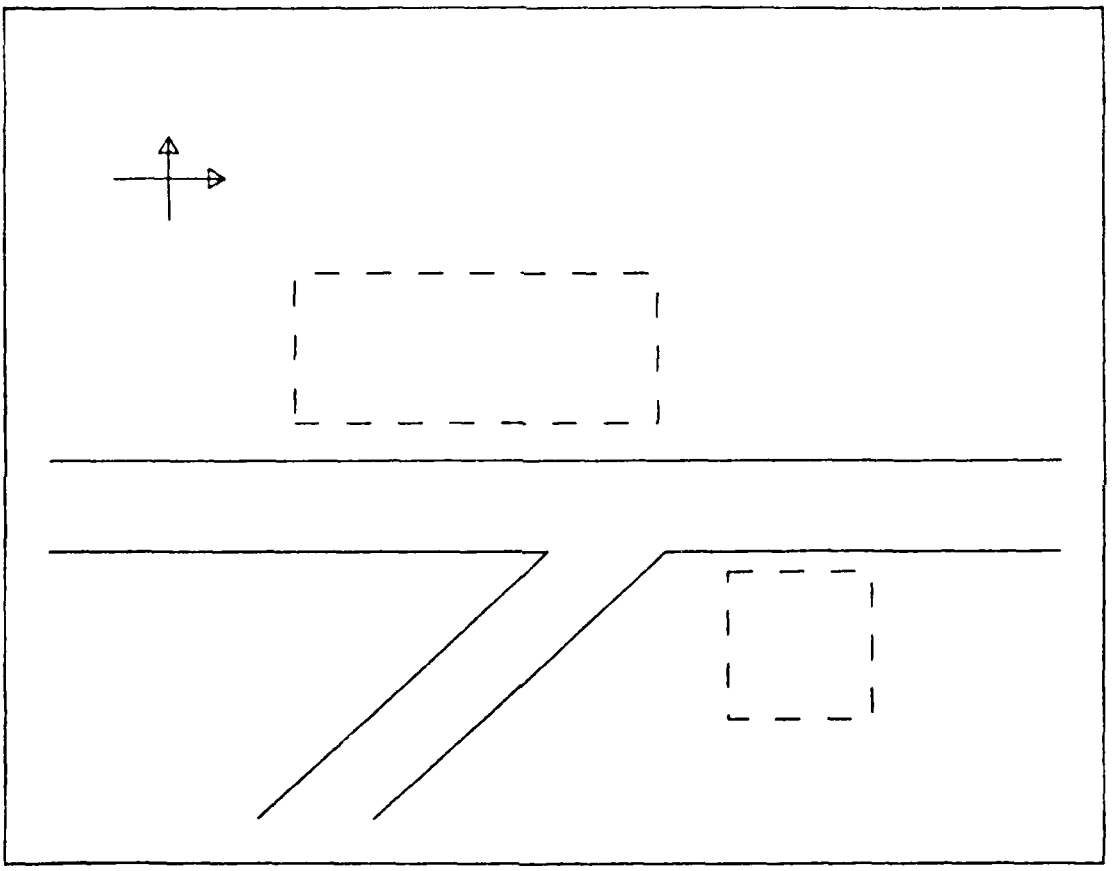


Figure 33. Example 3.6

E X A M P L E 3.8

```
C
C SAMPLE PROGRAM TO GENERATE THE SAME DISPLAY AT VARIOUS
C LOCATIONS ON THE DEVICE.  THIS IS WRITTEN FOR A
C TEKTRONIX 4014/4015.
C
C INITIALIZE GCS AND OUTLINE THE DEFAULT DEVICE AREA.
C
      CALL UERASE
      CALL UOUTLN
C
C FOR EACH OF THE THREE PASSES, DEFINE A DEVICE AREA,
C OUTLINE THE DEVICE AREA AND DRAW THE FIGURE.
C
      DO 10 I = 1, 3
      IF (I .EQ. 1) CALL UDAREA (5.,9.,0.,4.)
      IF (I .EQ. 2) CALL UDAREA (10.,14.,1.,5.)
      IF (I .EQ. 3) CALL UDAREA (6.,10.,6.,10.)
      CALL UOUTLN
C
C CALL THE SUBROUTINE TO DRAW THE FIGURE
C
      CALL GRAFIT
   10 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
      SUBROUTINE GRAFIT
C
C SUBROUTINE USED TO DRAW THE FIGURE.
C
      CALL UMOVE (10.,10.)
      CALL USET ('LINE')
      CALL UPEN (90.,10.)
      CALL UMOVE (20.,10.)
      CALL UPEN (30.,70.)
      CALL UPEN (70.,70.)
      CALL UPEN (80.,10.)
      CALL UMOVE (30.,70.)
      CALL USET ('DOUBLE ARROWHEAD LINE')
      CALL UPEN (40.,10.)
      CALL UPEN (50.,70.)
      CALL UPEN(60.,10.)
      CALL UPEN (70.,70.)
      CALL UEND
      STOP
      END
```

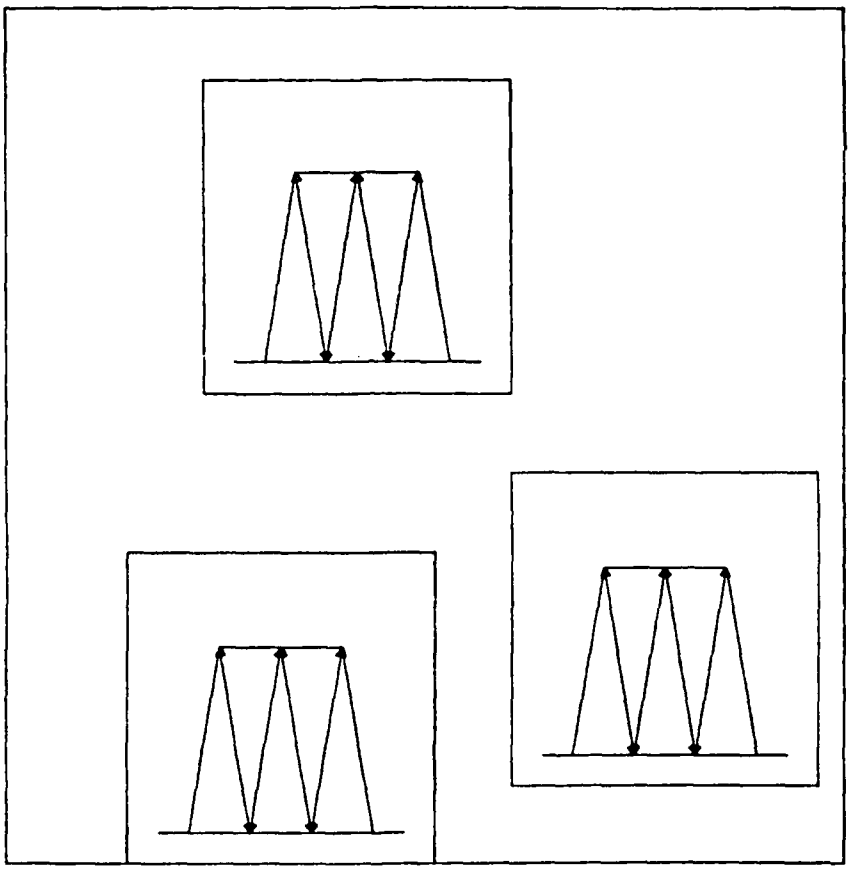


Figure 34. Example 3.8

E X A M P L E 3.9

```
C
C THIS PROGRAM GENERATES A FIGURE, A VERTICALLY
C DISTORTED VERSION OF THE FIGURE AND A
C HORIZONTALLY DISTORTED VERSION OF THE FIGURE.
C THIS IS WRITTEN FOR A TEKTRONIX 4010/4013.
C
C INITIALIZATION
C
      CALL USTART
C
C FOR EACH OF THE THREE PASSES, DEFINE A NEW DEVICE
C PLOTTING AREA, OUTLINE THE AREA AND DRAW THE FIGURE.
C
      XO = -2.1
      DO 10 I = 1, 3
      XO = XO + 2.5
      CALL UDAREA (XO,XO+2.,2.,3.5)
      IF(I.EQ.2) CALL UDAREA (XO,XO+2.,2.3,3.2)
      IF (I.EQ.3) CALL UDAREA (XO+.5,XO+1.5,2.,3.5)
      CALL UOUTLN
      CALL UMOVE (50.,20.)
      CALL UPEN1 (50.,80.,'DOUBLEARROW')
      CALL UMOVE (10.,90.)
      CALL UPEN (50.,65.)
      CALL UPEN (100.,65.)
      CALL UMOVE (10.,90.)
      CALL UPEN (50.,50.)
      CALL UPEN (100.,50.)
      CALL UMOVE (10.,90.)
      CALL UPEN (50.,35.)
      CALL UPEN (100.,35.)
   10 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

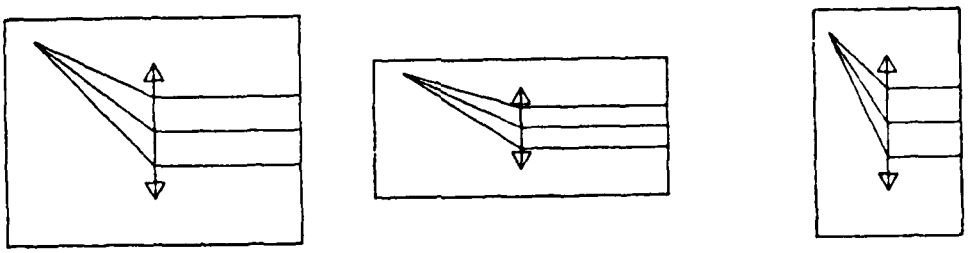


Figure 35. Example 3.9

E X A M P L E 4.1

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE OPTIONS AVAILABLE
C THROUGH 'UPRINT' AND 'UWRITE'.  THE DEFAULT VIRTUAL
C WINDOW AND DEVICE AREA WILL BE USED.
C
      DIMENSION COORD (2)
      DATA COORD/25.,0.999E7/
C
C INTIALIZE GCS.  CHANGE THE DEFAULT TERMINATOR (\)
C FOR 'TEXT' TO A ';' AND USE 'EXTRALARGE' CHARACTERS.
C OUTPUT A LINE OF TEXT AT (0.,75.).  NOTE THAT THE
C TEXT STRING IS CLIPPED AT THE DEVICE AREA BOUNDARY.
C
      CALL USTART
      CALL UPSET ('TERMINATOR CHARACTER',';')
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPRINT (0.,75.,'THIS IS A SAMPLE LINE OF OUTPUT TEXT
     & WHICH WILL BE CLIPPED;')
C
C SPECIFY 'REALNUMBER' MODE OF OPERATION AND USE
C 'UWRITE' TO OUTPUT THE NUMBER 100.
C
      CALL USET ('REALNUMBER')
      CALL UWRITE (50.,25.,100.)
C
C SPECIFY 'INTEGER' MODE AND OUTPUT THE NUMBER
C -123456789.  NOTICE THAT SINCE ALL GCS MUST BE
C REAL NUMBERS, EVEN THIS INTEGER MUST BE PASSED
C AS A REAL NUMBER.
C
      CALL USET ('INTEGER')
      CALL UPRINT (75.,0.,-123456789.)
C
C SPECIFY 'XYCOORDINATES' MODE.  NOTE THE VARIED
C FORM OF OUTPUT OF REAL NUMBERS WITH 'G' FORMAT.
C
      CALL USET ('XYCOORDINATES')
      CALL UWRITE (25.,50.,COORD)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

```
THIS IS A SAMPLE LINE OF OUTPUT TEXT WHICH WILL BE CLIP

(25.,.9990E+7)

100.

                                                -123456789
```

Figure 36. Example 4.1

E X A M P L E 4.:

```
C
C SAMPLE PROGRAM TO ILLUSTRATE THE OPTIONS AVAILABLE
C THROUGH MARGINNING.  DEFAULT VALUES WILL BE USED IN THIS
C EXAMPLE WITH ADDITIONAL CALLS TO 'UMARGN' TO ADJUST
C THE ALPHANUMERIC MARGIN BOUNDARIES.
C
C SET UP A 300 CHARACTER ARRAY NAMED SAMPLE AND INITIALIZE.
C NOTE THE USE OF A SEMICOLON (;) AS THE LAST CHARACTER.
C
      CHARACTER SAMPLE*300
      DATA SAMPLE/'THIS IS A LINE OF OUTPUT TEXT WHICH IS LONG
     & ENOUGH TO CAUSE THE ALPHANUMERIC OUTPUT TO WRAP-AROUND. NOTE THE
     & EFFECTS WHICH THE DEFAULT MARGINS HAVE UPON OUTPUT;'/
C
C INITIALIZE GKS.  SET ALL (X,Y) COORDINATES TO DEVICE UNITS
C AND CHANGE THE DEVICE UNITS TO PERCENTUNITS.  REMEMBER
C THAT MARGINING ONLY WORKS IN 'DEVICE' COORDINATE SPACE.
C ALSO CHANGE THE TEXT STRING TERMINATOR TO A ';' AND
C USE 'EXTRALARGE' CHARACTERS.
C
      CALL USTART
      CALL USET ('DEVICE UNITS')
      CALL UPSET ('TERMINATOR CHARACTER',';')
      CALL USET ('PERCENTUNITS')
      CALL USET ('EXTRALARGE CHARACTERS')
C
C OUTPUT THE TEXT STRING AND NOTE THE WRAP-AROUND.
C
      CALL UPRINT (31.,25.,SAMPLE)
C
C SPECIFY 'FONTUNITS' AS THE DEVICE UNIT OF MEASURE. THIS
C ALLOWS YOU TO SET MARGINS BASED UPON A SINGLE CHARACTER
C HEIGHT AND WIDTH.  REMEMBER THE RULES FOR MARGINNING
C IF THE STARTING POSTION OF THE TEXT STRING IS
C ABOVE THE TOP MARGIN.
C
C OUTPUT THE TEXT STRING.  NOTICE THE ';'.
C
      CALL USET ('FONTUNITS')
      CALL UMARGN (35.,36.,1.,35.)
      CALL UPRINT (0.,4000.,'HELLO THERE';')
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

THIS IS A LINE OF OUTPUT TEXT WHICH IS LONG ENOUGH
TO CAUSE THE ALPHANUMERIC OUTPUT TO WRAP-AROUND. NOTE THE EFFECTS WHICH
THE DEFAULT MARGINS HAVE UPON OUTPUT

Figure 37. Example 4.2

E X A M P L E 4.3

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE SINGLE CHARACTER
C OUTPUT AND LINE TERMINATOR OPTIONS AVAILABLE
C THROUGH 'UPEN'.  TWO GRAPHS WILL BE PLOTTED.
C
C SET UP X, Y AND Z ARRAYS.  PRESTORE DATA IN THEM.
C
      DIMENSION X(11), Y(11), Z(11)
      DATA X/0.,10.,20.,30.,40.,50.,60.,70.,80.,90.,100./
      DATA Y/0.,10.,20.,30.,40.,50.,60.,70.,80.,90.,100./
      DATA Z/0.,1.,4.,9.,16.,25.,36.,49.,64.,81.,100./
C
C INITIALIZE GCS AND OUTLINE THE DEFAULT DEVICE AREA.
C SET THE CHARACTER SIZE TO 'EXTRALARGE'.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UOUTLN
C
C SPECIFY THAT AN 'A' TERMINATOR WILL BE DRAWN AFTER
C EACH 'UPEN'.  MOVE TO AN INITIAL (X,Y) POINT.
C
      CALL USET ('LA')
      CALL UMOVE (X(1),Y(1))
C
C DRAW 11 LINE SEGMENTS AND NOTICE THE 'A'.
C
      DO 1 I = 1, 11
    1 CALL UPEN (X(I),Y(I))
C
C SPECIFY THAT 'NULL' LINES WILL BE DRAWN WITH A 'B'
C AT THE END OF EACH INVISIBLE LINE SEGMENT.
C
      CALL USET ('NB')
C
C MOVE TO THE FIRST (X,Z) POINT, THEN PLOT 11 (X,Z)
C VALUES.
C
      CALL UMOVE (X(1),Z(1))
      DO 2 I = 1, 11
    2 CALL UPEN (X(I),Z(I))
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

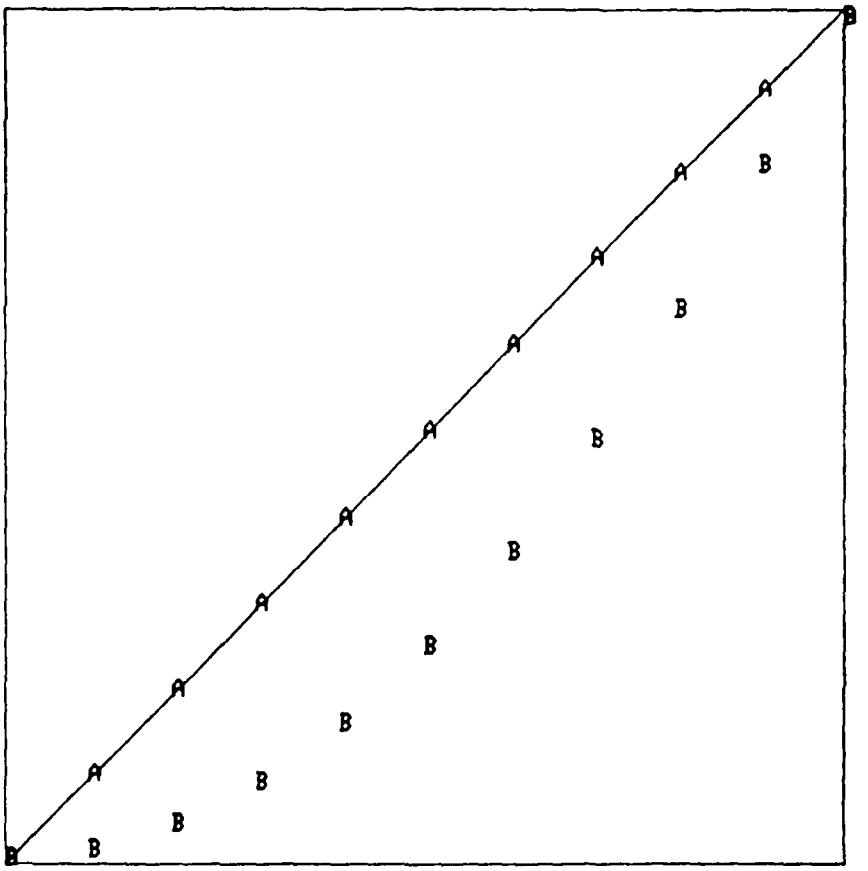


Figure 38. Example 4.3

E X A M P L E 4.4

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE CHARACTER
C TERMINATOR AND ALPHANUMERIC OUTPUT USED IN
C CONJUNCTION WITH GRAPHICAL OUTPUT.  THIS PROGRAM
C WAS WRITTEN FOR A TEKTRONIX 4014/4015 TERMINAL.
C
C ALLOCATE ARRAYS AND INITIALIZE ALL VARIABLES
C THAT WILL BE USED
C
      CHARACTER ROUTINE*7(4)
      INDEX = 0
C
      ROUTINE(1)='UPEN;'
      ROUTINE(2)='UAOUT;'
      ROUTINE(3)='UPRNT1;'
      ROUTINE(4)='UWRIT1;'
C
C INITIALIZE GCS.  DIVIDE THE DEVICE PLOTTING
C AREA INTO FOUR EQUAL PARTS, CHOOSE ONE PART
C AND OUTLINE THE 'UDAREA'.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('TERMINATOR CHARACTER',';')
      DO 5 I = 1,4
      INDEX = INDEX + 1
      IF (I.EQ.1) CALL UDAREA(2.,7.,5.5,10.5)
      IF (I.EQ.2) CALL UDAREA(7.5,12.5,5.5,10.5)
      IF (I.EQ.3) CALL UDAREA(2.,7.,0.,5.)
      IF (I.EQ.4) CALL UDAREA(7.5,12.5,0.,5.)
      CALL UOUTLN
C
C MOVE TO (0.,0.) AND SPECIFY STANDARD LINE WITH
C NO TERMINATOR.  IF 'UPEN' OPTION IS IN EFFECT
C SPECIFY AN 'A' AS THE TERMINATOR.
C
      CALL UMOVE (0.,0.)
      CALL USET ('LNULL')
      IF (INDEX .EQ. 1) CALL USET ('LA')
C
C DRAW A LINE THEN BRANCH TO ONE OF THE FOUR
C ROUTINES TO PRINT AN 'A' AT THE END OF THE LINE.
C
      DO 4 K = 1, 4
      CALL UPEN ((25.*FLOAT(K)),(25.*FLOAT(K)))
      GO TO (4,1,2,3), INDEX
    1 CALL UAOUT ('A;')
      GO TO 4
    2 CALL UPRNT1 ('A;','TEXT')
      GO TO 4
    3 CALL UWRIT1 ('A;','TEXT')
    4 CONTINUE
C
C OUTPUT THE NAME OF THE ROUTINE USED IN THE
C BOTTOM RIGHT CORNER.  REMEMBER THE WINDOW
C BOUNDARIES ARE THE SAME FOR ALL FOUR DEVICE AREAS.
C
      CALL UPRINT (75.,2.,ROUTINE(INDEX))
    5 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

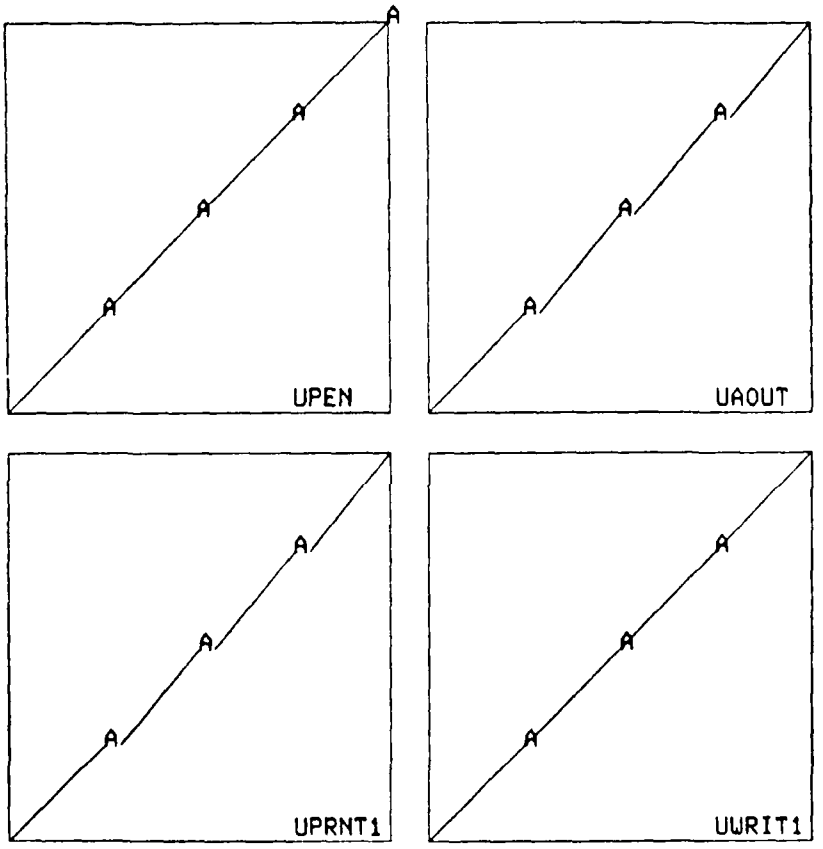


Figure 39. Example 4.4

E X A M P L E 4.5

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE ALPHANUMERIC OUTPUT
C THROUGH 'SOFTWARE' CHARACTER OPTION IN GCS.  THESE
C EXAMPLES SHOW: SOFTWARE CHARACTER OUTPUT USING ITALICIZED
C CHARACTERS, GOTHIC (DEFAULT) OUTPUT WITH CHARACTERS REORIENTED
C AND REDUCED AND ROTATED SOFTWARE CHARACTER OUTPUT.
C
      CHARACTER OUTPT*50
      CHARACTER OUT*1(50)
      CHARACTER FORMT*13
      DATA OUTPT/'REDUCED AND ROTATED CHARACTERS '
      DATA DEG/45./
C
C INITIALIZE GCS AND SPECIFY 'SOFTWARE' CHARACTER .
C
      CALL USTART
      CALL UOUTLN
      CALL USET ('SOFTWARE GENERATED CHARACTER '
C
C OUTPUT TEXT WITH 'ITALICS' SOFTWARE CHARACTER
C
      CALL USET ('ITALICS')
      CALL UPRINT (10.,5.,'SAMPLE OF ITALICS '
C
C SPECIFIED 'GOTHIC' (DEFAULT) CHARACTERS AND
C ALSO ROTATE THE TEXT STRING.  NOTE THAT SETTING
C ONLY TO 'SOFTWARE' CHARACTERS AND EACH CHARACTER
C IS REORIENTED.
C
      CALL USET ('GOTHIC')
      CALL UPSET ('ROTATION',DEG)
      CALL UPRINT (10.,90.,'DEFAULT TEXT SIZE '
C
C CHANGE SOFTWARE CHARACTER SIZE AND ROTATE THE ENTIRE
C TEXT STRING INSTEAD OF EACH CHARACTER.
C
      XSIZE = 2.
      YSIZE = 3.
      CALL UPSET ('HORIZONTAL CHARACTER WIDTH',XSIZE
      CALL UPSET ('VERTICAL CHARACTER HEIGHT',YSIZE
C
C DETERMINE THE NUMBER OF CHARACTERS
C
      CALL UCOUNT (OUTPT,COUNT)
      ICNT = COUNT
C
C USE FORTRAN 'ENCODE' TO BUILD A FORMAT
C
      I1 = ICNT/10
      I2 = ICNT - I1*10
      ENCODE (FORMT,100) '(',I1,I2,'A1)'
  100 FORMAT(A1,I1,I1,A3)
C
C USE FORTRAN 'DECODE' TO SEPARATE CHARACTERS
C
      DECODE (OUTPT,FORMT) (OUT(I),I=1,ICNT)
C
C OUTPUT TEXT STRING
C
      CDEG = 3.1416/180.
      X = XSIZE * COS (DEG*CDEG)
      Y = XSIZE * SIN (DEG*CDEG)
      XX = 25.
      YY = 25.
      DO 300 I = 1, ICNT
      CALL UMOVE (XX,YY)
      CALL UAOUT (OUT(I))
      XX = XX + X
      YY = YY + Y
  300 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

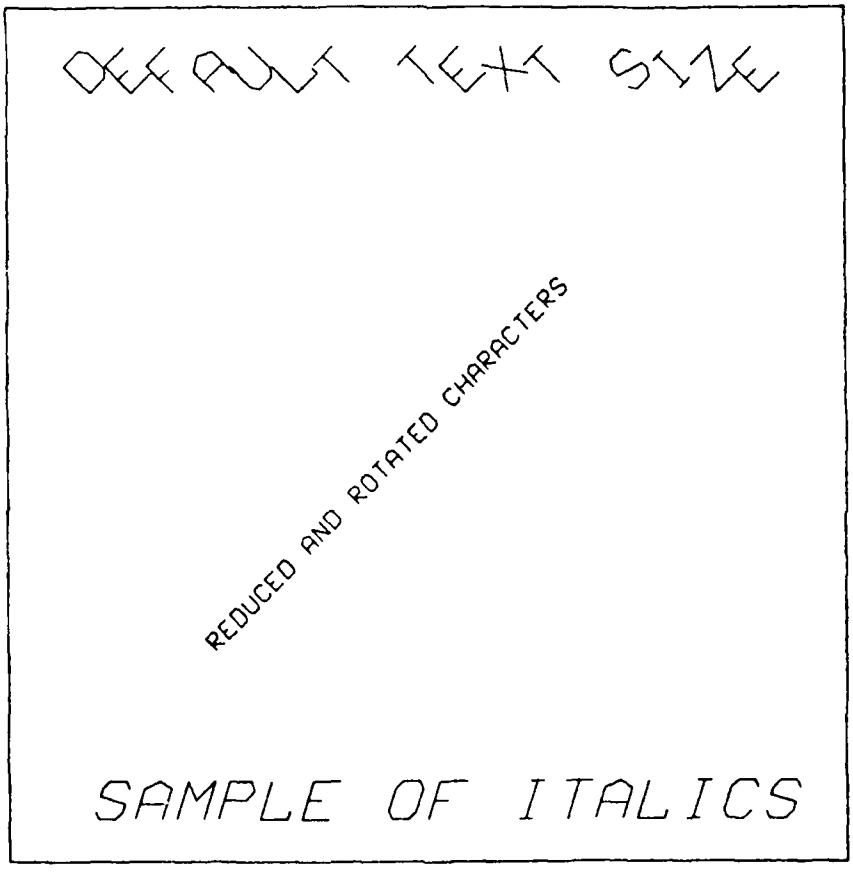


Figure 40. Example 4.5

E X A M P L E 5.1

```
C
C SAMPLE PROGRAM WHICH ILLUSTRATES GRAPHICS INPUT
C THROUGH 'UGRIN'.  THREE TYPES OF CASES ARE
C HANDLED: SOLID 'S', INVISIBLE 'I' AND DASHED
C 'D' LINES. THE DESIRED OPTION FOR THE LINE IS
C ENTERED AS A SINGLE CHARACTER WHEN THE CURSORS
C HAVE BEEN POSITIONED.  THIS PROGRAM WAS WRITTEN
C FOR A TEKTRONIX TERMINAL AND IT MAY BE
C NECESSARY TO PUSH THE RETURN KEY AFTER THE
C SINGLE CHARACTER IS ENTERED.  DEFAULT VALUES
C OF WINDOW, DEVICE AREA AND DASH SPECIFICATION ARE USED.
C AN 'E' WILL TERMINATE THE PROGRAM.
C
C
      CHARACTER CHAR * 1
C
C INITIALIZE GCS AND OUTLINE THE DEFAULT DEVICE AREA.
C
      CALL USTART
      CALL UOUTLN
C
C ENABLE THE CURSORS, POSITION THEM WHERE DESIRED
C AND ENTER A SINGLE CHARACTER.  THE (X,Y) LOCATION
C AND THE SINGLE CHARACTER ENTERED WILL BE RETURNED
C TO THE PROGRAM.
C
    1 CALL UGRIN (X,Y,CHAR)
C
C CHECK IF THE SINGLE CHARACTER IS AN 'S', 'I' OR 'D'.
C PERFORM THAT GRAPHICS FUNCTION IF YES. IF AN 'E' IS
C ENTERED, STOP THE PROGRAM.
C
      IF (CHAR .EQ. 'S') CALL UPEN1 (X,Y,'LINE')
      IF (CHAR .EQ. 'I') CALL UMOVE (X,Y)
      IF (CHAR .EQ. 'D') CALL UPEN1 (X, Y, 'DASH')
      IF (CHAR .EQ. 'E') GO TO 2
      GO TO 1
    2 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

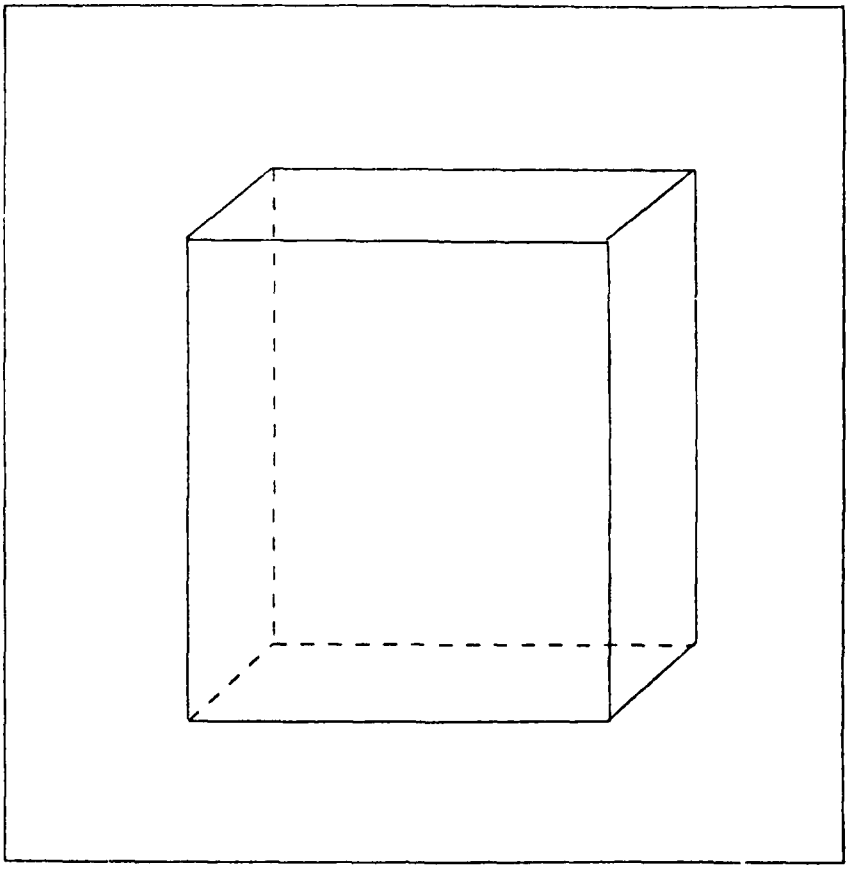


Figure 41. Example 5.1

E X A M P L E 5.2

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE USE OF 'UINPUT' TO
C ACCEPT ALPHANUMERIC INPUT FROM A USER, EDIT IT INTO
C THE PROPER FORMAT, STORE IT IN A DATA ARRAY AND PRINT
C THE DATA AT A DIFFERENT LOCATION ON THE DISPLAY.
C
C DEFINE AND INITIALIZE SOME DATA ARRAYS
C
      CHARACTER OPTION*12(4)
      DIMENSION COUNT(4),DATA(6),INDEX(4)
      COUNT(1)=5.
      COUNT(2)=1.
      COUNT(3)=1.
      COUNT(4)=1.
      INDEX(1)=1
      INDEX(2)=3
      INDEX(3)=4
      INDEX(4)=5
      OPTION(1)='TEXT'
      OPTION(2)='REALNUMBER'
      OPTION(3)='INTEGER'
      OPTION(4)='XYCOORDINATE'
      X = 5.
      Y = 90.
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE AND
C OUTLINE THE DEFAULT 'UDAREA'.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UOUTLN
C
C DEFINE A LOOP TO ILLUSTRATE THE FOUR INPUT AND
C OUTPUT OPTIONS.
C
      DO 1 I = 1, 4
C
C POSITION BEAM/PEN TO INITIAL LOCATION
C
      CALL UMOVE (X,Y)
C
C ALERT THE USER THAT INPUT IS DESIRED, THEN ACCEPT DATA
C
      CALL UPRNT1 ('ENTER: \','TEXT')
      CALL UINPUT (DATA(INDEX(I)),COUNT(I),FLAG,OPTION(I))
C
C IF 'TEXT' OPTION, INSERT TERMINATOR CHARACTER AT END
C OF INPUTTED TEXT STRING.
C
      IF (I.EQ.1) CALL UAPEND (COUNT(1),DATA(INDEX(1)),DATA(INDEX(1)))
C
C SET THE CORRECT OPTION TO OUTPUT THE DATA THE USER
C HAS JUST ENTERED AND PRINT THE DATA.
C
      CALL USET (OPTION(I))
      CALL UPRINT (X,10.,DATA(INDEX(I)))
C
C UPDATE COORDINATE LOCATIONS FOR NEXT INPUT PROMPT.
C
      X = X + 22.5
      Y = Y - 20.
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

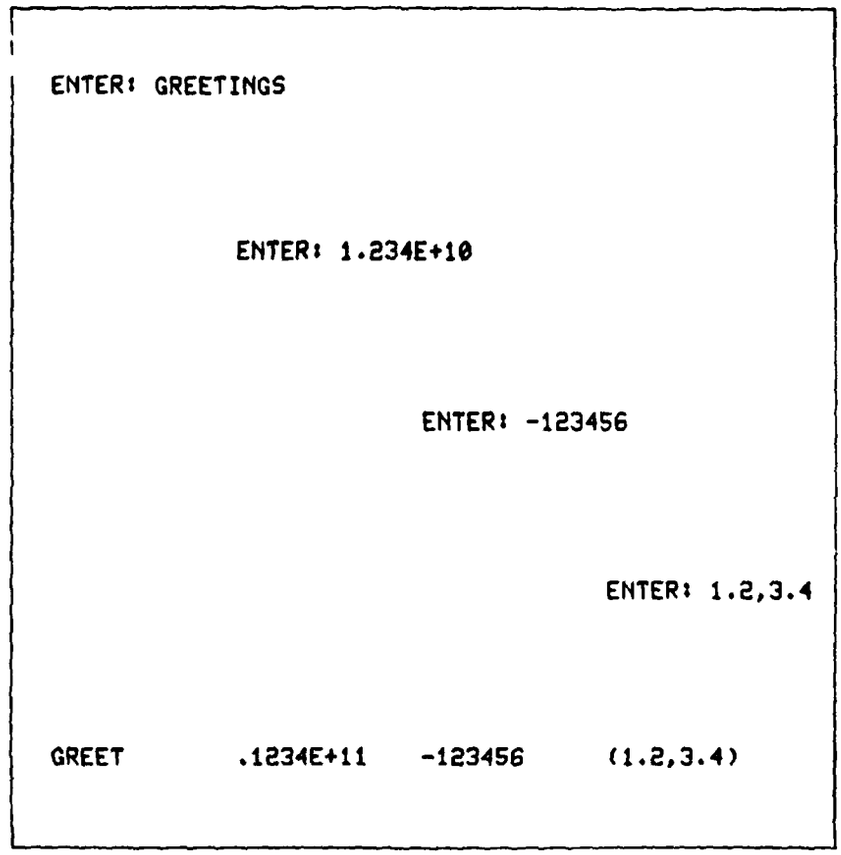


Figure 42. Example 5.2

E X A M P L E 5.3

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE MENUING.  THE CHARACTER
C ARRAY 'OPTION' CONTAINS THE LABELS TO BE PRINTED UNDER
C EACH OF THE MENU SELECTION BOXES.  THE NUMBER OF THE
C BOX WHICH WAS SELECTED BY THE USER POSITIONING THE
C CROSSHAIRS AND ENTERING A CHARACTER IS RETURNED IN THE
C PARAMETER 'CHOICE'.  THIS WAS WRITTEN FOR A TEKTRONIX
C TERMINAL.  THE DEFAULT VALUES OF WINDOW AND DEVICE AREA
C ARE USED.
C
C INITIALIZE LABELS FOR THE MENU CHOICES
C
C
      CHARACTER OPTION * 8(9)
      DATA OPTION/'OPTION 1','OPTION 2','OPTION 3','OPTION 4',
     &            'OPTION 5','OPTION 6','OPTION 7','OPTION 8',
     &            'OPTION 9'/
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE AND
C OUTLINE THE DEFAULT DEVICE AREA (UDAREA).
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UOUTLN
C
C CALL 'UMENU' TO DRAW A MENUBOARD OF 9 OPTIONS AND ACCEPT
C THE USER'S SELECTION.  PUSH A SINGLE CHARACTER AND THE
C RETURN KEY.
C
      CALL UMENU (9.0,OPTION,CHOICE)
C
C PRINT WHAT MENU BOX SELECTED
C
      CALL UPRINT (25.,25.,'THE MENU BOX SELECTED WAS \')
      CALL UPRNT1 (CHOICE,'INTEGER')
C
C CALL 'UMENU' AGAIN, BUT USE A MINUS (-) SIGN TO SPECIFY
C THAT THE MENUBOARD IS NOT TO BE REDRAWN.  ENTER ANOTHER
C SINGLE CHARACTER AND THE RETURN KEY.
C
      CALL UMENU (-9.0,OPTION,CHOICE)
C
C PRINT WHAT MENU BOX SELECTED
C
      CALL UPRINT (25.,50.,'THE MENU BOX SELECTED WAS \')
      CALL UPRNT1 (CHOICE,'INTEGER')
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

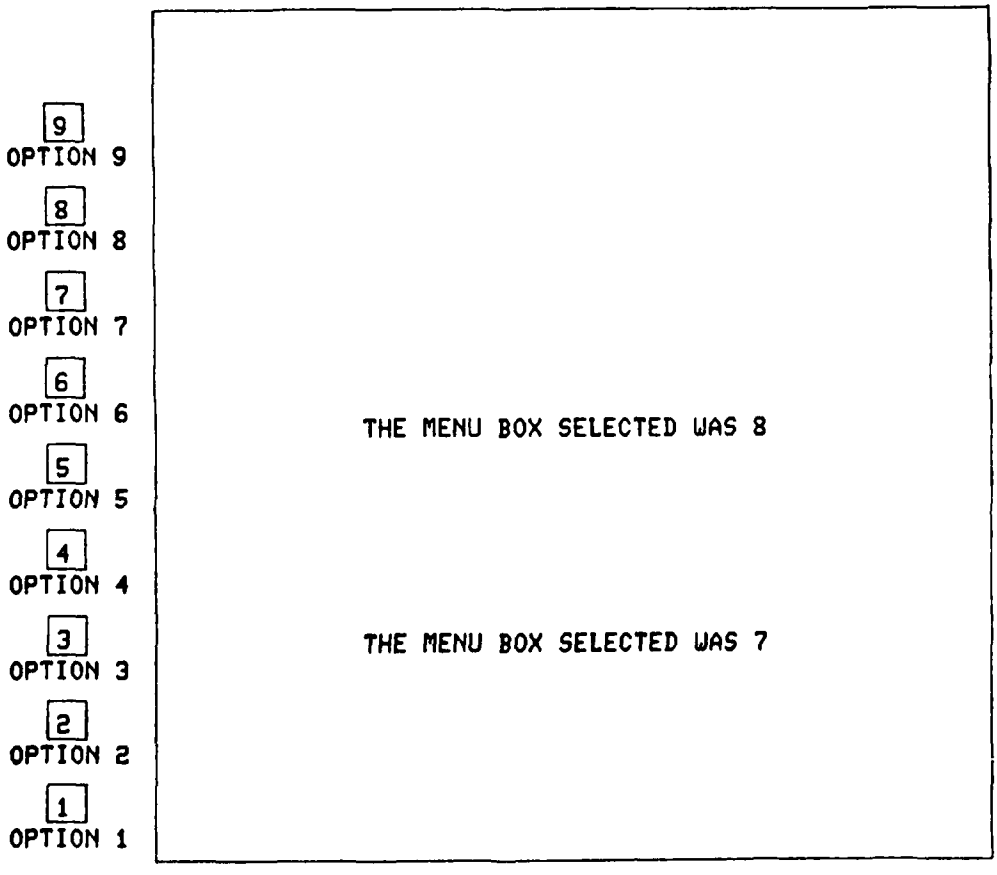


Figure 43. Example 5.3

E X A M P L E 6.1

```
C     PROBLEM SOLUTION I
      CALL USTART
      CALL UPSET('SPEED',120.)
      CALL USTART
      CALL AREA1
      CALL SQUARE
      CALL AREA2
      CALL TRIAN
      CALL COMMND
      CALL UEND
      STOP
      END
      SUBROUTINE AREA1
      CALL UDAREA(0.,7.,0.,7.)
      CALL UOUTLN
      CALL UEND
      STOP
      END
      SUBROUTINE AREA2
      CALL UDAREA(7.3,14.3,0.,7.)
      CALL UOUTLN
      CALL UEND
      STOP
      END
      SUBROUTINE SQUARE
      CALL UMOVE(20.,20.)
      CALL UPEN(80.,20.)
      CALL UPEN(80.,80.)
      CALL UPEN(20.,80.)
      CALL UPEN(20.,20.)
      CALL UPRINT(5.,5.,'SQUARE\')
      CALL UEND
      STOP
      END
      SUBROUTINE TRIAN
      CALL UMOVE(20.,20.)
      CALL UPEN(80.,20.)
      CALL UPEN(50.,80.)
      CALL UPEN(20.,20.)
      CALL UPRINT(5.,5.,'TRIANGLE\')
      CALL UEND
      STOP
      END
      SUBROUTINE COMMND
      CHARACTER CHAR*1
  100 CALL UAIN(CHAR)
      CALL UERASE
      IF(CHAR.EQ.'B') GO TO 200
      IF(CHAR.EQ.'S') GO TO 300
      IF(CHAR.EQ.'T') GO TO 400
      IF(CHAR.EQ.'R') GO TO 500
      CALL ERROR
      GO TO 100
  200 CALL AREA1
      CALL SQUARE
      CALL AREA2
      CALL TRIAN
      GO TO 100
  300 CALL AREA
      CALL SQUARE
      GO TO 100
  400 CALL AREA
      CALL TRIAN
      GO TO 100
      CALL UEND
  500 STOP
      END
```

E X A M P L E 6.1 (continued)

```
      SUBROUTINE ERROR
      CALL UHOME
      CALL UALPHA
      PRINT,'B - BOTH FIGURES'
      PRINT,'S - SQUARE'
      PRINT,'T - TRIANGLE'
      PRINT,'R - RETURN'
      CALL UEND
      STOP
      END
      SUBROUTINE AREA
      CALL UDAREA(0.,10.,0.,10.)
      CALL UOUTLN
      CALL UEND
      STOP
      END
```

```
6 - BOTH FIGURES
5 - SQUARE
7 - TRIANGLE
0 - RETURN
```

Figure 44. Example 6.1

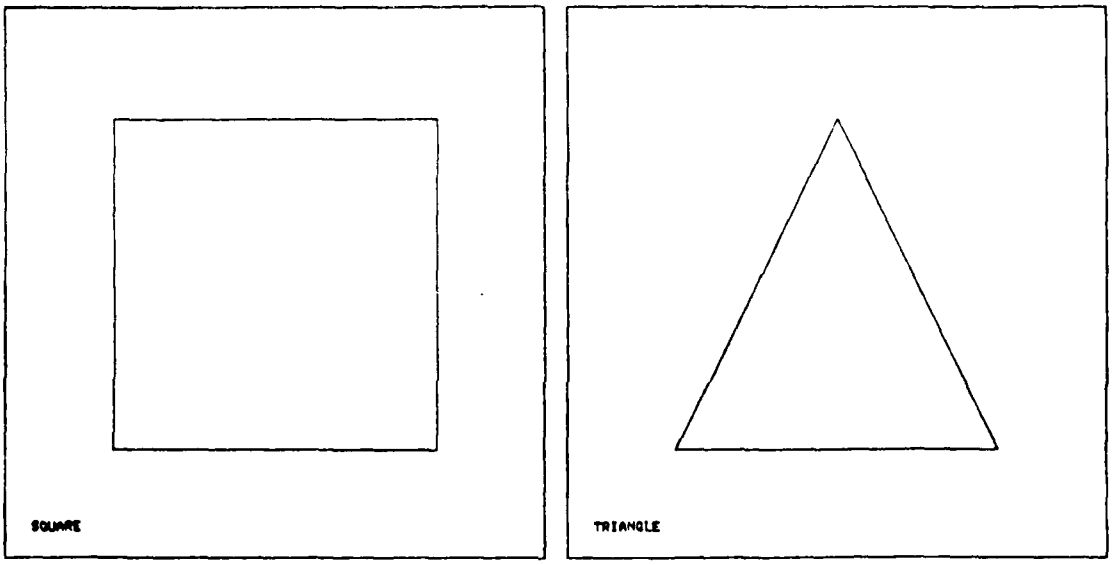


Figure 45. Example 6.1 (continued)

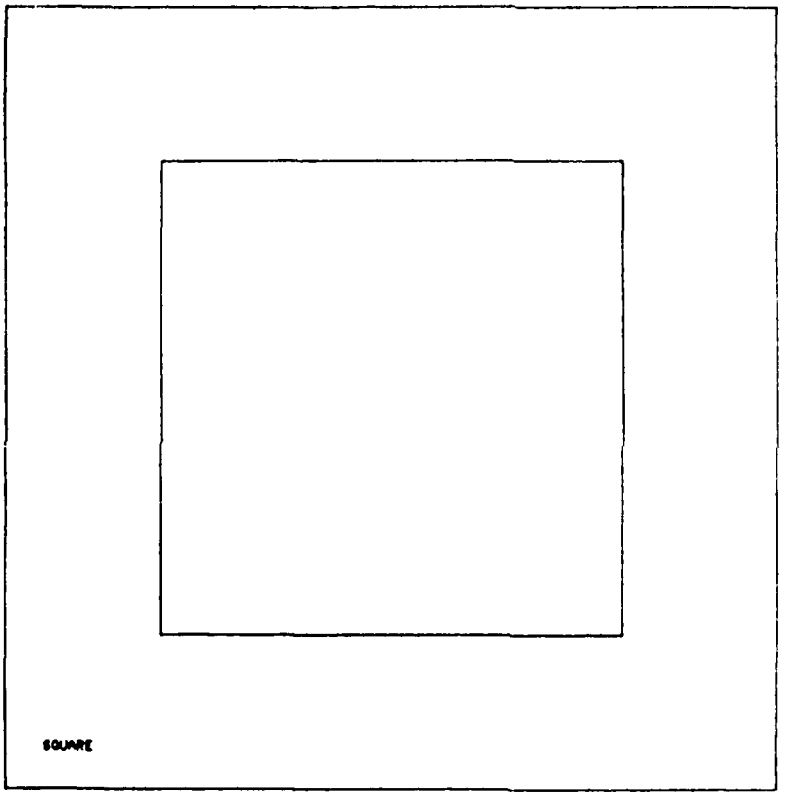


Figure 46. Example 6.1 (continued)

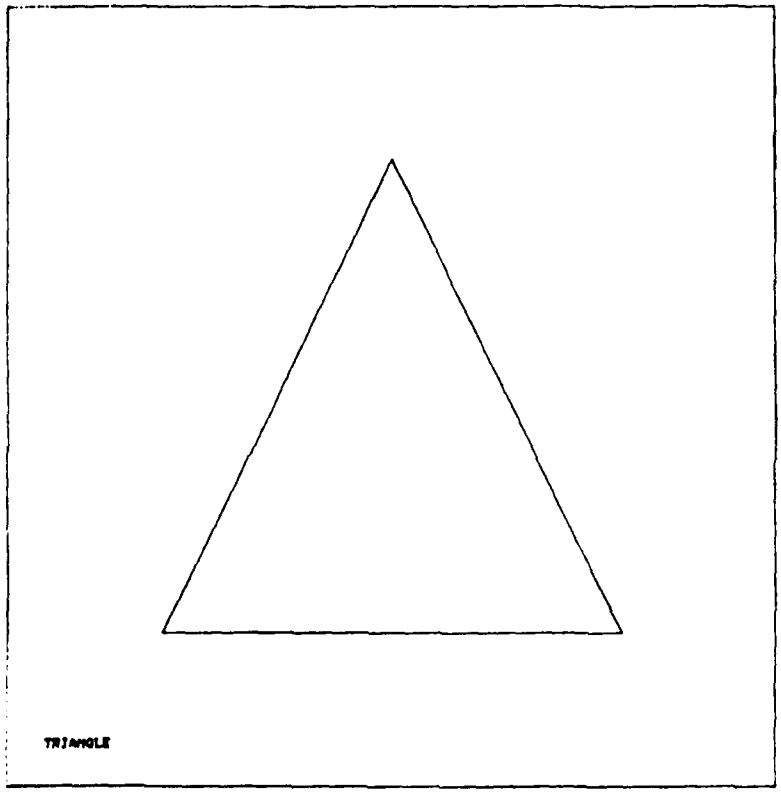


Figure 47. Example 6.1 (continued)

E X A M P L E 7.1

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE 'UCRCLE'.
C DEFAULT WINDOW AND DEVICE AREA ARE USED.  NOTE
C THE EFFECT OF 'CLIPPING' DUE TO THE VIRTUAL
C WINDOW'S RESTRICTING OF ALL GRAPHICAL
C INFORMATION TO RESIDE WITHIN THE REGION DEFINED
C BY 100.0 BY 100.0 SQUARE.
C
C INITIALIZE GCS AND OUTLINE THE DEFAULT DEVICE AREA.
C
      CALL USTART
      CALL UOUTLN
C
C DRAW A SERIES OF CIRCLES WITH RADIUS OF 50.
C
      CALL UCRCLE (0.,0.,50.)
      CALL UCRCLE (50.,0.,50.)
      CALL UCRCLE (100.,0.,50.)
      CALL UCRCLE (100.,50.,50.)
      CALL UCRCLE (100.,100.,50.)
      CALL UCRCLE (50.,100.,50.)
      CALL UCRCLE (0.,100.,50.)
      CALL UCRCLE (0.,50.,50.)
C
C DRAW A CIRCLE OF RADIUS 20.7 WITH A CENTER AT (50.,50.).
C
      CALL UCRCLE (50.,50.,20.7)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

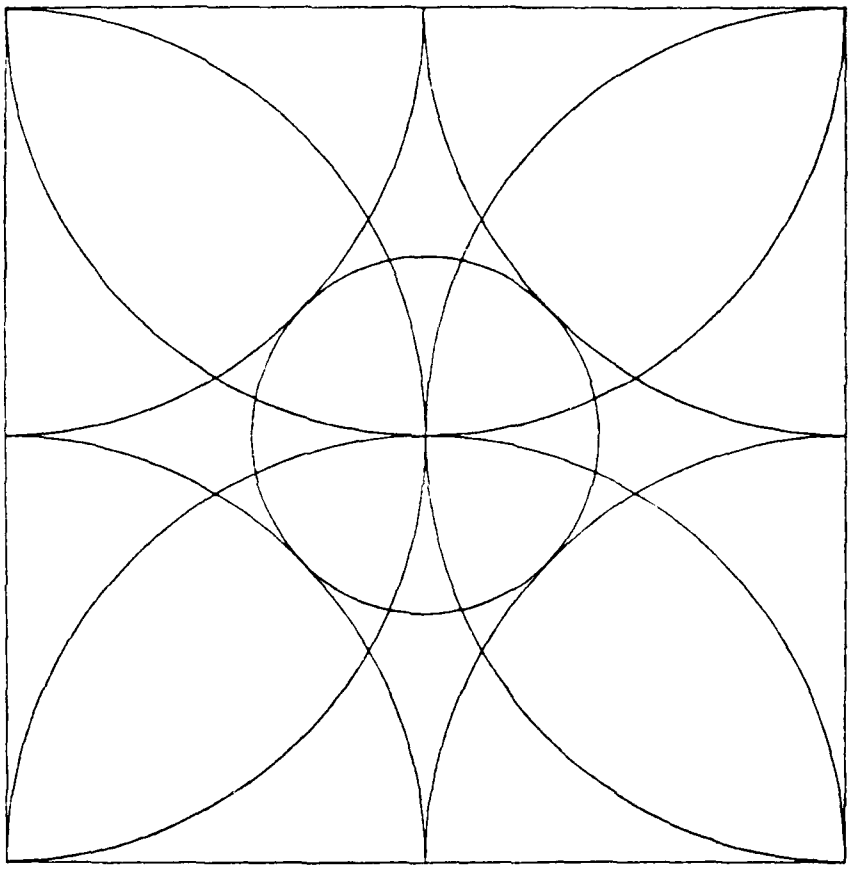


Figure 48. Example 7.1

E X A M P L E 7.2

```
C
C SAMPLE PROGRAM USED TO DEMONSTRATE HOW 'UARC'
C DRAWS AN ARC CENTERED AT (40.,20.) AND OF
C RADIUS 30.  THE ARC WILL BEGIN AT (40.,50.) AND
C TERMINATE AT (10.,20.).
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE
C AND OUTLINE THE DEFAULT 'UDAREA'.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UOUTLN
C
C MOVE TO WHERE THE ARC IS TO BEGIN AND LABEL
C
      CALL UPEN1 (40.0,50.0,'NCOORDINATES')
      CALL UMOVE (40.0,50.0)
C
C CALL 'UARC' TO GENERATE THE ARC WITH AN ANGULAR
C SPAN OF 90.0 DEGREES.
C
      CALL UARC (40.0,20.0,90.0)
C
C DETERMINE WHERE 'UARC' COMPLETED THE ARC AND PRINT
C THE COORDINATES OF THIS LOCATION.  ALSO PRINT THE
C COORDINATES OF THE CENTER OF THE ARC.
C
      CALL UWHERE (X,Y)
      CALL UPEN1 (X,Y,'NCOORDINATES')
      CALL UPEN1 (40.0,20.0,'NCOORDINATES')
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

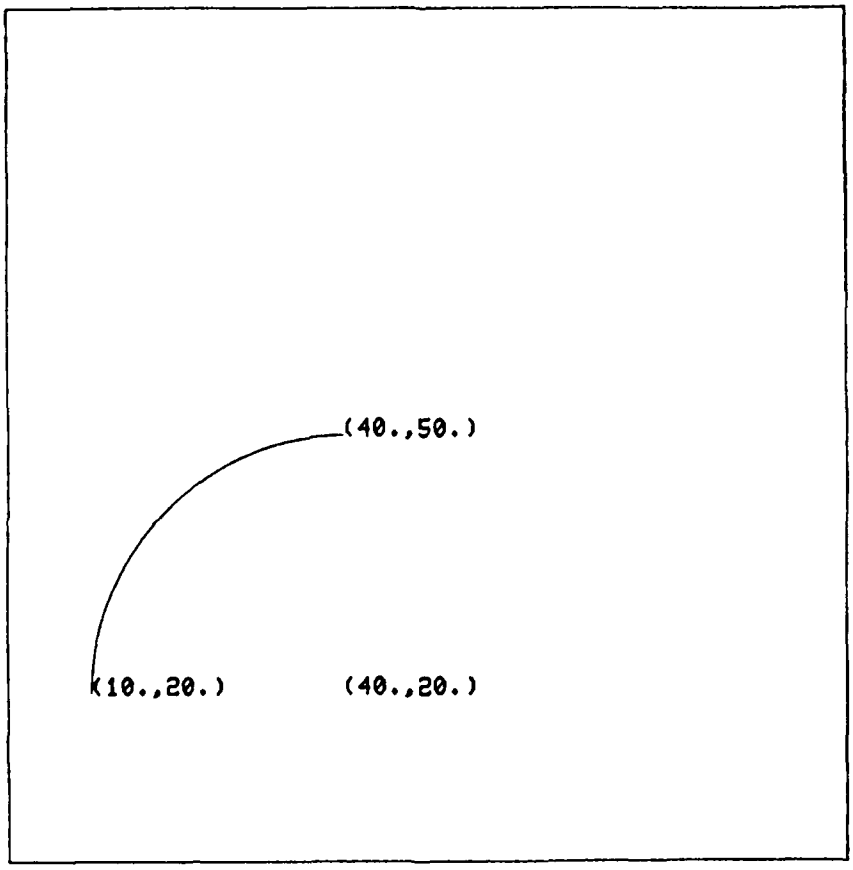


Figure 49. Example 7.2

E X A M P L E 7.3

```
C
C SAMPLE PROGRAM USED TO DEMONSTRATE USE OF 'UPLYGN'.
C POLYGONS OF FROM 2 TO 9 SIDES WILL BE DRAWN WITHIN
C THEIR OWN WINDOW, WHICH IS MAPPED TO DIFFERENT
C PORTIONS OF THE SCREEN.  THIS PROGRAM WAS WRITTEN
C FOR A TEKTRONIX 4014/4015 TERMINAL.
C
C INITIALIZE VARIABLES
C
      SIDES = 1.0
      YO = 9.4
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE
C AND CHANGE TEXT STRING TERMINATOR TO A SEMICOLON (;).
C SET VIRTUAL WINDOW TO -1.1 TO 1.1 FOR X AND Y. THIS
C SMALL WINDOW WILL BE DRAWN 8 TIMES; 2 ROWS OF FOUR,
C EACH TIME CONTAINING A DIFFERENT POLYGON WITH THE
C DEVICE AREA OUTLINED.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('TERMINATOR CHARACTER',';')
      CALL UWINDO (-1.1,1.1,-1.1,1.1)
C
C THE FOLLOWING DO LOOPS SET UP THE 2 ROWS OF 4 DISPLAYS.
C
      DO 1 I = 1, 2
      XO = -2.6
      YO = YO - 3.6
      DO 1 J = 1, 4
      XO = XO + 3.3
C
C INITIALIZE NUMBER OF SIDES; 1 IS ADDED BEFORE EACH
C EXECUTION SO POLYGON SIDES START AT 2 AND GO TO 9
C IN 8 STEPS.
C
      SIDES = SIDES + 1.0
C
C SET UP THE DEVICE AREA AND OUTLINE IT.  AS XO AND YO
C CHANGE, IT WILL MOVE TO 8 DIFFERENT LOCATIONS.
C
      CALL UDAREA (XO,(XO+3.0),YO,(YO+3.0))
      CALL UOUTLN
C
C DRAW THE POLYGON
C
      CALL UPLYGN (0.0,0.0,SIDES,1.0)
C
C LABEL THE POLYGON
C
      CALL USET ('TEXT')
      CALL UPRINT (-1.0,-1.05,'SIDES:;')
      CALL USET ('INTEGER')
      CALL UPRINT (0.9,-1.05,SIDES)
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

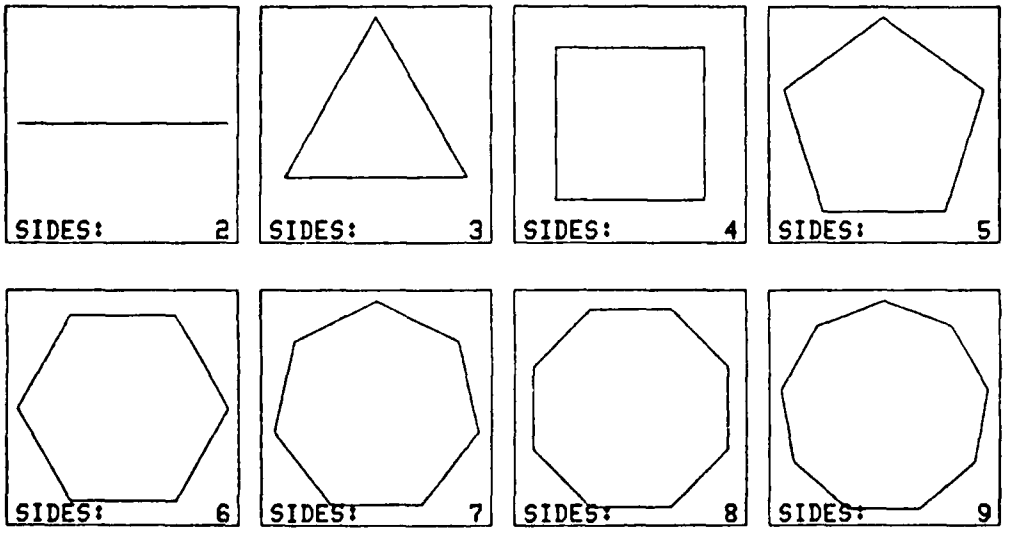


Figure 50. Example 7.3

E X A M P L E 7.4

```
C
C SAMPLE PROGRAM USED TO DEMONSTRATE USE OF 'UPLYGN'.
C A TRIANGLE WILL BE DRAWN USING 1 OF 12 POSIBLE PEN OPTIONS.
C THIS PROGRAM WAS WRITTEN FOR A TEKTRONIX 4014/4015.
C
C INITIALIZE CHARACTER ARRAY WITH OPTIONS TO BE USED BY
C 'USET' AND 'UPRINT'.  NOTE THAT A BACKSLASH (\) IS USED
C AS THE LAST CHARACTER.  'USET' CHECKS ONLY THE FIRST 4
C CHARACTERS, BUT 'UPRINT' MUST HAVE THE BACKSLASH.
C
      CHARACTER OPTION*16(12)
      DATA OPTION/'LNULL\','LARROW\','LBACKARROW\',
     & 'LDOUBLEARROW\','DNULL\','DARROW\','DBACKARROW\','DDOUBLEARROW\',
     & 'TNULL\','TARROW\','TBACKARROW\','TDOUBLEARROW\'/
      INDEX = 0
      Y0 = 10.9
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE AND
C SET NEW TICMARK INTERVAL.  DEFINE NEW VIRTUAL WINDOW.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('TICINTERVAL',0.25)
      CALL UWINDO (-1.1,1.1,-1.1,1.1)
C
C DRAW FIGURES IN 3 ROWS OF 4.
C
      DO 1 I = 1, 3
      X0 = -2.6
      Y0 = Y0 - 3.4
      DO 1 J = 1, 4
      X0 = X0 + 3.3
      INDEX = INDEX + 1
C
C DEFINE NEW DEVICE AREA AND OUTLINE IT.
C
      CALL UDAREA (X0,(X0+3.0),Y0,(Y0+3.0))
      CALL UOUTLN
C
C DRAW THE POLYGON USING 1 OF THE 12 POSSIBLE 12 OPTIONS
C WITHIN THE DEFINED 'UDAREA'.
C
      CALL USET (OPTION(INDEX))
      CALL UPLYGN (0.0,0.0,3.0,1.0)
C
C LABEL THE POLYGON
C
      CALL UPRINT (-1.0,-1.0,OPTION(INDEX))
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

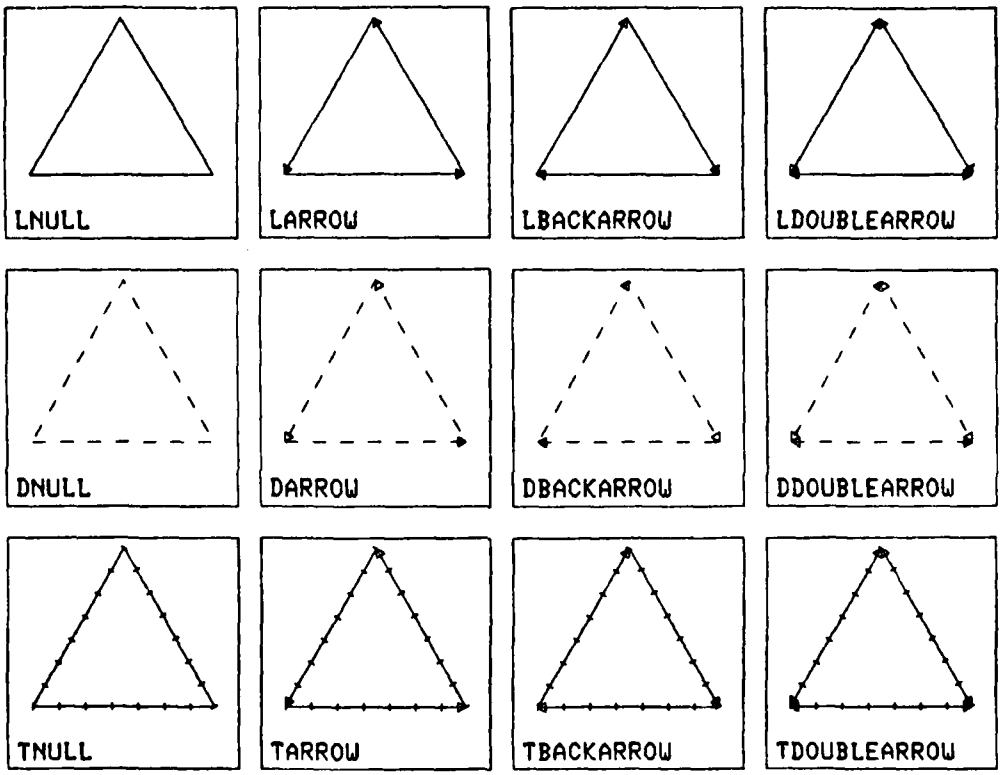


Figure 51. Example 7.4

E X A M P L E 7.5

```
C
C SUBROUTINE USED TO DEMONSTRATE 'UPLYGN'.  A TRIANGLE
C WILL BE DRAWN IN RELATIVE MODE AND ROTATED ABOUT ITS
C CENTER IN TEN DEGREE INCREMENTS.  THIS PROGRAM WAS
C WRITTEN FOR A TEKTRONIX 4014/4015 TERMINAL.
C
C INITIALIZE VARIABLES.
C
      DEGREE = 0.0
      Y0 = 10.9
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE AND
C DEFINE A NEW WINDOW.  DEFINE TWO 'UPEN' OPTIONS.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UWINDO (-1.1,1.1,-1.1,1.1)
      CALL USET ('INTEGER')
      CALL USET ('LARROW')
C
C DRAW FIGURE IN 3 ROWS OF 4.
C
      DO 1 I = 1, 3
      X0 = -2.6
      Y0 = Y0 - 3.4
      DO 1 J = 1, 4
      X0 = X0 + 3.3
      DEGREE = DEGREE + 10.0
C
C DEFINE NEW DEVICE AREA IN WHICH POLYGON WILL BE DRAWN.
C OUTLINE THE AREA.
C
      CALL UDAREA (X0,(X0+3.0),Y0,(Y0+3.0))
      CALL UOUTLN
C
C DRAW THE POLYGON WITHIN THE 'UDAREA', ROTATING IT IN
C RELATIVE MODE BY TEN DEGREE INCREMENTS.
C
      CALL UMOVE (0.0,0.0)
      CALL USET ('RELATIVE PLOTTING MODE')
      CALL UPSET ('ROTATE',DEGREE)
      CALL UPLYGN (0.0,0.0,3.0,1.0)
C
C RESET TO ABSOLUTE MODE AND LABEL THE POLYGON.
C
      CALL USET ('ABSOLUTE PLOTTING MODE')
      CALL UPRINT (-1.0,-1.0,DEGREE)
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

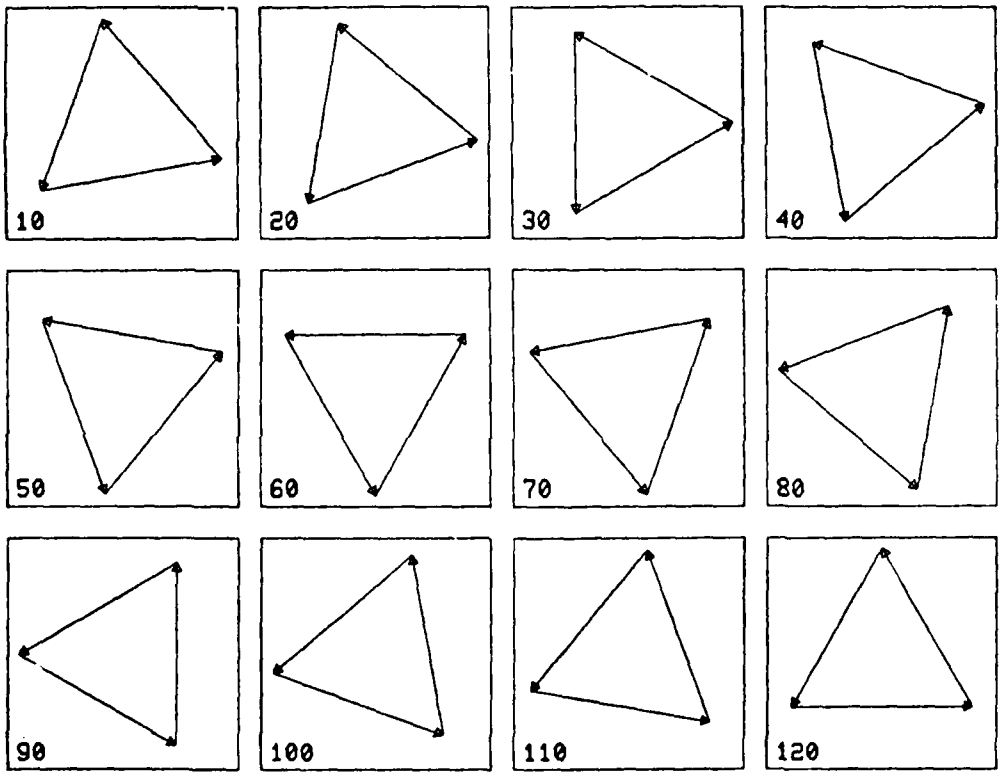


Figure 52. Example 7.5

E X A M P L E 7.6

```
C
C SAMPLE PROGRAM USED TO DEMONSTRATE 'URECT'.  A
C RECTANGLE WILL BE DRAWN IN RELATIVE MODE AND
C ROTATED IN 30 DEGREE INCREMENTS.
C
C INITALIZE VARIABLES
C
      DEGREE = 0.0
      Y0 = 10.9
C
C INITIALIZE GCS, SET THE CHARACTER SIZE TO EXTRALARGE AND
C DEFINE A NEW WINDOW.  SET 'UPRINT' OPTION TO INTEGER.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UWINDO (-1.1,1.1,-1.1,1.1)
      CALL USET ('INTEGER')
C
C DRAW THE RECTANGLE IN 3 ROWS OF 4 IN 30 DEGREE INCREMENTS.
C
      DO 1 I = 1, 3
      X0 = -2.6
      Y0 = Y0 - 3.4
      DO 1 J = 1, 4
      X0 = X0 + 3.3
      DEGREE = DEGREE + 30.0
C
C DEFINE DEVICE AREA WITHIN WHICH RECTANGLE WILL BE DRAWN.
C OUTLINE THE AREA.
C
      CALL UDAREA (X0,(X0+3.0),Y0,(Y0+3.0))
      CALL UOUTLN
C
C DRAW THE RECTANGLE WITHIN THE 'UDAREA', ROTATING IT IN
C RELATIVE MODE BY 30 DEGREE INCREMENTS.
C
      CALL UMOVE (0.0,0.0)
      CALL USET ('RELATIVE PLOTTING MODE')
      CALL UPSET ('ROTATE',DEGREE)
      CALL URECT (0.8,0.6)
C
C RESET TO ABSOLUTE MODE AND LABEL THE RECTANGLE.
C
      CALL USET ('ABSOLUTE PLOTTING MODE')
      CALL UPRINT (-1.0,-1.0,DEGREE)
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

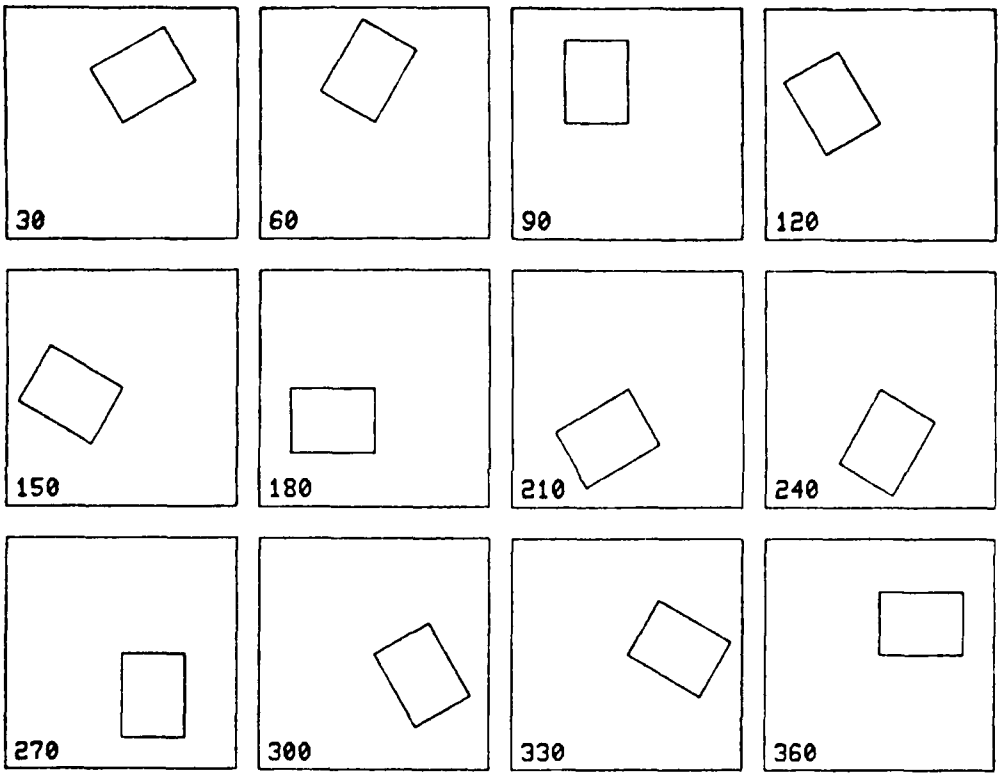


Figure 53. Example 7.6

E X A M P L E 7.7

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE APPLICATION OF 'UCONIC'
C TO GENERATE ELLIPSES AND HYPERBOLAE. FOUR FIGURES
C WILL BE DRAWN: 2 ELLIPSES ORIENTED ALONG THE X AND Y AXES;
C AND 2 HYPERBOLAE ORIENTED ALONG THE X AND Y AXES.  EACH
C ONE OF THE FIGURES IS DRAWN WITHIN ITS OWN REGION OF THE
C SCREEN BY REDEFINING THE DEVICE AREA PRIOR TO DRAWING THE
C FIGURE. THIS PROGRAM WAS WRITTEN FOR A TEKTRONIX 4014/4015.
C
C SET UP DATA ARRAYS FOR UCONIC
C
      DIMENSION X(4),Y(4),P(4),E(4)
C
      INDEX=0
      Y0=10.9
      X(1)=10.
      X(2)=50.
      X(3)=67.
      X(4)=50.
      Y(1)=50.
      Y(2)=10.
      Y(3)=50.
      Y(4)=67.
      P(1)=9.5
      P(2)=-9.5
      P(3)=9.
      P(4)=-9.
      E(1)=.9
      E(2)=-.9
      E(3)=1.44
      E(4)=-1.44
C
C INITIALIZE GCS
C
      CALL USTART
C
C LOOP TO DEFINE FOUR DEVICE AREAS AND OUTLINE EACH ONE.
C
      DO 1 I = 1, 4
      IF(I.EQ.1) CALL UDAREA(1.9,6.9,5.7,10.7)
      IF(I.EQ.2) CALL UDAREA(7.4,12.4,5.7,10.7)
      IF(I.EQ.3) CALL UDAREA(1.9,6.9,.2,5.2)
      IF(I.EQ.4) CALL UDAREA(7.4,12.4,.2,5.2)
      INDEX=INDEX+1
      CALL UOUTLN
C
C DRAW THE CONIC. NOTE THAT THE DEFAULT WINDOW IS MAPPED
C TO THE CURRENT DEVICE AREA SPECIFICATION.
C
      CALL UCONIC (X(INDEX),Y(INDEX),P(INDEX),E(INDEX),0.0,360.0)
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

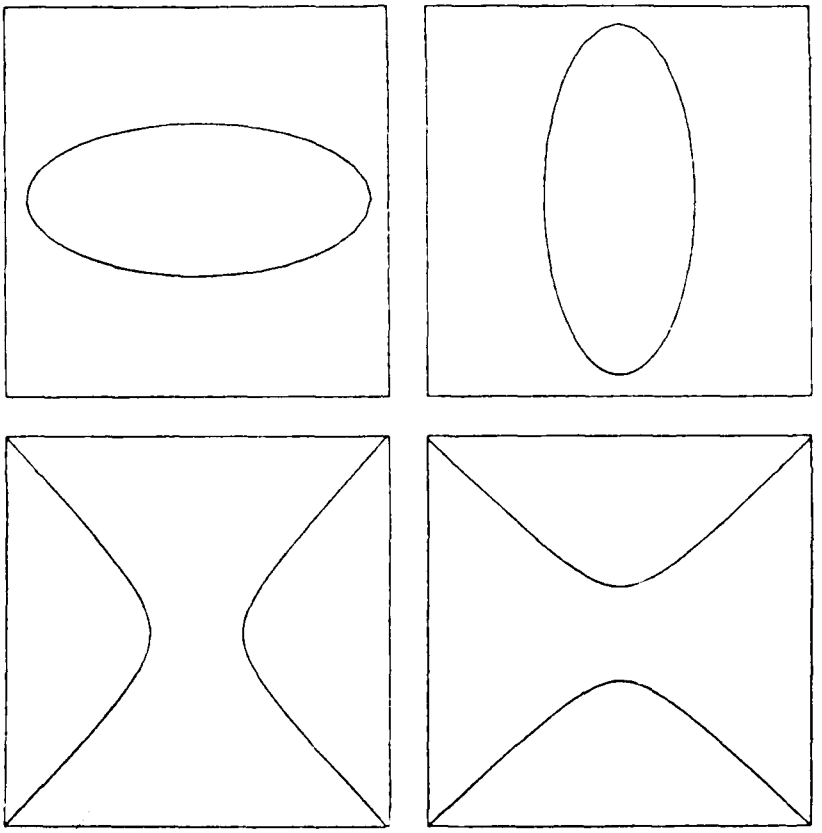


Figure 54. Example 7.7

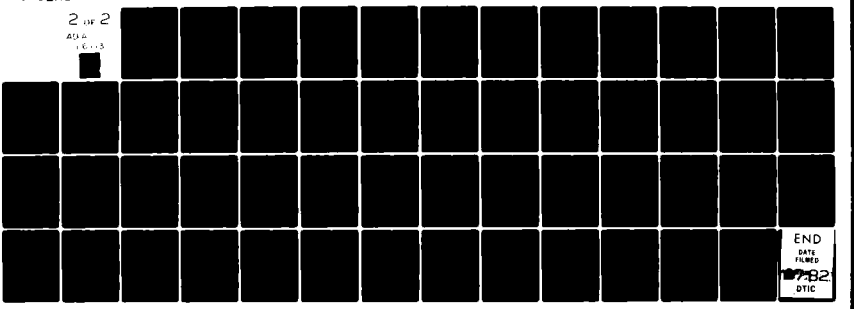
E X A M P L E 7.8

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE APPLICATION OF 'UCONIC'
C TO GENERATE PARABOLAE.  FOUR FIGURES WILL BE DRAWN: 2 WILL
C BE ORIENTED ALONG THE +X AND +Y AXES; AND 2 WILL BE ORIENTED
C ALONG THE -X AND -Y AXES.  EACH ONE OF THE PARABOLAE IS
C DRAWN WITHIN ITS OWN REGION OF THE SCREEN BY REDEFINING THE
C DEVICE AREA PRIOR TO DRAWING THE FIGURE.
C
C SET UP ARRAYS FOR UCONIC
C
      DIMENSION X(4),Y(4),P(4),E(4)
      INDEX=0
      Y0=10.9
      X(1)=10.
      X(2)=50.
      X(3)=90.
      X(4)=50.
      Y(1)=50.
      Y(2)=10.
      Y(3)=50.
      Y(4)=90.
      P(1)=13.
      P(2)=-13.
      P(3)=-13.
      P(4)=13.
      E(1)=1.
      E(2)=-1.
      E(3)=1.
      E(4)=-1.
C
C
C INITIALIZE GCS
C
      CALL USTART
C
C LOOP TO DEFINE FOUR DEVICE AREA AND OUTLINE EACH ONE.
C
      DO 1 I = 1, 4
      IF (I.EQ.1) CALL UDAREA (1.9,6.9,5.7,10.7)
      IF (I.EQ.2) CALL UDAREA (7.4,12.4,5.7,10.7)
      IF (I.EQ.3) CALL UDAREA (1.9,6.9,.2,5.2)
      IF (I.EQ.4) CALL UDAREA (7.4,12.4,.2,5.2)
      INDEX = INDEX + 1
      CALL UOUTLN
C
C DRAW THE CONIC.  NOTE THAT THE DEFAULT WINDOW IS MAPPED
C TO THE CURRENT DEVICE AREA SPECIFICATION.
C
      CALL UCONIC (X(INDEX),Y(INDEX),P(INDEX),E(INDEX),0.0,360.0)
     1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

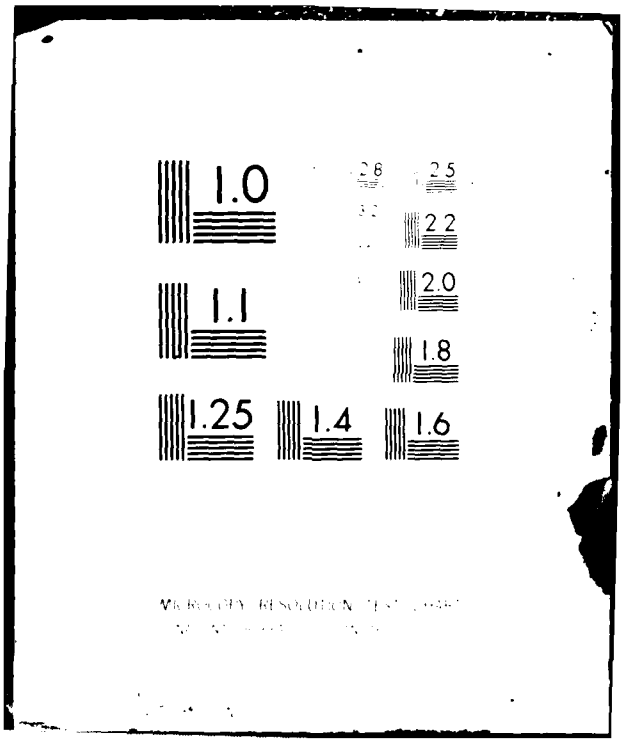
AD-A116 113 ARMY ENGINEER WATERWAYS EXPERIMENT STATION VICKSBURG MS F/G 9/2
NOTEBOOK FOR LESSONS ON THE GRAPHICS COMPATIBILITY SYSTEM (GCS)--ETC(U)
APR 82 D WARD, J M JONES, M E GEORGE

UNCLASSIFIED WES-INSTRUCTION-K-82-4 NL

2 OF 2



END
DATE FILMED
7-82
DTIC


1.0
1.1
1.25
1.4
1.6
2.0
1.8
2.2
2.5

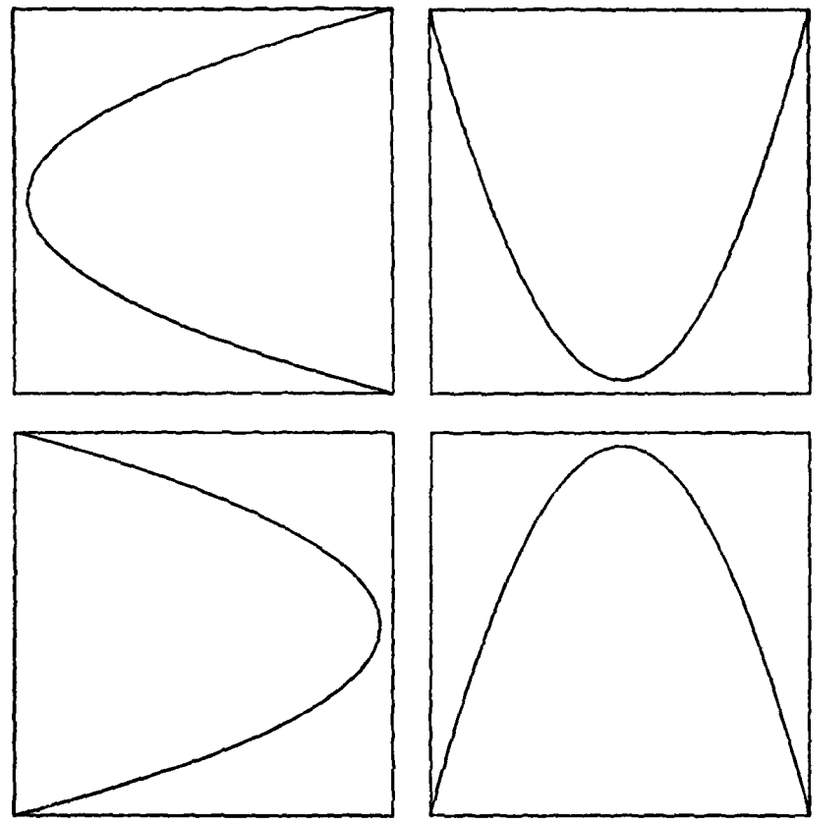


Figure 55. Example 7.8

E X A M P L E 7.9

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE APPLICATION OF 'ULINFT' TO
C CALCULATE THE SLOPE (S) AND Y-INTERCEPT (YI) OF A LINE WHICH
C REPRESENTS THE 'BEST' LINEAR FIT TO A SERIES OF DATA POINTS.
C SET UP DATA ARRAYS FOR ULINFT
C
      DIMENSION X(20),Y(20)
      DATA X/2.,5.,10.,15.,20.,25.,30.,35.,40.,45.,50.,55.,60.,65.,70.,75.,
     &       80.,85.,90.,95./
      DATA Y/2.,7.,10.,14.,25.,24.,31.,34.,40.,46.,50.,59.,58.,64.,70.,76.,
     &       82.,86.,91.,97./
C
C INITIALIZE GCS AND OUTLINE THE DEFAULT DEVICE AREA.
C PLOT THE DATA POINTS, CENTERED, WITH PLUS (+) SIGNS.
C NOTE THAT THE DATA LIES WITHIN THE DEFAULT WINDOW BOUNDARIES.
C
      CALL USTART
      CALL UOUTLN
      CALL USET ('ACENTER CHARACTERS')
      CALL USET ('N+')
      CALL ULINE (X,Y,20.)
C
C CALL 'ULINFT' TO CALCULATE THE LINE'S SLOPE AND Y-INTERCEPT.
C
      CALL ULINFT (X,Y,20.,S,YI)
C
C RESET THE PEN MODE TO DRAW SOLID LINES, MOVE TO THE
C Y-INTERCEPT AND GRAPH THE LINE USING YO=S*X+YI.
C
      CALL USET ('LINE')
      XMIN = 0.0
      XMAX = 100.0
      CALL UMOVE (XMIN,YI)
      YO = YI + S * XMAX
      CALL UPEN (XMAX,YO)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

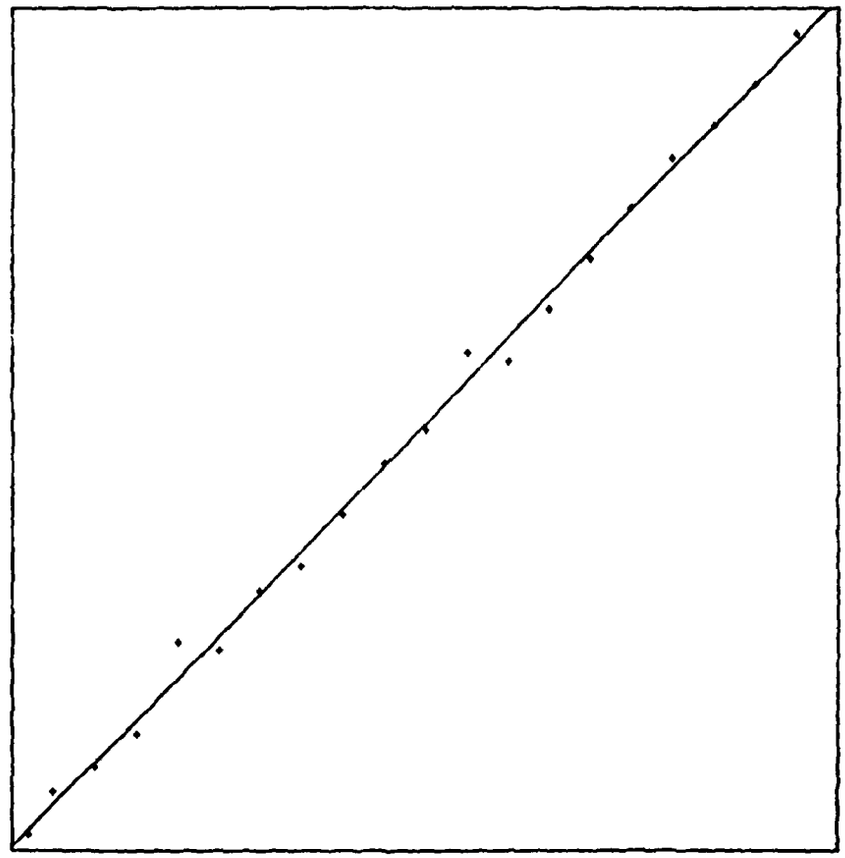


Figure 56. Example 7.9

E X A M P L E 7.10

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE APPLICATION OF 'ULSTSQ' TO
C CALCULATE THE COEFFICIENTS OF A POLYNOMIAL OF ORDER 6
C WHICH REPRESENTS THE 'BEST' FIT TO A SERIES OF DATA POINTS.
C
C SET UP DATA ARRAYS FOR ULSTSQ
C
      DIMENSION A(7),X(20),Y(20)
      DATA IDEGRE/7/
      DATA X/0.,5.,10.,15.,20.,25.,30.,35.,40.,45.,50.,55.,60.,65.,70.,75.,
     &       80.,85.,90.,95./
      DATA Y/0.,8.,20.,32.,46.,55.,60.,60.,56.,48.,40.,37.,35.,37.,42.,49.,
     &       58.,68.,82.,95./
C
C INITIALIZE GCS AND OUTLINE THE DEFAULT DEVICE AREA.
C
      CALL USTART
      CALL UOUTLN
C
C PLOT DATA POINTS, CENTERED, WITH PLUS (+) SIGNS.
C
      CALL USET ('N+')
      CALL USET ('ACENTER CHARACTERS')
      CALL ULINE (X,Y,20.)
C
C MOVE PEN TO ORIGIN AND COMPUTE LEAST SQUARES LINE WITH 6TH
C ORDER FIT.
C
      CALL UMOVE (0.0,0.0)
      CALL UPSET ('POLYNOMIAL DEGREE',FLOAT(IDEGRE-1))
      CALL ULSTSQ (X,Y,20.,A)
      CALL USET ('LINE')
C
C PLOT LEAST SQUARES LINE APPROXIMATING POINTS
C
      DO 5 I = 1, 101
      YO = A(1)
      XO = FLOAT(I-1)
      XK = XO
      DO 4 J = 2, IDEGRE
      YO = A(J) * XK + YO
      XK = XK * XO
    4 CONTINUE
      CALL UPEN (XO,YO)
    5 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

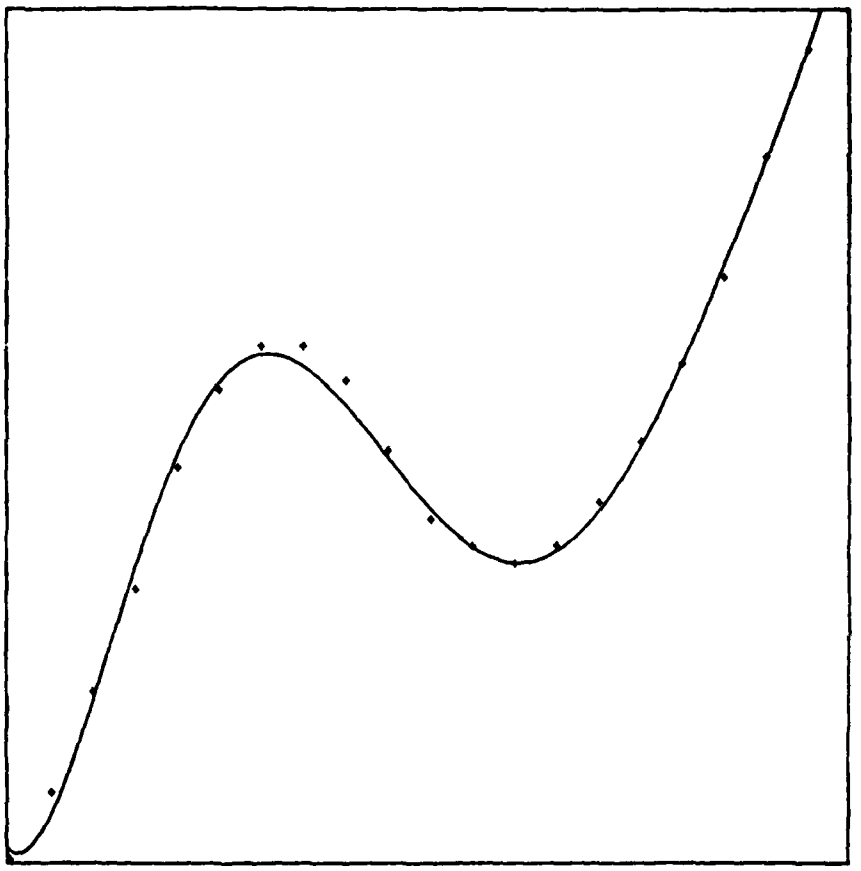


Figure 57. Example 7.10

E X A M P L E 8.1

```
C
C SAMPLE PROGRAM TO ILLUSTRATE OPTIONS AVAILABLE
C WITH 'UAXIS'.
C
C INITIALIZE GCS AND GENERATE AXES
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL USET ('XBOTHLABELS')
      CALL UPSET ('XLABEL','EDGEAXIS\')
      CALL UAXIS (-5.32,4.89,-63.4,156.459)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

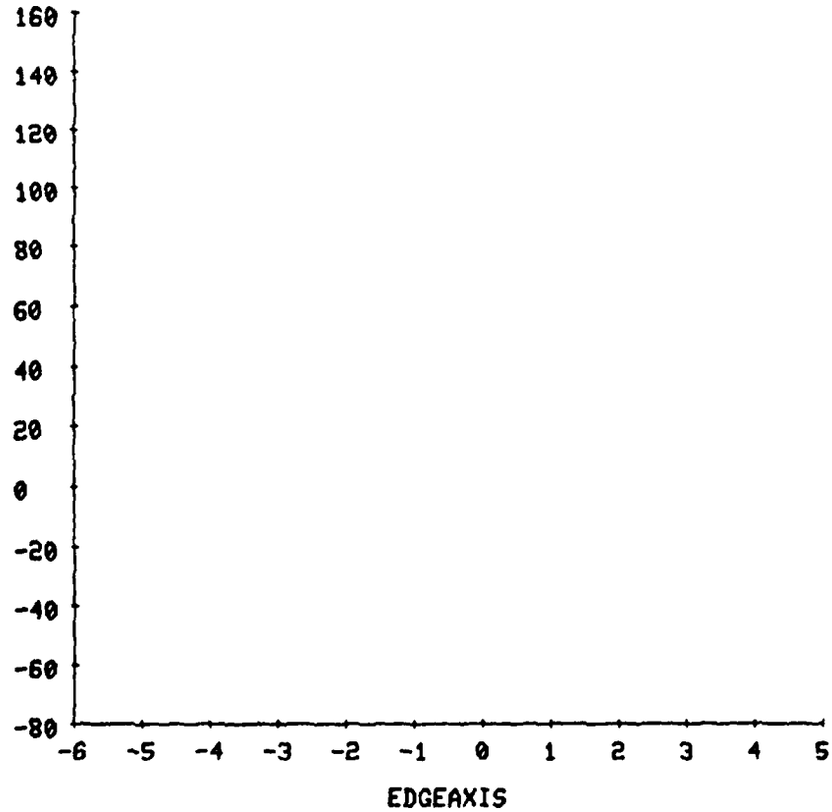


Figure 58. Example 8.1

E X A M P L E 8.2

```
C
C SAMPLE PROGRAM TO ILLUSTRATE OPTIONS AVAILABLE
C WITH 'UAXIS'.
C
C INITIALIZE GCS AND GENERATE AXES
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL USET ('ZEROAXES')
      CALL USET ('XBOTHLABELS')
      CALL UPSET ('XLABEL','ZEROAXES\')
      CALL UAXIS (-6.,2.,-5.,4.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

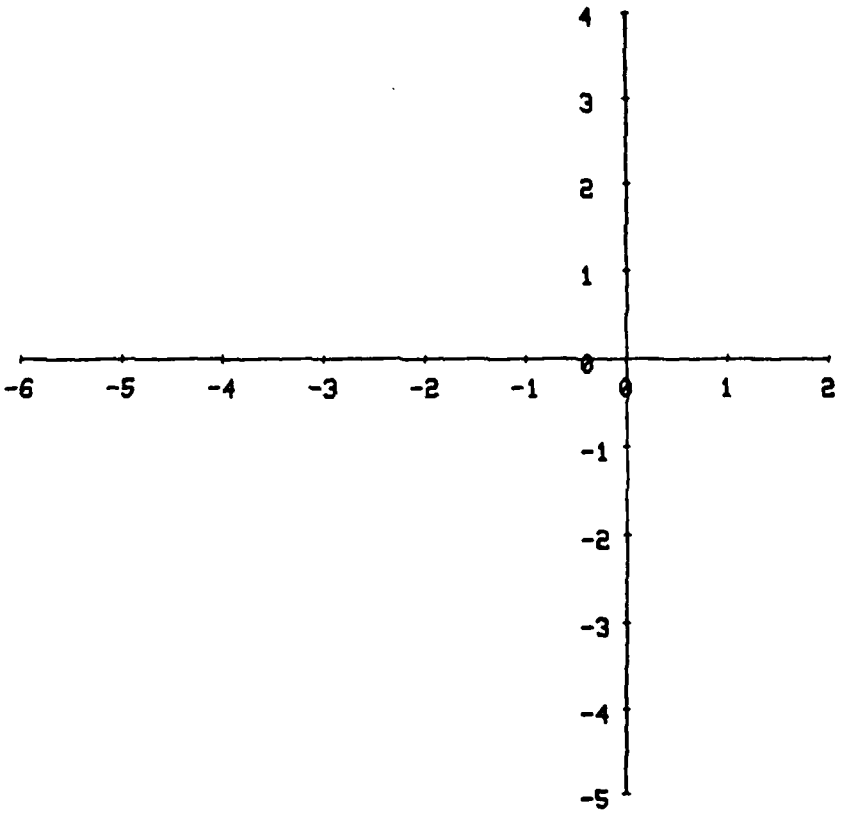


Figure 59. Example 8.2

E X A M P L E 8.3

```
C
C SAMPLE PROGRAM TO ILLUSTRATE OPTIONS AVAILABLE
C WITH 'UAXIS'.
C
C INITIALIZE GKS AND GENERATE AXES
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL USET ('PENAXIS')
      CALL USET ('XBOTHLABELS')
      CALL UPSET ('XLABEL','PENAXES\')
      CALL UPEN (-3.,2.)
      CALL UAXIS (-6.,2.,-5.,4.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

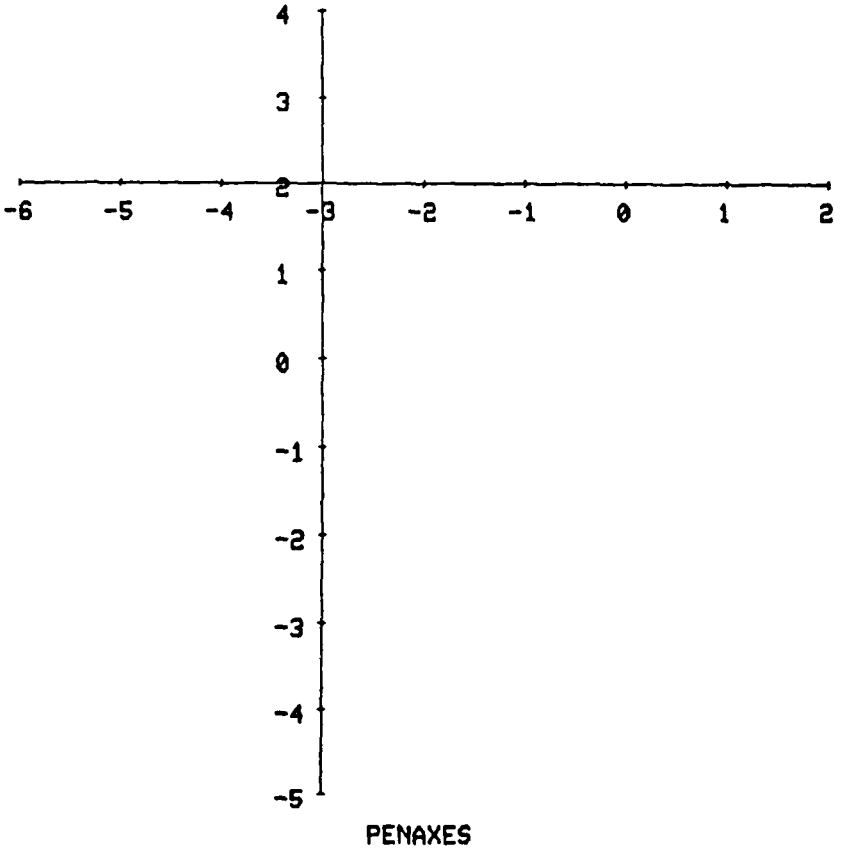


Figure 60. Example 8.3

E X A M P L E 8.4

```
C
C SAMPLE PROGRAM TO ILLUSTRATE OPTIONS AVAILABLE
C WITH 'UAXIS'.
C
C INITIALIZE GCS AND GENERATE AXES
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('XLABEL','POLAR GRID\')
      CALL USET ('POLARAXES')
      CALL USET ('GRID')
      CALL USET ('XBOTHLABELS')
      CALL UAXIS (-6.,6.,0.,360.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

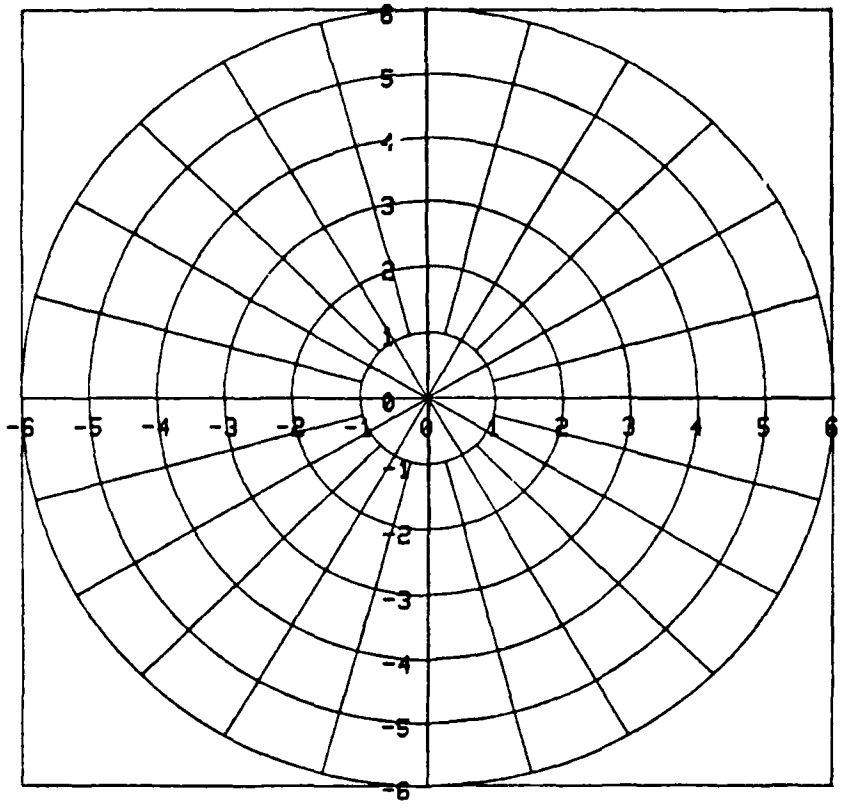


Figure 61. Example 8.4

E X A M P L E 8.5

```
C
C SAMPLE PROGRAM TO ILLUSTRATE THE USE OF 'UPLOT'.
C TWO CURVES WILL BE PLOTTED ON A SINGLE PAIR OF AXES.
C THIS PROGRAM WAS WRITTEN FOR A TEKTRONIX 4014/4015.
C
C SET UP DATA ARRAYS FOR 'UPLOT'.  NOTE THE SEMICOLON (;)
C AT THE END OF THE LABELS.
C
      DIMENSION X(12),Y(12),PARRAY(2)
      CHARACTER OPTARY*4(2)
      CHARACTER XLABEL*40,YLABEL*40
      DATA X/1.,2.,3.,4.,5.,1.,2.,2.5,3.,3.5,6.,7./
      DATA Y/1.,4.,9.,16.,25.,.5,1.,1.2,1.5,1.7,3.,3.5/
      DATA PARRAY/5.,7./
      DATA XLABEL/'THIS IS THE X LABEL;'/
      DATA YLABEL/'THIS IS THE Y LABEL;'/
      DATA OPTARY/'LINE','DASH'/
C
C INITIALIZE GCS, SET THE CHARACTER SIZE TO EXTRALARGE AND
C CHANGE THE TEXT STRING TERMINATOR TO A SEMICOLON (;).
C INITIALIZE AXES OPTIONS.  REMEMBER THAT 'UPLOT' CALLS
C 'UAXIS'.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('TERMINATOR CHARACTER',';')
      CALL UPSET ('XLABEL',XLABEL)
      CALL UPSET ('YLABEL',YLABEL)
      CALL USET ('GRIDAXES')
      CALL USET ('XBOTHLABEL')
      CALL USET ('YBOTHLABEL')
C
C X-AXIS WILL BE LINEAR (DEFAULT) AND Y-AXIS WILL BE
C LOGARITHMIC.
C
      CALL USET ('LOGYAXIS')
C
C DEFINE NEW DEVICE AREA AND OUTLINE IT.
C
      CALL UDAREA (4.,14.,0.,10.)
      CALL UOUTLN
C
C PLOT THE TWO CURVES
C
      CALL UPLOT (X,Y,2.,PARRAY,OPTARY)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

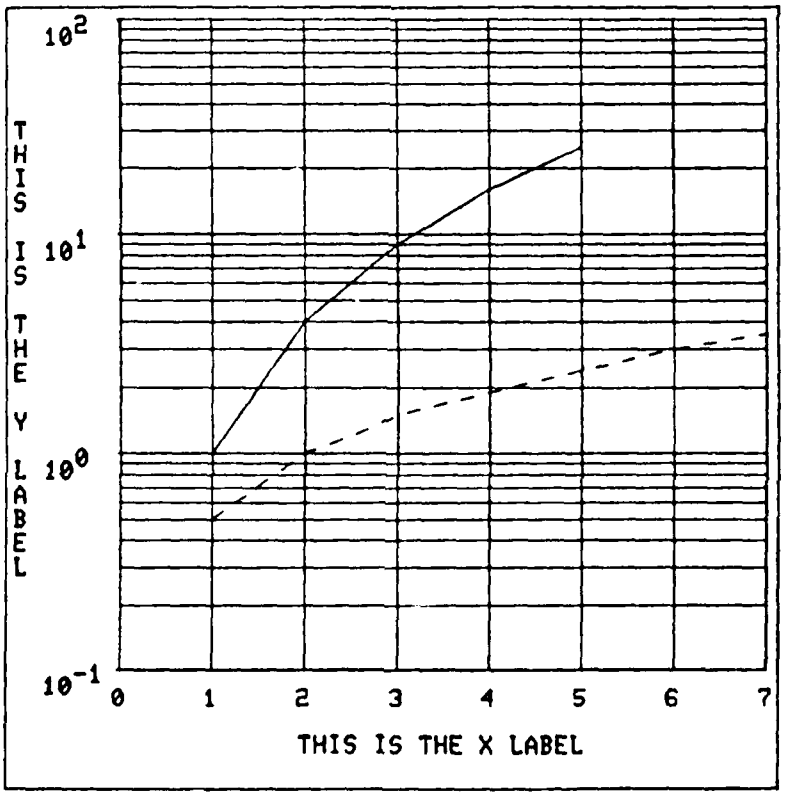


Figure 62. Example 8.5

E X A M P L E 8.6

```
C
C THIS PROGRAM ILLUSTRATES THE USE OF 'UPLOT1' TO PLOT
C A SINGLE CURVE THAT REPRESENTS A 4TH ORDER POLYNOMIAL FIT.
C THIS PROGRAM WAS WRITTEN FOR A TEKTRONIX 4014/4015.
C
C SET UP DATA ARRAYS FOR UPLOT1.
C
      DIMENSION DB(61),E2(61)
      CHARACTER XLABEL*40,YLABEL*40
      DATA XLABEL/'STEADY-STATE EXCITATION VOLTAGE (VOLTS)\'/
      DATA YLABEL/'EXCITATION REFERENCED TO 1V (DB)\'/
C
C GENERATE SOME DATA VALUES
C
      DO 1 I = 1, 61
      DB(I) = FLOAT(I-1)
    1 E2(I) = 10.0**(DB(I) / 20.0)
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE AND
C INITIALIZE 'UAXIS' OPTIONS.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('XLABEL',XLABEL)
      CALL UPSET ('YLABEL',YLABEL)
      CALL USET ('GRIDAXES')
      CALL USET ('XBOTHLABEL')
      CALL USET ('YBOTHLABEL')
C
C INITIALIZE 'UPLOT1' OPTIONS FOR A POLYNOMIAL FIT
C
      CALL USET ('FITPOLYNOMIAL')
      CALL UPSET ('POLYNOMIAL DEGREE',4.)
C
C DEFINE A NEW DEVICE AREA AND OUTLINE IT
C
      CALL UDAREA (4.,14.3,0.,10.9)
      CALL UOUTLN
C
C PLOT THE CURVE.  NOTE THAT THE Y-LABEL IS CLIPPED.
C
      CALL UPLOT1 (E2,DB,61.0)
C
C CONNECT THE ORIGINAL DATA POINTS WITH A 'HARDWARE' GENERATED
C DOTTED LINE.
C
      CALL USET ('DASHLINE')
      CALL UPSET ('SETDASH',9.)
      CALL ULINE (E2,DB,61.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

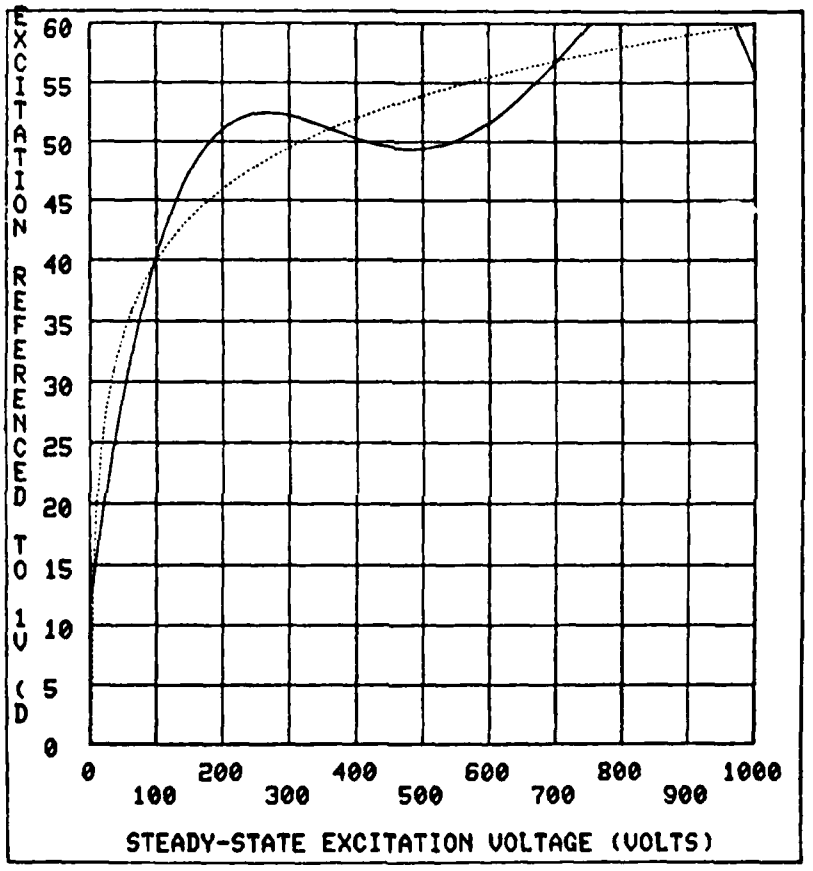


Figure 63. Example 8.6

E X A M P L E 8.7

```
C
C SAMPLE PROGRAM TO ILLUSTRATE APPLICATION OF 'UHISTO'.
C
C ALLOCATE VARIABLES FOR UHISTO
C
      DIMENSION DATA(1000)
      DATA XN /1000./
C
C BUILD DATA FROM SINE VALUES.
C
      DO 2 I = 1, 1000
    2 DATA(I) = SIN(FLOAT(I)/150.)
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE
C AND CHANGE TEXT STRING TERMINATOR TO A SEMICOLON (;).
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL UPSET ('TERMINATOR CHARACTER',';')
C
C INDICATE THAT BOTH NUMERIC AND ALPHABETIC LABELS ARE
C DESIRED FOR THE X-AXIS AND SET ALPHABETIC LABEL.
C
      CALL USET ('XBOTHLABELS')
      CALL UPSET ('XLABEL','DISTRIBUTION OF VALUES OF SINE;')
C
C CHANGE DEVICE UNITS TO PERCENTUNITS AND CHANGE
C DEVICE AREA TO ENTIRE SCREEN.
C
      CALL USET ('PERCENTUNITS')
      CALL UDAREA (0.,100.,0.,100.)
C
C INDICATE THAT YOU WILL PROVIDE YOUR OWN SCALE AND
C SET THE LIMITS WITH 'UWINDO'.
C
      CALL USET ('OWNSCALE')
      CALL UWINDO (0.,150.,-1.,1.)
C
C PROCESS THE DATA ARRAY USING 20 CELLS.
C
      CALL UHISTO (DATA,XN,20.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

.9	TO	1.	136
.8	TO	.9	57
.7	TO	.8	45
.6	TO	.7	40
.5	TO	.6	36
.4	TO	.5	34
.3	TO	.4	44
.2	TO	.3	47
.1	TO	.2	45
-.1863E-8	TO	.1	45
-.1	TO	-.1863E-8	30
-.2	TO	-.1	30
-.3	TO	-.2	31
-.4	TO	-.3	32
-.5	TO	-.4	34
-.6	TO	-.5	36
-.7	TO	-.6	39
-.8	TO	-.7	46
-.9	TO	-.8	58
-1.	TO	-.9	135

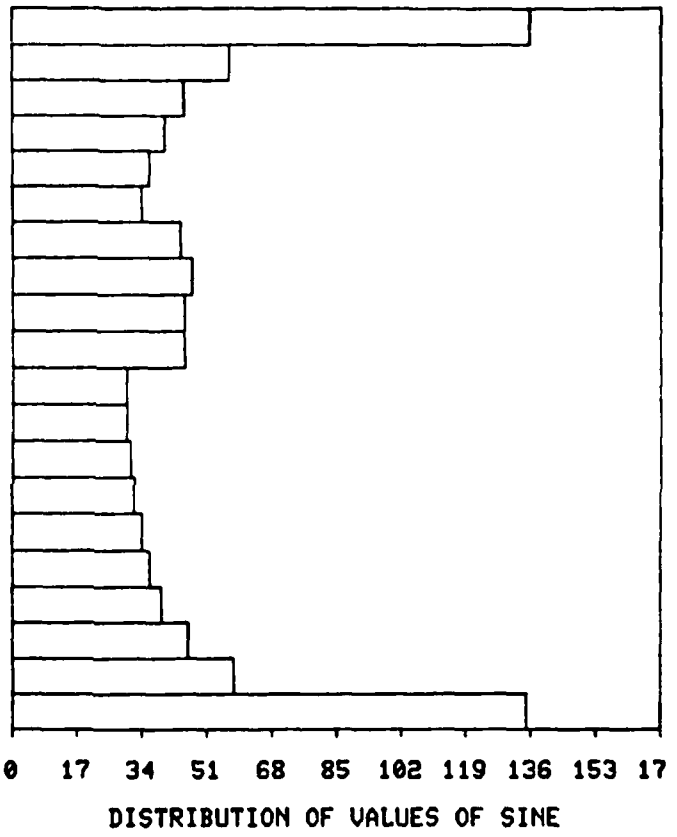


Figure 64. Example 8.7

E X A M P L E 8.8

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE 'UBAR'.
C
C ALLOCATE ARRAYS FOR UBAR
C
      DIMENSION DATA(7)
      CHARACTER LABELS*12(7)
      DATA DATA/4.,30.,48.,10.,2.,2.,4./
      DATA LABELS/'ALGOL\','COBOL\','FORTRAN\','GMAP\','JOVIAL\',
     &            'SIMSCRIPT\','SNOBOL\'/
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE AND
C INDICATE THAT BOTH X-AXIS LABELS ARE TO BE USED.
C
      CALL USTART
      CALL USET ('EXTRALARGE CHARACTERS')
      CALL USET ('XBOTHLABELS')
      CALL UPSET ('XLABEL','TYPICAL LANGUAGE UTILIZATION AT USMA\')
C
C GENERATE THE BARCHART WITH THE SPECIFIED DATA VALUES
C AND LABELS.
C
      CALL UBAR (DATA,7.,LABELS,12.)
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

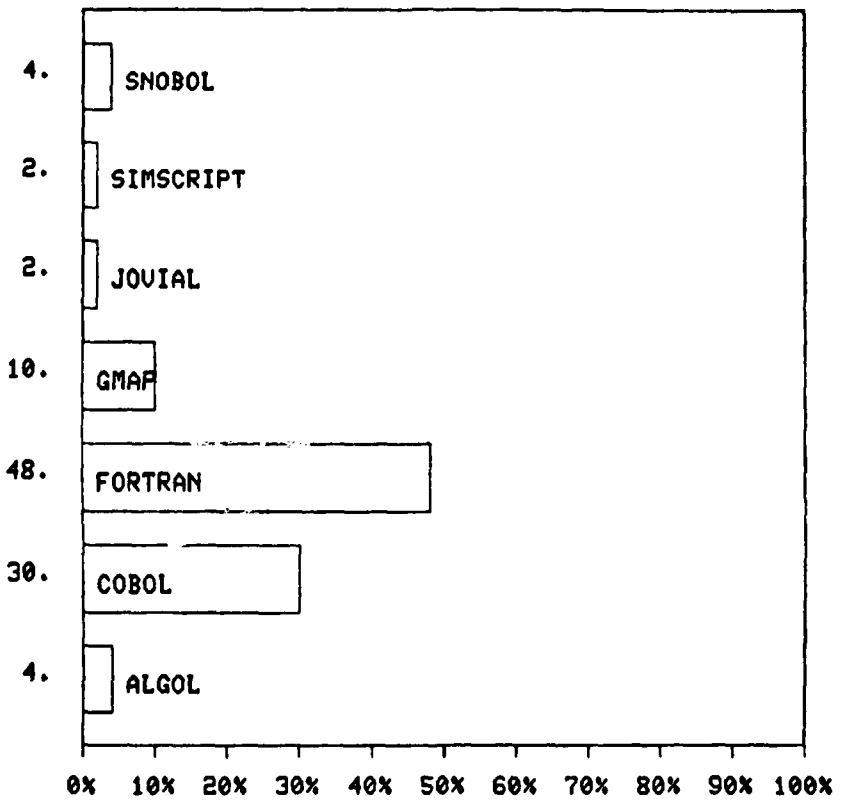


Figure 65. Example 8.8

E X A M P L E 8.9

```
C
C SAMPLE PROGRAM USED TO ILLUSTRATE 'UPIE'.
C IN ADDITION TO GENERATING A PIECHART, SOME
C ADDITIONAL GRAPHIC OUTPUT IS DONE BY 'UPRNT1'.
C
C ALLOCATE ARRAYS FOR UPIE.
C
      DIMENSION DATA(7)
      CHARACTER LABELS*12(7)
      DATA(1)=4.
      DATA(2)=30.
      DATA(3)=48.
      DATA(4)=10.
      DATA(5)=2.
      DATA(6)=2.
      DATA(7)=4.
      Y=100.
      LABELS(1)='ALGOL\'
      LABELS(2)='COBOL\'
      LABELS(3)='FORTRAN\'
      LABELS(4)='GMAP\'
      LABELS(5)='JOVIAL\'
      LABELS(6)='SIMSCRIPT\'
      LABELS(7)='SNOBOL\'
C
C INITIALIZE GCS, SET CHARACTER SIZE TO EXTRALARGE
C AND INDICATE THAT ONLY X-AXIS ALPHABETIC ARE NEEDED.
C
      CALL USTART
      CALL USET ('EXTRALARGE')
      CALL USET ('XALPHABETIC')
      CALL UPSET ('XLABEL','TYPICAL LANGUAGE UTILIZATION AT USMA\')
C
C GENERATE THE PIECHART
C
      CALL UPIE (DATA,7.,LABELS,12.)
C
C SET ADDRESSING MODE TO 'DEVICE/PERCENTUNITS' AND
C OUTPUT LABEL INFORMATION.  NOTE THAT IN 'DEVICE' MODE
C ALL (X,Y) COORDINATES ARE IN PERCENTUNITS.
C
      CALL USET ('DEVICE')
      CALL USET ('PERCENTUNITS')
      CALL UDAREA (0.,100.,0.,100.)
      DO 1 I = 1, 7
      Y = Y - (100. / FLOAT(7+1))
      CALL UMOVE (0.,Y)
      CALL UPRNT1 (LABELS(I),'TEXT')
      CALL UPRNT1 (' - \','TEXT')
      CALL UPRNT1 (DATA(I),'INTEGER')
      CALL UPRNT1 ('%\','TEXT')
    1 CONTINUE
C
C WRAP UP
C
      CALL UEND
      STOP
      END
```

ALGOL - 4%

COBOL - 30%

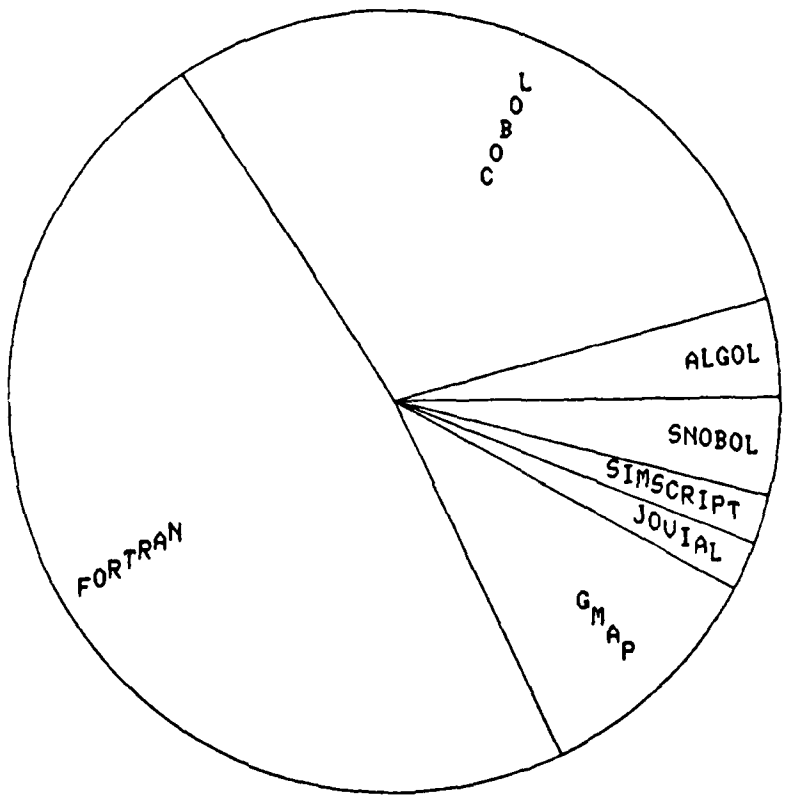
FORTRAN - 48%

GMAP - 10%

JOVIAL - 2%

SIMSCRIPT - 2%

SNOBOL - 4%



TYPICAL LANGUAGE UTILIZATION AT USMA

Figure 66. Example 8.9

E X A M P L E 9.1

```
C   PROBLEM SOLUTION II
      COMMON/PLT/ X(18),Y(18),PTS(3),OPT(3)
      CHARACTER FF2*21
      ENCODE(FF2,77)'GRAPHICS/LESSON/DATA;'
   77 FORMAT(A21)
      CALL ATTACH(11,FF2,1,0,ISTAT,)
      DO 100 I=1,18
      READ(11,110) X(I),Y(I)
  110 FORMAT(V)
  100 CONTINUE
      CALL INIT
      CALL TPLOT
      CALL CONTRL
      CALL UEND
      STOP
      END
      SUBROUTINE INIT
      COMMON/PLT/ X(18),Y(18),PTS(3),OPT(3)
      CHARACTER XLABEL*10,YLABEL*10
      DATA OPT/'LINE','DASH','N+'/
      DATA PTS/5.,7.,6./
      DATA XLABEL/'X VALUE\'/,YLABEL/'Y VALUE\'/
      CALL USTART
      CALL UPSET('XLABEL',XLABEL)
      CALL UPSET('YLABEL',YLABEL)
      CALL USET('GRIDAXES')
      CALL USET('XBOTHLABEL')
      CALL USET('YBOTHLABEL')
      CALL UEND
      STOP
      END
      SUBROUTINE TPLOT
      COMMON/PLT/ X(18),Y(18),PTS(3),OPT(3)
      CALL UDAREA(0.,10.,0.,10.)
      CALL UERASE
      CALL UPLOT(X,Y,3.,PTS,OPT)
      CALL UEND
      STOP
      END
      SUBROUTINE CONTRL
      CHARACTER CHAR*1
  100 CALL UAIN(CHAR)
      IF(CHAR.EQ.'T') GO TO 200
      IF(CHAR.EQ.'W') GO TO 300
      IF(CHAR.EQ.'R') GO TO 400
      CALL FAULT
      GO TO 100
  200 CALL USET('AUTOSCALE')
      CALL TPLOT
      GO TO 100
  300 CALL UGRIN(XL,YL,CHAR)
      CALL UGRIN(XU,YU,CHAR)
      CALL UWINDO(XL,XU,YL,YU)
      CALL USET('OWNSCALE')
      CALL TPLOT
      GO TO 100
      CALL UEND
  400 STOP
      END
      SUBROUTINE FAULT
      CALL UERASE
      CALL UHOME
      CALL UALPHA
      PRINT,'T - TOTAL'
      PRINT,'W - WINDOW'
      PRINT,'R - RETURN'
      CALL UEND
      STOP
      END
```

T - TOTAL
W - WINDOW
R - RETURN

Figure 67. Example 9.1

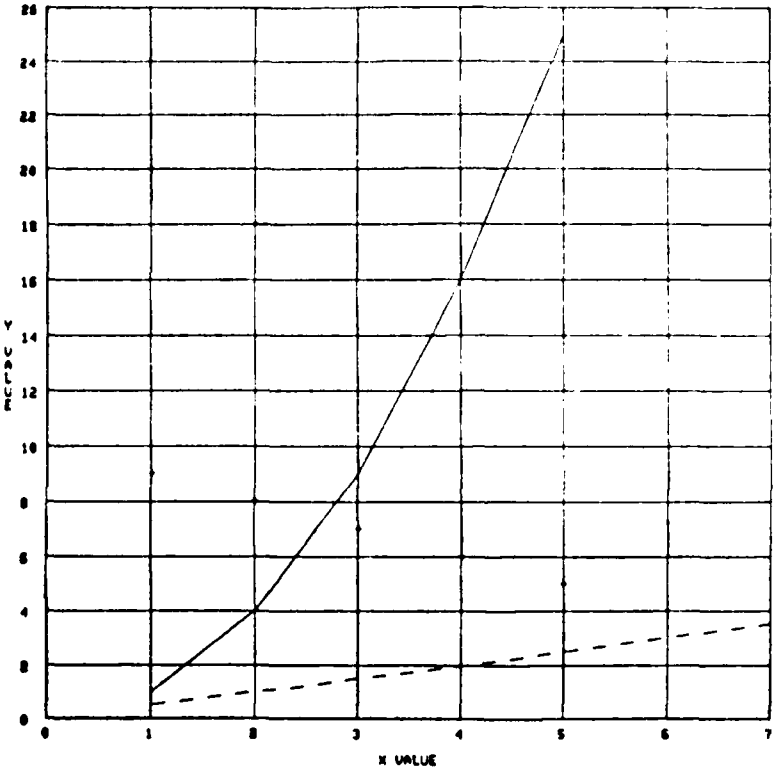


Figure 68. Example 9.1 (continued)

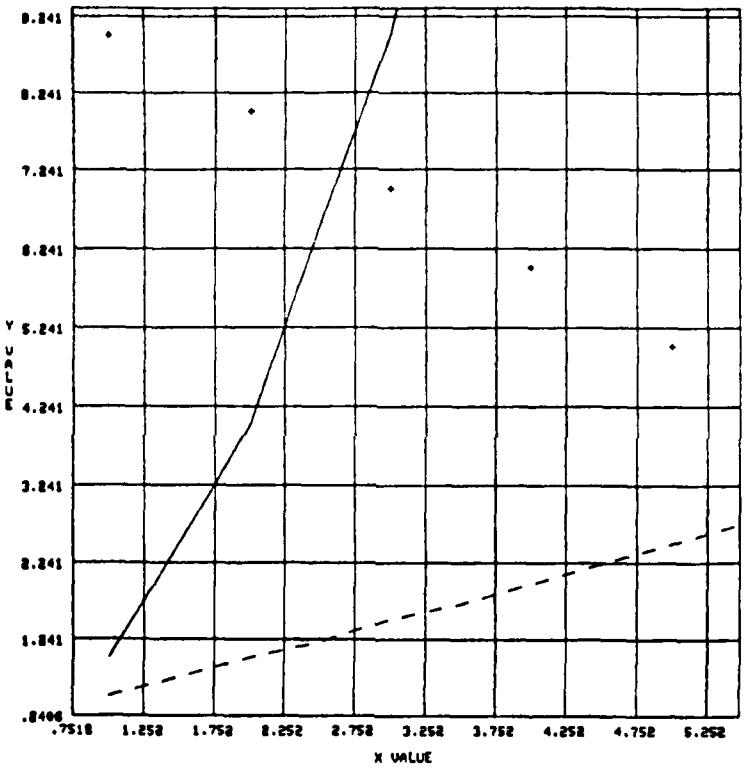


Figure 69. Example 9.1 (continued)

E X A M P L E 10.1

```
      CALL USTART
      CALL USET('WORKINGAXIS')
      CALL UWINDO (-10.,10.,-10.,10.)
      CALL UOUTLN
      CALL UMOVE (0.,0.)
      CALL UPEN (2.,0.)
      CALL UPEN (0.,2.)
      CALL UPEN (0.,0.)
      CALL UCOSYS (5.,4.,1.,1.,45.)
      CALL UMOVE (0.,0.)
      CALL URECT (3.,3.)
      CALL UEND
      STOP
      END
```

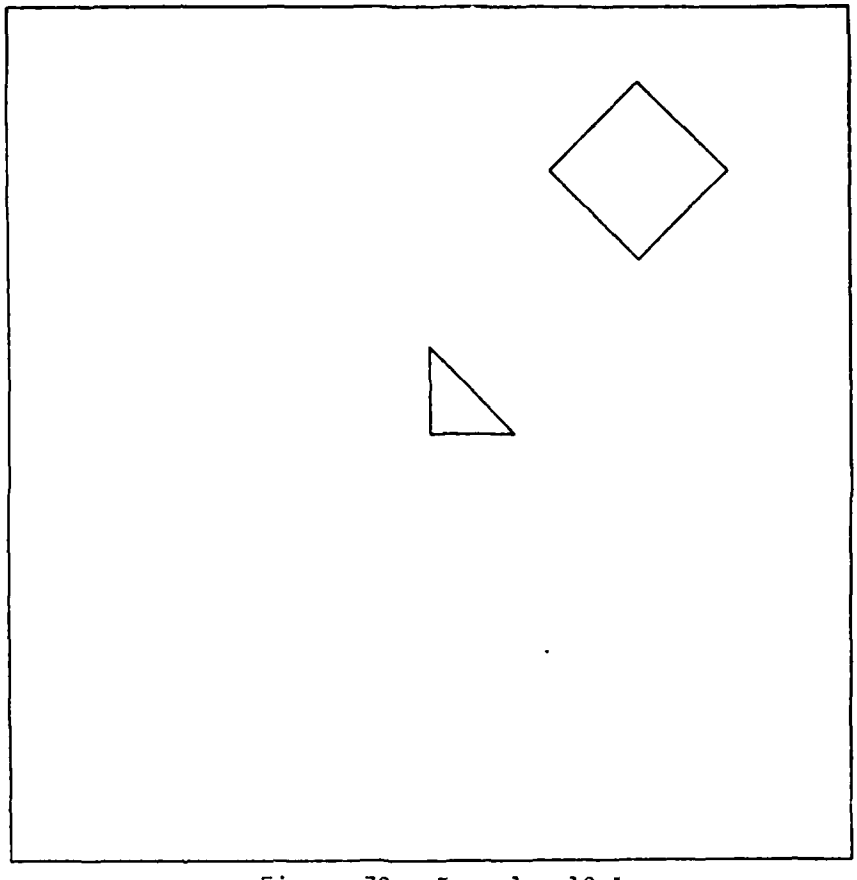


Figure 70. Example 10.1

E X A M P L E 10.2

```
      CALL USTART
      CALL USET ('WORKINGAXIS')
      CALL UWINDO (-2.,8.,-2.,8.)
      CALL UOUTLN
      CALL AXIS
      CALL UMOVE (0.,0.)
      CALL UCOSYS (2.,5.,1.,1.,20.)
      CALL UPEN1 (0.,0.,'DARROW')
      CALL AXIS
      CALL UCOSYS (5.,1.,1.,1.,45.)
      CALL USET ('SYSTEMAXIS')
      CALL UMOVE (0.,0.)
      CALL USET ('USERAXIS')
      CALL UPEN1 (0.,0.,'DARROW')
      CALL AXIS
      CALL UEND
      STOP
      END
      SUBROUTINE AXIS
      CALL UMOVE (-1.,0.)
      CALL UPEN1 (1.,0.,'LARROW')
      CALL UMOVE (0.,-1.)
      CALL UPEN (0.,1.)
      CALL UEND
      STOP
      END
```

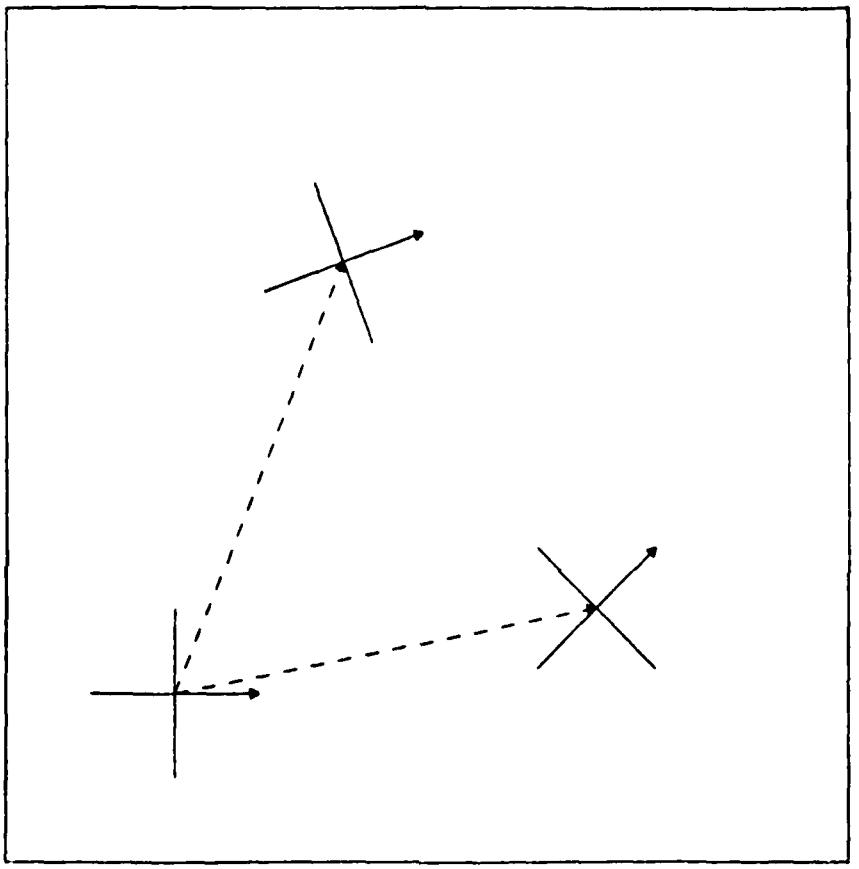


Figure 71. Example 10.2

E X A M P L E 10.3

```
      CALL USTART
      CALL UWINDO (-1.,9.,-1.,9.)
      CALL UOUTLN
      CALL USET ('REFERENCEAXIS')
      CALL AXIS
      CALL UMOVE (0.,0.)
      CALL UCOSYS (2.,5.,1.,1.,20.)
      CALL UPEN1 (0.,0.,'DARROW')
      CALL AXIS
      CALL UMOVE (0.,0.)
      CALL UCOSYS (5.,1.,1.,1.,45.)
      CALL UPEN1 (0.,0.,'DARROW')
      CALL AXIS
      CALL UEND
      STOP
      END
      SUBROUTINE AXIS
      CALL UMOVE (-1.,0.)
      CALL UPEN1 (1.,0.,'LARROW')
      CALL UMOVE (0.,-1.)
      CALL UPEN (0.,1.)
      CALL UEND
      STOP
      END
```

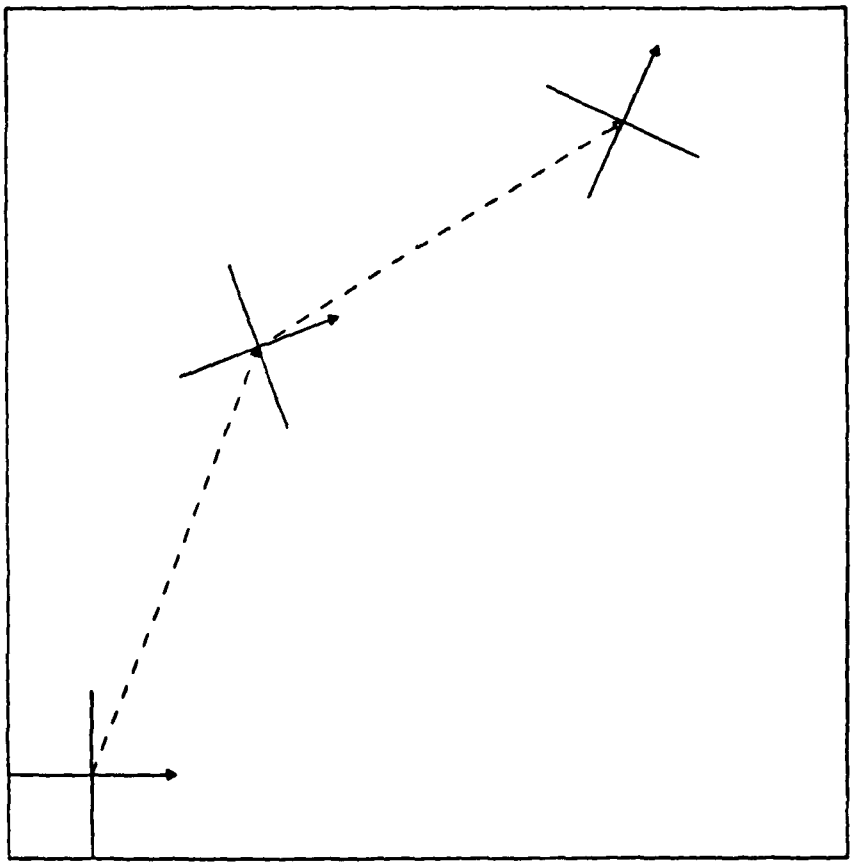


Figure 72. Example 10.3

E X A M P L E 10.4

```
      T=0.
      THETA=75.
      V0=107.
      X0=-180.
      Y0=0.
      CALL USTART
      CALL USET ('WORKINGAXIS')
      CALL UWINDO (-225.,225.,-15.,95.)
      CALL UOUTLN
      CALL UMOVE (X0,Y0)
      CALL USET ('SOFTWARE')
      CALL UPSET ('HORIZONTAL',6.)
      CALL UPSET ('VERTICAL',3.)
      CALL UPSET ('SETDASH',92.)
      DO 1 I = 1, 11
      X = (.707*V0*T) + X0
      Y = T * (.707 * V0 - (16.*T)) + Y0
      CALL UCOSYS (X,Y,1.,1.,THETA)
      CALL UPEN1 (0.,0.,'DASH')
      CALL UWINDO (0.,100.,0.,100.)
      CALL UWHERE (X,Y)
      CALL UPEN1 (X,Y,'DD')
      CALL UWINDO (-225.,225.,-15.,95.)
      T = T + 0.4758
      THETA = THETA + 57.
    1 CONTINUE
      CALL UEND
      STOP
      END
```

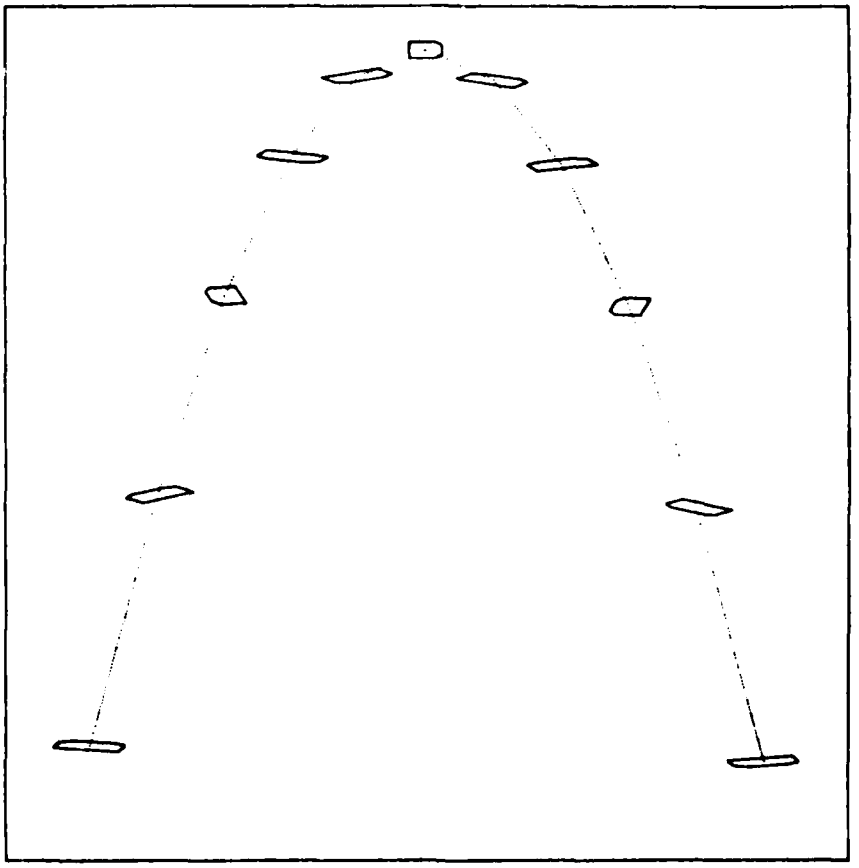


Figure 73. Example 10.4

E X A M P L E 10.5

```
      T=0.
      V0=107.
      X0=-180.
      Y0=0.
      CALL USTART
      CALL USET ('WORKINGAXIS')
      CALL UWINDO (-225.,225.,-15.,95.)
      CALL UOUTLN
      VX = .707 * V0
      ZETA = 11. * VX
      CALL UMOVE (X0,Y0)
      CALL UPSET ('SETDASH',92.)
      DO 1 I = 1, 11
      X = (VX*T) + X0
      VY = 2. * (VX - (32.*T))
      Y = T * (VX - (16.*T)) + Y0
      CALL UCOSYS (X,Y,1.,1.,0.)
      CALL UPEN1 (0.,0.,'DASH')
      CALL UWINDO (0.,ZETA,-ZETA,ZETA)
      CALL USET ('RELATIVE')
      CALL UPEN1 (VX,0.,'LARROW')
      CALL UMOVE (-VX,0.)
      CALL UPEN1 (0.,VY,'LARROW')
      CALL UMOVE (0.,-VY)
      CALL USET ('ABSOLUTE')
      CALL UWINDO (-225.,225.,-15.,95.)
      T = T + .4758
    1 CONTINUE
      CALL UEND
      STOP
      END
```

E X A M P L E 10.5

```
      T=0.
      V0=107.
      X0=-180.
      Y0=0.
      CALL USTART
      CALL USET ('WORKINGAXIS')
      CALL UWINDO (-225.,225.,-15.,95.)
      CALL UOUTLN
      VX = .707 * V0
      ZETA = 11. * VX
      CALL UMOVE (X0,Y0)
      CALL UPSET ('SETDASH',92.)
      DO 1 I = 1, 11
      X = (VX*T) + X0
      VY = 2. * (VX - (32.*T))
      Y = T * (VX - (16.*T)) + Y0
      CALL UCOSYS (X,Y,1.,1.,0.)
      CALL UPEN1 (0.,0.,'DASH')
      CALL UWINDO (0.,ZETA,-ZETA,ZETA)
      CALL USET ('RELATIVE')
      CALL UPEN1 (VX,0.,'LARROW')
      CALL UMOVE (-VX,0.)
      CALL UPEN1 (0.,VY,'LARROW')
      CALL UMOVE (0.,-VY)
      CALL USET ('ABSOLUTE')
      CALL UWINDO (-225.,225.,-15.,95.)
      T = T + .4758
    1 CONTINUE
      CALL UEND
      STOP
      END
```

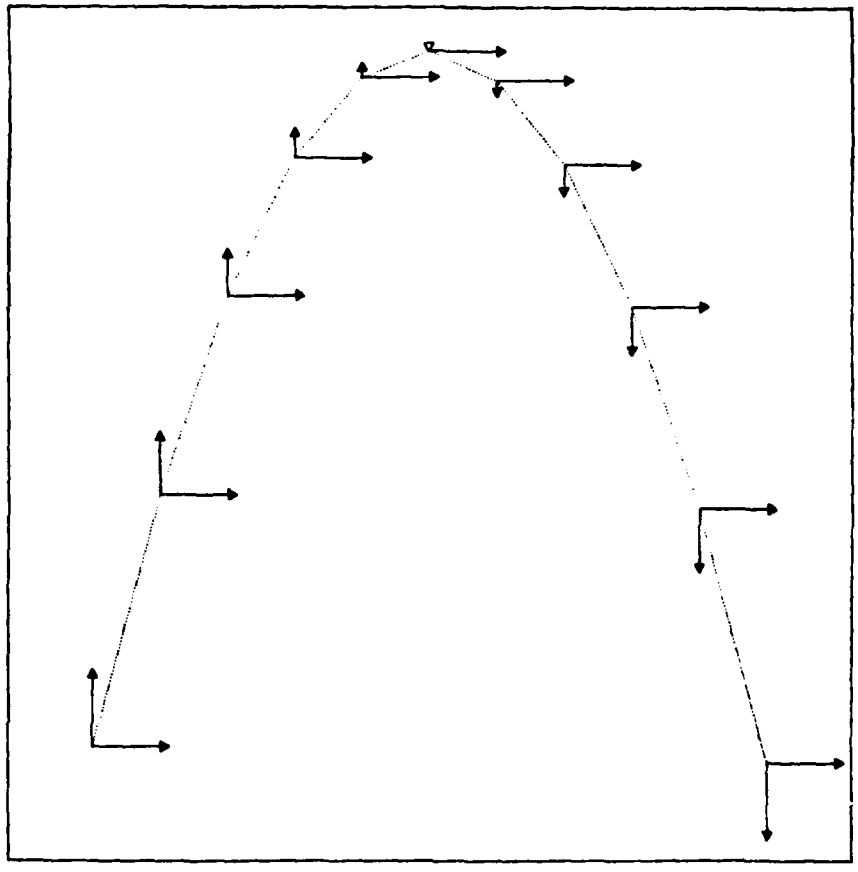


Figure 74. Example 10.5

E X A M P L E 10.6

```
      T=0.
      V0=107.
      X0=-180.
      Y0=0.
      CALL USTART
      CALL USET ('WORKINGAXIS')
      CALL UWINDO (-225.,225.,-225.,225.)
      CALL UOUTLN
      VX = .707 * V0
      CALL UMOVE (X0,Y0)
      CALL UPSET ('SETDASH',92.)
      CALL USET ('RADIANS')
      DO 1 I = 1, 11
      X = (VX*T) + X0
      Y = T * (VX - (16.*T)) + Y0
      DYDX = 1 - (32 * (X-X0) / VX**2)
      CALL UCOSYS (X,Y,1.,1.,ATAN(DYDX))
      CALL UPEN1 (0.,0.,'DASH')
      CALL UWINDO (0.,1.,-1.,0.)
      CALL USET ('RELATIVE')
      CALL UPEN1 (.05,0.,'LARROW')
      CALL UMOVE (-.05,0.)
      CALL UPEN1 (0.,-.05,'LARROW')
      CALL UMOVE (0.,.05)
      CALL USET ('ABSOLUTE')
      CALL UWINDO (-225.,225.,-225.,225.)
      T = T + .4758
    1 CONTINUE
      CALL UEND
      STOP
      END
```

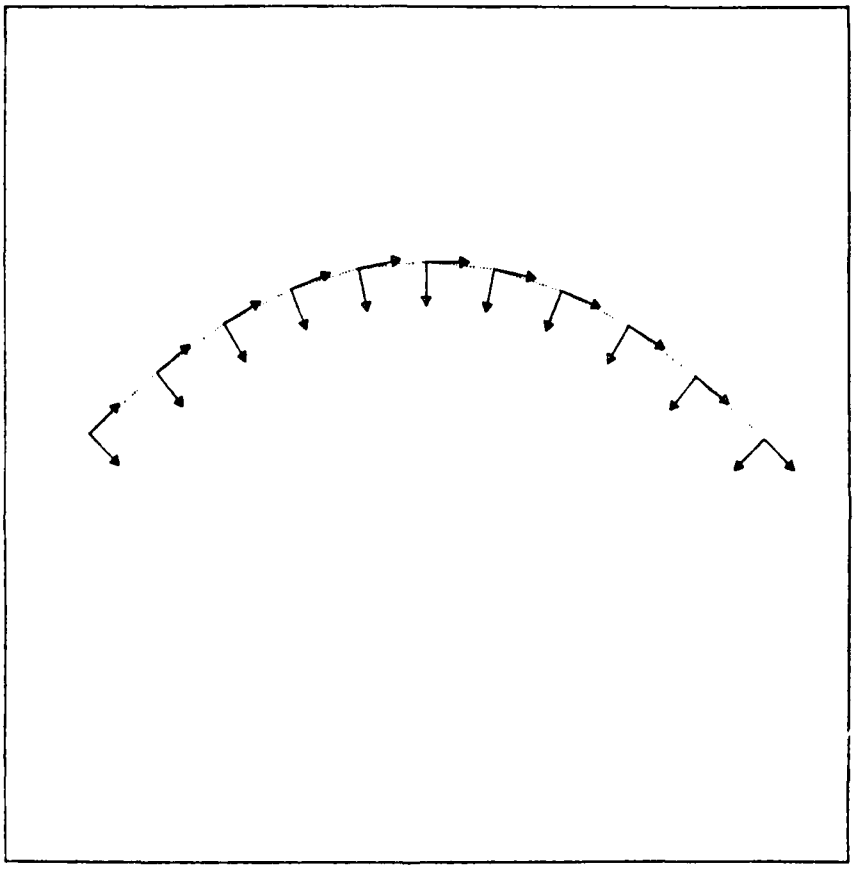


Figure 75. Example 10.6

E X A M P L E 12.1

```
      CALL ATTACH(1,'GRAPHICS/LESSON/FILE121;',3,0,ISTAT,)
      CALL ATTACH(8,'GRAPHICS/LESSON/FILE121S;',3,0,ISTAT,)
      CALL USTART
      CALL UPSET ('TERMINATOR',';')
      CALL UPSET ('SPEED',120.)
      CALL UPSET ('LIBRARY',1.)
      CALL UPSET ('SETDASH',1.)
      CALL USET ('PERCENTUNITS')
      CALL UDAREA (0.,100.,0.,100.)
      CALL UERASE
      CALL UOUTLN
      CALL UWINDO (-100.,100.,-100.,100.)
      CALL USTRCT ('AXIS ')
      CALL AXIS
      CALL UTERM ('AXIS ')
      CALL USET ('ORTHOGONAL')
      CALL U3CSYS (25.,25.,25.,1.,1.,1.,10.,10.,0.)
      CALL UVIEW (-20.,-20.,-150.,0.,0.,0.)
      CALL USTRCT ('BOX  ')
      CALL BOX
      CALL UTERM ('BOX  ')
      CALL USET ('PERSPECTIVE')
      CALL BOX
      CALL UTILTY ('SAVE',8.)
      CALL UTILTY('PURGE',1.)
      CALL UEND
      STOP
      END
      SUBROUTINE AXIS
      CALL USET ('DASH')
      CALL U3MOVE (-20.,-20.,-20.)
      CALL U3PEN (50.,-20.,-20.)
      CALL UPRNT1 ('X;','TEXT')
      CALL U3MOVE (-20.,-20.,-20.)
      CALL U3PEN (-20.,50.,-20.)
      CALL UPRNT1 ('Y;','TEXT')
      CALL U3MOVE (-20.,-20.,-20.)
      CALL U3PEN (-20.,-20.,100.)
      CALL UPRNT1 ('Z;','TEXT')
      CALL UEND
      STOP
      END
      SUBROUTINE BOX
      CALL USET ('LINE')
      CALL U3MOVE (0.,0.,0.)
      CALL U3PEN (10.,0.,0.)
      CALL U3PEN (10.,10.,0.)
      CALL U3PEN (0.,10.,0.)
      CALL U3PEN (0.,0.,0.)
      CALL U3MOVE (10.,0.,0.)
      CALL U3PEN (10.,0.,60.)
      CALL U3PEN (10.,10.,60.)
      CALL U3PEN (10.,10.,0.)
      CALL U3MOVE (0.,10.,0.)
      CALL U3PEN (0.,10.,60.)
      CALL U3PEN (10.,10.,60.)
      CALL U3MOVE (0.,0.,0.)
      CALL U3PEN (0.,0.,60.)
      CALL U3PEN (0.,10.,60.)
      CALL U3PEN (0.,10.,0.)
      CALL U3MOVE (0.,0.,60.)
      CALL U3PEN (10.,0.,60.)
      CALL UEND
      STOP
      END
```

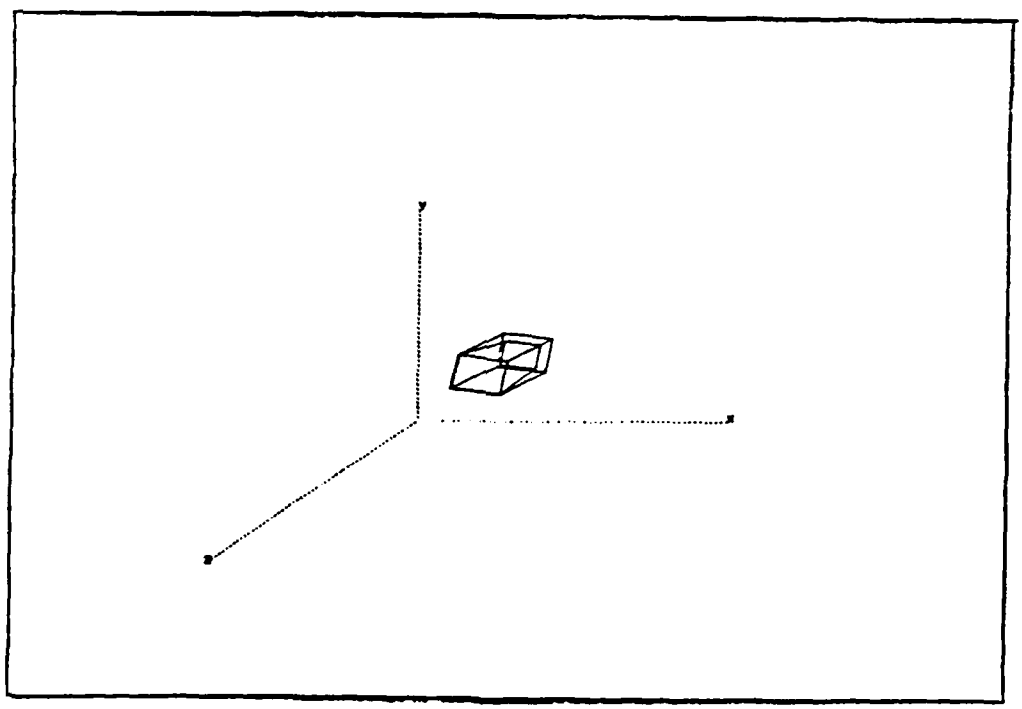


Figure 76. Example 12.1

E X A M P L E 12.4

```
      CALL ATTACH(1,'GRAPHICS/LESSON/FILE124;',3,0,ISTAT,)
      CALL ATTACH(8,'GRAPHICS/LESSON/VILLAG;',3,0,ISTAT,)
      CALL USTART
      CALL UPSET ('LIBRARY',1.)
      CALL UTILTY ('LOAD',8.)
      CALL UPSET('TERMINATOR',';')
      CALL USET ('VIEWDISTANCE')
      CALL UVWPRT (150.)
      CALL UWINDO (-100.,100.,-100.,100.)
      CALL UVIEW (-40.,200.,70.,-20.,20.,0.)
      CALL VILLAG
      CALL UEND
      STOP
      END
      SUBROUTINE VILLAG
      CALL USET ('XYZ ')
      CALL USET ('SYSTEMAXIS')
      CALL USET ('REFERENCEAXIS')
      CALL USET ('BLACK')
      CALL U3CALL (-50.,20.,0.,1.,1.,1.,90.,0.,0.,'CHURCH')
      CALL USET ('RED ')
      CALL U3CALL (24.,-19.,0.,1.,1.,1.,90.,-90.,0.,'SCHOOL')
      CALL USET ('BLUE')
      CALL U3CALL (70.,70.,0.,0.7,0.7,0.7,0.,0.,0.,'PIZZA')
      CALL USET ('BLACK')
      CALL U3CALL (0.,0.,0.,1.,1.,1.,0.,0.,0.,'ROAD ')
      CALL UEND
      STOP
      END
```

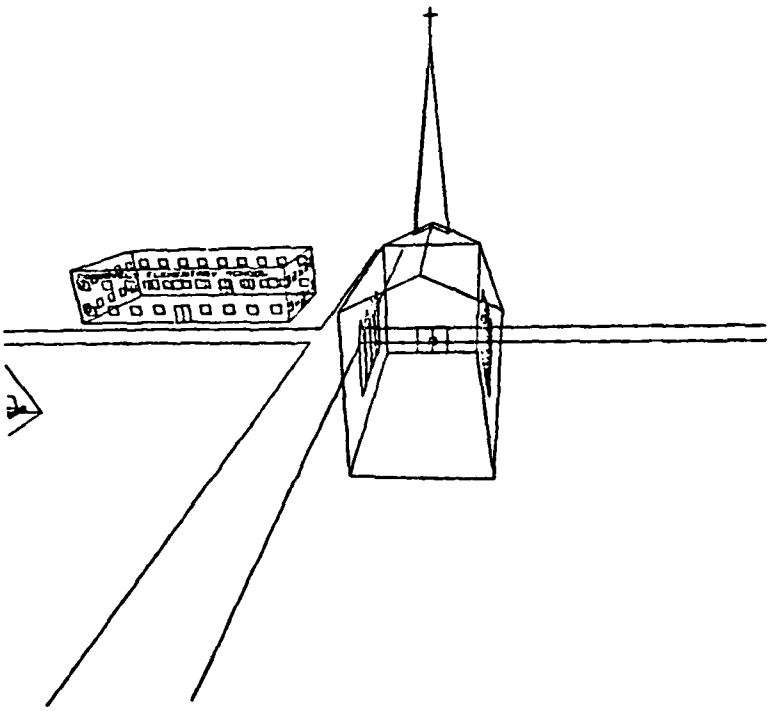


Figure 77. Example 12.4

E X A M P L E 13.1

```
      CALL ATTACH(1,'GRAPHICS/LESSON/FILE131;',3,0,ISTAT,)
      X = 10.0
      Y = 80.0
      CALL USTART
      CALL UPSET ('LIBRARY',1.)
      CALL UOUTLN
      DO 1 I = 1, 8
      CALL UFRAME ('TRIANGLE')
      CALL UMOVE (X,X)
      CALL UPEN (X,(X+5.0))
      CALL UPEN ((X+5.0),X)
      CALL UPEN (X,X)
      CALL UFREND ('TRIANGLE')
      CALL UFRAME ('SQUARE')
      CALL UMOVE (X,Y)
      X = X + 10.0
      Y = Y - 10.0
      CALL URECT ((X-5.0),(Y+15.0))
      CALL UFREND ('SQUARE')
    1 CONTINUE
      CALL UEND
      STOP
      END
```

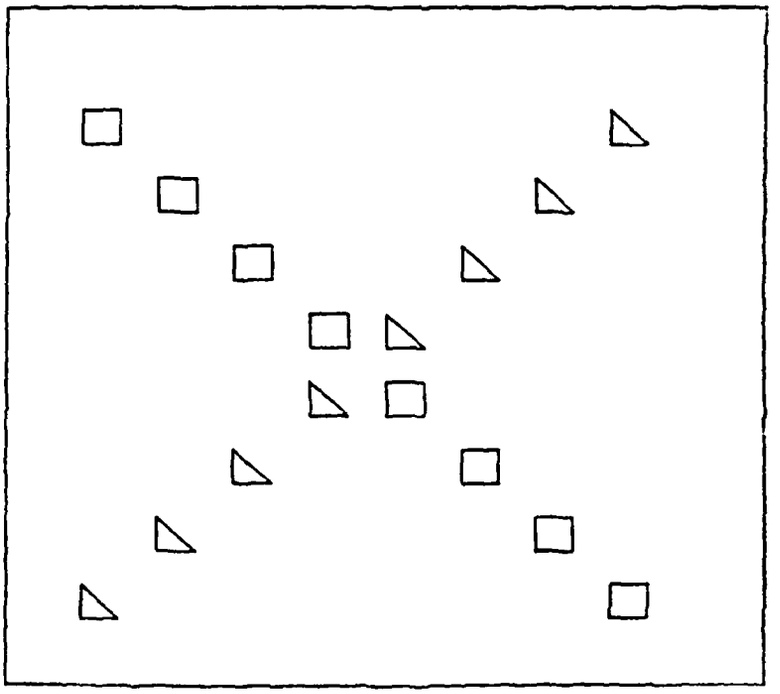


Figure 78. Example 13.1

E X A M P L E 13.2

```
      CHARACTER CHAR*1
      CALL ATTACH(1,'GRAPHICS/LESSON/FILE132;',3,0,ISTAT,)
      X=10.0
      CALL USTART
      CALL UPSET ('LIBRARY',1.)
      CALL UOUTLN
      DO 1 I = 1, 8
      CALL UFRAME ('TRIANGLE')
      CALL UMOVE (X,X)
      CALL UPEN (X,(X+5.0))
      CALL UPEN ((X+5.0),X)
      CALL UPEN (X,X)
      X = X + 10.0
      CALL UFREND ('TRIANGLE')
    1 CONTINUE
      CALL UNSHOW ('TRIANGLE')
    2 CALL UAIN (CHAR)
      IF (CHAR .NE. 'T') GO TO 3
      CALL USHOW ('TRIANGLE')
      GO TO 2
      CALL UEND
    3 STOP
      END
```

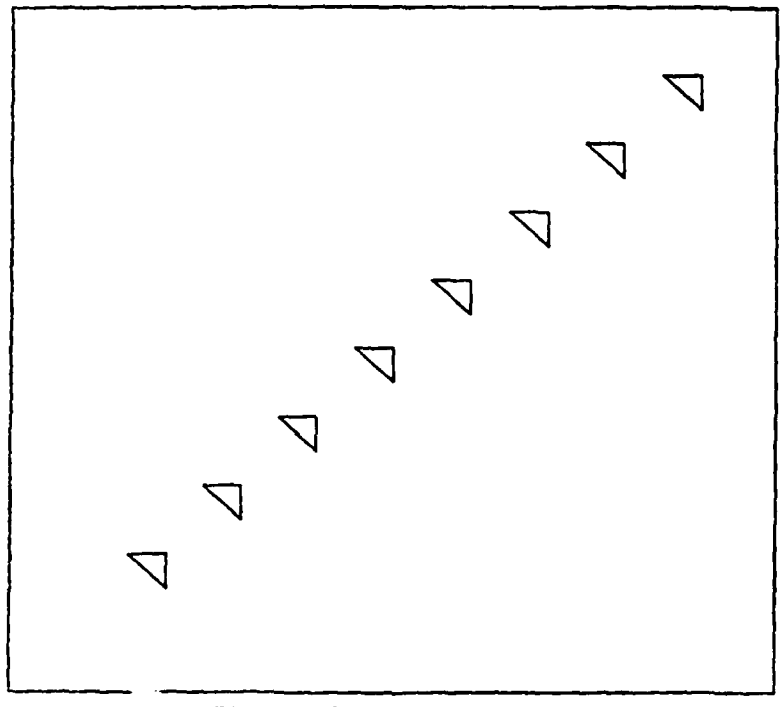


Figure 79. Example 13.2

E X A M P L E 13.3

```
      CALL ATTACH(1,'GRAPHICS/LESSON/FILE133;',3,0,ISTAT,)
      DEGREE=0.
      CALL USTART
      CALL UPSET ('LIBRARY',1.)
      CALL UWINDO (-50.,50.,-50.,50.)
      CALL UOUTLN
      CALL USET ('POLAR')
      DO 1 I = 1, 12
      DEGREE = DEGREE + 30.0
      CALL UFRAME ('SQUARE')
      CALL UMOVE (25.,DEGREE)
      CALL UPSET ('ROTATE',DEGREE)
      CALL USET ('RELATIVE')
      CALL UPLYGN (0.,0.,4.,5.)
      CALL USET ('ABSOLUTE')
      CALL UFREND ('SQUARE')
    1 CONTINUE
      CALL UEND
      STOP
      END
```

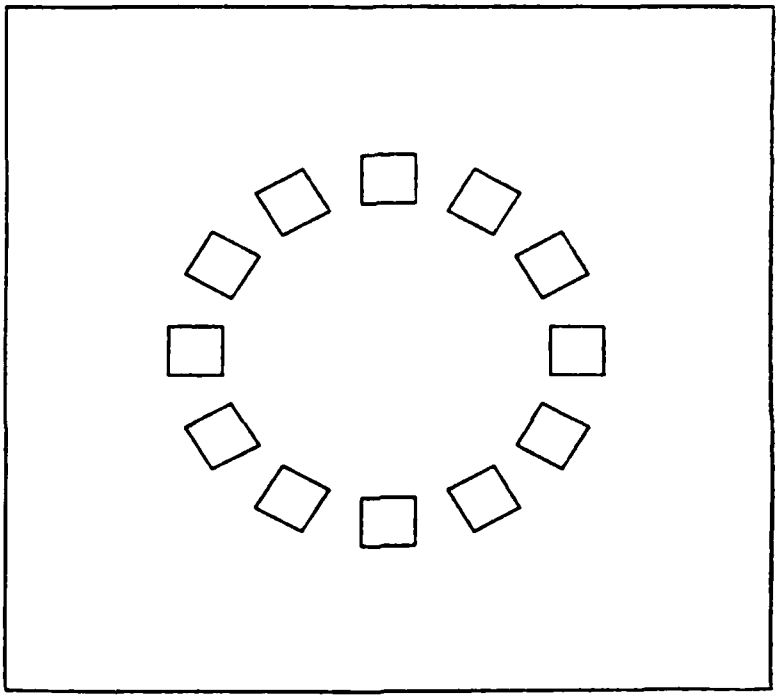


Figure 80. Example 13.3

E X A M P L E 13.4

```
      INTEGER FRAME(9)
      DATA FRAME/'A','B','C','D','E','F','G','H','I'/
      CALL ATTACH(1,'GRAPHICS/LESSON/FILE134;',3,0,ISTAT,)
      CALL USTART
      CALL UPSET ('LIBRARY',1.)
      CALL UOUTLN
      DO 1 I = 1, 9
      X = 10.0 * FLOAT(I)
      CALL UFRAME (FRAME(I))
      CALL UMOVE (X,X)
      CALL UPSET ('ROTATE',X)
      CALL USET ('RELATIVE')
      CALL UPLYGN (0.,0.,4.,FLOAT(I))
      CALL USET ('ABSOLUTE')
      CALL UFREND (FRAME(I))
      CALL UNSHOW (FRAME(I))
    1 CONTINUE
      DO 2 I = 1, 9
      J = MOD((I-1),9) + 1
      CALL USHOW (FRAME(J))
      CALL UNSHOW (FRAME(J))
    2 CONTINUE
      CALL UEND
      STOP
      END
```

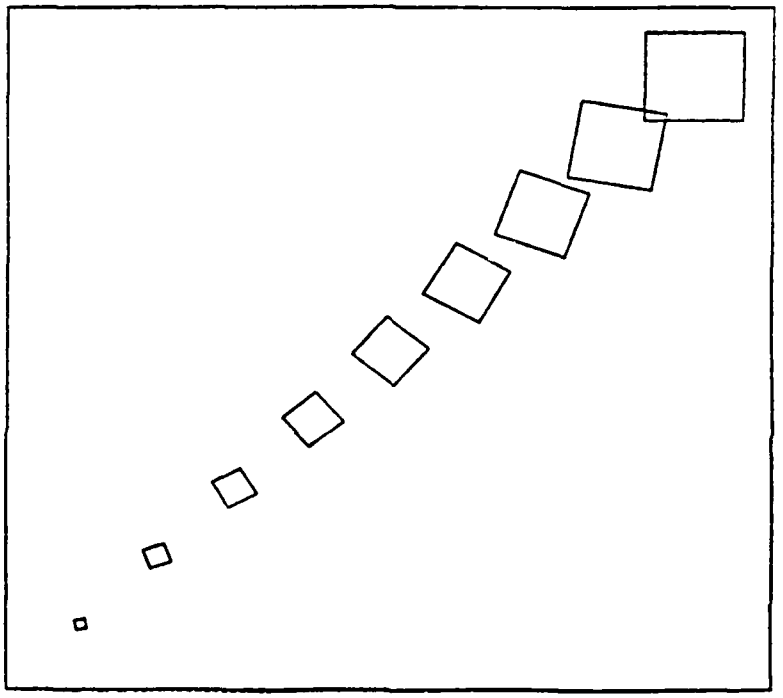


Figure 81. Example 13.4

In accordance with letter from DAEN-RDC, DAEN-ASI dated 22 July 1977, Subject: Facsimile Catalog Cards for Laboratory Technical Publications, a facsimile catalog card in Library of Congress MARC format is reproduced below.

Ward, Darrell
Notebook for lessons on the Graphics Compatability System (GCS) / by Darrell Ward, James M. Jones II, Michael E. George (Automatic Data Processing Center, U.S. Army Engineer Waterways Experiment Station). -- Vicksburg, Miss. : The Station ; Springfield, Va. : available from NTIS, 1982.
143 p. ; ill. ; 27 cm. -- (Instruction report ; K-82-4)
Cover title.
"April 1982."
Final report.
"Prepared for Office, Chief of Engineers, U.S. Army."

1. Computer graphics. 2. Computer program. 3. FORTRAN (Computer program language). 4. Programming (Electronic computers). I. Jones, James M. II. George, Michael E. III. United States. Army. Corps of Engineers. Office of the Chief of Engineers. IV. U.S. Army

Ward, Darrell
Notebook for lessons on the Graphics Compatability : ... 1982.
(Card 2)

Engineer Waterways Experiment Station. Automatic Data Processing Center. V. Title VI. Series; Instruction report (U.S. Army Engineer Waterways Experiment Station) ; K-82-4.
TA7.W34i no.K-82-4


FILMED
7-8