



Curriculum Units by Fellows of the Yale-New Haven Teachers Institute
1981 Volume VI: Computing

Graphing and the Computer

Curriculum Unit 81.06.06
by James Francis Langan

Computers may be used four ways in teaching: as teaching machines, as business machines, as producers of teaching aids, and as users of concepts taught. As a teaching machine the computer drills students on topics previously seen or it may initiate instruction in new areas. As a business machine the computer may keep and interpret records. As a producer of learning aids the computer may solve problems, print tables or draw figures. As a user of concepts the computer's program, how it accomplishes a task, becomes the point of study.

The first two uses are highly dependent upon hardware. As a teaching machine there must be as many machines as students to be taught. As a business machine the computer must be as available as one's marking book, if it is to be an improvement over present practice.

The second two uses are closely related. The results of a program or a program itself may be shown to a class with out the computer present. The program may be demonstrated on one day to motivate or introduce a topic, and then discussed on the computer,

This paper consists of two major programs. The first is a teaching machine. The second makes topics from the curriculum, The first teaches Cartesian graphing the stereograms of geometric solids. The user may orient the solid in any direction computer before the stereogram is printed. Listings of the programs are found below.

My personal goal in the unit was to write programs of use in teaching and at the same time explore the graphics capability of the computer. I have never oscilloscope before.

Let us look at the programs. Two dimensional Cartesian plotting may be taught early in the Algebra I curriculum. In fact, one test used the capable students who compare notes feel short changed. If there was a program available these students could use it independently before the topic came up in class. When the topic does come up in the normal time sequence, the least capable students find they need more practice. Again the machine is an answer. Many students find it fun to operate the computer. Thus having an incentive to get the practice.

The plotting program is in three major parts. One uses the oscilloscope to teach x-y plotting. The *next* part reading the graph. An "x" is placed o *n* a coordinate system by the plotter and the student is asked for its coordinates. The third part reverses the second by giving the coordinates and responding to commands that move the plotter pen to the correct position.

I have tried to make this program self-explanatory. The oscilloscope portion begins in line 1760. It sets up X and Y area on the scope just as the other parts will do on the plotter. It then selects a random point placing an "x" at that position. Then it prints the numbers along the x axis from zero to whatever is the x coordinate of the point. After a wait it prints the numbers from zero to whatever is the y coordinate of the point vertically from where the x numbers ended. The student is told that the last x number is the x coordinate and that the last y number is the y coordinate. The student may repeat this part as often as he likes getting new points each time. Lines 1880 PRINT "LESSON" and 2230 PRINT "REVIEW" provided space in which to write fuller explanations. I found the repeat option made explanations less necessary. The student discovers the concept.

The second part uses the plotter. The student can take his work home. The computer draws the axis, generates a point and asks the student for the coordinates. The student's answer is tested, if correct the score is increased and a new problem given. There are twelve problems to a set. If the student's answer is wrong the machine moves the pen to his point so he may see the difference between his answer and the machine's point. His answer is tested for typical errors: switching x and y coordinates, having one coordinate correct and the other wrong, having the magnitude correct but not the sign. There are three chances to get any one problem correct. After three tries the machine gives the correct coordinates. Part credit is given as follows: two thirds credit if two tries, one third credit if correct after three tries.

The third section draws a new coordinate system on a fresh page. The student is given a coordinate pair, the pen is placed at (9,9) and the student is required to move the pen to the proper position. Movement is caused by the "L","R","U", and "D" keys standing for left, right, up, and down respectively. The command is ended with the letter "P" to tell the computer the point is plotted. Scoring here is as in the previous section. If the student says "P" at the wrong position the computer gives the coordinates of the error. The student may move the pen out of the way to the edge of the board or proceed to another answer from the error. If the student should be wrong after three tries the machine will move to the correct point and pause for about five seconds. This allows the student to see and to check the solution before the another problem.

This no sophisticated techniques. Counts the number of problems done and the number of tries taken. The "CALL" statements are machine language subroutines provided by the system and are transparent to us.

The plotter's coordinate system has (0,0) as its lower left hand corner and (9999,9999) as its upper right hand corner. We want those points to be (-9,-9) and (9,9) respectively. Statement 2310 does that; it changes our coordinates into plotter coordinates. $FNS(-9) = 0$ and $FNS(9) = 18 * S = 9990$. S is set to 555 in line 2280. S is the number of machine coordinates between any one of our coordinates.

So the longer of the two programs is the less sophisticated. The teaching machine type program can be written by beginners. These programs are long only due to the explanations needed by the students. The length is another reason for us to *pool* these programs or we will end up writing them over and over again.

The second program uses concepts that will need to be explained. How can we draw pictures with the computer? How can we change the view point of these pictures? There are commercials on television where a designer has a line drawing of a car in perspective displayed on a CRT screen. By tapping keys he change the orientation of the car. That is what we do in the second program. A geometric solid is entered as DATA, is displayed on the scope, is rotated to a pleasing orientation and then drawn as a stereogram on the plotter. The concepts used here are taught in school.

In his book, *Mathematics in Western Culture* © 1953 Oxford University Press, Morris Kline has a chapter on "painting and Perspective". The chapter contains four wood cuts by Albrecht Dürer showing various

mechanical aids Renaissance artists used to develop a painting's perspective. The first, easel with a transparent panel having an eye sight point in front to position the artist's eye as he copies the scene behind onto the panel. The second illustration, "The Designer of the Lying Woman", uses a latticed frame instead of a transparent-panel. The artist has a correspondingly marked paper in front of him upon which he transfers what he sees. The lattice looks like coordinates. Durer died in 1528. Descartes was born in 1596. More evidence that mankind uses concepts without "knowing" them. The third figure, "The Designer of the Can", uses a transparent panel and a hand sight attached to a point on the wall behind the artist. Looking at the scene behind the panel through the sight establishes a point on the panel as if the artist's eye were on the wall farther away than the length of his arms. The next cut is the most mathematical. "The Designer of the Lute" uses an open frame and a sight line. Here the sight line passes to the object drawn, held by an assistant, while the artist measures the position where the string passes through the frame, transferring the measurements to the drawing. Here is a schematic of the situation.

(figure available in print form)

E is the position of the eye, PP' is the picture plane, F is the three dimensional object to be drawn. Ignore B for the moment. So to draw the figure the string is moved to each point of the figure and the corresponding points are found in the picture plane. A great deal of work. So how do we do it with the computer?

We must turn the object F into numbers, coordinates, change the sight lines into lines with equations. Solve for the points where the sight lines intersect the picture plane. Plot and connect the points to form the picture.

So we have an opportunity to talk about three dimensional coordinates, and the equation of a line in three space. The Advanced Algebra text I use teaches the vector equation of a line in two dimensions, I tell the students the situation is directly analogous in three space. I now have a concrete reason to show them.

Having set up coordinates we may use coordinate transformations to rotate the figure to different orientations. This is a topic that I have tried to interest students in. They seem to find it a useless activity this changing of coordinates. Why would anyone want to do it? Well now I have a useful example.

If a computer can do a task once it can do it any number of times. So let us do the same for the other eye. This gives us two images, a stereographic pair. This is done by changing the coordinates of the eye in the first case to the coordinates of the second eye for the second image. The points are plotted on the plotter and you have hard copy to take back to your class. Run a sequence of images showing different orientations.

The program lends itself to production of geometry illustrations. Often students have difficulty in seeing three dimensions in solid geometry proof illustrations. The teacher may make a model using pencils and pens as lines and pieces of paper as planes. However some students still say "That's not the picture in the book." Well now the program can turn the book illustration into a stereo image. "The picture in the book" becomes three dimensional.

Having the program we can explore the theory. We can see how changes in eye position effect the image. We can investigate curved lines. We can try various methods to enlarge the size of the image. This last question was the one that held up this program for me. As vectors the points of the figure may be enlarged by taking a scalar multiple of them. That is what I did. However, the computer said the coordinates were too large to fit on the plotter. Looking at the illustration you see that the figure on plane PP has to be smaller than the original figure. What was wrong? While I enlarged the figure I did not move the eye position, the eyes ended up inside the figure. There is an easy remedy. Looking at the illustration plane B gives another image of the figure, a larger image. So to enlarge your figure move the picture plane beyond the figure from the eyes. This is the

reason for lines 350 and 360 in the program. I gave up on scalar multiplication and moved the picture plane in statement 40 to $z = -10$. This was done by trial and error.

More specific explanation of the program is in order. The x , y and z coordinates are set as follows, the x axis is horizontal in the plane of the plotter, the y axis is vertical in the plane of the plotter and the z axis is perpendicular coming out of the plane of the plotter. The equation of a line of sight:

$$(P_x, P_y, P_z) - (E_x, E_y, E_z) + T * ((E_x, E_y, E_z) - (A_x, A_y, A_z))$$

or

(figure available in print form)

Where P is the point in the picture plane, E is the eye point, A is a point of the figure to be drawn and t is the parameter. Line 390 solves for t knowing the value of P the z coordinate of the picture plane. Lines 400 and 410 then solve for the x and y coordinates of the point in the picture plane. K is the count of the number of the points in the A array.

To draw another figure you must change the DATA. There may be more efficient ways to do this. If so, that is another area of exploration and learning motivated by this program. This program needs the following data in the following order. The first six numbers are the coordinates of the eyes, right eye then the left eye, The next number is the number of lines in the figure. Following that each line needs six numbers, the coordinates of each end point. This means that a single point will be listed as many times as there are lines intersecting it. This is what I meant by a "better way". Perhaps graph theory could be looked at here by an interested independent study student. The program is able to handle a maximum of 50 lines in a figure as it is *now* dimensioned.

So there is more that can be done. I only wanted a program to show a use of some of the topics covered *in* the curriculum, to reassure students their study was not in vain. The highly curious students may use this program as motivation to explore the topic further. Computers motivate.

The rotations of the figure is achieved by matrix multiplication. The formulae for rotating a system of points (x, y, z) in the x - y plane to get a new system (X, Y, Z) is

$$X = x \cos A + y \sin A$$

$$Y = -x \sin A + y \cos A$$

$$Z = z$$

where A is the angle of rotation. This is the same as the following matrix multiplication problem.

$$(X, Y, Z) = (x, y, z) \begin{pmatrix} \cos A & \sin A & 0 \\ -\sin A & \cos A & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Similar matrices are used for the other rotations. The matrices are defined in lines 100 through 250. They are used in lines 670 through 840. So another topic of discussion reveals itself, matrices.

I only had time to work with one geometric figure. The figures shown after the program listing were all drawn

from the same set of coordinates. The only difference is that the coordinates were rotated by the matrices. I previewed the figures on the oscilloscope, the ones I liked I had plotted. As the program now stands you have to move the paper to the right to keep the left eye image from writing over the right.

The figure consists of a pyramid with a square base and a tetrahedron both having unit length edges. The figure is drawn as if the faces were glass. The question is how many faces does the combined figure have.

Bibliography

The computer is an opportunity for discovery. As such one works with the computer itself instead of reading about working with the computer. One will need a manual for the machine being used. It is easier to do other people's problems than to think up your own so a problem book is convenient.

William Ralph Bennett, Jr., *Scientific and Engineering Problem Solving with the Computer*, Prentice- Hall, Inc., © 1976

John G. Kemeny and Thomas Kurtz, *Basic Programming*, John Wiley & Sons, Inc. © 1971.

Morris Kline, *Mathematics in Western Culture*, Oxford University Press, © 1953.

Level II Basic Reference Manual, © 1979 Radio Shack, A Division of Tandy Corporation.

```
REM TEACHING MACHINE FOR CARTESIAN GRAPHING
10 PRINT " '1' STANDS FOR YES ON ALL ANSWERS."
11 PRINT " ANY OTHER NUMBER IS 'NO'."
12 PRINT
20 REM DEFINE FUNCTIONS AND INITIALIZE VARIABLES
30 GOSUB 2270
40 PRINT "DO YOU KNOW HOW TO GRAPH",
50 PRINT "('1' MEANS 'YES' )";
55 INPUT A7
60 IF A7=1 THEN 80
70 GOSUB 1760
80 PRINT "WHAT IS YOUR CHOICE: 1— GRAPH READING OR"
89 PRINT
90 PRINT " 2— GRAPH PLOTTING";
99 PRINT
100 REM STUDENT MAY SELECT TO DO ONLY THE SECOND PART.
110 INPUT A9
111 PRINT "IS THE PLOTTER TURNED ON";
112 INPUT A7
120 IF A9=2 THEN 820
130 REM LINE 790 STARTS PLOTTING PORTION *****
140 GOSUB 2750
150 REM 2580 IS THE AXES DRAWING ROUTINE*****
160 REM
170 REM ***** READING A GRAPH *****
```

```

179 PRINT
180 PRINT " THE MACHINE WILL PRINT AN 'X' AT A "
190 PRINT "RANDOM POINT. YOUR ANSWERS WILL BE WHOLE NUMBERS." 200 LET K=0
210 IF T=12 THEN 650
220 GOSUB 2460
230 REM ***** COUNT THE NUMBER OF THE PROBLEM*****
240 LET T=T+1
250 REM ***** MOVE PEN OUT OF THE WAY *****
260 LET Y2=Y1
270 CALL (1,-1,1,9999,Y2)
280 PRINT
290 PRINT "WHAT ARE THE COORDINATES OF THE POINT"
300 INPUT P,Q
310 REM ***** MOVE PEN TO STUDENT'S POINT *****
320 CALL (1,-1,1,FNS(P),FNS(Q))
330 REM ***** COUNT THE TRIES *****
340 LET K=K+1
350 IF (X=P) AND (Y=Q) THEN 490
360 PRINT
369 PRINT
370 PRINT "THE PEN IS NOW AT YOUR ANSWER; (;P;";";Q;")."
380 IF K=3 THEN 440
390 IF (X=P) AND (Y*Q) THEN 550
400 IF (X*P) AND (Y=Q) THEN 570
410 IF (X=Q) AND (Y=P) THEN 590
420 IF (X=-P) OR (Y=-Q) THEN 610
430 IF K3 THEN 630
440 PRINT "THE CORRECT CO-ORDINATES ARE (;X;";";Y;")."
450 WAIT (5000)
460 PRINT "TRY ANOTHER."
470 GOTO 510
480 REM COUNT THE NUMBER OF PROBLEMS CORRECT
490 LET Z=Z+(4- K)/3
500 PRINT "CORRECT IN (;K;" TRIES."
510 CALL (1,-1,1,X1,Y1)
520 CALL (2,200,0,0,200)
530 PRINT " ";T
540 GOTO 200
550 PRINT "YOUR X CO—ORDINATE IS CORRECT. TRY AGAIN "
560 GOTO 290
570 PRINT "YOUR Y CO-ORDINATE IS CORRECT. TRY AGAIN."
580 GOTO 290
590 PRINT "YOU HAVE YOUR X AND Y CO-ORDINATES REVERSED. GET IT RIGHT.
600 GOTO 290
610 PRINT "SIGN ERROR."
620 GOTO 290

```

```

630 PRINT "TRY AGAIN."
640 GOTO 290
650 PRINT " THAT IS ALL.";
660 REM 2400 IS THE SCORING ROUTINE*****
670 GOSUB 2570
680 PRINT "IS THERE PAPER IN THE PLOTTER"
690 PRINT "AND IS THE PLOTTER TURNED ON ";
700 PRINT "( '1' MEANS YES )";
710 INPUT AS
720 IF A8#1 THEN 680
730 REM
740 REM ***** PLOTTING POINTS *****
750 PRINT
760 PRINT " HAVE YOU CHANGED THE PAPER";
770 PRINT "( '1' MEANS 'YES' )";
780 INPUT AS
790 LET T=0
800 LET Z=0
810 GOTO 820
820 PRINT
830 GOSUB 2750
840 REM 2580 IS THE AXES DRAWING ROUTINE
850 LET H1=H2=9
860 LET H=FNS(9)
870 REM ***** MOVE PEN TO (9,9) TO START *****
880 CALL (1,-1,1,H,H)
890 PRINT "YOU WILL BE GIVEN THE CO-ORDINATES OF A POINT."
900 PRINT "YOUR TASK IS TO MOVE THE PEN TO THE CORRECT POSITION."
910 PRINT "YOU WILL DO THIS BY TYPING THE FOLLOWING LETTERS:"
920 PRINT " 'R' TO MOVE THE PEN TO THE RIGHT"
930 PRINT " 'L' TO MOVE THE PEN TO THE LEFT"
940 PRINT " 'D' TO MOVE THE PEN DOWN"
950 PRINT " AND 'P' TO SHOW THE POINT IS 'PLOTTED'."
960 PRINT " IT IS YOUR 'END OF INSTRUCTION' COMMAND."
970 PRINT "HINT: START FORM THE ORIGIN ON THE"
980 PRINT "PLOTTER, COUNT TO THE REQUESTED POINT."
1000 PRINT "MATCH THE MARK "
1010 LET X=FNR(X)
1020 LET Y=FNR(Y)
1030 PRINT
1040 PRINT " FIND THE POINT (";X;" ";Y;")."
1050 LET K=0
1060 LET T=T+1
1070 PRINT "TYPE YOUR MOVES"
1080 CALL (22,C3,0)
1090 IF C3=85 THEN 1170

```

```

1100 IF C3=68 THEN 1230
1110 IF C3=76 THEN 1280
1120 IF C3=82 THEN 1330
1130 IF C3=80 THEN 1380
1140 PRINT "YOU MAY ONLY USE 'L','R','U','D' AND 'P'."
1150 GOTO 1080
1160 REM MOVE UP
1170 LET H2=H2+1
1180 IF H2=9 THEN 1580
1190 LET H2=H2-1
1200 PRINT
1210 PRINT "I WILL NOT MOVE OFF THE BOARD!!! START OVER."
1220 GOTO 1070
1230 REM MOVE DOWN ONE
1240 LET H2=H2-1
1250 IF H2<0 THEN 1580
1260 LET H2=H2+1
1270 GOTO 1210
1280 REM MOVE LEFT
1290 LET H1=H1-1
1300 IF H1<0 THEN 1580
1310 LET H1=H1+1
1320 GOTO 1210
1330 REM MOVE RIGHT
1340 LET H1=H1+1
1350 IF H1 = 9 THEN 1580
1360 LET H1=H1-1
1370 GOTO 1210
1380 REM PLOTTED
1390 PRINT
1400 LET K=K+1
1410 IF (X=H1) AND (Y=H2) THEN 1510
1420 PRINT "YOU ARE AT (";H1;";";H2;");" NOT (";X;";";Y;");"
1430 IF K=3 THEN 1620
1440 PRINT "DO YOU WANT TO MOVE FROM HERE";
1450 INPUT Q
1460 IF Q=1 THEN 1070
1470 LET H4=FNS(H2)
1480 CALL (1,-1,1,H,H4)
1490 LET H1=9
1500 GOTO 1070
1510 PRINT
1520 LET Z=Z+(4-K)/3
1530 PRINT "CORRECT IN ";K;" TRIES
1540 PRINT
1550 PRINT

```



```

1560 GOTO 1630
1570 REM ***** MOVE THE PEN ONE SPACE *****
1580 LET X1=FNS(H1)
1590 LET Y1=FNS(H2)
1600 CALL (1,-1,1,X1,Y1)
1610 GOTO 1080
1620 PRINT "THE PEN WILL NOW MOVE TO (";X;" ";Y;")"
1630 LET H3=FNS(X)
1640 LET H4=FNS(Y)
1650 LET H1=X
1660 LET H2=Y
1670 CALL (1,-1,1,H3,H4)
1680 CALL (2,200,0,0,200)
1690 PRINT " ";T
1700 CALL (1,-1,1,H3,H4)
1710 IF T12 THEN 1010
1720 PRINT "THAT IS ALL"
1730 GOSUB 2570
1740 REM 2400 IS THE SCORING ROUTINE*****
1750 STOP
1760 REM *****SUB ROUTINES *****
1770 PRINT "HAVE YOU TURNED ON THE OSCILLOSCOPE ";
1780 INPUT A7
1790 IF A7#1 THEN 1770
1800 LET W=-1
1810 CALL (50,W)
1820 DIM U[20,100]
1830 CALL (3.U[1,1],4000)
1840 REM DRAW AXES
1850 GOSUB 2750
1860 REM PICK A POINT
1870 GOSUB 2460
1880 PRINT "LESSON"
1890 REM CHECK TO SEE CO-ORD NEITHER TRIVIAL NOR IN THE WAY 1900 IF X#0 AND X*-1 AND
Y#0 AND Y#-1 THEN 1930
1910 GOSUB 2460
1920 GOTO 1900
1930 LET S1=1
1940 IF X>0 THEN 1960
1950 LET S1=-1
1960 FOR I=0 TO X STEP S1
1970 CALL (1,-1,1,FNS(I),FNS(.5))
1980 CALL (2,200,0,0,200)
1990 PRINT I
2000 NEXT I
2010 PRINT "SEE THE LAST NUMBER?"

```

```

2020 PRINT "WHAT IS YOUR X CO-ORDINATE."
2030 WAIT (5000)
2039 PRINT
2040 PRINT "NOW COUNT FOR YOUR Y CO-ORDINATE. 2050 LET S1=1
2060 IF Y>0 THEN 2080
2070 LET S1=-1
2080 FOR I=0 TO Y STEP S1
2090 CALL (1,-1,1,FNS(X+.5),FNS(I))
2100 CALL (2,200,0,0,200)
2110 PRINT I
2120 NEXT I
2130 PRINT "SEE THE NUMBER NEXT TO THE POINT.
2140 PRINT "THAT IS YOUR Y CO-ORDINATE. "
2149 PRINT
2150 PRINT "SO THE POINT'S X AND Y CO-ORDINATES"
2160 PRINT "ARE (";X;",";Y;")"
2170 PRINT
2180 PRINT "DO YOU WANT TO SEE ANOTHER EXAMPLE"; 2190 INPUT A7
2210 IF A7=1 THEN 1830
2220 CALL (50,1)
2230 PRINT "REVIEW"
2240 PRINT "NOW CONTINUE THE LESSON."
2250 RETURN
2260 STOP
2270 LET W=1
2280 LET S=555
2290 REM ***** SCALE FUNCTION CONVERTS CO-ORDINATES
2300 REM BETWEEN -9 & +9 TO PLOTTER CO-ORDINATES
2310 DEF FNS(X)=(X+9)*S
2320 REM ***** FUNCTION TO GENERATE RANDOM NUMBER BETWEEN -8 & 9.`
2330 DEF FNR(X)=INT(17*RND(0))-8
2340 LET Z=0
2350 LET K=T=0
2360 PRINT "WHAT IS YOUR FAVORITE NUMBER BETWEEN 143 AND 278";
2370 REM **** SO MACHINE WILL GIVE EACH STUDENT
2360 REM DIFFERENT PROBLEMS
2390 INPUT F
2400 FOR I=1 TO F
2410 LET X=RND(0)
2420 NEXT I
2430 RETURN
2440 REM
2450 REM***** PICK A POINT *****
2460 LET X=FNR(1)
2470 LET Y=FNR(1)
2480 LET X1=FNS(X)

```

```

2490 LET Y1=FNS(Y)
2500 LET X1=X1-100
2510 LET Y1=Y1-100
2520 CALL (1,-1,1,X1,Y1)
2530 CALL (2,200,0,0,200)
2540 PRINT "X"
2550 RETURN
2560 REM ***** SCORING SUBROUTINE *****
2570 IF Z9 THEN 2590
2580 PRINT "YOUR SCORE IS ";Z*8.33333;"%. GOOD WORK."
2590 PRINT
2600 CALL (1,-1,1,0,0)
2610 CALL (2,200,0,0,200)
2620 PRINT F;Z
2630 CALL (1,-1,1,9999,0)
2640 RETURN
2650 REM A SUBROUTINE TO DRAW A LINE SEGMENT FROM
2660 REM (X1,Y1) TO (X2,Y2)
2670 LET X1=FNS(X1)
2680 LET X2=FNS(X2)
2690 LET Y1=FNS(Y1)
2700 LET Y2=FNS(Y2)
2710 CALL (1,- 1,1,X1,Y1)
2720 CALL (1,1,1,X2,Y2)
2730 RETURN
2740 REM SUBROUTINE TO DRAW AND MARK X & Y AXES
2750 FOR W=1 TO 2
2760 READ X1,Y1,X2,Y2
2770 DATA -9,0,9,0,0,9,0,-9
2780 GOSUB 2670
2790 NEXT W
2800 FOR X=-9 TO 9
2810 LET X1=X2=X
2820 LET Y1=0
2830 LET Y2=5
2840 GOSUB 2670
2850 NEXT X
2860 FOR Y=-9 TO 9
2870 LET Y2=Y1=Y
2880 LET X1=0
2890 LET X2=.5
2900 GOSUB 2670
2910 NEXT Y
2920 RESTORE
2930 RETURN
2940 END

```

```

5 REM PROGRAM TO DRAW STEREOGRAMS
10 DIM U[10,200]
20 CALL (3,U[1,1],4000)
30 DIM E[6],X[2],Y[2]
40 LET S1=-10
50 LET S=500
60 DEF FNS(A)=(A+8)*S
70 CALL (50,-1)
80 DIM A[100,3]B[100,3]
90 DIM V[3,3],W[3,3]
100 DIM T[3,3],R[3,3],S[3,3],Q[3,3]
110 MAT T=IDN
120 LET T[2,1]=SIN(3,14159/12)
130 LET T[1,2]=-T[2,1]
140 LET T[1,1]=T[2,2]=COS(3,14159/12)
150 MAT R=TRN(T)
160 MAT S=IDN
170 LET S[1,1]=S[3,3]=T[1,1]
180 LET S[1,3]=T[1,2]
190 LET S[3,1]=T[2,1]
200 MAT(S)
210 MAT V=IDN
220 LET V[2,2]=V[3,3]=T[1,1]
230 LET V[3,2]=T[2,1]
240 LET V[2,3]=T[1,2]
250 MAT W=TRN(V)
260 FOR I=1 TO 6
270 READ E[I]
280 NEXT I
290 READ L1
300 LET D1=2*L1
310 MAT B=ZER[D1,3]
320 MAT P=ZER[D1,2]
330 MAT READ A[D1,3]
340 MAT B=A
350 LET M=1
360 MAT A=(M)*A
370 REM
380 FOR K=1 TO D1
390 LET T=(S1-E[3])/(E[3]-A[K,3])
400 LET P[K,1]=E[1]+T*(E[1]-A[K,1])
410 LET P[K,2]=E[2]+T*(E[2]-A[K,2])
420 NEXT K
430 GOSUB 1050
440 PRINT "YOU NOW HAVE EIGHT CHOICES:"
450 PRINT " 1 ROTATE CLOCKWISE ABOUT Z AXIS"

```

```

460 PRINT " 2 ROTATE COUNTER-CLOCKWISE ABOUT Z AXIS"
470 PRINT " 3 ROTATE CLOCKWISE ABOUT THE Y AXIS"
480 PRINT " 4 ROTATE COUNTER-CLOCKWISE ABOUT Y AXIS"
490 PRINT " 5 ROTATE CLOCKWISE ABOUT X AXIS"
500 PRINT " 6 ROTATE COUNTER-CLOCKWISE ABOUT X AXIS"
510 PRINT " 7 DRAW STEREOGRAM ON PLOTTER"
520 PRINT " 8 QUIT"
530 PRINT
540 PRINT " WHAT IS YOUR CHOICE";
550 INPUT Q
560 CALL (3.U[1,1],4000)
570 IF Q=1 THEN 670
580 IF Q=2 THEN 700
590 IF Q=3 THEN 760
600 IF Q=4 THEN 730
610 IF Q=5 THEN 790
620 IF Q=6 THEN 820
625 REM
626 REM
630 IF Q=7 THEN 850
640 IF Q=8 THEN 1280
650 PRINT "1,2,3,4,5,6,7, OR 8 ARE THE ONLY CHOICES."
660 GOTO 540
670 MAT B=A*T
680 MAT A=B
690 GOTO 380
700 MAT B=A*R
710 MAT A=B
720 GOTO 370
730 MAT B=A*S
740 MAT A=B
750 GOTO 370
760 MAT R=A*Q
770 MAT A=B
780 GOTO 370
790 MAT B=A*V
800 MAT A=B
810 GOTO 370
820 MAT B=A*W
830 MAT A=B
840 GOTO 370
850 CALL (50,1)
860 PRINT "IS PLOTTER TURNED ON?"
870 PRINT "ORIENTED FOR RIGHT EYE";
880 INPUT I
890 GOSUB 1050

```

```

900 FOR K=1 TO D1
910 LET T=(S1-E[6])/(E[6]-A[K,3])
920 LET P[K,1]=E[4]+T*(E[4]-A[K,1])
930 LET P[K,2]=E[5]+T*(E[5]-R[K,2])
940 NEXT K
950 PRINT "PLOTTER ORIENTATED FOR LEFT EYE";
960 INPUT I
970 GOSUB 1050
980 CALL (50,-1)
990 PRINT
1000 PRINT "YOU MAY NOT PLOT A DUPLICATE OF THE"
1010 PRINT "STEREOGRAM. THE RIGHT EYE IMAGE HAS BEEN"
1020 PRINT "WRITTEN OVER. MAKE ANOTHER CHOICE AND "
1030 PRINT "REVERSE IT BEFORE MAKING A NEW PLOT."
1040 GOTO 440
1050 FOR K=1 TO D1 STEP 2
1060 LET X1=FNS(P[K,1])
1070 LET Y1=FNS(P[K,2])
1080 LET X2=FNS(P[K+1,1])
1090 LET Y2=FNS(P[K+1,2])
1100 CALL (1,-1,1,X1,Y1)
1110 CALL (1,1,1,X2,Y2)
1120 CALL (1,-1,1,X2,Y2)
1130 NEXT K
1140 RETURN
1150 DATA .25,.5,3,-.25,.5,3
1160 DATA 11
1170 DATA .5,0,.5,.5,0,-.5
1180 DATA .5,0,-.5,-.5,0,-.5
1190 DATA -.5,0,-.5,-.5,0,.5
1200 DATA -.5,0,.5,.5,0,.5
1210 DATA 0,.707,0,.5,0,.5
1220 DATA 0,.707,0,.5,0,-.5
1230 DATA 0,.707,0,-.5,0,.5
1250 DATA 0,.707,0,-.5,0,-.5
1240 DATA 0,.707,0,-.5,0,.5
1250 DATA 0,.707,0,1,.707,0
1260 DATA .5,0,.5,1,.707,0
1270 DATA .5,0,-.5,1,.707,0
1280 END

```

(figure available in print form)

(figure available in print form)

(figure available in print form)

(figure available in print form)

<https://teachersinstitute.yale.edu>

©2019 by the Yale-New Haven Teachers Institute, Yale University

For terms of use visit <https://teachersinstitute.yale.edu/terms>