

# APPENDIX

```

10 SCREEN 0:WIDTH 80:CLS:KEY OFF
20 PRINT "*****"
30 PRINT"*
40 PRINT"*
50 PRINT"*
60 PRINT"*
70 PRINT"*
80 PRINT"*
90 PRINT"*
100 PRINT"*****"
110 '***** Main Program *****"
120 PRINT:PRINT
130 PRINT:PRINT"          OUTPUT OPTIONS":PRINT
140 PRINT:PRINT"          1. PRINT FINAL RESULTS ONLY"
150 PRINT:PRINT"          2. PRINT STEP-BY-STEP RESULTS"
160 PRINT:PRINT:INPUT"ENTER YOUR CHOICE(1-2)"; RQ
170 IF RQ<1 OR RQ>2 THEN GOTO 160
180 GOSUB 990
190 GOSUB 2050
200 PRINT CHR$(12)
210 FQ=1:EPS=.001:N=NFC
220 FOR ITER=1 TO 30
230 PRINT:PRINT"INTEGRATING. PLEASE WAIT"
240 GOSUB 2650
250 IF RQ=1 THEN PRINT "ITERATION=";ITER
260 IF RQ=2 THEN PRINT CHR$(12);"ITERATION=";ITER
270 IF RQ=2 THEN GOSUB 3060
280 'Construct the Jacobian matrix and the DELTAY vector
290 K=NSE+NIC*NFC
300 IF RQ=2 THEN PRINT:PRINT"JACOBIAN MATRIX:":PRINT
310 FOR I=1 TO NFC
320 FOR J=1 TO NFC
330 K=K+1
340 JAC(I,J)=A(L,K)
350 AA(I,J)=JAC(I,J) : BB(I,J)=JAC(I,J)
360 IF RQ=2 THEN PRINT JAC(I,J),
370 NEXT J
380 IF RQ=2 THEN PRINT
390 DELTAY(I)=- (A(L,NIC+I)-YF(NIC+I))
400 NEXT I
410 'Invert the Jacobian
420 GOSUB 1590
430 IF RQ=2 THEN PRINT:PRINT"INVERSE OF JACOBIAN MATRIX:":PRINT
440 FOR I=1 TO NFC
450 FOR J=1 TO NFC
460 JACINV(I,J)=AA(I,NFC+J)
470 IF RQ=2 THEN PRINT JACINV(I,J),
480 NEXT J
490 IF RQ=2 THEN PRINT
500 NEXT I
510 'Calculate the correction vector DELGAMMA
520 IF RQ=2 THEN PRINT
530 FOR I=1 TO NFC
540 SUM=0
550 FOR J=1 TO NFC

```

```

560 SUM=SUM + JACINV(I,J)*DELTAY(J)
570 NEXT J
580 DELGAMMA(I)=SUM
590 IF RQ=2 THEN PRINT"DELTAY(";I;")=";DELTAY(I)
600 IF RQ=2 THEN PRINT"DELGAMMA(";I;")=";DELGAMMA(I)
610 NEXT I
620 'Check for convergence
630 COUNT=0
640 FOR I=1 TO NFC
650 IF YF(NIC+I)=0 THEN CCC=DELTAY(I) ELSE CCC=DELTAY(I)/YF(NIC+I)
660 IF ABS(CCC)<=CONV THEN COUNT=COUNT+1
670 IF COUNT=NFC THEN PRINT "***** CONVERGED *****":GOTO 850
680 NEXT I
690 'Correct the guessed initial conditions(use 1/2 step)
700 FOR I = 1 TO NFC
710 Y9(NIC+I)=Y9(NIC+I)+DELGAMMA(I)/2
720 NEXT I
730 'Reset all initial conditions
740 A(0,0)=X9 : X=X9
750 FOR I=1 TO N1 : Y(I)=Y9(I) : A(0,I)=Y9(I) : NEXT I
760 NEXT ITER
770 PRINT CHR$(12);" ITERATION LIMIT EXCEEDED. RESULTS MAY NOT BE CORRECT."
780 PRINT:PRINT" OPTIONS IN CASE OF NONCONVERGENCE":PRINT
790 PRINT:PRINT" 1. SEE THE NONCONVERGED RESULTS "
800 PRINT:PRINT" 2. RESET INTEGRATION PARAMETERS, AND RECALCULATE"
810 PRINT:PRINT" 3. END PROGRAM"
820 PRINT:PRINT:INPUT" ENTER YOUR CHOICE(1-3)"; NRRQ
830 IF NRRQ<1 OR NRRQ>3 THEN GOTO 820
840 ON NRRQ GOTO 850,950,980
850 PRINT CHR$(12);"ITERATION=";ITER :GOSUB 3060
860 SWAP N1,NSE: GOSUB 3250: SWAP N1,NSE
870 REM GOSUB 9000
880 PRINT:PRINT" RERUN OPTIONS":PRINT
890 PRINT:PRINT" 1. SHOW SAME RESULTS AGAIN"
900 PRINT:PRINT" 2. RESET INTEGRATION PARAMETERS AND RECALCULATE"
910 PRINT:PRINT" 3. END PROGRAM"
920 PRINT:PRINT:INPUT" ENTER YOUR CHOICE(1-3)"; RRQ
930 IF RRQ<1 OR RRQ>3 THEN GOTO 920
940 ON RRQ GOTO 850,950,980
950 PRINT:PRINT:PRINT:PRINT" NEW CONDITIONS"
960 GOSUB 2090 'Enter Subroutine 3 after the dimension statement
970 GOTO 220
980 END
990 '***** Subroutine 1: Input equations *****
1000 '
1010 PRINT CHR$(12)
1020 PRINT:PRINT" DIFFERENTIAL EQUATIONS"
1030 PRINT:PRINT" *****"
1040 PRINT:PRINT"THE SYSTEM AND VARIATIONAL DIFFERENTIAL EQUATIONS, AND OTHER "
1050 PRINT:PRINT"ASSOCIATED ALGEBRAIC EQUATIONS, MAY BE ENTERED HERE."
1060 PRINT:PRINT"EACH LINE-STATEMENT MUST OBEY THE RULES OF BASIC PROGRAMMING."
1070 PRINT:PRINT"THE LINES ARE AUTOMATICALLY NUMBERED, STARTING AT 4011."
1080 PRINT:PRINT"THESE STATEMENTS ARE MERGED INTO THE PROGRAM IN SUBROUTINE 4."
1090 PRINT:PRINT" *****"
1100 PRINT:INPUT "HOW MANY SYSTEM DIFFERENTIAL EQUATIONS "; NSE

```

```

1110 PRINT:INPUT " HOW MANY OF THESE HAVE KNOWN INITIAL CONDITIONS "; NIC
1120 PRINT:INPUT " HOW MANY OF THESE HAVE KNOWN FINAL CONDITIONS "; NFC
1130 IF NFC=NSE-NIC THEN GOTO 1160
1140 PRINT:PRINT"***** ERROR *****"
1150 PRINT"TOTAL NUMBER OF BOUNDARY CONDITIONS MUST BE =";NSE:GOTO 1100
1160 PRINT:INPUT "HOW MANY VARIATIONAL DIFFERENTIAL EQUATIONS "; NVE
1170 IF NVE=NSE*NFC THEN GOTO 1200
1180 PRINT:PRINT"***** ERROR *****"
1190 PRINT"THE NUMBER OF VARIATIONAL EQUATIONS MUST BE =";NSE*NFC:GOTO 1160
1200 N1=NSE+NVE
1210 DIM Y(N1),Y9(N1),YF(N1),Y1(N1),C(50),G(N1),B1(N1),B2(N1),B3(N1),B4(N1)
1220 DIM JAC(NFC,NFC),JACINV(NFC,NFC),DELTAY(NFC),DELGAMMA(NFC)
1230 DIM AA(NFC,2*NFC),BB(NFC,NFC),CC(NFC,NFC)
1240 PRINT:INPUT "HAVE YOU PREVIOUSLY ENTERED THE EQUATIONS(Y/N)";Q1$
1250 IF Q1$="Y" OR Q1$="y" THEN GOTO 1280
1260 IF Q1$="N" OR Q1$="n" THEN GOTO 1320
1270 GOTO 1240
1280 PRINT:PRINT"THE FOLLOWING EQUATION-FILES EXIST ON YOUR DISK:":PRINT
1290 FILES "*.EQU"
1300 INPUT"GIVE NAME OF FILE TO BE USED (DO NOT ENTER THE EXTENSION): ",FIL$
1310 FIL$=FIL$+".EQU":GOTO 1560
1320 PRINT:INPUT"NAME OF FILE FOR STORING THE DIFFERENTIAL EQUATIONS: ",FIL$
1330 FIL$=FIL$+".EQU"
1340 OPEN FIL$ FOR OUTPUT AS #1
1350 CLS:PRINT:PRINT"*****"
1360 PRINT:PRINT"ENTER THE DIFFERENTIAL EQUATIONS IN THE FORM SHOWN BELOW:"
1370 PRINT:PRINT" G(i)= f[x, y(1), y(2),...,y(n)]"
1380 PRINT:PRINT"THE EQUATIONS MUST BE NUMBERED IN THE FOLLOWING ORDER:"
1390 PRINT:PRINT" First: System equations with known initial conditions"
1400 PRINT:PRINT" Second: System equations with unknown initial conditions"
1410 PRINT:PRINT" Last: Variational equations in the exact order as shown"
1420 PRINT" in this example"
1430 PRINT:PRINT"*****":PRINT
1440 INPUT"GIVE THE EXACT NUMBER OF LINES OF INPUT YOU WILL ENTER(1-980)";LN1
1450 PRINT:PRINT"SEPARATE EACH STATEMENT WITH THE ENTER(RETURN) KEY."
1460 PRINT#1,"4010 'The differential equations are merged in lines 4011-4990
1470 FOR I=1 TO LN1
1480 PRINT:PRINT " ";4010+I;
1490 LINE INPUT ;EQUATION$
1500 IF EQUATION$="" THEN GOTO 1480
1510 PRINT#1,4010+I; EQUATION$
1520 PRINT
1530 NEXT I
1540 PRINT#1,"4995 'End of space allocated for the differential equations
1550 CLOSE #1
1560 PRINT CHR$(12)
1570 CHAIN MERGE FIL$,180,ALL,DELETE 2560-2630
1580 RETURN
1590 '***** Subroutine 2: Matrix inversion *****
1600 'Augment with the identity matrix
1610 FOR I = 1 TO N
1620 FOR J = N+1 TO 2*N
1630 AA(I,J) = 0
1640 NEXT J
1650 AA(I,I+N) = 1

```

```

1660 NEXT I
1670 'Beginning of the Gauss-Jordan reduction procedure
1680 FOR K = 1 TO N
1690 'Apply partial pivoting strategy
1700 MAXPIVOT = ABS(AA(K,K)):NPIVOT=K
1710 FOR I = K TO N
1720 IF MAXPIVOT > = ABS(AA(I,K)) GOTO 1740
1730 MAXPIVOT=ABS(AA(I,K)) : NPIVOT=I
1740 NEXT I
1750 IF MAXPIVOT > = EPS GOTO 1770
1760 PRINT"CAUTION: PIVOT ELEMENT SMALLER THAN";EPS;". MATRIX MAY BE SINGULAR."
1770 IF NPIVOT = K GOTO 1810
1780 FOR J=K TO 2*N
1790 SWAP AA(NPIVOT,J),AA(K,J)
1800 NEXT J
1810 DD=AA(K,K)
1820 FOR J = 2*N TO K STEP -1
1830 AA(K,J) = AA(K,J)/DD
1840 NEXT J
1850 FOR I = 1 TO N
1860 IF I=K GOTO 1910
1870 MULT=AA(I,K)
1880 FOR J = 2*N TO K STEP -1
1890 AA(I,J) = AA(I,J) - MULT* AA(K,J)
1900 NEXT J
1910 NEXT I:NEXT K
1920 'Check the product of matrix and inverse
1930 FOR I = 1 TO N
1940 FOR J = 1 TO N
1950 CC(I,J)=0
1960 FOR K = 1 TO N
1970 CC(I,J)=CC(I,J) + BB(I,K)*AA(K,N+J)
1980 NEXT K
1990 IF I=J AND ABS(CC(I,J)-1)<EPS THEN GOTO 2020
2000 IF I<>J AND ABS(CC(I,J))<EPS THEN GOTO 2020
2010 PRINT "CAUTION: INVERSE MAY BE INCORRECT."
2020 NEXT J
2030 NEXT I
2040 RETURN
2050 '***** Subroutine 3: Input integration parameters *****
2060 PRINT:PRINT "          INTEGRATION PARAMETERS"
2070 PRINT:PRINT "NUMBER OF INTEGRATION STEPS:";: INPUT L
2080 DIM A(L,N1)
2090 PRINT:PRINT "INITIAL VALUE OF INDEPENDENT VARIABLE:";: INPUT X9
2100 PRINT:PRINT "FINAL VALUE OF INDEPENDENT VARIABLE: ";: INPUT XF
2110 D=(XF-X9)/L : X = X9 : A(0,0) = X9
2120 PRINT:PRINT "INCREMENT IN INDEPENDENT VARIABLE IS="; D
2130 PRINT:PRINT"ENTER KNOWN INITIAL CONDITIONS:"
2140 FOR K = 1 TO NIC
2150 PRINT:PRINT "INITIAL Y(";K;")";: INPUT Y9(K)
2160 Y(K) = Y9(K)
2170 A(0,K) = Y9(K)
2180 NEXT K
2190 PRINT:PRINT"ENTER KNOWN FINAL CONDITIONS:"
2200 FOR K = NIC+1 TO NSE

```

```

2210 PRINT:PRINT "FINAL Y(";K;")";: INPUT YF(K)
2220 NEXT K
2230 PRINT:PRINT"ENTER GUESSED INITIAL CONDITIONS:"
2240 FOR K = NIC+1 TO NSE
2250 PRINT:PRINT "INITIAL Y(";K;")";: INPUT Y9(K)
2260 Y(K) = Y9(K)
2270 A(O,K) = Y9(K)
2280 NEXT K
2290 PRINT
2300 INPUT"GIVE THE CONVERGENCE CRITERION FOR THE BOUNDARY CONDITIONS";CONV
2310 PRINT
2320 'Set the initial conditions for the variational equations
2330 PRINT"INITIAL CONDITIONS FOR VARIATIONAL EQUATIONS ARE SET AS FOLLOWS:"
2340 K=NSE
2350 FOR I=1 TO NSE
2360 FOR J=NIC+1 TO NSE
2370 K=K+1
2380 Y9(K)=0
2390 IF I=J THEN Y9(K)=1
2400 Y(K)=Y9(K) : A(O,K)=Y9(K)
2410 PRINT:PRINT "INITIAL Y(";K;")";Y9(K)
2420 NEXT J
2430 NEXT I
2440 PRINT
2450 PRINT"PLEASE CHECK THE ABOVE TO MAKE SURE THAT THEY CORRESPOND TO YOUR"
2460 PRINT"VARIATIONAL EQUATIONS.":PRINT
2470 PRINT"DO YOU WANT TO ENTER CONSTANTS IN THE DIFFERENTIAL EQUATIONS(Y/N)";
2480 INPUT VC# : IF VC#="N" OR VC#="n" THEN GOTO 2540
2490 PRINT:INPUT"HOW MANY CONSTANTS";CO
2500 PRINT:PRINT "GIVE VALUES OF CONSTANTS":PRINT
2510 FOR K = 1 TO CO
2520 PRINT "CONSTANT ";K;: INPUT C(K)
2530 NEXT K
2540 RETURN
2550 '***** Subroutine 4: Differential equations *****
2560 'The differential equations are merged in lines 4011-4990
2570 G(1)=Y(2)
2580 G(2)=Y(3)
2590 G(3)=(-4.5/X)*Y(3) - (2.0665/X^2)*Y(2)
2600 G(4)=Y(5)
2610 G(5)=Y(6)
2620 G(3)=(-4.5/X)*Y(6) - (2.0665/X^2)*Y(5)
2630 'End of space allocated for the differential equations
2640 RETURN
2650 '***** Subroutine 5: Integration methods *****
2660 '
2670 ON FQ GOTO 2680, 2910
2680 'Runge-Kutta 4th order method
2690 FOR J2 = 1 TO L
2700 FOR K2 = 1 TO N1 : Y1(K2) = Y(K2) : NEXT K2
2710 X1 = X : GOSUB 2550
2720 FOR K3 = 1 TO N1
2730 B1(K3) = G(K3):Y(K3) = Y1(K3) + D * G(K3) / 2
2740 NEXT K3
2750 X = X1 + D / 2 : GOSUB 2550

```

```

2760 FOR K4 = 1 TO N1
2770 B2(K4) = G(K4):Y(K4) = Y1(K4) + D * G(K4) / 2
2780 NEXT K4
2790 GOSUB 2550
2800 FOR K5 = 1 TO N1
2810 B3(K5) = G(K5):Y(K5) = Y1(K5) + D * G(K5)
2820 NEXT K5
2830 X = X1 + D : GOSUB 2550
2840 FOR K6 = 1 TO N1
2850 Y(K6) = Y1(K6) + D * (B1(K6) + G(K6) + 2 * (B2(K6) + B3(K6))) / 6
2860 A(J2,K6) = Y(K6)
2870 NEXT K6
2880 A(J2,0) = X
2890 NEXT J2
2900 RETURN
2910 'Euler predictor-corrector method
2920 FOR J2 = 1 TO L
2930 FOR K2 = 1 TO N1 : Y1(K2) = Y(K2) : NEXT K2
2940 X1 = X : GOSUB 2550
2950 FOR K3 = 1 TO N1
2960 Y(K3) = Y1(K3) + D * G(K3)
2970 NEXT K3
2980 X = X1 + D : GOSUB 2550
2990 FOR K6 = 1 TO N1
3000 Y(K6) = Y1(K6) + D * G(K6)
3010 A(J2,K6) = Y(K6)
3020 NEXT K6
3030 A(J2,0) = X
3040 NEXT J2
3050 RETURN
3060 '***** Subroutine 6: Print table of results *****
3070 '
3080 PRINT
3090 PRINT "          TABLE OF RESULTS"
3100 PRINT:PRINT "INDEPENDENT          DEPENDENT VARIABLES"
3110 ST=INT(L/10) 'Prints every ST(th) point
3120 FOR J2 = 0 TO L STEP ST
3130 FOR J1 = 0 TO 3 : PRINT A(J2,J1), " : NEXT J1
3140 PRINT
3150 NEXT J2
3160 INPUT "give the name of data file,no ext";DFIL$
3170 DFIL$=DFIL$+".dat"
3180 OPEN DFIL$ FOR OUTPUT AS #2
3190 FOR J2=0 TO L STEP ST
3200 FOR J1=0 TO 3 : PRINT #2,A(J2,J1),
3210 NEXT J1,J2
3220 CLOSE #2
3230 ' PRINT:INPUT "PRESS THE ENTER KEY TO CONTINUE",KK$
3240 RETURN
3250 '***** Subroutine 7: Plotting options *****
3260 '
3270 PRINT CHR$(12):IF N1=1 THEN GOTO 3550
3280 PRINT:PRINT "          PLOTTING OPTIONS"
3290 PRINT:PRINT "          1. PLOT EACH VARIABLE SEPARATELY"
3300 PRINT:PRINT "          2. PLOT ALL VARIABLES ON SAME GRAPH"

```

```

3310 PRINT:INPUT"ENTER YOUR CHOICE(1-2)"; G1
3320 IF G1<1 OR G1>2 THEN GOTO 3310
3330 ON G1 GOTO 3340,3550
3340 FOR IP=1 TO N1
3350 NS=IP : NF=IP
3360 YTIT$="Y"+STR$(IP) : XTIT$="X"
3370 PRINT:PRINT
3380 PRINT:PRINT"          SCALING OPTIONS "
3390 PRINT:PRINT"      1. MANUAL CHOICE OF MAXIMA AND MINIMA"
3400 PRINT:PRINT"      2. AUTOMATIC CHOICE OF MAXIMA AND MINIMA"
3410 PRINT:INPUT"ENTER YOUR CHOICE(1-2)"; G2
3420 IF G2<1 OR G2>2 THEN GOTO 3410
3430 IF G2=2 THEN PRINT CHR$(12): GOSUB 3740: GOTO 3500
3440 PRINT:PRINT"          SCALES FOR VARIABLE ";IP
3450 PRINT:INPUT"MINIMUM X=";MINX :PRINT:INPUT"MAXIMUM X=";MAXX
3460 PRINT:LINE INPUT"SPECIFY X-AXIS TITLE:",XTIT$
3470 PRINT:INPUT"MINIMUM Y=";MINY :PRINT:INPUT"MAXIMUM Y=";MAXY
3480 PRINT:LINE INPUT"SPECIFY Y-AXIS TITLE:",YTIT$
3490 PRINT CHR$(12): GOSUB 4020
3500 FOR PAUSE=1 TO 10000:NEXT PAUSE
3510 LOCATE 25,1:INPUT"PRESS THE ENTER KEY TO SEE NEXT GRAPH",KK$
3520 SCREEN 0:WIDTH 80:PRINT CHR$(12)
3530 NEXT IP
3540 GOTO 3730
3550 NS=1: NF=N1
3560 YTIT$="Y" : XTIT$="X"
3570 PRINT:PRINT
3580 PRINT:PRINT"          SCALING OPTIONS "
3590 PRINT:PRINT"      1. MANUAL CHOICE OF MAXIMA AND MINIMA"
3600 PRINT:PRINT"      2. AUTOMATIC CHOICE OF MAXIMA AND MINIMA"
3610 PRINT:INPUT"ENTER YOUR CHOICE(1-2)"; G2
3620 IF G2<1 OR G2>2 THEN GOTO 3610
3630 IF G2=2 THEN PRINT CHR$(12): GOSUB 3740: GOTO 3700
3640 PRINT:PRINT"          SCALES FOR ALL VARIABLES"
3650 PRINT:INPUT"MINIMUM X=";MINX :PRINT:INPUT"MAXIMUM X=";MAXX
3660 PRINT:LINE INPUT"SPECIFY X-AXIS TITLE:",XTIT$
3670 PRINT:INPUT"MINIMUM Y=";MINY :PRINT:INPUT"MAXIMUM Y=";MAXY
3680 PRINT:LINE INPUT"SPECIFY Y-AXIS TITLE:",YTIT$
3690 GOSUB 4020
3700 FOR PAUSE=1 TO 10000:NEXT PAUSE
3710 LOCATE 25,1:INPUT"PRESS THE ENTER KEY TO CONTINUE",KK$
3720 SCREEN 0:WIDTH 80:PRINT CHR$(12)
3730 RETURN
3740 '***** Subroutine 8: Plotting *****
3750 '
3760 '
3770 'Locate maxima and minima for scaling axes
3780 MINX=A(0,0):MAXX=A(0,0):MINY =A(0,NS):MAXY =A(0,NS)
3790 FOR K = 0 TO L
3800 IF A(K ,0) < MINX THEN MINX = A(K ,0)
3810 IF A(K ,0) > MAXX THEN MAXX = A(K ,0)
3820 FOR J = NS TO NF
3830 IF A(K ,J) < MINY THEN MINY = A(K ,J)
3840 IF A(K ,J) > MAXY THEN MAXY = A(K ,J)
3850 NEXT J:NEXT K

```



```

3850 IF MAXX=0 GOTO 3900
3870 FACT=INT(LOG(ABS(MAXX))/LOG(10))
3880 TEMP=MAXX/(10^FACT)
3890 MAXX=(INT(TEMP+.999))*(10^FACT)
3900 IF MINX=0 GOTO 3940
3910 FACT=INT(LOG(ABS(MINX))/LOG(10))
3920 TEMP=MINX/(10^FACT)
3930 MINX=(INT(TEMP))*(10^FACT)
3940 IF MAXY=0 GOTO 3980
3950 FACT=INT(LOG(ABS(MAXY))/LOG(10))
3960 TEMP=MAXY/(10^FACT)
3970 MAXY=(INT(TEMP+.999))*(10^FACT)
3980 IF MINY=0 GOTO 4020
3990 FACT=INT(LOG(ABS(MINY))/LOG(10))
4000 TEMP=MINY/(10^FACT)
4010 MINY=(INT(TEMP))*(10^FACT)
4020 CLS:SCREEN 1
4030 VIEW (56,1)-(296,161)
4040 IF MINY=MAXY THEN MINY=MINY/2: MAXY=2*MAXY
4050 IF MINY=MAXY AND MINY=0 THEN MINY=-1:MAXY=1
4060 WINDOW (MINX,MINY)-(MAXX,MAXY)
4070 'Draw the axes
4080 DX=(MAXX-MINX)/10: DY=(MAXY-MINY)/10
4090 IF MINY/MAXY > 0 THEN YZ=MINY ELSE YZ=0
4100 X=MINX : PSET(X,YZ)
4110 FOR I=1 TO 10
4120 X=X+DX : YT=YZ + ABS(DY/10)
4130 LINE -(X,YZ) : LINE -(X,YT) : LINE -(X,YZ)
4140 NEXT I
4150 IF MINX/MAXX > 0 THEN XZ=MINX ELSE XZ=0
4160 Y=MINY
4170 PSET(XZ,Y)
4180 FOR I=1 TO 10
4190 Y=Y+DY : XT=XZ + ABS(DX/10)
4200 LINE -(XZ,Y) : LINE -(XT,Y) : LINE -(XZ,Y)
4210 NEXT I
4220 'Label the axes
4230 DELX=(MAXX-MINX) : DELY=(MAXY-MINY)
4240 FOR I=0 TO 10 STEP 2
4250 LBX=MINX+I*DELX/10 : LBX=INT(LBX*100)/100
4260 LN=LEN(STR$(LBX))
4270 IF I=10 THEN LOCATE 22,40-LN:PRINT LBX:GOTO 4290
4280 LOCATE 22,6+I*3:PRINT LBX
4290 NEXT I
4300 FOR I=0 TO 10
4310 LBY=MINY+I*DELY/10 : LBY=INT(LBY*100)/100
4320 LN=LEN(STR$(LBY))
4330 IF LN>=7 THEN LN=LN-1 : GOTO 4330
4340 LOCATE 21-I*2,7-LN:PRINT LBY
4350 NEXT I
4360 'Center vertical axis title
4370 IF LEN(YTIT$)>16 GOTO 4430
4380 V1=3+(14-LEN(YTIT$))/2
4390 FOR I=1 TO LEN(YTIT$)
4400 LOCATE V1+I,1:PRINT MID$(YTIT$,I,1)

```

```
4410 NEXT I
4420 GOTO 4460
4430 FOR I=1 TO 16
4440 LOCATE 3+I,2:PRINT MID$(YTIT$,I,1)
4450 NEXT I
4460 'Center horizontal title
4470 IF LEN(XTIT$)>35 GOTO 4510
4480 LOCATE 23,5+(36-LEN(XTIT$))/2
4490 PRINT XTIT$
4500 GOTO 4520
4510 LOCATE 23,5:PRINT MID$(XTIT$,1,36)
4520 'Draw the function
4530 FOR J=NS TO NF
4540 X=A(0,0):Y=A(0,J)
4550 PSET(X,Y)
4560 FOR I=1 TO L
4570 X=A(I,0) :Y=A(I,J)
4580 LINE -(X,Y)
4590 NEXT I
4600 NEXT J
4610 RETURN
```