

Graphics Toolkit

TALENT has been setting the standard of QL software for more than a year. Its major successes include **GraphiQL**, **Cartridge Doctor**, **Akul** and **West**. In the summer of 1985 the company's talented programming team was set the task of producing a professional quality utility for the *Sinclair User Annual*.

The **Talent Graphics Toolkit** is the result of two months hard work. You can use it to experiment with the new science of Fractals, which deals with the replication of natural shapes to form landscapes, rock formations and even buildings. The techniques are similar to those used by Walt Disney Productions to create *Tron*. Enter a new world of computer generated graphics and a new area of programming.

FRACTALS were defined by Mandelbrot who conceived and developed 'a new geometry of nature'. Natural shapes, such as clouds, mountains, trees, coastlines, exhibit a totally different level of complexity to that of standard geometric shapes. Fractals describe many of the fragmented and irregular patterns found in nature and the degree of that irregularity is identical at all levels. For example, a coastline can be represented on a large-scale map as a series of bays or inlets. A more detailed map will reveal that the bays are themselves broken up into smaller bays and indentations.

A simple example of a fractal shape is the snowflake shown in figure one. The basic frame is an equilateral triangle (A). The next shape (B) has been formed by adding equilateral triangles, one third of the size of the original, on the central third of each side. The same process is repeated in (C) and can be repeated ad infinitum until the detail is too fine to see.

TALENT's fractal program allows you to choose one of twenty stored frame shapes or to define your own. The segment, or method by which each line is to be modified, can also be selected from a library of twenty stored shapes or can be user-defined. Finally, the computer draws the frame on screen at any level or generation of complexity you choose.

Program Design

The program has been carefully engineered. The interface should allow you to obtain pleasing results quickly and easily. You should not need to understand the basic mathematical concepts. All help is provided on-screen and no manual is necessary.

Main Menu — nearly always on

screen. Options are highlighted in turn by using the cursor keys and selected with Space. When an option is chosen, a drop-down sub-menu is displayed.

Sub-Menu — brought down when an option is chosen. Options are selected in turn with the Space key. All sub-menus have an ESCape option

which returns you to the main menu.

Segment Window — always on screen. This window displays the current segment selected out of a library of 20. The first 10 of those are defined. The remainder are stored as straight lines which can be picked up

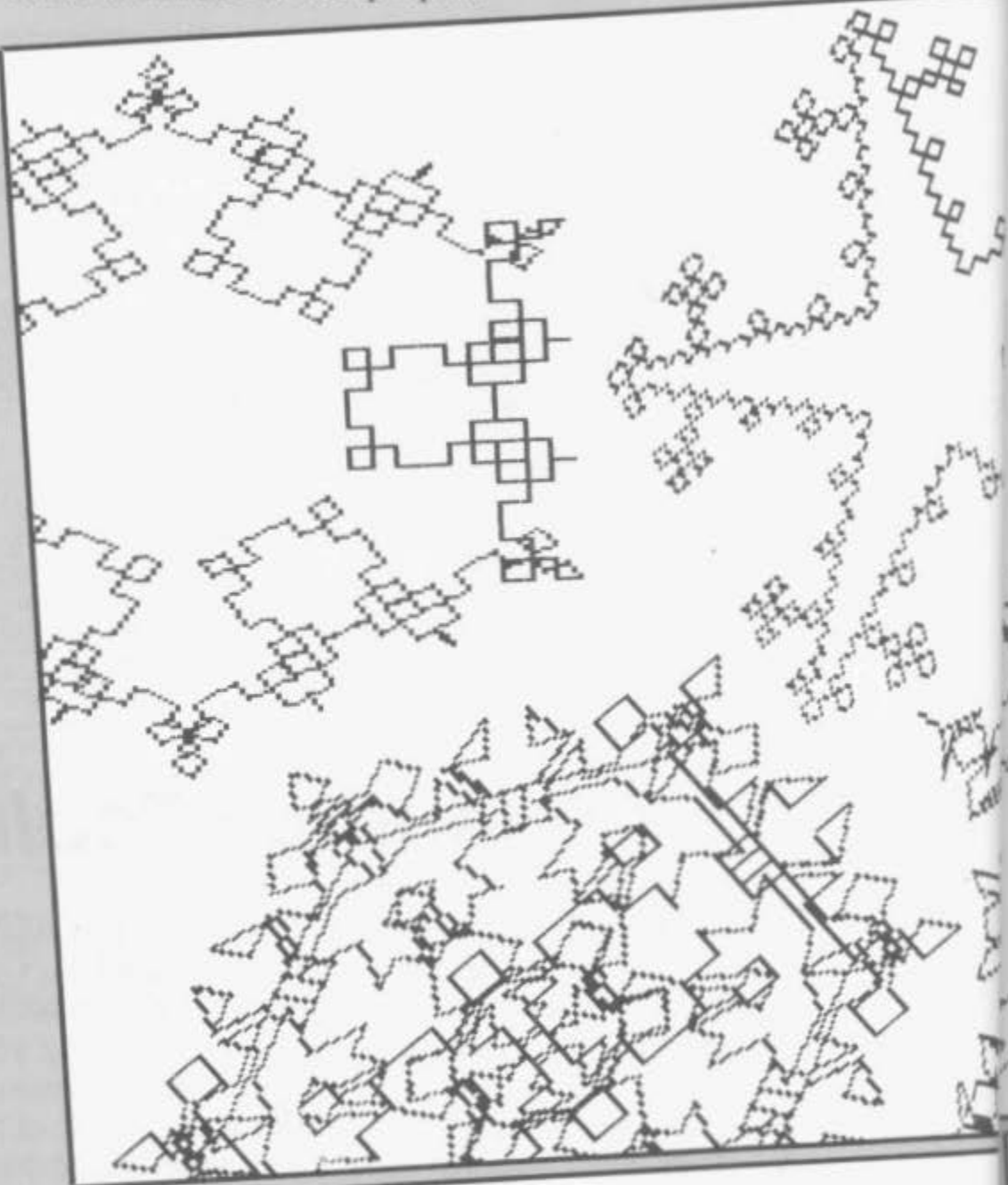
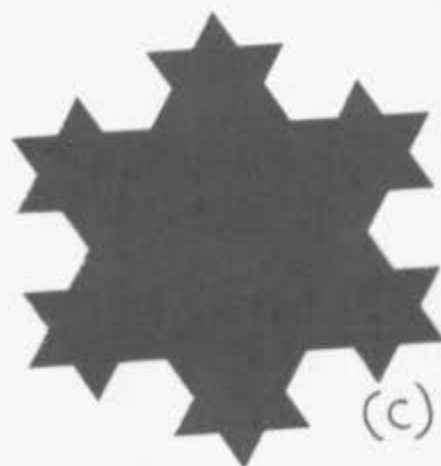
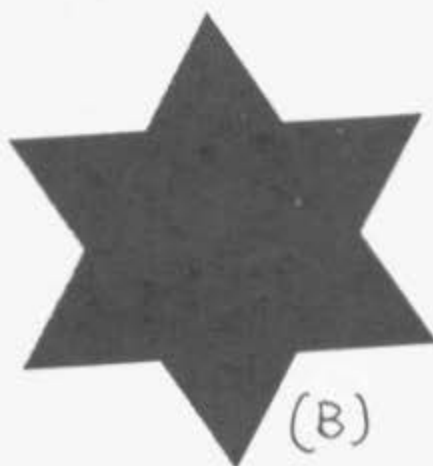
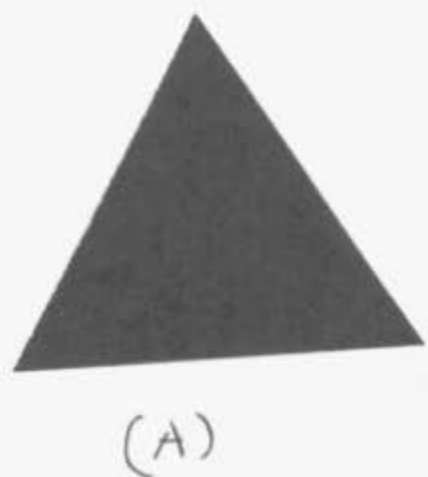


Figure 1. Snowflake fractal



and edited to become any shape you wish.

Frame Window — the same as segment window. Displays the current frame selected from the frame library. The first 10 are defined, the remainder can be edited in the same way as

segments.

Input/Error/Message Window — this window is used when input is required — filenames. It will also display errors and other messages.

Summary of functions

After Segments/Frames

On-screen edit — used to change the segment or frame on-screen. After the segment/frame is drawn, you move the cursor to a point on the line. Then pick up the point and pull it to a new position using the cursor keys. The space bar is used to register the change. That is useful for specifying irregular shapes.

Numerical data — you are asked to input a series of numbers—distance, angle—to specify the segment or frame. That is useful for specifying regular shapes. Distance is scaled proportionately, for example, a standard length for the frame is 50 and for the segment, 10. The angles start at 0 degrees and work clockwise — 90 degrees, 180 degrees, 270 degrees and back to 0 degrees.

Change Segment/Frame

The current segment or frame is changed to another in store — 1-20.

Draw Fractal

When this item is specified on the main menu, the sub-menu will ask for various data which you must enter as numbers on the keyboard. The options are mode, paper, ink, x co-ordinate, y co-ordinate, depth and scale.

Permanent Store

Format device will format a specified device.

Directory device will give a directory of the specified device.

Save fractals will save the present store arrays.

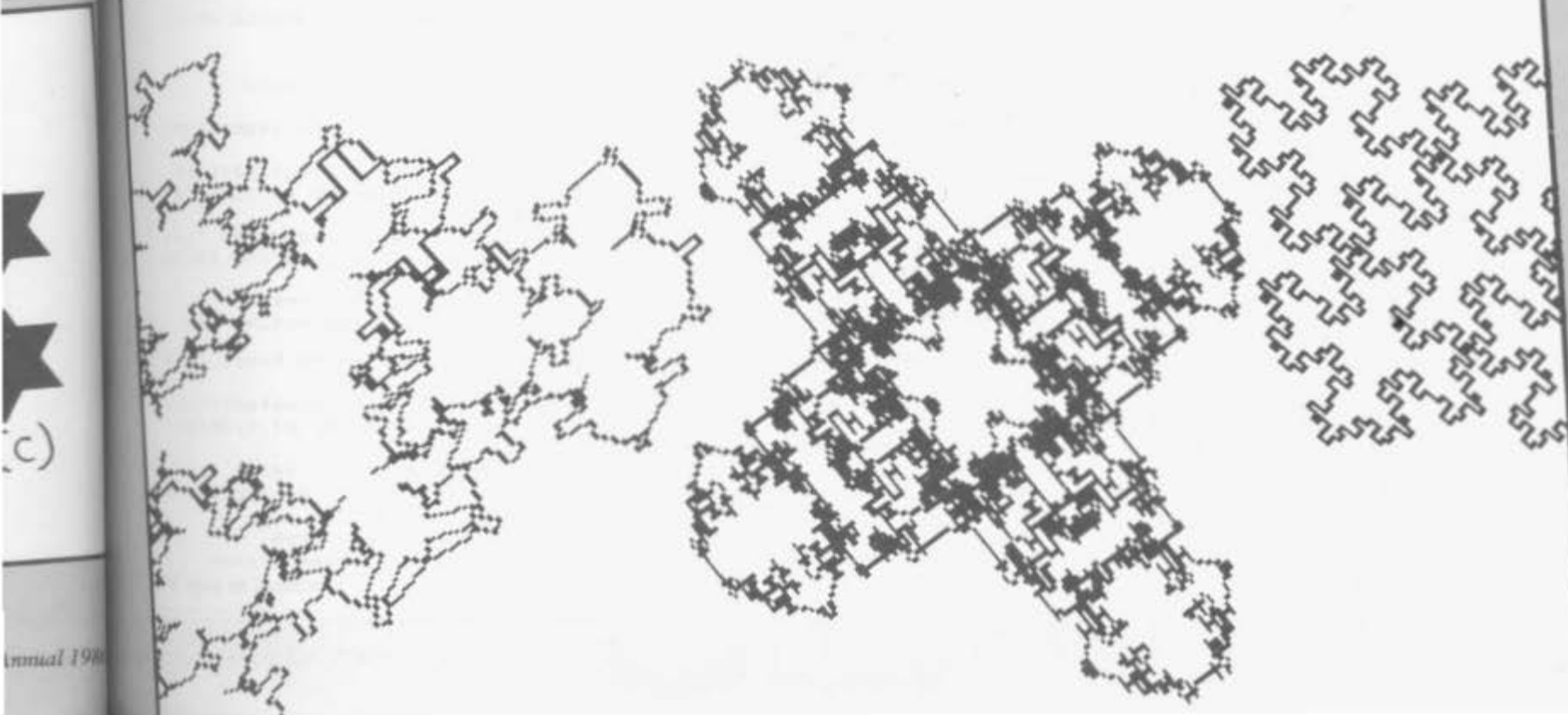
Load fractals will load the store arrays.

Additional information

Occasionally values — paper, ink — must be entered in the input window. The default value — printed in brackets after the option — is printed first. If that is to be changed, it must first be erased using the Sinclair erase sequence of CTRL and back arrow. Then the new value must be typed in, followed by ENTER.

An error message will be generated if a segment is 'closed' — the start and end point are put in the same place. If the start and end point of a segment are too close together, the segment will not fit into the window. It will also run off the main fractal drawing screen. It should be changed using the on-screen edit to make it fit.

Talent Computer Systems will be happy to supply copies of this program on request. Send a formatted microdrive cartridge, name and address and £1.50 handling charge to Talent Computer Systems, Curran Building, 101 St James Road, Glasgow G4 0NS.




```

240 :
1000 FOR a=3 TO 8:OPEN #a,scr:CL
OSE #a
1010 FOR a=0 TO 2:PAPER #a,0:BOR
DER #a,0
1020 MODE 4:INK 7:CSIZE 2,1:PRIN
T "Please wait. Setting fracta
l data."
1030 init_main:main_screen:logo
1040 REPEAT forever
1050 select_option 1,3,1,6,mpos
,1,-1:m_sel=selection
1060 pull_menu m_sel
1070 selection=0:scrn=2
1080 REPEAT sub_menu
1090 sell=selection-1
1100 IF scrn=2 THEN sell=-1
1110 select_option 0,6,0,max,s
election,0,sell
1120 IF selection=0 THEN EXIT
sub_menu
1130 IF m_sel=0
1140 SELECT ON selection
1150 =1
1160 edit_s:scrn=2
1170 =2
1180 get_seg_data:scrn=2
1190 END SELECT
1200 END IF
1210 IF m_sel=1
1220 SELECT ON selection
1230 =1
1240 edit_f:scrn=2
1250 =2
1260 get_frm_data:scrn=2
1270 END SELECT
1280 END IF
1290 IF m_sel=2
1300 SELECT ON selection
1310 =1
1320 choose_seg:scrn=1
1330 =2
1340 g_val pres_seg,2
1350 IF input_$>20 THEN inp
ut_$=pres_seg
1360 pres_seg=input_$
1370 change_seg pres_seg:di
sp_seg pres_seg:scrn=1
1380 END SELECT
1390 END IF
1400 IF m_sel=3
1410 SELECT ON selection
1420 =1
1430 choose_frm:scrn=1
1440 =2
1450 g_val pres_frm,2
1460 IF input_$>20 THEN inp
ut_$=pres_frm
1470 pres_frm=input_$
1480 change_frm pres_frm:di
sp_frm pres_frm:scrn=1
1490 END SELECT
1500 END IF
1510 IF m_sel=4
1520 SELECT ON selection
1530 =1
1540 draw_fractal:scrn=2
1550 =2
1560 IF mode_=4 THEN mode_=
8:ELSE mode_=4
1570 UPDATE_FRACTAL_MENU se
lection,mode_:scrn=0
1580 =3
1590 g_val paper_.3
1600 IF input_$>255 THEN in
put_$=paper_
1610 paper_=input_$
1620 UPDATE_FRACTAL_MENU se
lection,paper_:scrn=1
1630 =4
1640 g_val ink_.3
1650 IF input_$>255 THEN in
put_$=ink_
1660 ink_=input_$
1670 UPDATE_FRACTAL_MENU se
lection,ink_:scrn=1
1680 =5
1690 g_val x_coord,3:x_coord
d=input_$

```

```

1700 UPDATE_FRACTAL_MENU se
lection,x_coord:scrn=1:disp_frm
pres_frm
1710 =6
1720 g_val y_coord,3:y_coord
d=input_$
1730 UPDATE_FRACTAL_MENU se
lection,y_coord:scrn=1:disp_frm
pres_frm
1740 =7
1750 g_val depth_,2:depth_=
input_$
1760 UPDATE_FRACTAL_MENU se
lection,depth_:scrn=1
1770 =8
1780 g_val scale_,3:scale_=
input_$
1790 UPDATE_FRACTAL_MENU se
lection,scale_:scrn=1:disp_frm p
res_frm
1800 END SELECT
1810 END IF
1820 IF m_sel=5
1830 SELECT ON selection
1840 =1
1850 directory:scrn=2
1860 =2
1870 format_device:scrn=1
1880 =3
1890 save_fractals:scrn=1
1900 =4
1910 load_fractals:scrn=1
1920 END SELECT
1930 END IF
1940 IF scrn=1 THEN logo
1950 IF scrn=2 THEN main_scee
n:logo:pull_menu m_sel
1960 END REPEAT sub_menu
1970 push_menu
1980 END REPEAT forever
1990 :
2000 DEFINE PROCEDURE logo
2010 prompt 1,0:prompt 10,1
2020 END DEFINE logo
2030 :
2040 DEFINE PROCEDURE g_val (def
ault,g_len)
2050 command:prompt 8,0:_input
8,0,10,g_len,default,1,1
2060 END DEFINE g_val
2070 :
2080 DEFINE PROCEDURE main_scee
n
2090 OPEN #3,scr_512x256a0x0
2100 PAPER #3,7,0,1:CLS #3:CLOS
E #3
2110 frame_window pres_frm
2120 segment_window pres_seg

```

```

2130 main_menu:command
2140 OPEN #1,scr_433x200a36x18
2150 END DEFINE main_screen
2160 :
2170 DEFINE PROCEDURE pull_menu
(no)
2180 LOCAL a,option,a$,depth
2190 RESTORE no*10+10000
2200 OPEN #5,scr_200x92a36x120
2210 OPEN #6,scr_200x4a36x131
2220 PAPER #5,2:INK #5,0
2230 PAPER #6,0:INK #6,7:CSIZE
#6,1,0
2240 depth=4:READ title$
2250 AT #5,0,0:CLS #5,3:CSIZE #
5,2,0:PRINT #5:title$
2260 READ option
2270 FOR a=1 TO option
2280 depth=depth+10
2290 WINDOW #6,200,depth,36,12
9:BORDER #6,2,2
2300 AT #6,a-1,0:CLS #6,3:SCRO
LL #6,10
2310 READ a$:a$=" "a$
2320 AT #6,0,0:PRINT #6:a$
2330 END FOR a
2340 max=option
2350 END DEFINE pull_menu
2360 :
2370 DEFINE PROCEDURE push_menu
2380 LOCAL a
2390 OPEN #6,scr_200x92a36x131:
PAPER #6,7,0,1
2400 FOR a=1 TO 10:SCROLL #6,-1
0
2410 CLOSE #6
2420 END DEFINE push_menu
2430 :
2440 DEFINE PROCEDURE segment_wi
ndow (no)
2450 OPEN #4,scr_220x94a248x18
2460 PAPER #4,4:INK #4,0:CLS #4
:CSIZE #4,2,0
2470 PRINT #4:' SEGMENT ':no
2480 WINDOW #4,213,80,252,29:PA
PER #4,0:INK #4,7
2490 disp_seg pres_seg
2500 END DEFINE segment_window
2510 :
2520 DEFINE PROCEDURE frame_wind
ow (no)
2530 OPEN #7,scr_220x94a248x120
2540 PAPER #7,4:INK #7,0:CSIZE
#7,2,0:CLS #7
2550 PRINT #7:' FRAME ':no
2560 WINDOW #7,213,80,252,131:P
APER #7,0:INK #7,7
2570 disp_frm pres_frm
2580 END DEFINE frame_window

```

continued on page 97

continued from page 34

```

2590 :
2600 DEFINE PROCEDURE main_menu
2610 LOCAL a
2620 OPEN #3,scr_200x94a36x18
2630 PAPER #3,2:INK #3,0:CSIZE
#3,2,0:CLS #3
2640 PRINT #3:' MAIN MENU'
2650 WINDOW #3:193,80,40,29
2660 PAPER #3,0:INK #3,7:CSIZE
#3,1,0:CLS #3
2670 PRINT #3:" Alter segme
nt" " Alter frame" " Chang
e segment" " Change frame" "
Draw fractal" " Permanent
store"
2680 END DEFINE main_menu
2690 :
2700 DEFINE PROCEDURE select_opt
ion (menu2, strm, strt, tot, mpos, va
nish, old_vanish)
2710 LOCAL a, pos
2720 pos=mpos: opos=-1
2730 REPEAT loop
2740 OVER #strm, -1
2750 IF old_vanish<>-1 THEN BL
OCK #strm, 192, 10, 0, ((strt+old_va
nish+1)*10), 7: old_vanish=-1
2760 IF opos>-1 THEN BLOCK #st
rm, 192, 10, 0, ((strt+opos)*10), 7
2770 BLOCK #strm, 192, 10, 0, ((st
rt+pos)*10), 7: OVER #strm, 0
2780 opos=pos: IF menu2 THEN ca
ll_menu pos
2790 REPEAT loop2
2800 i$=INKEY$(-1)
2810 pos=pos+((i$=CHR$(216))*
(pos<(tot-1)))-((i$=CHR$(208))*
(pos>0))
2820 IF i$=CHR$(32) THEN sele
ction=pos: EXIT loop2
2830 IF pos<>opos THEN EXIT l
oop2
2840 END REPEAT loop2
2850 IF i$=CHR$(32) THEN EXIT l
oop
2860 END REPEAT loop
2870 OVER #strm, -1
2880 IF vanish THEN BLOCK #strm
, 192, 10, 0, ((strt+pos)*10), 7
2890 OVER #strm, 0: mpos=pos
2900 END DEFINE select_option
2910 :
2920 DEFINE PROCEDURE call_menu
(no)
2930 LOCAL title$
2940 OPEN #5,scr_200x10a36x120
2950 PAPER #5,2:INK #5,0
2960 RESTORE no*10+10000: READ t
itle$
2970 CSIZE #5,2,0:CLS #5:PRINT
#5:title$
2980 END DEFINE call_menu
2990 :
3000 DEFINE PROCEDURE UPDATE_FRA
CTAL_MENU (option, value)
3010 PAPER #6,7:INK #6,0: AT #6.
option, 20:CLS #6,4
3020 PRINT #6:value:'): PAPER #
6,0:INK #6,7
3030 END DEFINE UPDATE_FRACTAL_M
ENU
3040 :
3050 DEFINE PROCEDURE command
3060 OPEN #8,scr_433x26a36x226
3070 BORDER #8,2,2: PAPER #8,0: C
LS #8
3080 END DEFINE command
3090 :
3100 DEFINE PROCEDURE prompt (no
, line)
3110 RESTORE no+20000: READ text
$
3120 AT #8, line, 0:CLS #8,3:PRINT
#8;text$:
3130 END DEFINE prompt
3140 :
3150 DEFINE PROCEDURE directory
3160 command:prompt 2,0:_input
8,0,15,5,dir_$,0,5
3170 OPEN #9,scr_200x196a36x18

```

```

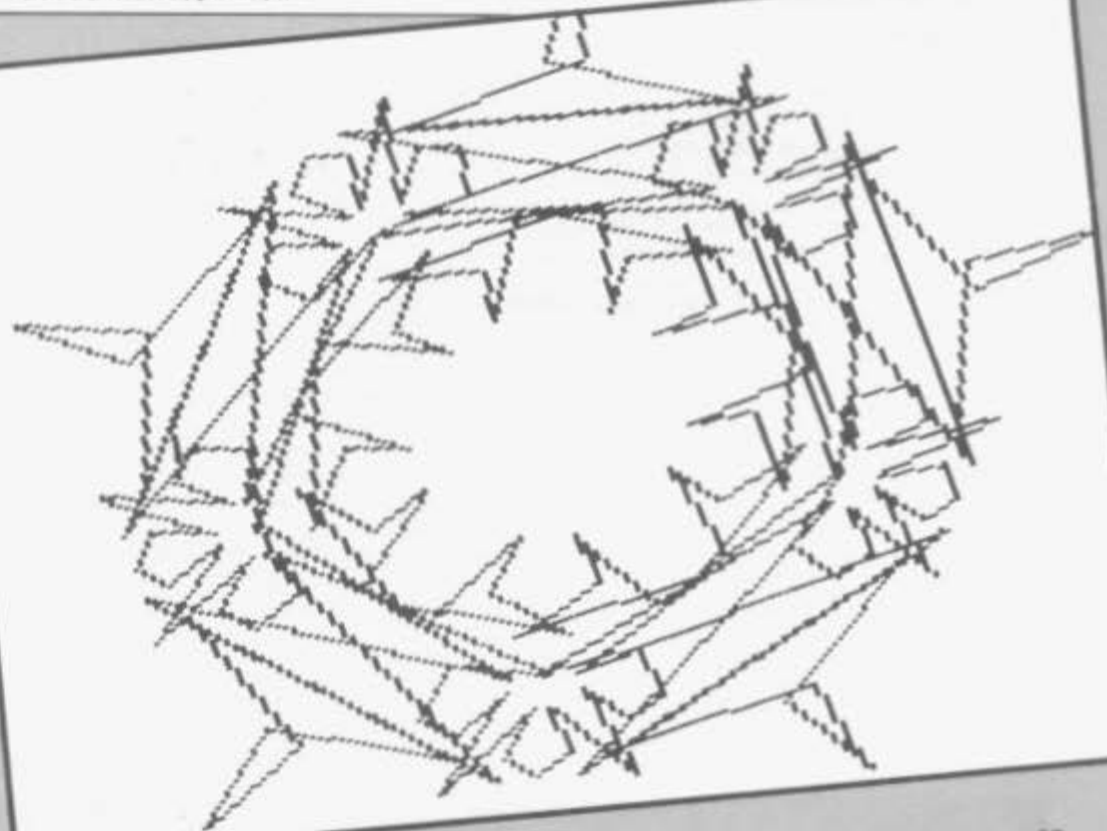
3180 PAPER #9,2:INK #9,0:CSIZE
#9,1,0:CLS #9
3190 PRINT #9:' DIRECTORY OF '
:_input_$
3200 WINDOW #9,193,182,40,29
3210 PAPER #9,0:INK #9,7:CSIZE
#9,0,0:CLS #9
3220 DIR #9,_input_$:dir_$=_input
_$
3230 command:prompt 4,1:PAUSE:C
LOSE #9
3240 END DEFINE directory
3250 :
3260 DEFINE PROCEDURE _input (st
rm, px, py, leng, default$, num, min)
3270 LOCAL inp$, i, ok
3280 ok=0: _input_$=default$
3290 REPEAT i_loop
3300 AT #strm, px, py:CLS #strm,
4:PRINT #strm:_input_$
3310 inp$=INKEY$(-1):i=CODE(inp
$)
3320 SELECT ON i
3330 =10
3340 IF LEN(_input_$)>min TH
EN ok=1
3350 =194
3360 IF LEN(_input_$)>0 THEN
_input_$=_input_$ (1 TO (LEN(_input_
$)-1))
3370 =65 TO 90, 95, 97 TO 122
3380 IF num=0 AND LEN(_input_
$)<leng THEN _input_$=_input_$&inp
$
3390 =48 TO 57
3400 IF LEN(_input_$)<leng TH
EN _input_$=_input_$&inp$
3410 END SELECT
3420 IF ok THEN EXIT i_loop
3430 END REPEAT i_loop
3440 END DEFINE _input
3450 :
3460 DEFINE PROCEDURE format_dev
ice
3470 command:prompt 2,0:_input
8,0,15,15,form$,0,5
3480 form$=_input_$:command
3490 PRINT #8:" Formatting ":fo
rm$: '
Please wait'
3500 AT #8,0,13+LEN(form$):FORM
AT #8;form$
3510 AT #8,0,12+LEN(form$):PRIN
T #8;' ':AT #8,0,1:PRINT #8:'
':AT #8,0,44:CLS #8,4
3520 prompt 4,1:PAUSE:logo
3530 END DEFINE format_device
3540 :
3550 DEFINE PROCEDURE get_dims
3560 LOCAL xinc,yinc,n

```

```

3570 xinc=0:yinc=0
3580 FOR n=1 TO num_seg
3590 xinc=xinc+dist_s(n)*SIN(a
ngle_s(n))
3600 yinc=yinc+dist_s(n)*COS(a
ngle_s(n))
3610 END FOR n
3620 len_seg=SQRT(xinc^2+yinc^2
)
3630 IF yinc+1=-1
3640 IF xinc>0:horiz=PI/2
3650 IF xinc<0:horiz=-PI/2
3660 ELSE
3670 IF xinc+1=-1
3680 IF yinc>0:horiz=0
3690 IF yinc<0:horiz=PI
3700 ELSE
3710 horiz=ATAN(xinc/yinc)
3720 IF yinc<0:horiz=horiz+PI
3730 END IF
3740 END IF
3750 END DEFINE get_dims
3760 :
3770 DEFINE PROCEDURE draw(strm,
dist, theta)
3780 LOCAL lx, ly
3790 lx=cx+dist*SIN(theta):ly=c
y+dist*COS(theta)
3800 LINE #strm, cx, cy TO lx, ly
3810 cx=lx:cy=ly
3820 END DEFINE draw
3830 :
3840 DEFINE PROCEDURE fractal(st
rm, depth, ex, ey, EX, EY)
3850 LOCAL scal, theta, n, x1, y1, x
2, y2
3860 IF depth=0
3870 LINE #strm, ex, ey TO EX, EY
3880 RETURN
3890 END IF
3900 scal=SQRT((ex-EX)^2+(ey-EY
)^2)/len_seg
3910 IF ey=ey
3920 IF EX>ex THEN theta=PI/2
3930 IF EX<ex THEN theta=-PI/2
3940 ELSE
3950 IF EX=ex
3960 IF ey>ey: theta=0
3970 IF ey<ey: theta=PI
3980 ELSE
3990 theta=ATAN((EX-ex)/(ey-ey
))
4000 IF ey<ey THEN theta=thet
a+PI
4010 END IF
4020 END IF
4030 theta=theta-horiz
4040 IF depth=1
4050 cx=ex:cy=ey

```




```

4060 FOR n=1 TO num_seg
4070 draw strm,dist_s(n)*scal
,angle_s(n)+theta
4080 IF strm=1 AND (CODE(INKE
Y$)=27 OR esc=1) THEN esc=1:RETU
rn
4090 END FOR n
4100 ELSE
4110 x1=xx:y1=yy
4120 FOR n=1 TO num_seg
4130 x2=x1+scal*dist_s(n)*SIN
(angle_s(n)+theta)
4140 y2=y1+scal*dist_s(n)*COS
(angle_s(n)+theta)
4150 fractal strm,depth-1,x1,
y1,x2,y2
4160 IF strm=1 AND (CODE(INKE
Y$)=27 OR esc=1) THEN esc=1:RETU
rn
4170 x1=x2:y1=y2
4180 END FOR n
4190 END IF
4200 END DEFINE fractal
4210 :
4220 DEFINE PROCEDURE init_main
4230 LOCAL num,n,angle,dist
4240 RESTORE
4250 DIM store_s(2,20,20),store
_f(2,20,20)
4260 DIM angle_s(20),dist_s(20)
,angle_f(20),dist_f(20)
4270 num_seg=0:num_frm=0:esc=0:
pres_seg=4:pres_frm=3
4280 mode_=4:ink_=7:paper_=0:x_
coord=50:y_coord=50:depth_=2
4290 scale_=100:mpos=0:main=0:d
ir_#='mdv1_':form#='mdv1_'
4300 FOR num=1 TO 10
4310 FOR n=1 TO num
4320 READ dist,angle
4330 store_s(1,num,n)=dist:st
ore_s(2,num,n)=angle*PI/180
4340 END FOR n
4350 FOR n=1 TO num
4360 READ dist,angle
4370 store_f(1,num,n)=dist:st
ore_f(2,num,n)=angle*PI/180
4380 store_f(2,num,n)=angle*P
I/180
4390 END FOR n
4400 END FOR num
4410 FOR num=11 TO 20
4420 FOR n=1 TO num
4430 store_s(1,num,n)=5:store
_f(1,num,n)=5
4440 store_s(2,num,n)=PI/2:st
ore_f(2,num,n)=PI/2
4450 END FOR n
4460 END FOR num
4470 END DEFINE init_main
4480 :
4490 DEFINE PROCEDURE change_seg
(num)
4500 LOCAL n
4510 FOR n=1 TO num
4520 dist_s(n)=store_s(1,num,n)
:angle_s(n)=store_s(2,num,n)
4530 END FOR n
4540 num_seg=num:get_dime
4550 END DEFINE change_seg
4560 :
4570 DEFINE PROCEDURE change_frm
(num)
4580 LOCAL n
4590 FOR n=1 TO num
4600 dist_f(n)=store_f(1,num,n)
:angle_f(n)=store_f(2,num,n)
4610 END FOR n
4620 num_frm=num
4630 END DEFINE change_frm
4640 :
4650 DEFINE PROCEDURE disp_seg (
num)
4660 OPEN#1,scr_220x94a248x18:P
APER 4:INK 0:CSIZE 2,0
4670 AT 0,12:PRINT num;':CLS
#4
4680 change_seg num
4690 fractal 4,1,50,50,150,50
4700 change_seg pres_seg

```

```

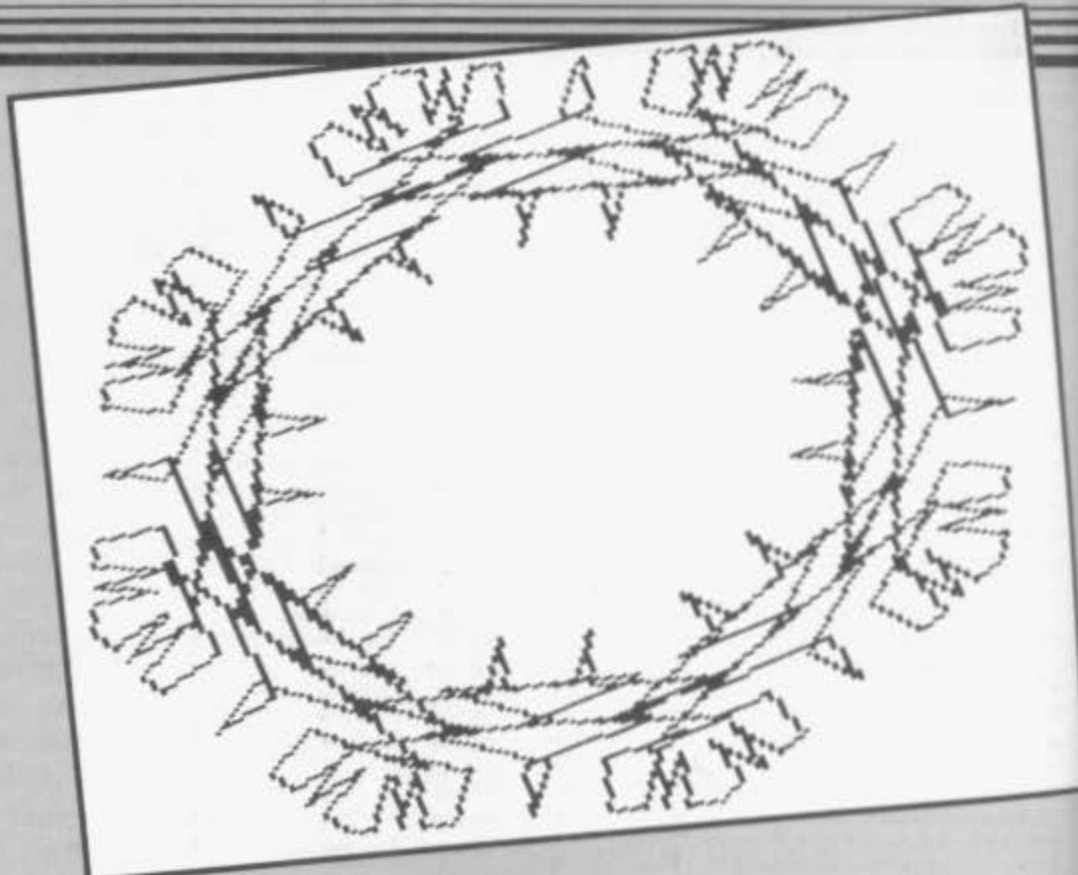
4710 END DEFINE disp_seg
4720 :
4730 DEFINE PROCEDURE disp_frm (
num)
4740 OPEN#1,scr_220x94a248x120:
PAPER 4:INK 0:CSIZE 2,0
4750 AT 0,11:PRINT num;':CLS
#7
4760 CLS #7
4770 change_frm num
4780 dx=x_coord:dy=y_coord:SCAL
E #7,scale_,0,0
4790 FOR n=1 TO num
4800 x=dx+dist_f(n)*SIN(angle_
f(n))
4810 y=dy+dist_f(n)*COS(angle_
f(n))
4820 fractal 7,0,dx,dy,x,y:dx=
x:dy=y
4830 END FOR n
4840 change_frm pres_frm
4850 END DEFINE disp_frm
4860 :
4870 DEFINE PROCEDURE choose_seg
4880 LOCAL i
4890 command:prompt 9,0:num=pre
s_seg
4900 REPEAT choose
4910 i=CODE(INKEY$(-1))
4920 SELECT ON i
4930 =192
4940 IF num>1 THEN num=num-1
:disp_seg num
4950 =200
4960 IF num<20 THEN num=num+
1:disp_seg num
4970 =32
4980 EXIT choose
4990 END SELECT
5000 END REPEAT choose
5010 change_seg num:pres_seg=num
5020 END DEFINE choose_seg
5030 :
5040 DEFINE PROCEDURE choose_frm
5050 LOCAL i
5060 command:prompt 9,0:num=pre
s_frm
5070 REPEAT choose
5080 i=CODE(INKEY$(-1))
5090 SELECT ON i
5100 =192
5110 IF num>1 THEN num=num-1
:disp_frm num
5120 =200
5130 IF num<20 THEN num=num+
1:disp_frm num

```

```

5140 =32
5150 EXIT choose
5160 END SELECT
5170 END REPEAT choose
5180 change_frm num:pres_frm=num
5190 END DEFINE choose_frm
5200 :
5210 DEFINE PROCEDURE store_seg(
num)
5220 LOCAL n
5230 FOR n=1 TO num
5240 store_s(1,num,n)=dist_s(n)
5250 store_s(2,num,n)=angle_s(
n)
5260 END FOR n
5270 END DEFINE store_seg
5280 :
5290 DEFINE PROCEDURE store_frm(
num)
5300 LOCAL n
5310 FOR n=1 TO num
5320 store_f(1,num,n)=dist_f(n)
5330 store_f(2,num,n)=angle_f(
n)
5340 END FOR n
5350 END DEFINE store_frm
5360 :
5370 DEFINE PROCEDURE edit_ (num)
5380 LOCAL n,i,apt,pt,ept
5390 FOR n=3 TO 8:OPEN #n,scr:C
LOSE #n
5400 OPEN#1,scr_433x200a36x18
5410 PAPER 0:INK 7:CLS:BORDER 2
,2:command:prompt 12,0
5420 POINT 60,50:OVER -1
5430 FOR n=1 TO num
5440 LINE TO points(1,n),point
s(2,n)
5450 END FOR n
5460 pt=0
5470 REPEAT m_loop
5480 REPEAT loop
5490 CURSOR points(1,pt),poin
ts(2,pt),-3,-6:PRINT"o"
5500 i=CODE(INKEY$(-1))
5510 SELECT ON i
5520 =192
5530 CURSOR points(1,pt),po
ints(2,pt),-3,-6:PRINT"o":IF pt>
0:pt=pt-1
5540 =200
5550 CURSOR points(1,pt),po
ints(2,pt),-3,-6:PRINT"o":IF pt<
num:pt=pt+1
5560 =32

```



```

5570 EXIT loop
5580 =27
5590 EXIT m_loop
5600 =REMAINDER
5610 CURSOR points(1,pt),po
ints(2,pt),-3,-6:PRINT"o"
5620 END SELECT
5630 END REPEAT loop
5640 CURSOR points(1,pt),point
s(2,pt),-3,-6:PRINT"o"
5650 spt=pt-1:ept=pt+1
5660 IF pt=0 THEN spt=pt
5670 IF pt=num THEN ept=pt
5680 move_pt points(1,spt),poi
nts(2,spt),points(1,pt),points(2
,pt),points(1,ept),points(2,ept)
5690 END REPEAT m_loop
5700 CURSOR points(1,pt),points
(2,pt),-3,-6:PRINT"o"
5710 FOR n=num TO 0 STEP -1
5720 points(1,n)=points(1,n)-p
oints(1,0)
5730 points(2,n)=points(2,n)-p
oints(2,0)
5740 END FOR n:OVER 0
5750 END DEFINE edit_
5760 :
5770 DEFINE PROCEDURE edit_s
5780 LOCAL points(2,num_seg),x,
y,n
5790 REPEAT check
5800 x=0:y=0
5810 FOR n=0 TO num_seg
5820 x=x+dist_s(n)*SIN(angle_
s(n)+PI/2-horiz)
5830 y=y+dist_s(n)*COS(angle_
s(n)+PI/2-horiz)
5840 points(1,n)=x+60:points(
2,n)=y+50
5850 END FOR n
5860 edit_num_seg
5870 FOR n=1 TO num_seg
5880 x=points(1,n)-points(1,n
-1)
5890 y=points(2,n)-points(2,n
-1)
5900 dist_s(n)=SQRT(x^2+y^2)
5910 IF y+1=1
5920 IF x>=0:angle_s(n)=PI/2
5930 IF x<0:angle_s(n)=-PI/2
5940 ELSE
5950 IF x+1=1
5960 IF y>0:angle_s(n)=0
5970 IF y<0:angle_s(n)=PI
5980 ELSE
5990 angle_s(n)=ATAN(x/y)
6000 IF y<0:angle_s(n)=angl
e_s(n)+PI
6010 END IF
6020 END IF
6030 END FOR n
6040 get_dims:IF len_seg>=1:st
ore_num_seg:EXIT check
6050 command:prompt 15,0:PAUSE
6060 END REPEAT check
6070 END DEFINE edit_s
6080 :
6090 DEFINE PROCEDURE edit_f
6100 LOCAL points(2,num_frm),x,
y,n
6110 x=0:y=0
6120 FOR n=0 TO num_frm
6130 x=x+dist_f(n)*SIN(angle_f
(n))
6140 y=y+dist_f(n)*COS(angle_f
(n))
6150 points(1,n)=x+60:points(2
,n)=y+50
6160 END FOR n
6170 edit_num_frm
6180 FOR n=1 TO num_frm
6190 x=points(1,n)-points(1,n
-1)
6200 y=points(2,n)-points(2,n
-1)
6210 dist_f(n)=SQRT(x^2+y^2)
6220 IF y+1=1
6230 IF x>=0:angle_f(n)=PI/2
6240 IF x<0:angle_f(n)=-PI/2
6250 ELSE
6260 IF x+1=1

```

```

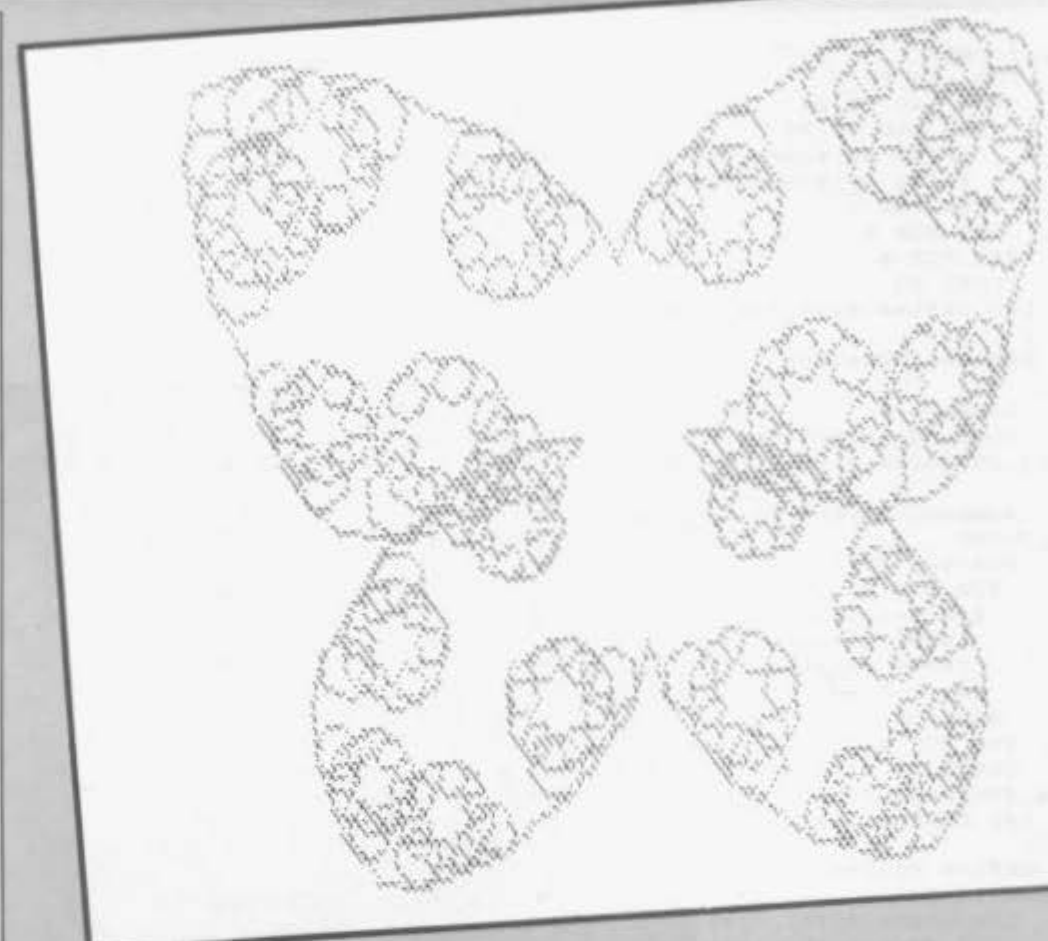
6270 IF y>0:angle_f(n)=0
6280 IF y<0:angle_f(n)=PI
6290 ELSE
6300 angle_f(n)=ATAN(x/y)
6310 IF y<0:angle_f(n)=angle
_f(n)+PI
6320 END IF
6330 END IF
6340 END FOR n
6350 store_frm num_frm
6360 END DEFINE edit_f
6370 :
6380 DEFINE PROCEDURE move_pt(sx
,sy,x,y,EX,ey)
6390 LOCAL i
6400 command:prompt 11,0
6410 OVER -1
6420 LINE sx,sy TO x,y TO EX,ey
6430 REPEAT loop
6440 LINE sx,sy TO x,y TO EX,e
y
6450 i=CODE(INKEY$(-1))
6460 SELECT ON i
6470 =208
6480 LINE sx,sy TO x,y TO EX
,ey:y=y+1
6490 =209
6500 LINE sx,sy TO x,y TO EX
,ey:y=y+5
6510 =216
6520 LINE sx,sy TO x,y TO EX
,ey:y=y-1
6530 =217
6540 LINE sx,sy TO x,y TO EX
,ey:y=y-5
6550 =192
6560 LINE sx,sy TO x,y TO EX
,ey:x=x-1
6570 =193
6580 LINE sx,sy TO x,y TO EX
,ey:x=x-5
6590 =200
6600 LINE sx,sy TO x,y TO EX
,ey:x=x+1
6610 =201
6620 LINE sx,sy TO x,y TO EX
,ey:x=x+5
6630 =32
6640 EXIT loop
6650 =REMAINDER
6660 LINE sx,sy TO x,y TO EX
,ey

```

```

6670 END SELECT
6680 END REPEAT loop
6690 prompt 12,0
6700 END DEFINE move_pt
6710 :
6720 DEFINE PROCEDURE draw_fract
al
6730 LOCAL n,x,sx,y,sy,esc
6740 FOR n=4 TO 8:OPEN #n,scr:C
LOSE #n
6750 IF mode=-8 THEN MODE 8:ELS
E push_menu
6760 OPEN #3,scr_512x256a0x0:PA
PER #3,7,0,1:CLS #3:CLOSE #3
6770 OPEN #1,scr_433x200a36x18
6780 INK ink:PAPER paper:CLS:
BORDER 2,2:command:prompt 7,0
6790 SCALE scale,0,0:sx=x_coor
d:sy=y_coord:esc=0
6800 FOR n=1 TO num_frm
6810 x=sx+dist_f(n)*SIN(angle_
f(n))
6820 y=sy+dist_f(n)*COS(angle_
f(n))
6830 fractal 1,depth_,sx,sy,x,
y:IF esc=1:EXIT n
6840 sx=x:sy=y
6850 END FOR n
6860 save_scrn:IF mode=-8 THEN
MODE 4
6870 END DEFINE draw_fractal
6880 :
6890 DEFINE PROCEDURE save_scrn
6900 LOCAL a$
6910 command:prompt 5,0
6920 a$=INKEY$(-1):IF a$="a"
6930 command:prompt 3,0:_input
8,0,13,15,form$.0,5:form$=input
_
6940 BORDER #8,0,0:PAPER #8,7,
0,1:CLS #8
6950 SBYTES form$,131072,32768
6960 END IF
6970 command
6980 END DEFINE save_scrn
6990 :
7000 DEFINE PROCEDURE save_fract
als
7010 LOCAL a,b,c
7020 command:prompt 3,0:_input
8,0,13,25,form$.0,5:form$=input
_
7030 command:prompt 13,0:OPEN_N

```




```

EW #9:form$
7040 FOR a=0 TO 2
7050 FOR b=0 TO 20
7060 FOR c=0 TO 20
7070 PRINT #9:store_s(a,b,c)
7080 PRINT #9:store_f(a,b,c)
7090 END FOR c
7100 END FOR b
7110 END FOR a
7120 CLOSE #9
7130 END Define save_fractals
7140 :
7150 Define PROCEDURE load_fract
als
7160 LOCAL a,b
7170 command:prompt 3,0:_input
8,0,13,25,form$,0,5:form$=input_
$
7180 command:prompt 14,0:OPEN_I
N #9,form$
7190 FOR a=0 TO 2
7200 FOR b=0 TO 20
7210 FOR c=0 TO 20
7220 INPUT #9:store_s(a,b,c)
7230 INPUT #9:store_f(a,b,c)
7240 END FOR c
7250 END FOR b
7260 END FOR a
7270 CLOSE #9:disp_seg pres_seg
:disp_frm pres_frm
7280 END Define load_fractals
7290 :
7300 Define PROCEDURE input_ (f$
,current,total)
7310 DIM angle_x(20),dist_x(20)
7320 OPEN #9,con_200x196a36x18
7330 PAPER #9,2:INK #9,0:CSIZE
#9,0,0:CLS #9
7340 PRINT #9:' NUMERICAL DATA
FOR 'f$
7350 WINDOW #9,193,182,40,29
7360 PAPER #9,0:INK #9,7:CSIZE
#9,0,0:CLS #9:PRINT #9:'Point
Distance Angle'
7370 command:prompt 16,0
7380 FOR a=1 TO total
7390 AT #9,a+1,0:PRINT #9:'(':
a:'))'
7400 AT #9,a+1,11:PRINT #9:'?'
:_input 9,a+1,12,3,'',1,1:AT #9
,a+1,11:PRINT #9:' ':dist_x(a)=i
nput_$
7410 AT #9,a+1,25:PRINT #9:'?'
:_input 9,a+1,26,3,'',1,1:AT #9,

```

```

a+1,25:PRINT #9:' ':angle_x(a)=i
nput_$*PI/180
7420 END FOR a
7430 END Define input_
7440 :
7450 Define PROCEDURE get_seg_da
ta
7460 LOCAL a
7470 REPEAT check
7480 input_ "SEGMENT "&pres_se
g,pres_seg,num_seg
7490 FOR a=1 TO num_seg
7500 dist_s(a)=dist_x(a)
7510 angle_s(a)=angle_x(a)
7520 END FOR a
7530 get_dims:IF len_seg>=1 TH
EN store_seg pres_seg:EXIT check
7540 command:prompt 15,0
7550 END REPEAT check
7560 END Define get_seg_data
7570 :
7580 Define PROCEDURE get_frm_da
ta
7590 LOCAL a
7600 input_ "FRAME "&pres_frm,p
res_frm,num_frm
7610 FOR a=1 TO num_frm
7620 dist_f(a)=dist_x(a)
7630 angle_f(a)=angle_x(a)
7640 END FOR a
7650 store_frm pres_frm
7660 END Define get_frm_data
7667 :
7668 REMARK FRACTALS DATA
7669 :
7670 DATA 10,90
7680 DATA 80,90
7690 DATA 10,45,10,135
7700 DATA 80,90,80,270
7710 DATA 10,45,20,135,10,45
7720 DATA 70,60,70,180,70,300
7730 DATA 10,90,10,30,10,150,10,
90
7740 DATA 50,45,50,135,50,225,50
,315
7750 DATA 10,90,10,0,10,90,10,18
0,10,90
7760 DATA 40,54,40,126,40,198,40
,270,40,342
7770 DATA 10,45,10,135,10,135,10
,45,10,45,10,135
7780 DATA 40,30,40,90,40,150,40,
210,40,270,40,330
7790 DATA 10,90,10,0,10,90,20,18
0,10,90,10,0,10,90

```

```

7800 DATA 35,12,5,35,64,35,115,5
,35,167,35,218,5,35,270,35,321,5
7810 DATA 10,90,10,0,10,90,10,18
0,10,180,10,90,10,0,10,90
7820 DATA 30,22,30,67,30,112,30,
157,30,202,30,247,30,292,30,337
7830 DATA 10,90,10,0,10,90,10,18
0,10,90,10,180,10,90,10,0,10,90
7840 DATA 25,30,25,70,25,110,25,
150,25,190,25,230,25,270,25,310,
25,350
7850 DATA 10,45,10,135,10,45,10,
315,10,45,10,135,10,225,10,135,1
0,45,10,135
7860 DATA 20,18,20,54,20,90,20,1
26,20,162,20,198,20,234,20,270,2
0,306,20,342
9997 :
9998 REMARK MENU DATA
9999 :
10000 DATA " ALTER SEGMENT",3,"
Numerical data","On-screen edit"
,"Main menu"
10010 DATA " ALTER FRAME",3,"N
umerical data","On-screen edit"
,"Main menu"
10020 DATA " CHANGE SEGMENT",3,"
Segment number","Cursor keys","M
ain menu"
10030 DATA " CHANGE FRAME",3,"F
rame number","Cursor keys","Main
menu"
10040 DATA " DRAW FRACTAL",9,"C
hange scale ("&scale_&"),"Ch
ange depth ("&depth_&"),"Cha
nge Y co_ord ("&y_coord&"),"Cha
nge X co_ord ("&x_coord&"),"Cha
nge ink ("&ink_&"),"Change
paper ("&paper_&"),"Change
mode ("&mode_&"),"Draw frac
tal","Main menu"
10050 DATA " PERMANENT STORE",5,
"Load fractals","Save fractals",
"Format device","Directory devic
e","Main menu"
19997 :
19998 REMARK PROMPT DATA
19999 :
20001 DATA " Use (CTRL+SHIFT+6)
and (CTRL+SHIFT+-) to select op
tion. Press SPACE to ch
oose...."
20002 DATA " Device name ? "
20003 DATA " File name ? "
20004 DATA " PRES
S ANY KEY TO RETURN TO THE MENU"
20005 DATA " Press 'S' to sa
ve screen, or ESC to ret
urn to menu...."
20006 DATA " Format name ? "
20007 DATA " Drawing fra
ctal Press ESC t
o abort...."
20008 DATA " Value ?"
20009 DATA " Use (CTRL+\) and (
CTRL+]) to view options.
Press SPACE to select...."
20010 DATA " Copyrig
ht Talent Computer Systems 198
5"
20011 DATA " Use (CTRL+SHIFT+6
), (CTRL+SHIFT+-), (CTRL+\), (CT
RL+]) and ALT to move point.
Press SPACE to fix...."
20012 DATA "
Use (CTRL+\) and (CTRL+]) to cha
nge points.
Press SPACE to move poi
nt or ESC to return to menu...."
20013 DATA " SAVING FRACTALS :
"&form$
20014 DATA " LOADING FRACTALS :
"&form$
20015 DATA " ERROR
:- YOU CANNOT CLOSE A SEGME
NT
PLEASE REDO SEGMENT"
20016 DATA " Type numer
ical value then press ENTER..."

```

