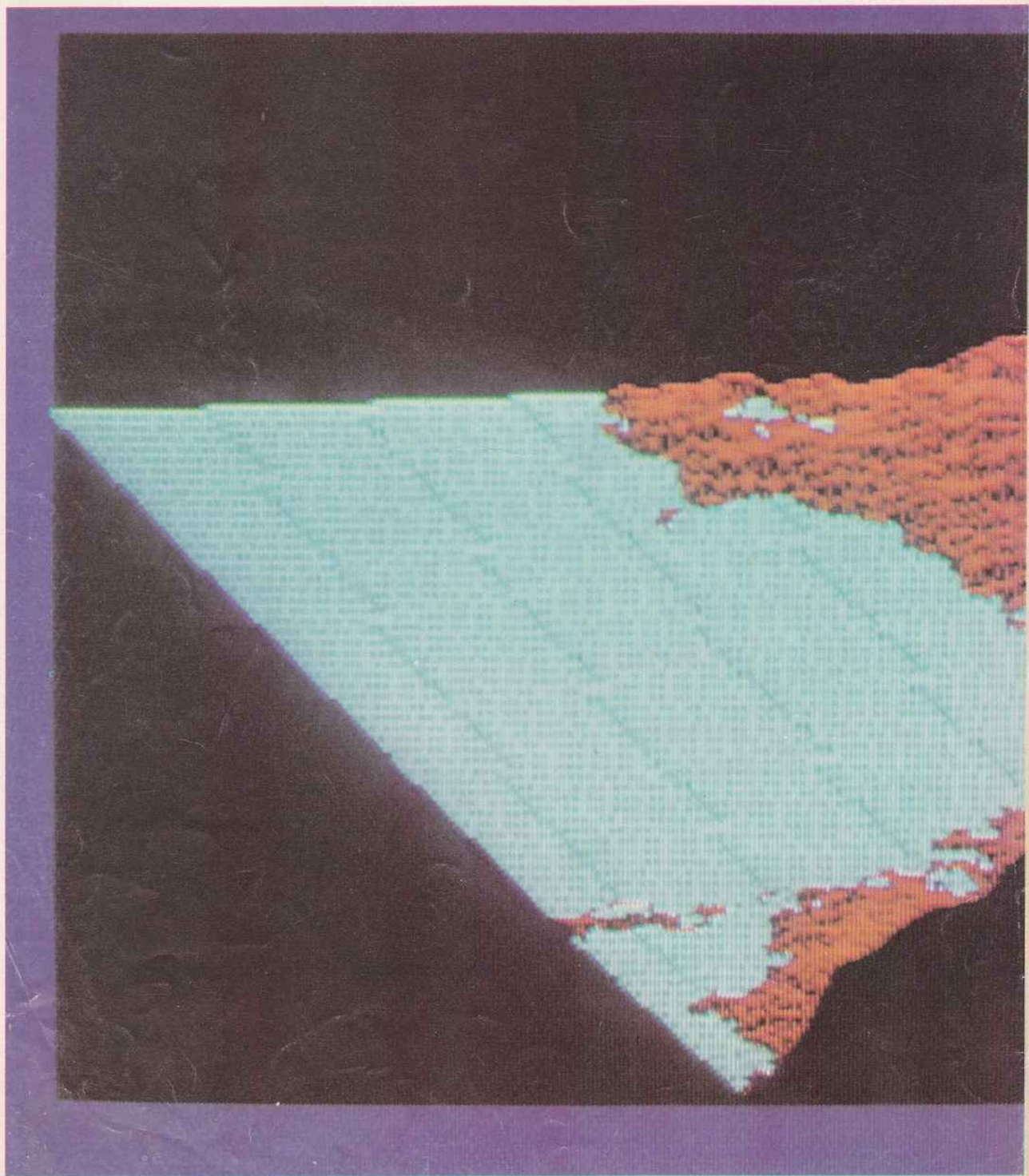
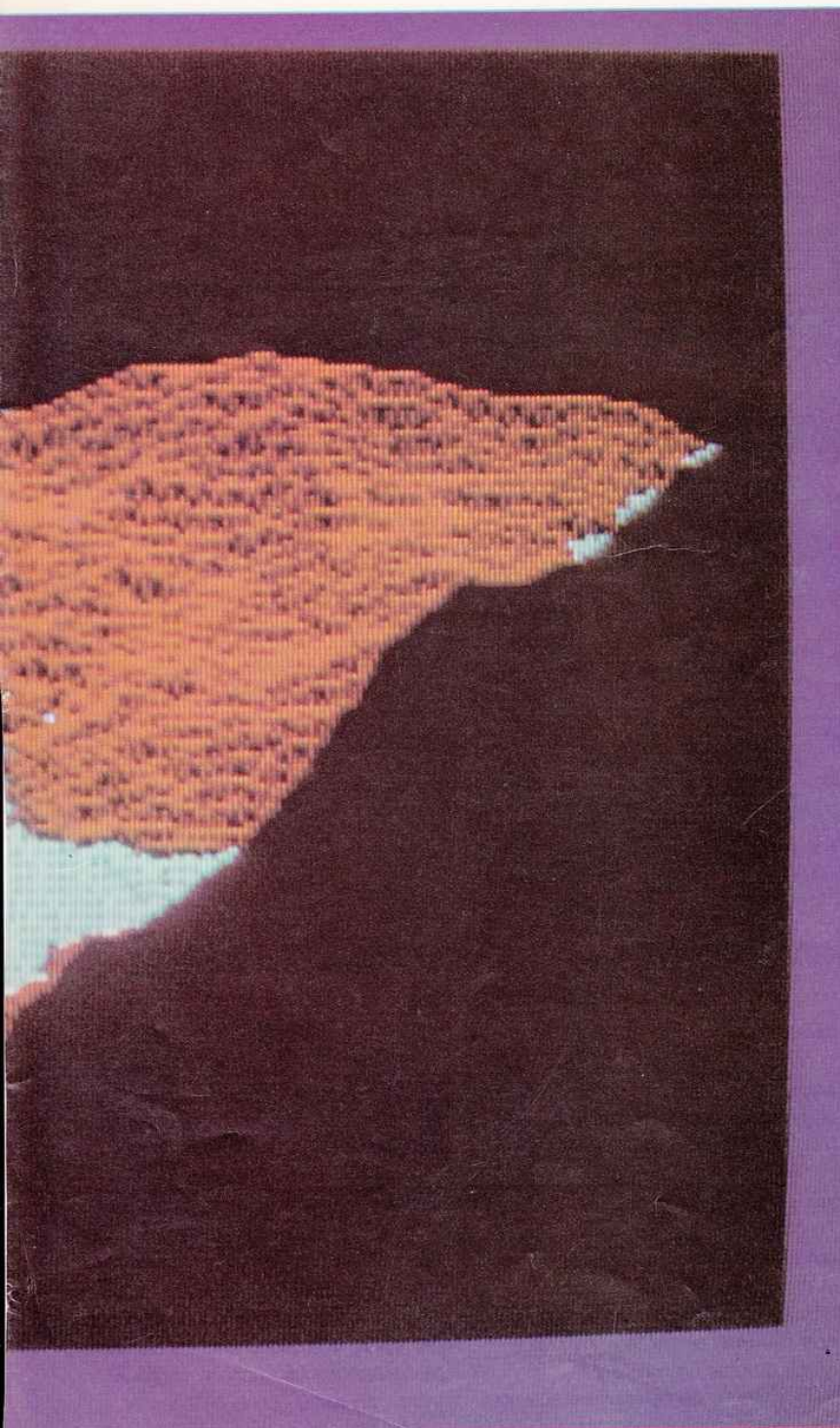


PAISAJES EXTRE



RATERRESTRES



Los muchachos de LucasFilm son grandes investigadores de la síntesis de imágenes por ordenador. En los intervalos entre batallas galácticas estudian cómo mejorar sus efectos especiales por ordenador. De su laboratorio salió el algoritmo que ha inspirado el programa BASIC que se presenta a continuación. Sirve para dibujar paisajes extraterrestres en tres dimensiones, basándose en curiosidades matemáticas como la geometría fractal.

Las fractales son superficies que se definen de una manera recursiva, generalmente por repetición de sí mismas. Si, por ejemplo, se dibuja un triángulo y se utiliza la parte central de cada lado como base para un nuevo triángulo (lógicamente de menores dimensiones), nunca acabaremos de trazar la curva, ya que siempre quedan triángulos por rellenar. Según avanzamos en el dibujo, éste se parecerá cada vez más a una parte de un copo de nieve, con forma de estrella. La propiedad más interesante de este tipo de curvas es que su dimensión, calculada por ciertos métodos, es mayor que uno (curvas) y menos que dos (superficies). Su nombre surge de ahí, de «dimensión FRACcional».

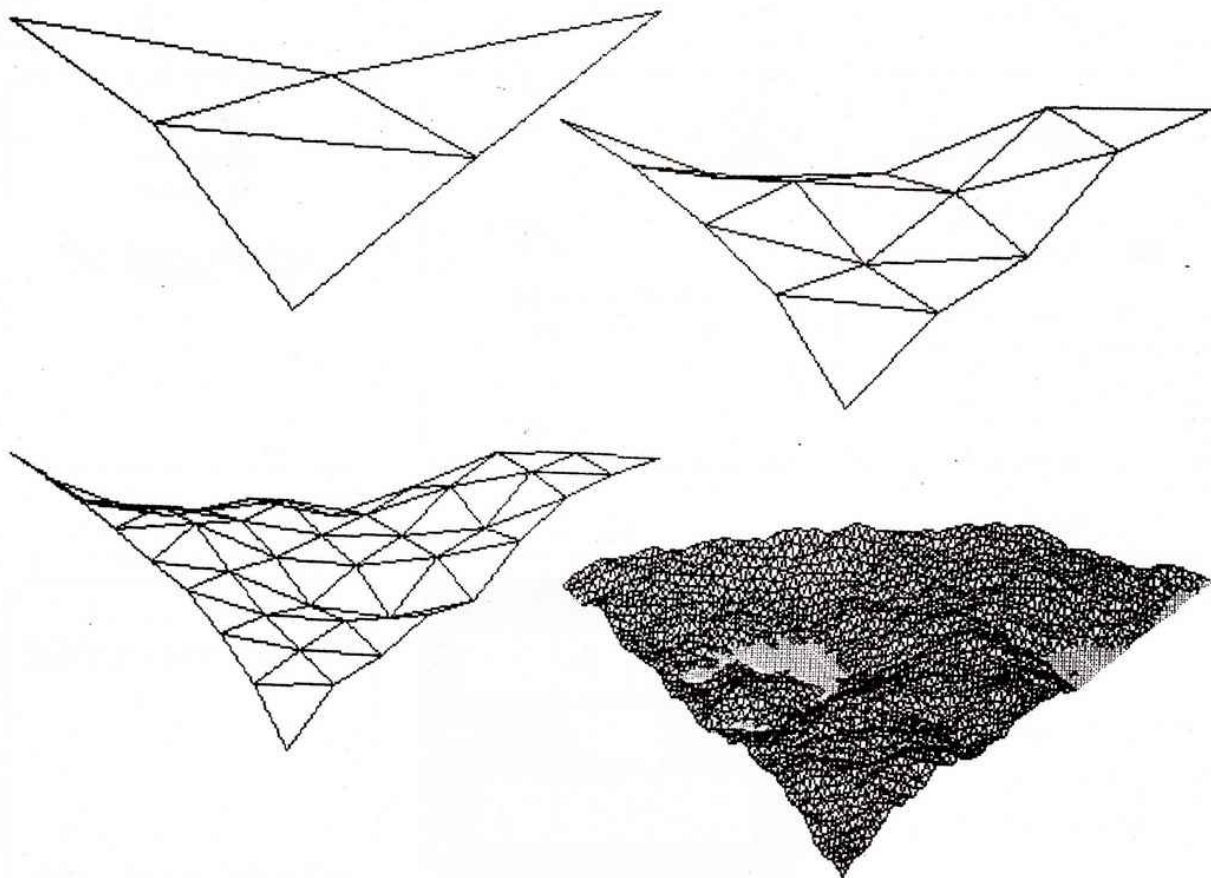
El programa que presentamos realiza una proyección en perspectiva de una superficie fractal. En este caso la definición es muy intuitiva: imaginemos que tenemos un triángulo, y levantamos o bajamos al azar el punto medio de cada lado. Uniéndolos obtenemos una superficie formada por cuatro triángulos. Si repetimos el proceso con cada uno, tendremos dieciseis, y continuando con él, acabaremos por obtener una superficie parecida a un paisaje


```

560 ' Aqui va agua
570 zz=z3: yy=y3: xx=x3: GOSUB 960
580 f1=i4: z2=0: yy=yt: xx=xt: z2=zt: GOTO 630
590 ' EmERGE del agua
600 zz=z3: yy=y3: xx=x3: GOSUB 960
610 f1=i3: z2=zt: yy=yt: xx=xt
620 z2=zz
630 x2=xx: y2=yy: RETURN
640 ' presentacion en pantalla
650 CLS
660 xs=0.05: ys=0.05: zs=0.05: ' Factores de escala
670 FOR ax=0 TO mx: x0=x0-999: FOR ay=0 TO ax
680 GOSUB 380: zz:= yy=ay/mx*10000: xx=ax/mx*10000-yy/2
690 GOSUB 950: NEXT ay: NEXT ax
700 FOR ay=0 TO mx: x0=x0-999: FOR ax=ay TO mx
710 GOSUB 380: zz:= yy=ay/mx*10000: xx=ax/mx*10000-yy/2
720 GOSUB 950: NEXT ax: NEXT ay
730 FOR ex=0 TO mx: x0=x0-999: FOR ey=0 TO mx-ex
740 ax=ex+ey: ay=ey: GOSUB 380: zz:= yy=ay/mx*10000
750 xx=mx/mx*10000-yy/2: GOSUB 950: NEXT ey: NEXT ex
760 GOTO 1080: 'Acabo y sale del bucle
770 ' Rotar
780 IF xx<0 THEN i80
790 IF yy<0 THEN r80=PI/2: GOTO 830
800 r80=PI/2: GOTO 830
810 r80=ATN(yy/xx)
820 IF xx<0 THEN r80=PI
830 r1=r80+rh: r=d=SQR(xx*xx+yy*yy)
840 xx=r*d*COS(r1): yy=r*d*SIN(r1)
850 RETURN
860 ' Perspectiva
870 rd=SQR(zz*zz+xx*xx)
880 IF xx=0 THEN r80=PI/2: GOTO 910
890 r80=ATN(zz/xx)
900 IF xx<0 THEN r80=PI
910 r1=r80-vt
920 xx=r*d*COS(r1): xx: z2=rd*SIN(r1)
930 RETURN
940 ' Moverse a (xp,yp)
950 GOSUB 480
960 xxx=xx*xs: yy=yy*ys: zz=zz*zs
970 GOSUB 780: 'Rotar
980 GOSUB 870: 'Perspectiva
990 xp=(INT((y1+c)*k1.1) : yp=INT(zz)+Z60
1000 IF (x0-999 OR i1=1) THEN x8=yp: y8=p
1010 FLOT x8,y8,f1,0: DRAW xp,yp,f1,0: x8=xp: y8=p: x0=xp
1020 RETURN
1030 'dibujar
1040 =xp:k1.1: yp=p*Z60: IF p=H OR i1=1 THEN x8=yp: y8=p
1050 FLOT x8,y8,f1,0: DRAW -p,yp,f1,0: x8=xp: y8=p: x0=xp
1060 RETURN
1070 ' Salida
1080 a=INKEY$: WHILE LEFT(a)=0: a=INKEY$: WEND
1090 STOP

```

EXTRATERRESTRES



valores aleatorios, y se hace **RANDOMIZE TIME** antes de comenzar. Una forma de seleccionar buenas vistas es realizar un dibujo a nivel tres. Si parece interesante se puede repetir a un nivel mayor escribiendo **RANDOMIZE tiempo** y **GOTO 30**. Las vistas obtenidas se pueden salvar en disco o cassette sustituyendo la última línea por el **SAVE**.

Naturalmente, los tiempos necesarios para realizar un dibujo son función del nivel de recursión, y se multiplican por cuatro cada vez

que le añadimos uno al nivel. Si alguien se plantea ampliar el nivel máximo de recursión, debe pensar al mismo tiempo en cambiar el programa a un lenguaje compilado para ganar velocidad.

Por otra parte, debemos pensar que la aplicación de estas técnicas a problemas reales de animación por ordenador exige el uso de los mayores ordenadores existentes (CRAY XM-P o parecidos). Así que conviene no hacerse ilusiones: con nuestro Amstrad no podremos llegar mucho más allá del nivel 6 que, por otra parte, establece el

límite de lo realizable en BASIC en la mayor parte de los ordenadores.

Las superficies fractales son un descubrimiento matemático reciente, que ha encontrado aplicaciones insospechadas en campos como la mecánica teórica. B. Mandelbrot, creador de este campo, ha presentado muchos otros ejemplos de curvas y superficies fractales; entre ellas el conjunto que lleva su nombre, cuyas representaciones gráficas, de una gran belleza, se muestran en la revista «Investigación y Ciencia» (octubre 1985).

Soy un suscriptor de su revista, que me gusta mucho, pero en el número dos, en la sección *A fondo*, en el programa titulado «Paisajes extraterrestres», hay algunas líneas que vienen en mi ejemplar medio borradas y no lo pude introducir.

Les agradecería mucho que me enviaran el listado correcto o bien lo publicasen en la Sección Correo del próximo número. Muchas gracias por todo.

Julio Roche Castán
Madrid

No has sido el único que has sufrido ese problema, extensivo a buena parte de la tirada de nuestra revista, por ello queremos aprovechar esta petición, así como la de otros lectores, para volver a publicar el listado borroso. Esperamos que no vuelva a ocurrir y pedimos disculpas por este problema técnico.

```

10 DEFINT a-n:INK 0,0:CLS
20 DIM d(64,32): zzz=TIME:RANDOMIZE zzz
30 INPUT "Nivel de recursion ?":ile
40 MODE 1
50 ds=2:FOR n=1 TO ile:ds=ds+2^(n-1):NEXT
  n
60 mx=ds-1: my=mx/2: rh=PI*30/180: vt=r
  h*1.2
70 FOR n=1 TO ile: l=10000/1.8^n
80 PRINT "Trabajando en el nivel ";n
90 ib=mx/2^n:sk=ib*2
100 GOSUB 160: 'Alturas a lo largo de x
110 GOSUB 230: 'Alturas a lo largo de y
120 GOSUB 300: 'Alturas en la diagonal
130 NEXT n
140 GOTO 650: 'Dibujo
150 ' Alturas en direccion x
160 FOR ye=0 TO mx-1 STEP sk
170 FOR xe=ib+ye TO mx STEP sk
180 ax=xe-ib:ay=ye:GOSUB 380:d1=d:ax=xe
  +ib:GOSUB 380: d2=d
190 d=(d1+d2)/2 + RND(1)*1/2 - 1/4: ax=
  xe:ay=ye:GOSUB 430
200 NEXT xe
210 NEXT ye:RETURN
220 ' Alturas en el eje y
230 FOR xe=mx TO 1 STEP -sk
240 FOR ye=ib TO xe STEP sk
250 ax=xe: ay=ye+ib: GOSUB 380: d1=d:
  ay = ye - ib: GOSUB 380: d2=d
260 d=(d1+d2)/2+RND(1)*1/2-1/4:ax=xe:ay
  =ye:GOSUB 430
270 NEXT ye
280 NEXT xe:RETURN
290 ' Alturas en la diagonal
300 FOR xe=0 TO mx-1 STEP sk
310 FOR ye=ib TO mx-xe STEP sk
320 ax=xe+ye-ib: ay=ye-ib: GOSUB 380: d
  1=d
330 ax=xe+ye+ib: ay=ye+ib: GOSUB 380: d
  2=d
340 ax=xe+ye: ay=ye: d=(d1+d2)/2+ RND(1)
  *1/2-1/4: GOSUB 430
350 NEXT ye
360 NEXT xe: RETURN
370 ' Obtencion de datos de la matriz
380 IF ay>my THEN 400
390 by=ay: bx=ax: GOTO 410
400 by=mx+1-ay: bx=mx-ax
410 d=d(bx,by): RETURN
420 ' Escritura en la matriz
430 IF ay>my THEN 450
440 by=ay: bx=ax: GOTO 460
450 by=mx+1-ay: bx=mx-ax
460 d(bx,by)=d: RETURN
470 ' Aqui se situa el nivel del mar
480 IF x0<-999 THEN 510
490 IF zz<0 THEN GOSUB 1080: z2=zz: zz=0:
  GOTO 630
500 GOSUB 1100: GOTO 620
510 IF z2>0 AND zz>0 THEN 620
520 IF z2<0 AND zz<0 THEN z2=zz: zz=0: G
  OTO 630
530 w3=zz/(zz-z2): x3=(x2-xx)*w3+xx:y3=(
  y2-yy)*w3+yy: z3=0
540 zt=zz: yt=yy: xt=xx
550 IF zz>0 THEN 600
560 ' Aqui va agua
570 zz=z3: yy=y3: xx=x3: GOSUB 960
580 GOSUB 1080: zz=0: yy=yt: xx=xt: z2=z
  t: GOTO 630
590 ' Emerge del agua
600 zz=z3: yy=y3: xx=x3: GOSUB 960
610 GOSUB 1100: zz=zt: yy=yt: xx=xt
620 z2=zz
630 x2=xx: y2=yy: RETURN
640 ' presentacion en pantalla
650 GOSUB 1120: 'Inicializa la pantalla
660 xs=0.05: ys=0.05: zs=0.05: 'Factore
  s de escala
670 FOR ax=0 TO mx: x0=-999: FOR ay=0 TO T
  O mx
680 GOSUB 380: zz=d: yy=ay/mx*10000: xx=
  ax/mx*10000-yy/2
690 GOSUB 950: NEXT ay: NEXT ax
700 FOR ay=0 TO mx: x0=-999: FOR ax=ay T
  O mx
710 GOSUB 380: zz=d: yy=ay/mx*10000: xx=
  ax/mx*10000-yy/2
720 GOSUB 950: NEXT ax: NEXT ay
730 FOR ex=0 TO mx: x0=-999: FOR ey=0 TO
  mx-ex
740 ax=ex+ey: ay=ey: GOSUB 380: zz=d: yy
  =ay/mx*10000
750 xx=ax/mx*10000-yy/2: GOSUB 950: NEXT
  ey: NEXT ex
760 GOTO 1140: 'Acabo y sale del bucle
770 ' Rotar
780 IF xx<0 THEN 810
790 IF yy<0 THEN ra=-PI/2: GOTO 830
800 ra=PI/2: GOTO 830
810 ra=ATN(yy/xx)
820 IF xx<0 THEN ra=ra+PI
830 r1=ra+rh:rd=SQR(xx*xx+yy*yy)
840 xx=rd*COS(r1):yy=rd*SIN(r1)
850 RETURN
860 ' ?????
870 rd=SQR(zz*zz+xx*xx)
880 IF xx=0 THEN ra=PI/2: GOTO 910
890 ra=ATN(zz/xx)
900 IF xx<0 THEN ra=ra+PI
910 r1=ra-vt
920 xx=rd*COS(r1)+xx:zz=rd*SIN(r1)
930 RETURN
940 ' Moverse a (xp,yp)
950 GOSUB 480
960 xx=xx*xs: yy=yy*ys: zz=zz*zs
970 GOSUB 780: 'rotar
980 GOSUB 870: '?????
990 IF x0=-999 THEN pr$="M" ELSE pr$="D"
1000 xp=INT(yy)+cx: yp=INT(zz)
1010 GOSUB 1040
1020 RETURN
1030 ' dibujar
1040 xp=xp*1.1: yp=yp+260: IF pr$="M" OR
  f1=1 THEN x8=xp: y8=yp
1050 PLOT x8,y8,f1,0: DRAW xp,yp,f1,0: x
  8=xp: y8=yp: x0=xp
1060 RETURN
1070 ' Color del mar
1080 f1=14: RETURN
1090 ' Color de tierra
1100 f1=3: RETURN
1110 ' Inicializar pantalla o plotter
1120 CLS: RETURN
1130 ' Salida
1140 as=INKEY$: WHILE LEN(as)=0: as=INKE
  Y$: WEND
1150 STOP

```