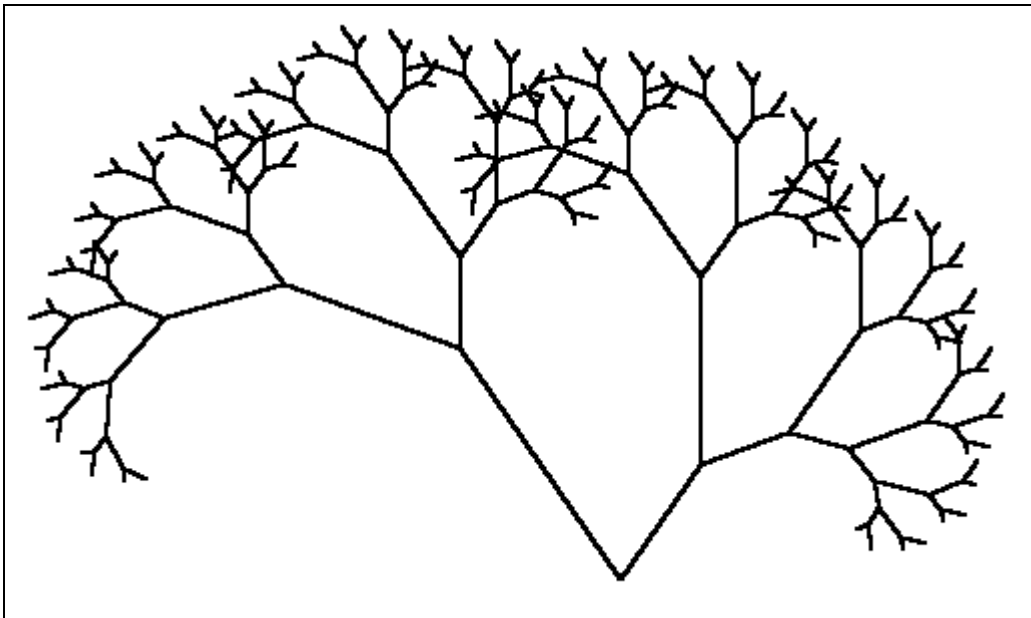


ATRACCIÓN FRACTAL



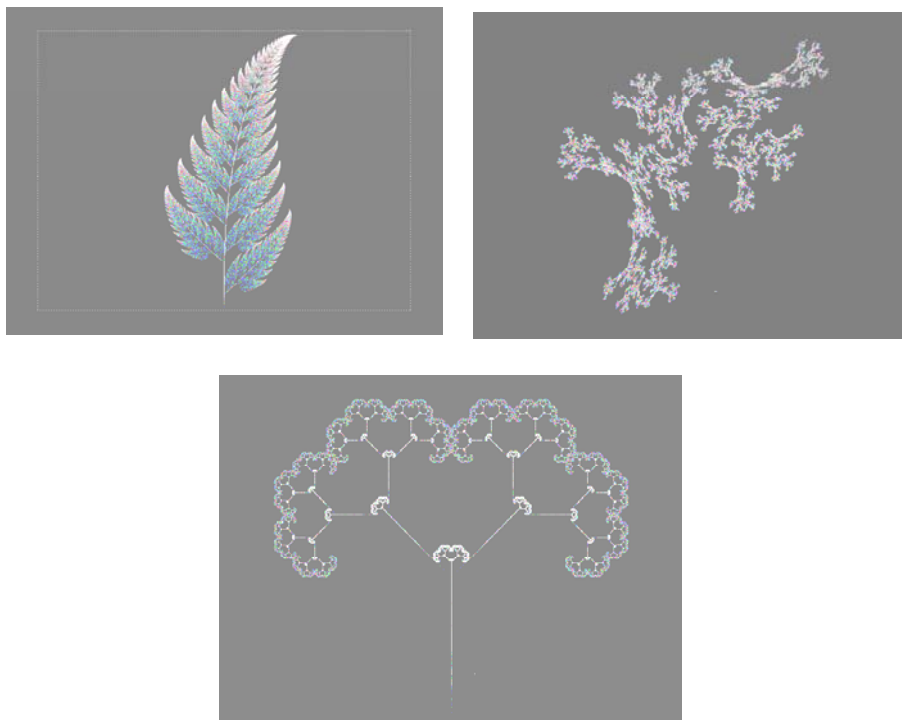
Mauricio Contreras

DISEÑO DE ESTRUCTURAS FRACTALES

1. Introducción

El crecimiento de muchas estructuras de la naturaleza parece seguir pautas fractales. Así ocurre especialmente en el mundo vegetal. Es frecuente encontrar especies de helechos, algas y árboles que, de manera aproximada, cumplen la propiedad de autosemejanza. Esta propiedad consiste en la presencia en el sistema de estructuras que permanecen constantes al variar la escala de observación; es decir: las partes, por pequeñas que sean, se parecen al todo. También la evolución de un bosque puede modelizarse mediante estructuras fractales.

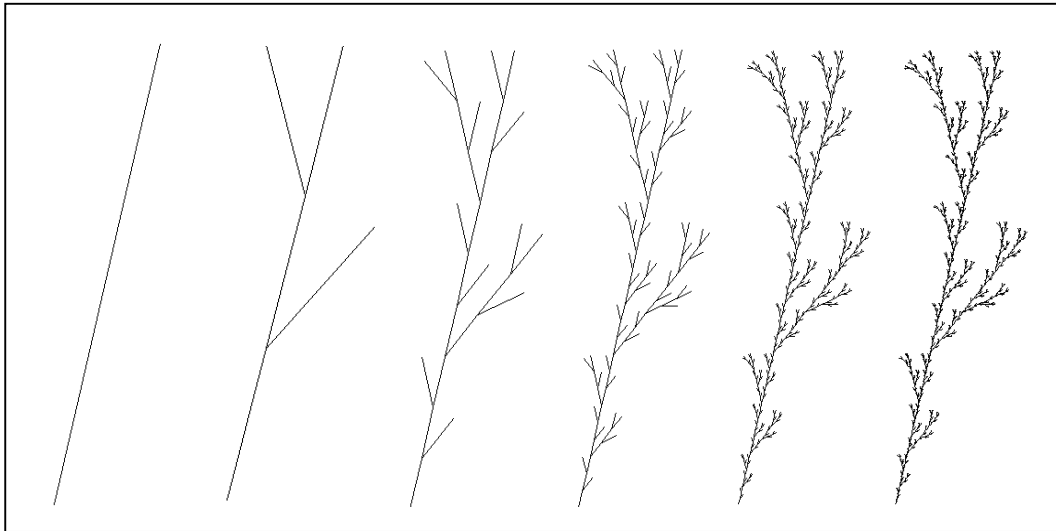
A continuación analizaremos algunas estructuras fractales tipo árbol y las generaremos usando distintos programas (Fractree y MsWinlogo). También usaremos la calculadora gráfica TI-83 para generar estructuras fractales.



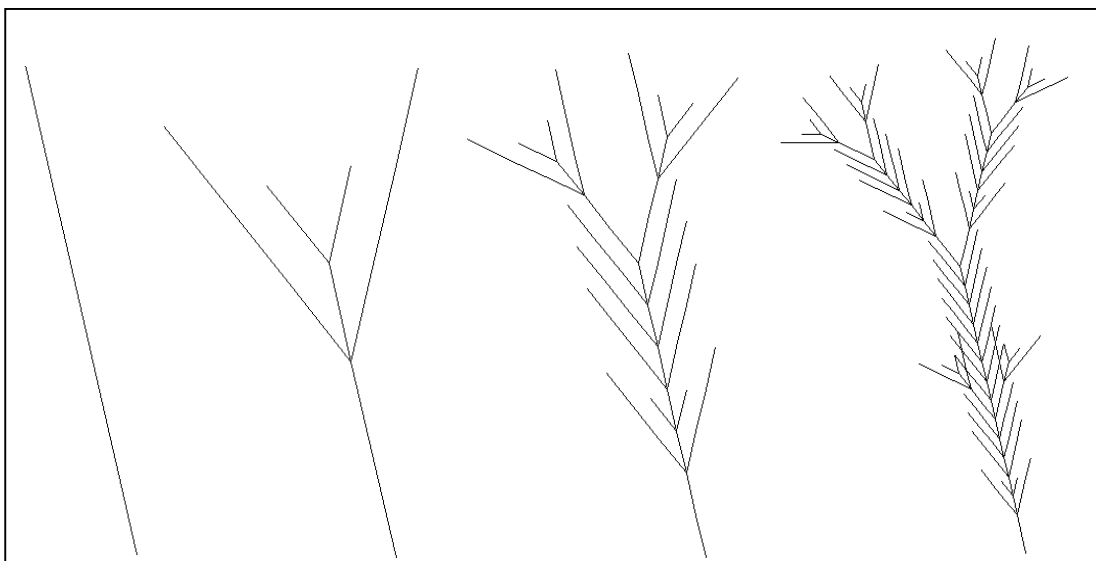
2. Fractales tipo árbol con Fractree

- Inicia el programa Fractree, abriendo la carpeta Fractree y haciendo doble clic sobre el archivo Fractree.exe.
- Selecciona el comando File / Open. En la lista desplegable de la carpeta c:\fractree, selecciona el archivo bush1.fra y haz clic en Aceptar.
- En la siguiente ventana se muestra como está diseñado el fractal. Concretamente, se muestran los axiomas y las reglas de construcción. Haz clic en OK.
- Selecciona el comando Display / Statge 1 y observa el resultado.

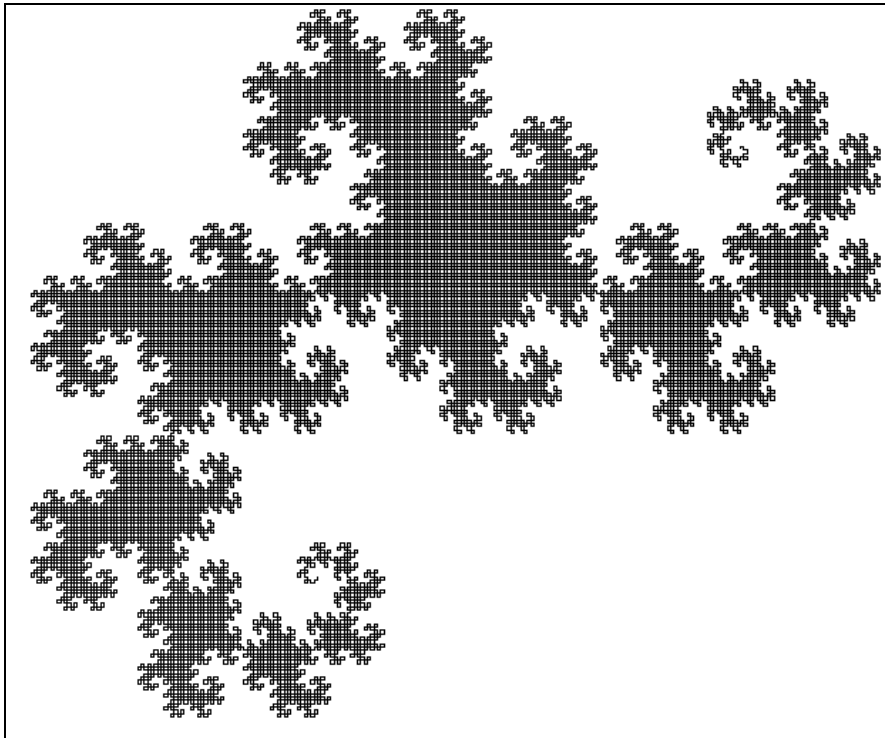
- Selecciona el comando Display / Statge 2 y observa el resultado. Repite el mismo procedimiento para visualizar las pantallas sucesivas. Observa el proceso iterativo y el fractal que se forma. También puedes seleccionar el comando Display / Next Statge o pulsar CTRL+N para pasar de un estado al siguiente.



- Selecciona el comando File / Open. En la lista desplegable de la carpeta c:\fractree, selecciona el archivo bush2.fra y haz clic en Aceptar.
- En la siguiente ventana observa cómo se genera el fractal, prestando atención a las casillas Axiom y Rules. Haz clic en OK.
- Selecciona Display / Statge 1 y observa el resultado.
- Selecciona Display / Statge 2 y observa el resultado. Repite el mismo procedimiento para visualizar las sucesivas pantallas. Selecciona Display / Next Statge o pulsa CTRL+N para pasar de un estado al siguiente.



- Utiliza el mismo procedimiento para explorar los fractales situados en los archivos cross, dragon, hilbert, island, koch, peano.



• **CÓMO GENERA FRACTALES FRACTREE**

Fractree utiliza un procedimiento recursivo para generar fractales. En este proceso, cada carácter en la cadena de entrada que actúa de axioma es sustituida por la correspondiente regla. La regla contiene una cadena sustitutiva para cada carácter ASCII. Algunas cadenas son equivalentes a los correspondientes caracteres en reglas sencillas como la cadena "A" que es sustituida por "A". Otras cadenas especifican reglas no triviales como en el caso de la curva de Koch, en la que la cadena "F" es sustituida por "F-F++F-F".

La cadena inicial se llama axioma. Para la curva de Koch, el axioma es "F", pero puede ser más complicada para otros fractales. El axioma es el estado 0 del fractal. El estado n del fractal se obtiene por sustitución de todos los caracteres de la cadena del estado n-1 de acuerdo con las reglas. Este es un proceso recursivo, porque genera cada vez cadenas más y más complicadas. Estas cadenas son interpretadas por una tortuga que va dibujando la estructura fractal.

A continuación se muestran los primeros estados de la curva de Koch, que hemos tomado como ejemplo:

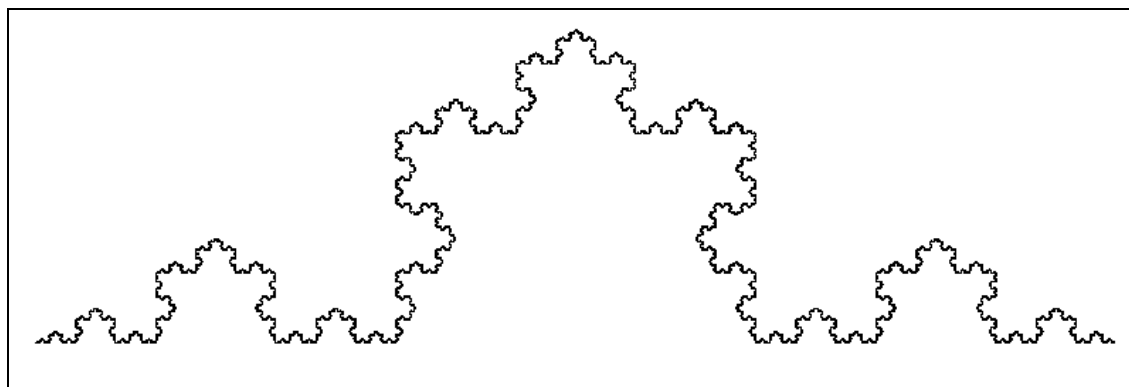
Estado 0: "F"

Estado 1: "F-F++F-F"

Estado 2: "F-F++F-F-F-F++F-F++F-F++F-F-F-F++F-F"

Estado 3: "F-F++F-F-F-F++F-F++F-F++F-F-F-F++F-F-F-F++F-F-F-F++
 F-F++F-F++F-F-F-F++F-F++F-F++F-F-F-F++F-F++F-F++F-F-F-F++F-F-
 F-F++F-F-F-F++F-F ++F-F++F-F-F-F++F-F"

- Para ver este fractal, abre el archivo koch.fra y selecciona el estado en el menú Display.



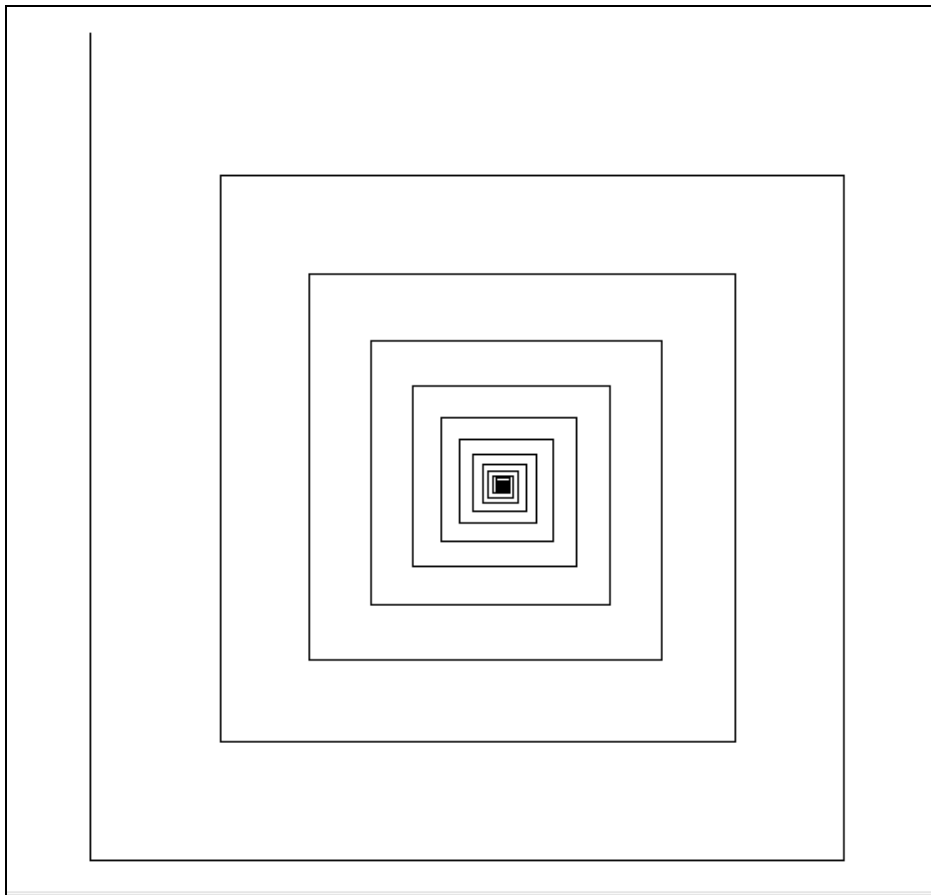
- **Tortuga gráfica**

Los gráficos son dibujados por una tortuga. La imaginaria tortuga interpreta una cadena de caracteres y cambia su posición o dirección de acuerdo con el carácter leído. Los siguientes caracteres son reconocidos por la tortuga como comandos:

"F"	Avanza un paso (del tamaño actual) y dibuja una línea
"f"	Avanza un paso sin dibujar una línea
"+"	Gira un ángulo δ en el sentido de las agujas del reloj
"_"	Gira un ángulo δ en sentido contrario a las agujas del reloj
" "	Gira 180 grados (da media vuelta).
"["	Guarda el estado de la tortuga en memoria; crea un nudo de una rama.
"]"	Restaura el estado anterior de la tortuga desde la memoria. Esto ocasiona que la tortuga vuelva al punto donde arranca la rama.
"*" (asterisco)	Incrementa el tamaño del paso en un 10%.
"/" (barra)	Disminuye el tamaño del paso en un 10%.
"," (coma)	Cambia el tamaño del paso aleatoriamente.

Nota: Otros caracteres especiales, como (!@#\$%^&) tienen usos especiales en futuras versiones de Frac Tree. Los caracteres no citados en la tabla anterior no tienen significado para la tortuga, pero son usados para reescribir cadenas del sistema. En la adición de estos comandos, el comportamiento de la tortuga está determinado por dos parámetros: el tamaño de paso y el ángulo de giro δ . El tamaño del paso se ajusta automáticamente para que la imagen resultante complete la pantalla. El tamaño de la imagen está determinado por la configuración del tamaño del paso y la interpretación de la cadena de comando, dibujando la imagen resultante. Durante este proceso, el cursor cambia al tamaño adecuado.

- Selecciona el comando File / New. A continuación elige el comando Edit / Directions y en la caja Number of Directions, escribe 4. Haz clic en OK.
- Selecciona Edit / Axiom. Escribe F y haz clic en OK.
- Selecciona Edit / Rules. Escribe la regla $F \rightarrow FXF$ y haz clic en el botón Add. Escribe la regla $X \rightarrow +*$ y haz clic en Add. Haz clic en OK.
- Para guardar el archivo, selecciona el comando File / Save rules as. En la caja Nombre de archivo escribe: c:\fractree\espi1.fra. y haz clic en Aceptar.
- Selecciona File / Open. En la lista desplegable de archivos situados en la carpeta c:\fractree, selecciona el archivo espi1.fra y haz clic en Aceptar.
- Utilizando el menú Display muestra distintos estados del fractal que has creado y observa el resultado.



- Utiliza el procedimiento anterior para definir un axioma, una regla y el número de direcciones que permitan construir un fractal. Diseña fractales utilizando distintos axiomas, reglas y números de direcciones.

3. La tortuga fractal: estructuras arborescentes

a) Árboles con MsWLogo

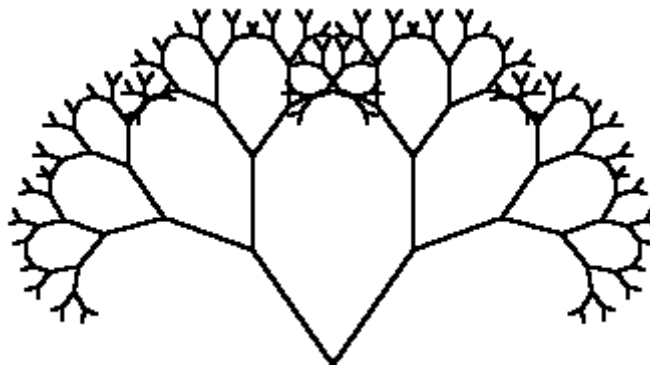
- Inicia el programa MsWLogo, haciendo clic en Inicio / Programas / MsWLogo / MsWinlogo. Con este programa puedes crear procedimientos que hagan que la tortuga se desplace por la pantalla dibujando las figuras geométricas que tú elijas.
- Al iniciar el programa, aparecen dos ventanas: la pantalla de MsWlogo y la ventana Trabajo. En la ventana Trabajo se muestra una ventana de historia, donde aparecen los comandos y procedimientos que se van ejecutando. Debajo de esta área se encuentra la línea de entrada de comandos.
- Para comprobar el funcionamiento del programa, vamos a definir un procedimiento que permite dibujar un cuadrado de cualquier lado. Para ello haz clic en el menú Fichero de la pantalla de MsWlogo y selecciona el comando Editar. En la siguiente ventana introduce el nombre del procedimiento, CUADRADO :LADO y haz clic en OK.
- Aparece la ventana Editor, que muestra la primera y última línea del procedimiento. Teclea en esta ventana el contenido del procedimiento:

```
PARA CUADRADO :LADO
REPITE 4 [AV :LADO GD 90]
FIN
```

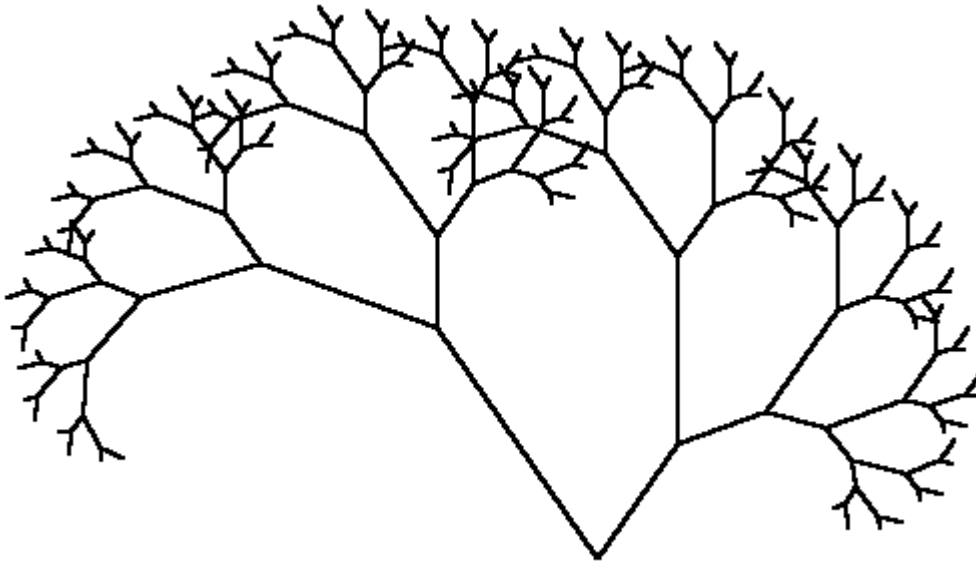
- Haz clic en Fichero, en la barra de menús de la ventana Editor, y selecciona Guardar y salir.
- Observa que el cursor aparece en la línea de comandos, debajo del área de historia (en la ventana de Trabajo). Para comprobar el funcionamiento del procedimiento, teclea:

```
CUADRADO 90 y haz clic en Ejecutar.
CUADRADO 70 y haz clic en Ejecutar.
CUADRADO 50 y pulsa ENTER.
CUADRADO 100 y pulsa ENTER.
```

- Borra la pantalla, tecleando BP y haciendo clic en Ejecutar.
- Vamos a definir los procedimientos ARBOL y ARBOLB que permiten dibujar los siguientes árboles:



ARBOL



ARBOLB

- Haz clic en el menú Fichero de la pantalla de MsWlogo y selecciona el comando Editar. En la siguiente ventana introduce el nombre del procedimiento ARBOL :L :A y haz clic en OK o pulsa ENTER. Las variables :L y :A representan, respectivamente, LADO y ÁNGULO.
- En la ventana Editor teclea el contenido del procedimiento ARBOL:

```

PARA ARBOL :L :A
SI 5 > :L [ALTO]
GI :A / 2
AV :L
ARBOL :L * 2 / 3 :A
RE :L
GD :A
AV :L
ARBOL :L * 2 / 3 :A
RE :L
GI :A / 2
FIN
    
```

- Haz clic en Fichero, en la barra de menús de la ventana Editor, y selecciona Guardar y salir.
- El cursor aparece en la línea de comandos, debajo del área de historia (en la ventana de Trabajo). Para comprobar el funcionamiento del procedimiento, teclea:

ARBOL 70 70 y haz clic en Ejecutar.

- Prueba otros valores para el lado :L y el ángulo :A
- Haz clic en el menú Fichero de la pantalla de MsWlogo y selecciona el comando Editar. En la siguiente ventana introduce el nombre del procedimiento ARBOLB :L :A y pulsa ENTER. Las variables :L y :A representan, respectivamente, LADO y ÁNGULO.
- En la ventana Editor teclea el contenido del procedimiento ARBOLB:


```

PARA ARBOLB :L :A
SI 5 > :L [ALTO]
GI :A / 2
AV :L * 2
ARBOLB :L * 2 / 3 :A
RE :L * 2
GD :A
AV :L
ARBOLB :L * 2 / 3 :A
RE :L
GI :A / 2
FIN
    
```

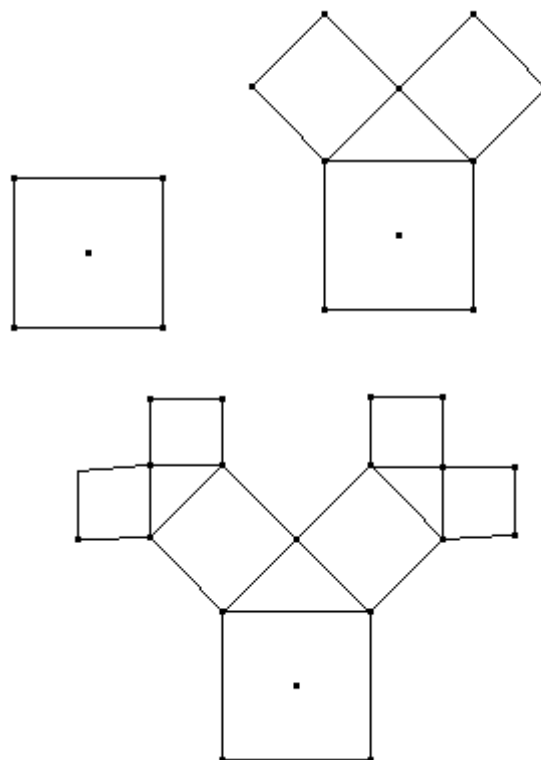
- Haz clic en Fichero, en la barra de menús de la ventana Editor, y selecciona Guardar y salir.
- El cursor aparece en la línea de comandos, debajo del área de historia (en la ventana de Trabajo). Para comprobar el funcionamiento del procedimiento, teclea:

ARBOLB 70 70 y haz clic en Ejecutar.

- Prueba otros valores para el lado :L y el ángulo :A

b) Ejercicio: Árboles Pitagóricos con MsWLogo

- Define un procedimiento de MsWlogo que permita generar árboles pitagóricos como el siguiente:



4. Fractales con la calculadora gráfica

a) Cómo crear y editar un programa con la calculadora TI-83

Para crear un programa, sigue estos pasos:

- Pulsa [PRGM] [◀] para visualizar el menú PRGM NEW.
- Pulsa [ENTER] para seleccionar el comando 1: Create New. Se mostrará el indicador Name= y se activa el bloqueo alfabético.
- Introduce un nombre para el programa, como máximo de ocho caracteres. El primero de ellos debe ser forzosamente una letra.
- Pulsa [ENTER]. Se mostrará el editor de programas.
- Introduce uno o más mandatos de programa. Una línea de mandatos puede contener cualquier instrucción o expresión que se pueda ejecutar desde la pantalla principal. Cada línea de mandato empieza con un signo de dos puntos (:). Los mandatos siempre van separados por dos puntos (:). En el editor de programas se pueden visualizar menús y seleccionar elementos de los mismos. Cuando termines una línea de mandato, pulsa [ENTER]: el cursor se desplazará a la siguiente línea.
- No es necesario introducir ningún comando especial para indicar el final del programa. Para salir del editor de programas y regresar a la pantalla principal, hay que pulsar [2nd] [QUIT].

b) Cómo ejecutar un programa

Para ejecutar un programa, comienza en una línea en blanco de la pantalla principal y sigue estos pasos:

- Pulsa [PRGM] para visualizar el menú PRGM EXEC.
- Selecciona el nombre del programa en el menú PRGM EXEC. Se copiará prgmnombre en la pantalla principal (por ejemplo, prgmHOLA).
- Pulsa [ENTER] para ejecutar el programa. Mientras se ejecuta el programa, se visualiza en pantalla el indicador intermitente de actividad.
- Para detener la ejecución del programa, pulsa [ON]. Se mostrará el menú ERR:BREAK. Para regresar a la pantalla principal, selecciona 1: Quit. Para ir al punto en que se produjo la interrupción, selecciona 2: Goto.

c) Algunas instrucciones útiles en la creación de programas

Los siguientes mandatos se pueden encontrar en el menú PRGM CTL (instrucciones de control), que únicamente se puede visualizar al pulsar [PRGM] desde el editor de programas. Estos elementos del menú controlan el flujo de un programa en ejecución. Facilitan la repetición o la omisión de un grupo de mandatos durante la ejecución de un programa.

If	Crea una prueba condicional, de manera que si la condición es falsa, entonces se omite el mandato que sigue inmediatamente a If; si la condición es verdadera, entonces se ejecuta el siguiente mandato.
Then	Ejecuta mandatos cuando If es verdadero.
Else	Ejecuta mandatos cuando If es falso.
For(Crea un bucle incremental : For(variable, principio, fin, incremento).
While	Crea un bucle condicional, ejecutando un grupo de mandatos mientras la condición sea verdadera.
Repeat	Crea un bucle condicional, repitiendo un grupo de mandatos hasta que la condición sea verdadera.
End	Indica el final de un bloque.
Pause	Realiza una pausa en la ejecución del programa.
Lbl	Define una etiqueta.
Goto	Va a una etiqueta.
IS > (Incrementa e ignora si es mayor que.
DS < (Disminuye e ignora si es menor que.
Menu (Define elementos y opciones de menú.
prgm	Ejecuta un programa como una subrutina.
Return	Regresa desde una subrutina.
Stop	Detiene la ejecución.
DelVar	Borra una variable desde un programa.
GraphStyle(Designa el estilo del gráfico que se dibuja.

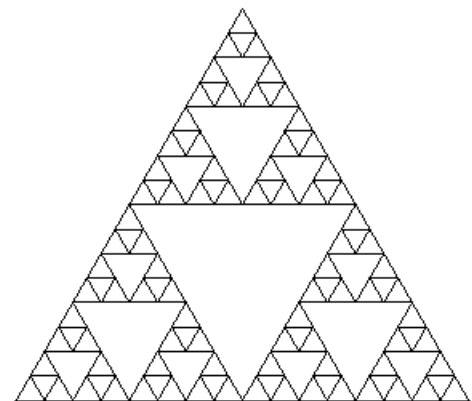
TRIÁNGULO DE SIERPINSKI

El siguiente programa crea un dibujo de un famoso fractal, el triángulo de Sierpinski, y lo almacena en una imagen. Para empezar, pulsa [PRGM] [▶] [▶] [1]. Asigna al programa el nombre SIERPINS y pulsa [ENTER]. Se mostrará el editor de programas.

PROGRAM: SIERPINS

```

:FnOff :ClrDraw
:PlotsOff
:AxesOff
:0 → Xmin :1 → Xmax }
:0 → Ymin :1 → Ymax } → Define la ventana de visualización
:rand → X :rand → Y
:For (K,1,3000) }
:rand → N } → Inicio del grupo For
:If N ≤ 1/3 }
:Then }
: .5X → X } → Grupo If / Then
: .5Y → Y }
:End }
:If 1/3 < N and N ≤ 2/3 }
:Then }
: .5(.5 + X) → X } → Grupo If / Then
: .5(1 + Y) → Y }
:End }
:If 2/3 < N }
:Then }
: .5(1 + X) → X } → Grupo If / Then
: .5Y → Y }
:End }
:Pt-On(X, Y) → Dibuja un punto
:End → Fin del grupo For
:StorePic 6 → Almacena la imagen
    
```



Después de ejecutar este programa, puedes recuperar y visualizar la imagen mediante la instrucción RecallPic 6.

DOBLANDO PAPEL

El programa PAPERDIS se usa para ver los cinco primeros estados de la curva “doblando papel”. El usuario puede experimentar con diferentes factores de escala. La curva real usa un factor de escala de 0,5; La ley de crecimiento usa un factor de escala de 0,71.

- Enciende la calculadora y pulsa **PRGM**. Usa el cursor [\downarrow] para bajar el cursor hasta el programa **PAPERDIS**.
- Pulsa **ENTER** dos veces para iniciar el programa.
- Primero el programa pide al usuario que introduzca el factor de escala. Elige un valor menor o igual que 0,8. Tecllea 0,5 y pulsa **ENTER**. El primer estado de la secuencia “doblando papel” aparece en la pantalla. El número de estado se muestra en la esquina superior izquierda de la pantalla. En la esquina inferior izquierda de la pantalla se muestra la distancia entre los puntos inicial y final de la curva (distancia máxima).
- Continúa pulsando **ENTER** para ver los siguientes estados.
- Después de ver el sexto estado, pulsa **ENTER** y aparecerá un nuevo menú. La Opción 1 ejecutará de nuevo el programa permitiendo al usuario seleccionar un factor de escala diferente. La Opción 2 mostrará en pantalla la distancia para cada estado (redondeando con dos decimales). Pulsa **ENTER** para regresar al menú del programa. La Opción 3 permite salir del programa.

PROGRAM: PAPERDIS (TI - 83)

```

:Disp "MOMENT PLEASE.."
:AxesOff:FnOff
:PlotsOff
:GridOff
:Fix 2
:0->Xmin:0->Ymin
:1-> $\Delta X$ :1-> $\Delta Y$ 
:63->dim (L1)
:1->L1(1)
:For(I,1,5)
:2^I->P
:1->L1(P)
:For(J,P+1,2P-1)
:1-L1(2P-J)->L1(J)
:End
:End
:Lbl N
:Input "SCALE( $\leq 8$ )",Q
:If Q>.8:Goto N
:32/Q->D
:6->dim (L2)
:For(S,1,6)
:ClrDraw
:D*Q->D:round(D,0)->L
:2L+4->U
:If S<6
:Then:62-L->V
:Else:62-2L->V
:End
:U->A:V->E

```

```

:Text(0,0,S)
:0->R
:1->B:Goto F
:Lbl A
:2^S-1->G
:For(I,1,G)
:If L1(I)=1
:Then:.25->T
:Else:.75->T
:End
:fPart ((R+T)->R
:2->B:Goto F
:Lbl B
:Text(57,0,"DIST:")
:round(sqrt((A-U)2+(E-V)2),2)->L2(S)
:Text(57,18,L2(S))
:Pause
:End
:Lbl M
:Menu("WHAT TO DO NOW?","NEW
SCALE",N,"SHOW DATA",S,"QUIT",E)
:Lbl F
:If R=0:Then
:Line(U,V,U,V+L)
:V+L->V
:End
:If R=.25:Then
:Line(U,V,U-L,V)
:U-L->U
:End
:If R=.5:Then
:Line(U,V,U,V-L)
:V-L->V
:End
:If R=.75:Then
:Line(U,V,U+L,V)
:U+L->U
:End
:If B=1:Goto A
:End
:If B=2:Goto B
:End
:Lbl S
:ClrHome
:For(I,1,6)
:Disp round(L2(I),2)
:End
:Pause
:ClrHome
:Goto M
:Lbl E
:Float

```

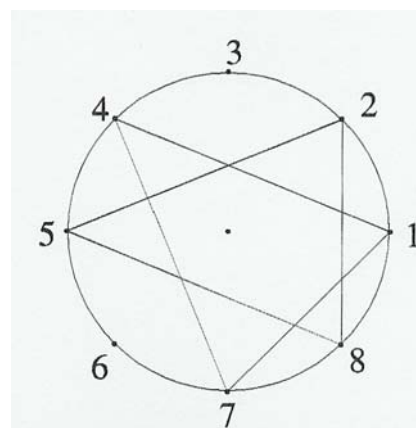
CÍRCULO

El programa CIRCLE contiene un ejercicio de multiplicación y aritmética modular. Se necesitan dos números. El primer número es el módulo, M. Elige números tales que M-1 divida los 360 grados de la circunferencia en partes iguales, por ejemplo M=9, ya que 360 se puede dividir fácilmente en nueve partes. El segundo número es el número multiplicador N, con el cual multiplicaremos cada uno de los números desde 1 hasta M-1. Cada producto se divide por M y se calcula el resto (módulo M). Para este ejemplo, tomamos M=9 y N=4.

Tabla para M = 9 y N = 4

números 1 hasta M - 1	1	2	3	4	5	6	7	8
veces N	4	8	12	16	20	24	28	32
resultado mod M	4	8	3	7	2	6	1	5

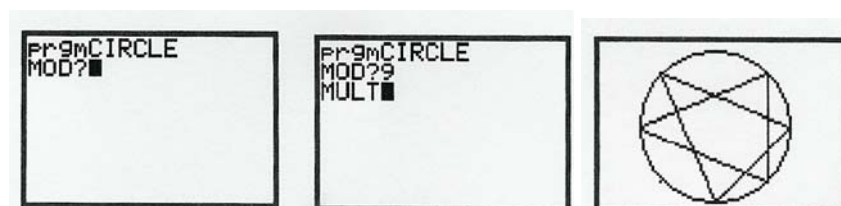
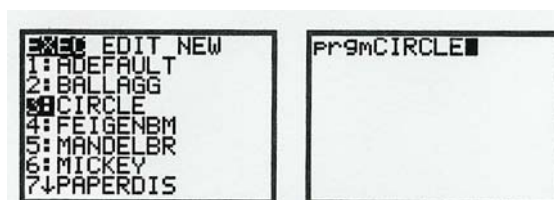
Dibuja una circunferencia y divide el perímetro en M-1 partes iguales y asigna los números a los puntos. Dibuja un segmento entre cada número de la fila 1 de la tabla y el correspondiente número módulo de la fila 3. Si los números en las filas 1 y 3 son iguales, no se dibuja el segmento.



El conjunto generado es siempre simétrico, por tanto es un test débil para chequear la multiplicación modular.

El programa CIRCLE dibuja el conjunto. De esta forma, la calculadora se puede usar para chequear la multiplicación modular.

- Enciende la calculadora y pulsa **PRGM**. Usa la tecla de cursor [↓] para seleccionar el programa **CIRCLE**.
- Pulsa **ENTER** para hacer la selección.
- Pulsa **ENTER** de nuevo para iniciar el programa.
- Un cursor pide al usuario que introduzca el número módulo. Elige **9** y pulsa **ENTER**.
- A continuación, el cursor pide que introduzcas el multiplicador. Elige **4** y pulsa **ENTER**. El programa dibuja el conjunto.



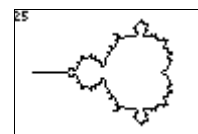
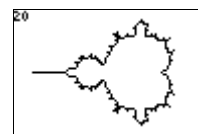
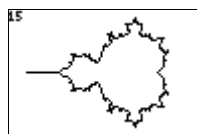
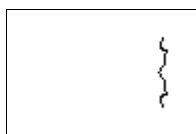
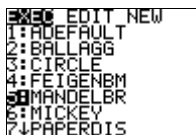
```

PROGRAM:CIRCLE (TI-83 y TI-82)
:ClrDraw:FnOff
:AxesOff
:PlotsOff
:GridOff
:0->Xmin:1->ΔX
:0->Ymin:1->ΔY
:
:Degree
:30->R
:Input "MOD?",M
:Input "MULT",F
:Circle(45,30,30)
:For(U,1,(M-1))
:(U*F/M)-int(U*F/M)->V
:V*M->V
:int((R*cos(360*(U-1)/(M-1))+R+15)->A
:int((R*sin(360*(U-1)/(M-1))+R)->B
:int((R*cos(360*(V-1)/(M-1))+R+15)->C
:int((R*sin(360*(V-1)/(M-1))+R)->D
:Line(A,B,C,D)
:End
    
```

MANDELBROT

El programa MANDELBR dibuja una aproximación del conjunto de Mandelbrot. Este famoso fractal fue el primero que se mostró en 1979 por el padre de los fractales, Benoit Mandelbrot.

- Enciende la calculadora y pulsa **PRGM**.
- Selecciona la opción **MANDELBR** y pulsa **ENTER** dos veces para ejecutar el programa.
- Un cursor aparece preguntando por el estado y la iteración. Este estado debe ser un entero mayor que 4. El mayor de los números, la frontera es la aproximación al conjunto de Mandelbrot. Sin embargo, el más alto es el número de iteraciones, el más extenso el tiempo se toma para completar la figura. Un número entre 10 y 25 está bien elegido.
- El programa entonces crea algunas variables. Entonces comienza dibujando una línea alrededor del conjunto de Mandelbrot. Se necesitan varios minutos para completar la figura.
- Después que la figura está completada, el estado de la iteración se muestra en la esquina superior izquierda de la pantalla.



PROGRAM:MANDELBR (TI-83)

```

:3->dim(LX)
:3->dim(LY)
:FnOff
:AxesOff
:PlotsOff
:GridOff
:-2.31->Xmin
:.035->ΔX
:-1.085->Ymin
:ΔX->ΔY
:Input "STAGE(≥4):",K
:If K<4:4->K
:Disp "MOMENT PLEASE.."
:ClrDraw
:0->T:3->N
:0.24->LX(N)
:0->LY(N)
:
:Lbl F
:1->Z:Goto M
:
:Lbl 1
:If T=0:Then
:LX(N)+ΔX->LX(N)
:Goto F
:End
:
:1->J:2->P
:LX(N)-ΔX->LX(J)
:LX(N)->LX(P)
:LY(N)->LY(J)
:LY(N)->LY(P)
:LY(N)+ΔY->LY(N)
:Pt-On(LX(J),LY(J))
:
:LX(J)->U
:LY(J)->V
:LX(P)->W
:LY(P)->A
:
:2->Z
:
:Lbl S
:Goto M
:
:Lbl 2
:If T=0
:Then
:Pt-On(LX(N),LY(N))
:Pt-On(LX(N),-LY(N))
:J->R:N->J
:Else
:P->R:N->P
:End
:R->N
:LX(J)+LX(P)-LX(R)->LX(N)
:LY(J)+LY(P)-LY(R)->LY(N)
:If LY(N)≥0:Goto S
:
:Lbl Q
:Text(0,0,K)
:Stop
:
:Lbl M
:LX(N)->X
:LY(N)->Y
:0->T
:If X<Xmin or X>Xmax or Y<Ymin or Y>Ymax
:1->T
:1->I
:While I<K and T=0
:I+1->I
:X^->Q:Y^->S
:If Q+S>5
:Then
:1->T
:Else
:2XY+LY(N)->Y
:Q-S+LX(N)->X
:End
:End
:If Z=1:Goto 1
:If Z=2:Goto 2

```

SPIROLT3

El programa SPIROLT3 contiene un ejercicio de espirales, donde una palabra se transforma en un conjunto. Esta actividad puede conducir a alguna investigación lógica o matemática. Los estudiantes necesitan papel cuadriculado y una tabla con el alfabeto. El número de filas puede ser varias y abre un ancho campo de conjuntos diferentes. Dos ejemplos de cómo iniciar una tabla se muestran a continuación.

Tabla 1

1	A	G	M	S	Y
2	B	H	N	T	Z
3	C	I	O	U	
4	D	J	P	V	
5	E	K	Q	W	
6	F	L	R	X	

Tabla 2

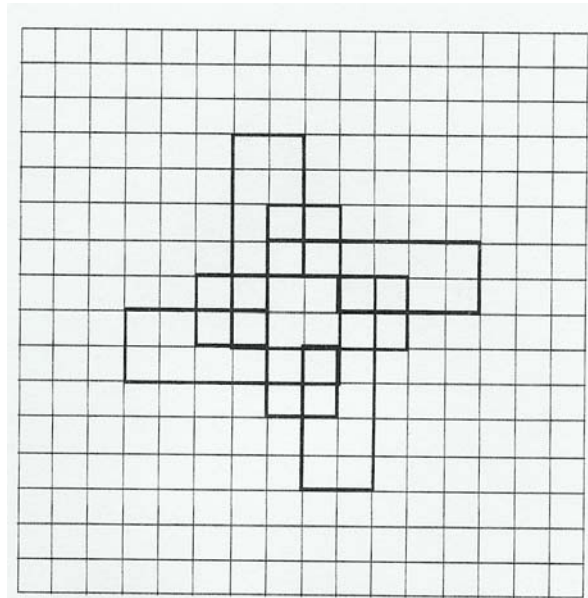
1	A	I	Q	Y
2	B	J	R	Z
3	C	K	S	
4	D	L	T	
5	E	M	U	
6	F	N	V	
7	G	O	W	
8	H	P	X	

Usando una de estas tablas una palabra puede ser transformada en una secuencia de números. Por ejemplos, usando la Tabla 1, la palabra “PATTERN” se transforma en “4122562”. El dibujo empieza en un punto en medio del papel. El primer número de la secuencia indica la longitud del segmento (número de cuadrados) que debe ser dibujado hacia el norte (hacia arriba). Al final de cada segmento se hace un giro hacia la derecha. La longitud del segundo segmento se indica por el segundo número de la secuencia y así sucesivamente. Una vez completada la secuencia, empezamos de nuevo por el principio (de forma circular). Continuamos haciendo esto hasta que el conjunto comienza a repetirse por sí mismo. En este ejemplo, la secuencia se ha usado cuatro veces. Después de esto, el conjunto empieza a repetirse a sí mismo.

El programa **SPIROLT3** dibuja estos conjuntos. Puede usarse para investigar conjuntos dibujados a mano o para producir e investigar un gran número de conjuntos.

- Enciende la calculadora y pulsa **PRGM**. Usa la tecla de cursor [↓] para seleccionar el programa SPIROLT3.
- Pulsa ENTER para hacer la selección.
- Pulsa ENTER de nuevo para iniciar el programa.

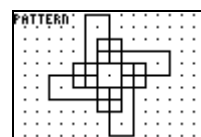
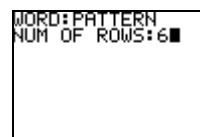
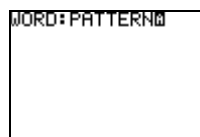
- Aparece un cursor en pantalla que pregunta por una palabra. Hay dos opciones para la entrada:
 - a) con [2nd] [A-LOCK] inicia el modo alfabético y teclea una palabra
 - b) teclea una secuencia de números directamente
- Pulsa **ENTER** para introducir la entrada.



- Teclea la palabra **PATTERN**. Aparece un cursor preguntando por el número de filas de la tabla que deben ser introducidas.
- Por ejemplo, introduce 6 y pulsa **ENTER**. Si se introduce una secuencia de números, después del cursor de la fila siempre introduce 9.
- El programa puede necesitar algún tiempo para completar los cálculos necesarios antes de dibujar el conjunto.

A continuación se muestran dos problemas para los estudiantes:

1. Teclea tres números. El conjunto muestra un cuadrado en el centro. ¿Cómo podemos calcular el tamaño de este cuadrado a partir de los tres números?
2. Teclea cuatro números. El conjunto se mueve en una dirección (por ejemplo hacia la izquierda). ¿Cómo podemos determinar esta dirección a partir de los cuatro números?



PROGRAM:SPIROLT3 (TI-83)

```

:-47->Xmin:1->ΔX
:-31->Ymin:1->ΔY
:ClrDraw
:PlotsOff
:FnOff
:AxesOff
:GridOn
:
:"ABCDEFGHIJKLMNOPQRSTUVWXYZ"->Str2
:"123456789"->Str3
:1->dim(L1)
:ClrHome
:
:Input "WORD:",Str1
:length(Str1)->N
:4->W
:If fPart(N/4)=.5
:2->W
:If fPart(N/4)=0
:6->W
:Input "NUM OF ROWS:",R
:Disp "MOMENT PLEASE.."
:For(I,1,N)
:inString(Str2,sub(Str1,I,1))->L
:inString(Str3,sub(Str1,I,1))->K
:L+K->L
:While L>R
:L-R->L
:End
:L->L1(I)
:End
:
:0->L:0->R:0->U:0->V
:0->A:0->B:0->X:0->Y
:1->C
:0->S
:
:Lbl S
:1->D:1->M
:For(Q,1,W*N)
:If D=0
:Y+C*L1(M)->B
:If D=1
:X+C*L1(M)->A
:If D=2
:Y-C*L1(M)->B
:If D=3
:X-C*L1(M)->A
:

```

```

:If S=0
:Then
:If A>R:A->R
:If A<L:A->L
:If B>V:B->V
:If B<U:B->U
:Else
:Line(X,Y,A,B)
:End
:A->X:B->Y
:D+1->D
:If D=4:0->D
:M+1->M
:If M>N:1->M
:End
:round((R-L)/2,0)*2->T
:iPart(94/T)->C
:round((V-U)/2,0)*2->T
:min({C,iPart(62/T)})->C
:If C<2:2->C
:If S=0:Then
:C->Xscl:C->Yscl
:Text(0,0,Str1)
:-round((R+L)/2,0)*C->X
:-round((U+V)/2,0)*C->Y
:X->A:Y->B
:1->S
:Goto S
:End

```