EJEMPLO DE EXPLORACIÓN DE LA GEOMETRÍA DESDE LA ÓPTICA LOGO: GENERACIÓN DE POLÍGONOS REGULARES (**)

RICARDO LUENGO GONZÁLEZ (*)(&)(•)
LUIS Mª CASAS GARCIA (*)
LUIS MÁRQUEZ ZURITA (*)

RESUMEN

La Geometría diferencial de la Tortuga se presta muy convenientemente a la exploración del espacio desde un punto de vista distinto a la Geometría Euclídea. En el presente artículo se presenta un ejemplo de micromundo que puede servir para comprender ese punto de vista y el isomorfismo existente entre las dos Geometrías, a la vez que abre un potente camino de exploración para compartir con nuestros alumnos en el proceso de enseñanza-aprendizaje.

Se parte del "teorema del viaje completo" y del "camino cerrado simple" para generar un micromundo de polígonos regulares, estudiar la circunferencia como caso límite y mostrar algunos ejemplos de aplicación, que ilustren la práctica en el micromundo diseñado.

SUMMARY

AN EXAMPLE OF HOW TO EXPLORE THE GEOMETRY FROM LOGO OPTICS: GENERATION OF REGULAR POLYGONS

The Differential Geometry of the TURTLE is adapted to the exploration of space

(*) Grupo Beta / Sociedad Extremeña de Educación Matemática "Ventura Reyes Prósper"
(&) ICE Universidad de Extremadura
(•) Depto. de Didáctica de las C. Experimentales y de las Matemáticas
(**) Conferencia pronunciada en Mérida en la presentación pública de la Sociedad Extremeña de Educación Matemática "Ventura Reyes Prósper" el 26 de Febrero de 1991.

from a different point of view to that of Euclidea Geometry. In this article, we present an example of micromundo, which can be used to understand that point of view and the isomorphism between the two Geometries. At the same time it opens the right door to exploring how to share with our pupils in the teaching-learning process.

We start from the "TEOREMA DEL VIAJE COMPLETO" and the "CAMINO CERRADO SIMPLE" in order to generate a "micromundo" of regular polygons and study the circumference as a last resort and show some examples of how to apply these in designing "micromundo".

1. TEOREMA DEL VIAJE COMPLETO

La Geometría de la Tortuga es una Geometría diferencial muy intuitiva y sintónica con el espacio que vivimos (Luengo G. y otros, 1984). Uno de sus Teoremas fundamentales es el "teorema del viaje completo", a partir de él vamos a diseñar un micromundo, paradigma de otros muchos que puede ilustrar el isomorfismo existente entre las dos Geometrías. Comencemos con algunos conceptos básicos necesarios antes de abordar el teorema.

1.1. VIAJES COMPLETOS Y PROCEDIMIENTOS TRANSPARENTES.

Llamamos *viaje completo* a la trayectoria realizada por una tortuga que llega al mismo punto de donde partió y queda (al terminar el viaje) con la misma orientación inicial (Ver figura 1).

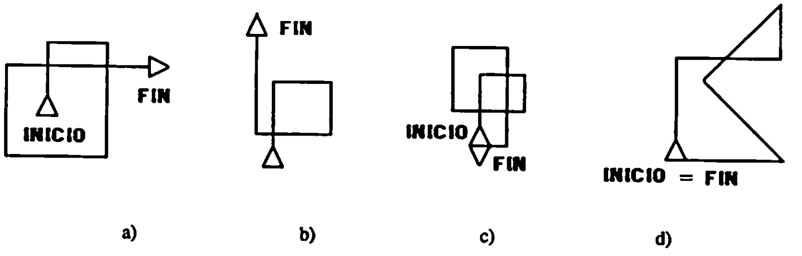


FIGURA 1

En el caso a) de la figura 1 la tortuga realiza un viaje que no es completo, pues los puntos iniciales y finales no son los mismos y además la tortuga al terminar el viaje tiene distinta orientación a la inicial. Análogamente b) y c) tampoco serían viajes completos mientras que c) si lo sería. La tabla siguiente resume los cuatro casos.

Caso	INICIO=FIN?	=ORIENTACIÓN?	TIPO DE VIAJE
a)	No	No	No completo
b)	No	Si	No completo
c)	Sí	No	No completo
d)	Sí	Sí	Viaje Completo

* *Procedimientos Transparantes.*

Se dice que un procedimiento es transparente cuando su efecto es un viaje completo y además deja en el mismo estado los parámetros propios de la tortuga (como si está oculta o nó, tiene o nó levantada la pluma, diseño del trazo de la pluma al terminar es el mismo que al empezar, etc...)

Se debe programar en general con procedimientos transparentes. Si no se hace así, después de la ejecución de un procedimiento no se sabe el estado de la tortuga y en caso de enlazar varios procedimientos, los efectos pueden ser distintos de los esperados. Esto no quiere decir que en un caso concreto no convenga diseñar intencionadamente un procedimiento no transparente, pero eso no debe ser la tónica general.

Programando con procedimientos transparentes, si la tortuga comienza la sesión con orientación Norte y con la pluma baja, se conserva este estado al final de las ejecuciones de los procedimientos y siempre sabremos cómo está la tortuga a fin de enlazar unos procedimientos con otros.

1.2. TEOREMA DEL VIAJE COMPLETO. ÍNDICE DE ROTACIÓN.

El Teorema del viaje completo dice lo siguiente: "El *giro neto total* realizado a lo largo de cualquier viaje completo es 0° o un número entero múltiplo de 360°".

Cuando decimos "giro neto total" nos referimos a la suma algebráica de los giros realizados por la tortuga al recorrer ese camino, considerando un giro positivo (por ejemplo) el de las agujas del reloj (para la tortuga un giro a la derecha) y negativo al contrario (para la tortuga giro a la izquierda).

El giro total se expresa más suscintamente a través del entero múltiplo de 360 que expresa el nº de vueltas que ha dado la tortuga. A ese número entero se le llama *índice de rotación* del camino completo.

Hay que hacer notar que el "giro total" es una propiedad intrínseca del camino; es decir que no depende de los puntos ni inicial ni final del viaje realizado por la tortuga, como tampoco depende de la orientación inicial de la tortuga al comenzar el viaje completo.

Como ejemplo, en la Figura 2, se muestran varios viajes completos indicando en cada caso el índice de rotación. Observese que en todos los casos (menos en b) cambiaría el signo del indice de rotación, si cambiaramos el sentido del recorrido; por eso se suele asignar a un camino completo que no indica sentido de recorrido el doble signo ($\pm$n).

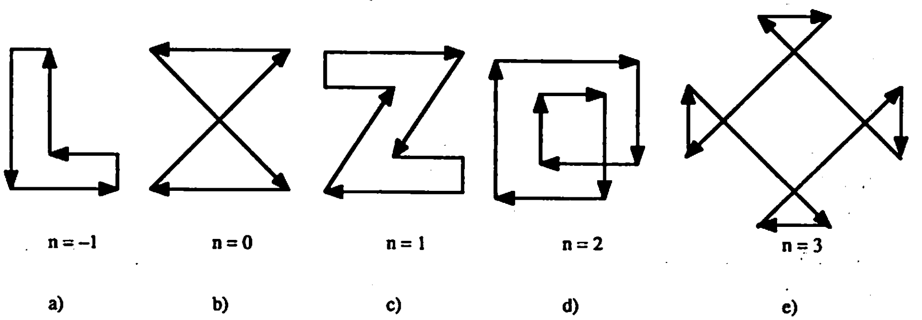


FIGURA 2

1.3. TEOREMA DEL CAMINO CERRADO SIMPLE.

Se llama *camino cerrado simple* al realizado en un viaje completo por la tortuga con la condición de que el camino no se cruce consigo mismo. Son "caminos cerrados simples" los de la Figura 2, casos a) y c); no lo sería b) y d) que tienen un punto de cruce y e) que tiene 4 puntos de cruce.

Una restricción muy importante del teorema del viaje completo lo constituye el *"teorema del camino cerrado simple"* que pasamo a enunciar:

"El índice de rotación de un camino cerrado simple es simpre +1 o -1".

En otras palabras, la veracidad del teorema asegura que el "giro total neto" en un camino cerrado simple es siempre $\pm$ 360° (360° a la derecha o a la izquierda).

Este teorema va a ser fundamental en el siguiente apartado para la generación de los polígonos regulares.

1.4. ISOMORFISMO DE LA GEOMETRÍA EUCLÍDEA Y DE LA GEOMETRÍA DIFERENCIAL DE LA TORTUGA.

Hasta ahora en la enseñanza hemos estado más familiarizados con la Geometría Euclídea que con otras Geometrías; pero no obstante no parece que sea ésta la manera más intuitiva ni más "sintónica" con nuestra experiencia de comprender el espacio y las figuras geométricas. En todo caso la Geometría de la tortuga aporta un punto de vista distinto de describir y representar figuras Geométricas. La diferencia entre ambas Geometrías puede sintetizarse (Abelson H. y diSessa A., 1986) en tres puntos:

1. Intrínseco frente a extrínseco.

Se dice que una propiedad de figura es intrínseca cuando no depende de su relación con un sistema de referencia, sino únicamente de la propia figura. Un triángulo equilátero tiene sus tres lados iguales; esto es una propiedad intrínseca porque no depende de ningún sistema de referencia externo, siempre tiene esa propiedad referido a cualquier sistema. Que un triángulo equilátero tenga una lado horizontal no es una propiedad intrínseca, ya que hace falta tener unos ejes de referencia externos a dicha figura para definir el concepto de "horizontalidad".

Mientras que la Geometría Euclídea se basa toda ella en las distancias de los puntos integrantes de las figura (coordenadas) a unos ejes externos de referencia, la Geometría diferencial de la tortuga se basa en las descripciones intrínsecas de las figuras. Las figuras de la Geometría Euclídea están "clavadas" en un sistema de referencia, mientras que las figuras de la Geometría de la tortuga están "libres" de todo sistema de referencia.

Por ejemplo, la descripción de un triángulo a partir de sus vértices, en la Geometría Euclídea deja clavado al mismo en su sistema de referencia, mientras que la descripción de un cuadrado en la Geometría de la Tortuga produce un triángulo que puede ser dibujado en cualquier posición y orientación y que no depende de ningún sistema de referencia externo.

2. Local frente a Global.

La Geometría Euclídea hace un estudio global de las figuras, mientras que la Geometría de la Tortuga hace un estudio más local de las figuras. Veamos el ejemplo clásico de la descripción de una circunferencia por ambas Geometrías. La ecuación de la Geometría Euiclídea que describe una circunferencia $x^2+y^2=r^2$ está relacionada con un conjunto global de espacio pues define en la misma como un conjunto de puntos que equidistan de otro punto fijo llamado centro. Por el contrario, la descripción que hace la Geometría de la tortuga no necesita para nada ese punto interior llamado centro, sino que opera cada vez en un pequeño trozo de espacio describiendo cada paso a partir del punto anterior de la trayectoria (es decir a partir de la orientación y posición de un punto próximo).

3. Procedimientos frente a ecuaciones.

La manera de describir las líneas que componen las figuras en la Geometría Euclídea es mediante ecuaciones que relacionan las variables que toman los valores de las coordenadas; es un punto de vista estático. La manera de describir las figuras en la Geometría de la Tortuga es mediante procedimientos (secuencia de pasos que debe realizar la tortuga para dibujar las figuras) y el dibujo se consigue a partir del "rastro" que deja la tortuga al moverse; es un punto de vista dinámico ideal para la exploración matemática.

A pesar de las diferencias apuntadas entre las dos Geometrías, ambas aportan descripciones de las mismas realidades geométricas. Las dos Geometrías son isomorfas, de manera que propiedades y teoremas de una de las Geometrías tienen sus propiedades y teoremas análogos en la otra. Por ejemplo, busquemos un teorema análogo al del "camino cerrado simple" antes enunciado:

Recordemos que dicho teorema establece que el "giro total neto" en un camino cerrado simple es siempre $\pm 360^{o}$. Un polígono convexo constituye un camino cerrado simple, de manera que la aplicación del teorema a un triángulo diría: El giro neto total realizado por una tortuga al describir un triángulo es de $\pm 360^{o}$.

En la Figura 3 vemos un triángulo recorrido por la tortuga en el sentido de las agujas del reloj (indice de rotación n = 1). Observese que al llegar a un vértice el giro que debe hacer la tortuga para seguir describiendo la figura viene dado por el ángulo externo.

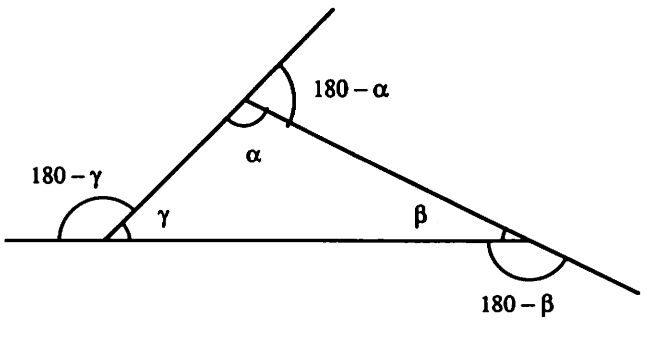


FIGURA 3

Sean α, β, γ los tres ángulos internos y por tanto $(180-\alpha)$, $(180-\beta)$, $(180-\gamma)$ los ángulos externos; en cumplimiento del teorema se verifica:

$(180-\alpha)+(180-\beta)+(180-\gamma)=360^{o}$
y por tanto: $540-(\alpha+\beta+\gamma)=360$.

Y en consecuencia, se obtiene el conocido teorema de la Geometría Euclídea que relaciona lo tres ángulos interiores de un triangulo:

$$\alpha+\beta+\gamma=180.$$

Esta isomorfía sugiere un gran camino de exploración e investigación; se puede partir de un teorema de una de las Geometrías y a partir de él buscar el teorema análogo en la otra. Queda abierta esa idea para llevar a la práctica en éste y otros micromundos geométricos.

2. GENERACIÓN DE POLÍGONOS REGULARES A PARTIR DE LOS TEOREMAS ANTERIORES

Los teoremas del apartado anterior nos sugieren un camino para construir procedimientos que generen polígonos regulares. Comencemos pensando en la descripción de la figura que no es más familiar (el cuadrado), para terminar extendiendo la idea un polígono regular de cualquier número de lados.

2.1. ÁNGULOS EXTERNOS Y ÁNGULOS INTERNOS: CUADRADO Y TRIÁNGULO.

El caso más sencillo es el de la generación del polígono regular de 4 lados: El cuadrado: Basta ordenar a la tortuga: REPEAT 4 [FD 20 RT 90] para que la tortuga genere un cuadrado . (Figura 4 a). Un procedimiento similar al anterior podríamos esperar que generara un triángulo: REPEAT 3 [FD 20 RT 60] y en cambio, genera la Figura 4 b).

Esta incorrección que cometen idefectiblemente los principiantes, comentada suficientemente en la bibliografía, es debida a que no se aplicó correctamente el teorema del camino cerrado simple. Si triángulo es un camino cerrado simple debe cumplirse que el giro neto total realizado por la tortuga deber ser de 360º; como el polígono es regular sus ángulos son iguales y ha girado 3 veces luego cada giro debió ser de 360/3, o sea de 120º, de manera que el procedimiento correcto para generar la figura 4 c) sería: REPEAT 3 [FD 20 RT 120].

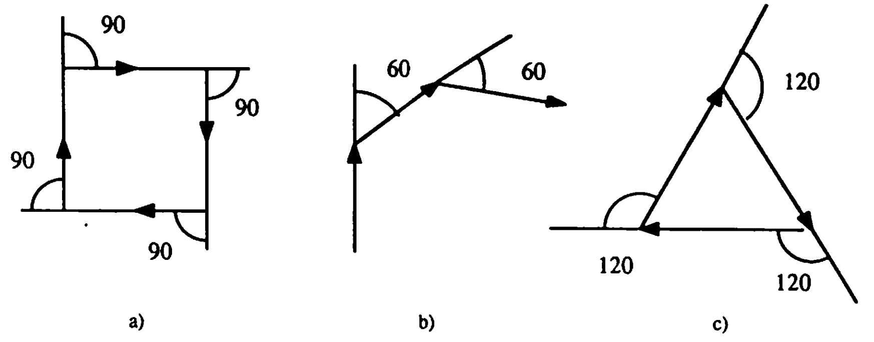


FIGURA 4

La clave de todo estaría en la correcta aplicación del Teorema del camino cerrado simple y en la consideración de que los ángulos de giro, en Geometría de la tortuga son los ángulos externos.

2.2. GENERALIZACIÓN A CUALQUIER NÚMERO DE LADOS.

Supongamos un polígono regular de un número n cualquiera de lados. Si el ángulo de giro de un lado a otro es α (ver figura 5) podemos relacionar n y α por medio del teorema del camino cerrado simple:

$n*\alpha=360$ de donde $\alpha=360/n$

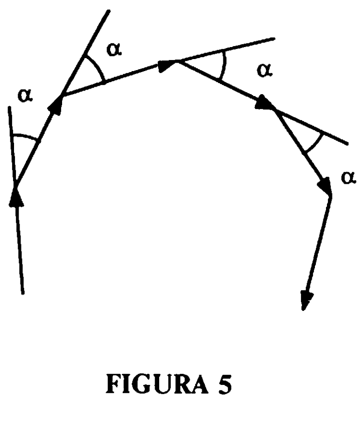


FIGURA 5

El siguiente procedimiento generaría polígonos regulares de lado 30:

```
TO POLI :N
REPEAT :N [FD 30 RT 360/:N]
END
```

Bastaría la adición de una nueva variable que controle el tamaño del lado para tener el procedimiento general:

```
TO POLIG :N :LADO
REPEAT :N [FD :LADO RT 360/:N]
END
```

Una segunda solución más elegante nos lleva a un procedimiento recursivo. La idea es la siguiente: Cuando se describe la tortuga un polígono regular siempre hace lo mismo, avanza un número de pasos y gira un cierto ángulo (siempre el mismo). El procedimiento recursivo POLIR hace justamente esto, pero no se para nunca.

```
TO POLIR :N
FD 30
RT 360 / :N
POLIR
END
```

Bastaría añadir un procedimiento auxiliar con una condición de parada y una variable asignada al tamaño del lado para generar cualquier polígono regular con cualquier tamaño. Los procedimientos podrían ser los siguientes.

```
TO POL :N :LADO
IF :I=:N [STOP]
MAKE "I :I+1
FD :LADO
RT 360 / :N
POL :N :LADO
END
```

```
TO POLY :N :LADO
MAKE "I 0
POL :N :LADO
END
```

Por último se podrían emplear recursos propios de la programación estructurada, mediante la orden DOUNTIL que ejecuta rápidamente una lista de instrucciones hasta que se cumple una condición determinada. Esta orden solo la posee OBJECT-LOGO por lo que para este Logo se podría implementar el procedimiento POLYSTOP que genera igualmente polígonos regulares:

```
TO POLYSTOP :N :LADO
MAKE "GIRO 0
DOUNTIL [FD :LADO RT 360/:N  MAKE "GIRO  :GIRO + (360/:N)]
_  :GIRO = 360
END
```

La clave está en ir almacenando en la variable "GIRO el giro acumulado que va realizando la tortuga al ir describiendo el polígono. Cuando llega a acumular 360º, según el teorema del camino cerrado simple debe haber acabado el polígono y en ese caso DOUNTIL para el procedimiento.

3. LA CIRCUNFERENCIA COMO CASO LÍMITE

Si vamos aumentando el número de lados de un polígono, se va pareciendo cada vez más a una circunferencia. La circunferencia tiene un sentido de límite cuando el número de lados tiende a infinito. En ese caso límite la longitud de cada lado sería infinitesimal. Sin embargo las condiciones de contorno de Logo imponen un giro mínimo de 1 grado y un lado mínimo de 1 paso. La circunferencia más perfecta que pudiéramos construir sería un polígono de 360 lados de tal manera que el ángulo de giro de este polígono sería:

360/360 (nº de lados) = 1 (el mínimo giro posible)

De todas formas, a efectos prácticos, la resolución de los monitores aconseja aceptar como circunferencia un polígono a partir de un número mínimo de 36 lados (un aumento del número de lados no haría parecer al polígono "más circunferencia"). El procedimiento CIRCUN realizaría una circunferencia en las condiciones descritas:

```
TO CIRCUN :LADO
REPEAT 36 [FD :LADO RT 10]
END
```

* *Cambio del parámetro de control del tamaño: del paso al radio.*

Con un lado pequeño se pueden obtener aproximaciones a la circunferencia, pero el problema es que no conocemos el radio de la circunferencia, que es el parámetro que nos dá más idea del tamaño de la circunferencia. Tratemos de construir un procedimiento que dibuje la circunferencia utilizando como variable el radio de la misma, en vez del lado del polígono que la simula.

1ª Opción:

Intentando obtener la máxima precisión (teórica) fijaremos en 1 paso cada desplazamiento y desde este punto de vista el problema es calcular el ángulo de giro entre un paso y el siguiente para obtener la circunferencia completa.

Basta la siguiente regla de tres para obtener dicho ángulo:

$2\pi R$ (pasos) ----------> 360º

de donde $\quad$ 1 (paso) ----------> α

$\alpha = 360/2\pi R$ ----------> $\alpha = 180/\pi R$

Una vez calculado el ángulo, veamos cuantas veces debemos repetir el proceso para obtener la circunferencia completa. Si cada vez gira α y lo hace un número N de veces, para cumplir el Teorema del camino cerrado simple debe verificarse: $N \cdot \alpha = 360$

Despejando N y sustituyendo a en función de R obtenemos:

$$N = 180 \cdot R \cdot \pi$$

El procedimiento que dibujaría la circunferencia, en esta primera opción sería:

```
TO CIRCU :RADIO
LOCAL :ANGULO  LOCAL :N
MAKE "N :PI * :R * 180   MAKE "ANGULO 180 / (:PI * :R)
REPEAT :N [FD 1 RT :ANGULO]
END
```

Pero si deseamos obtener un arco de circunferencia de ángulo β, en vez de la circunferencía completa:

$360^\circ \longrightarrow$ N (veces)

$\beta^\circ \longrightarrow$ n

Despejando n y sustituyendo N por su valor en función de R tenemos:

$n = N \cdot \beta/360$ y por tanto $n = \pi \cdot R \cdot \beta/180$

Con todo ello el procedimiento ARCD dibuja una arco de radio R y ángulo β :

```
TO ARCD :R :BETA
LOCAL :ANGULO  LOCAL :N
MAKE "N :PI * :R * :BETA / 180   MAKE "ANGULO 180 / (:PI * :R)
```

```
REPEAT :N [FD 1 RT :ANGULO]
PR :N PR :ANGULO  PR :N * :ANGULO
END
```

El problema de estos procedimientos es que mientras los cálculos se hacen con bastante precisión y teóricamente el giro total para el procedimiento CIRCU es de 360º, en la práctica tenemos la limitación de que el giro mínimo es de 1 grado; el cálculo del ángulo mediante la fórmula ($\alpha = 180 / \pi R$) dará generalmente un número entero cuya parte decimal no se gira y se acumula a lo largo de los N giros efectuados, de tal manera que el procedimiento no es transparente. Por ejemplo (ver figura .6), partiendo del origen de coordenadas y para R = 2, la posición final en vez de [0 0] es de [0.22 -0.53] y la orientación en vez de 0º o 360º es de 343.73º, por lo que el ángulo de desfase es de 16.27º.

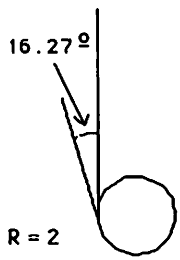


FIGURA 6

Vamos a tener que adoptar otro punto de vista, que consiga que los procedimientos sean transparentes; aunque tengamos que hacer teóricamente una aproximación menos exacta que la anterior, los resultados prácticos van a ser mejores.

2ª Opción:

Volvamos sobre la suposición de que la circunferencia es un polígono de 36 lados en primera aproximación. En la figura 7 se puede observar que, sin el supuesto polígono unimos los vértices con el centro de la circunferencia, a cada lado le corresponde un ángulo central de 360 / 36, es decir de 10º.

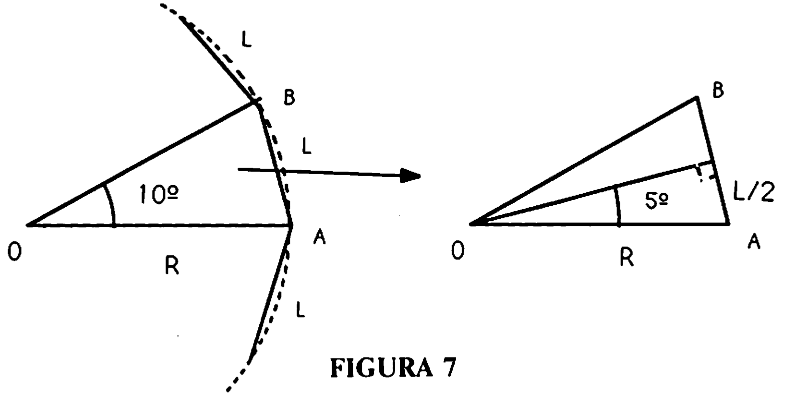


FIGURA 7

Relacionemos L en función de R, que pretendemos sea el dato de entrada:

$$L/2=R \cdot \text{sen } 5^\circ \longrightarrow L=2 \cdot R \cdot \text{sen} 5^\circ$$

En estas condiciones el siguiente procedimiento dibujaría con bastante aproximación una circunferencia de radio la entrada:

```
TO CIR :R
MAKE "LADO 2* :R * SIN 5
REPEAT 36 [RT 5 FD :LADO RT 5]
END
```

Razonando análogamente a como lo hicimos en la opción anterior podemos diseñar un procedimiento para la obtención de arcos de circunferencia de radio dado, determinados por su ángulo central:

$360^\circ \longrightarrow 36$ tramos

$\beta^\circ \longrightarrow n$ tramos

Con lo que el número de tramos a repetir es: $n=36 \cdot \beta/360= \beta/10$

Y el procedimiento sería ya el siguiente:

```
TO ARCODE :R :ANGULO
MAKE "LADO 2* :R * SIN 5    MAKE "N :ANGULO / 10
REPEAT :N [RT 5 FD :LADO RT 5]
END
```

El procedimiento CIR resulta siempre transparente y tiene otra ventaja respecto el procedimiento CIRCU de la primera opción, y es que para radios pequeños CIRCU simula la circunferencia con número pequeños de lados, mientras que CIR la simula siempre con 36 lados. Al contrario, para radios muy grandes, CIRCU tendrá un número muy grande de tramos por lo que se parecerá más a un arco de circunferencia que CIR que sólo tiene 36.

En todo caso en nuestro micromundo dispondremos de un factor de escala y, para un valor por defecto del mismo de 10, los valores habituales del radio de las circunferencias que manejaremos varían entre 2 y 50, por lo que es mucho más conveniente adoptar la segunda opción; de todas formas están implementadas las dos opciones.

5. PERÍMETRO Y AREA DE POLÍGONOS REGULARES

Sería conveniente disponer en nuestro micromundo de procedimientos que nos dieran el perímetro y el área de un polígono regular de cualquier número de lados.

El caso del perímetro es trivial, ya que el perímetro se define como la suma de sus lados, y por tanto:

$$P = N \cdot L$$

(siendo P el perímetro, N el nº de lados y L la longitud del lado)

El procedimiento PERIMETROPOLI nos daría el perímetro de un polígono regular a partir de el número de lados y la longitud de sus lados.

```
TO PERIMETROPOLI :N :LADO
OP :N * :LADO
END
```

Abordaremos ahora el caso del área. Supongamos un polígono regular de un número cualquiera de lados (ver figura 8); triangulemos el polígono mediante isósceles de base el lado del polígono y lados dos radios de la circunferencia que le circunscribe. Es evidente que:

Area Total = N•Area Triángulo

Calculemos el área del triángulo en función de nuestros datos, que son el número de lados y la longitud del lado: $\alpha = 360/2 \cdot N$

y por otra parte t; $\alpha = L/(2 \cdot H)$ $H = L/(2 \cdot tg\ \alpha)$

por tanto el área del triángulo valdrá: Area Triángulo= $L \cdot H / 2 = L^2/ 4 \cdot tg\ \alpha$

y el área total será: Area Total = $N \cdot LL^2/ (4 \cdot tg\ \alpha)$

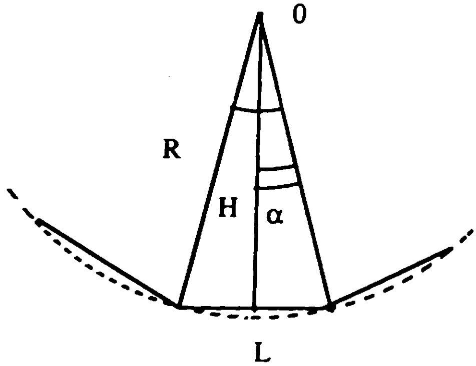


FIGURA 8

El procedimiento AREAPOLI nos daría el área de un polígono a partir del número de lados y de la longitud del lado.

```
TO AREAPOLI :N :LADO
MAKE "ANGULO 360 / (2 * :N)
MAKE "AREA :N * :LADO* :LADO / (4 * TAN :ANGULO)
OP :AREA
END
```

5. EL MICROMUNDO IMPLEMENTADO. ESTRUCTURA

Pretendemos implementar un micromundo que permita la manipulación de polígonos regulares en el plano. El tema ya ha sido tratado por otros autores como (Roanes Macías y Roanes Lozano E., 1988), (Arias J.M. y Bélanger J.E., 1988), (R. Roselló, 1986), (Rode, M. y Silva G, 1986), (Cajaraville, J.A., 1989), (Abelson H. diSessa A, 1986). Casi todos los libros de Logo incluyen procedimientos que generan polígonos regulares, otros incluyen procedimientos que calculan áreas y perímetros etc... ¿en qué se diferencia nuestro micromundo de las aportaciones precedentes?. Analicemos en primer lugar lo que hacen otros autores.

La mayoría programan en Logo utilizándolo como lenguaje de programación, como pudiera haber programado en BASIC, PASCAL o cualquier otro pero la previsible utilización por parte del usuario entra de lleno en el estilo de la enseñanza asistida por ordenador; es decir que el programa interroga al usuario, dialoga con él, le solicita datos y le apoya en la resolución de problemas según un plan previsto. Generalmente en estos casos los procedimientos son terminales, es decir que no se pueden enlanzar a otros nuevos para que el usuario siga complementando el micromundo.

Algunos otros autores utilizan estos procedimientos que generan polígonos regulares más para ilustrar el aprendizaje del propio Logo que para ser utilizados como un micromundo; es decir para explicar el funcionamiento de las estructuras iterativas, recursivas, para enseñar a explorar a modo "bricolage" etc...

En nuestro caso hemos tratado de implementar un *micromundo* para ser utilizado con el método y la filosofía Logo para el estudio de los polígonos regulares. No está pensado ni como un programa de enseñanza asistida, ni como una ilustración de las estructuras de Logo sino como una caja de herramientas:

Aquí vamos a encontrar un conjunto de procedimientos que definen figuras, pueden borrarlas, situarlas en diversos lugares de un espacio predefinido, hacer cálculos a partir de las propias características de las figuras etc... junto con una ayuda que contiene las instrucciones que indican cómo funciona cada herramienta. La mayoría de las herramientas se pueden ensamblar para formar una "macroherramiena" que puede hacer muchas más cosas. No hay un plan trazado a seguir en una clase entre profesores y alumnos para la utilización del micromundo; los ejemplos y aplicaciones con sugerencias de los múltiples caminos que pueden seguirse en el proceso de enseñanza-aprendizaje cuando se está utilizando el micromundo.

El micromundo está estructurado en siete secciones, tal como se muestra en la Figura 9, y utiliza los potentes recursos del Object-Logo que es el lenguaje de programación con el que se ha implementado; pasamos a comentar brevemente dichas secciones:

1. CREACIÓN DEL ESPACIO.

Consta de tres procedimientos (COMIENZO, CRA.AYUDA Y ESCALA) que tienen como misión preparar el espacio de la pantalla y crear un menú específico que contiene la ayuda on-line.

2. DEFINICIONES POLÍGONOS (Opción Vértice).

En esta sección se agrupan las primitivas que permiten la generación de polígonos regulares, a la manera que podemos denominar "clásica", es decir pintando el polígono a partir de un vértice desde el lugar en que se encuentre la tortuga. En la siguiente sección se tomará una opción distinta más propia del manejo de nuestro micromundo; en todo caso las primitivas de la sección

3 se apoyan en buena parte sobre las de la sección 2 y por ello la estructura de ambas secciones se presenta conjuntamente en la Figura 10.

3. DEFINICIONES POLÍGONOS (Opción Centro).

En esta sección se agrupan doce primitivas estructuradas tal como se ilustra en la Figura 10. Cada polígono regular se concibe aquí como un ente matemático dotado de ciertas propiedades que se asignan mediante su definición (nombre, nº de lados, longitud del lado y centro) y que se almacenan en una lista de propiedades y otras características calculables, en cada momento que se necesiten, a partir de las anteriores (radio, apotema, perímetro, área).

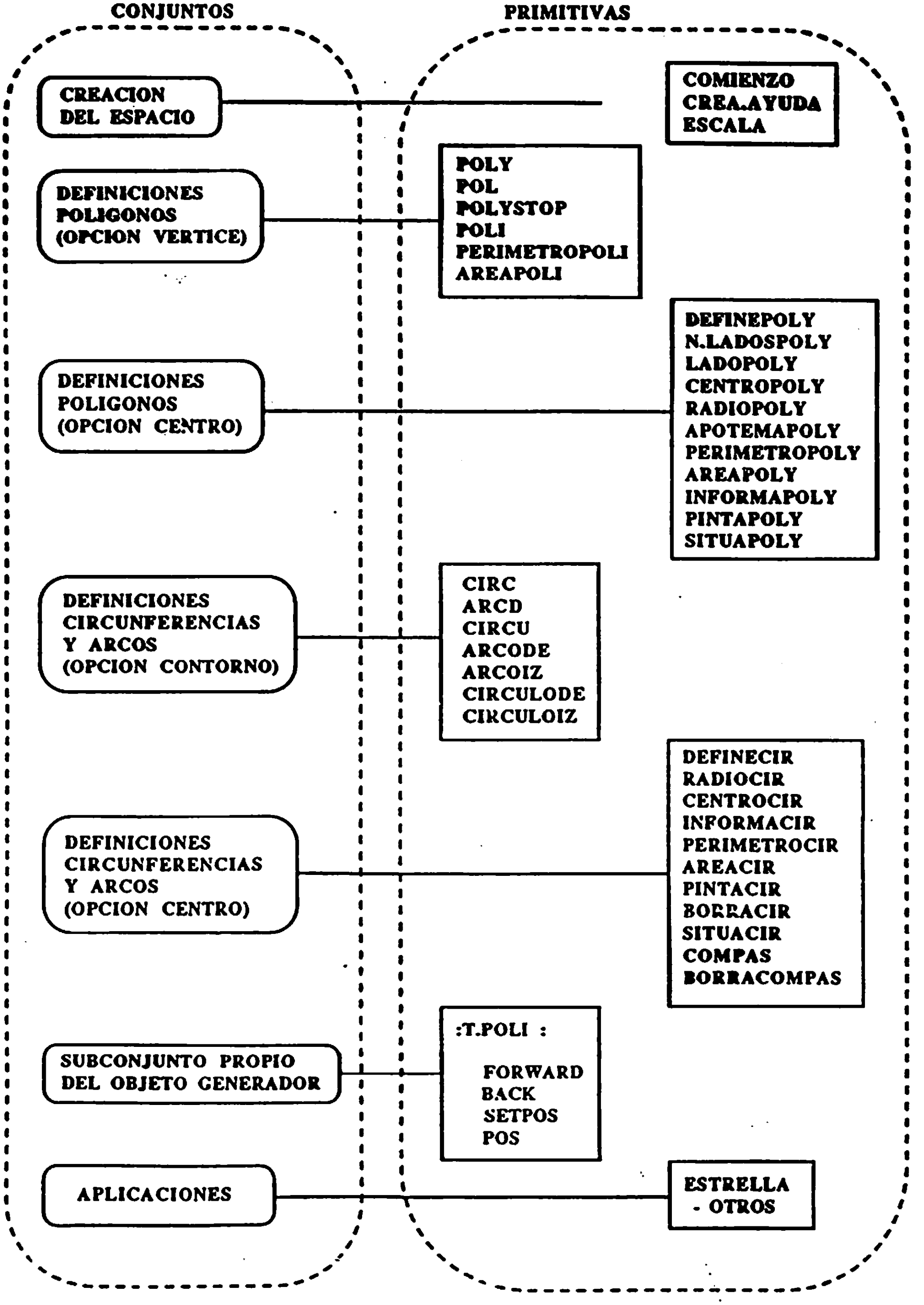
4. DEFINICIONES CIRCUNFERENCIAS Y ARCOS (Opción Contorno).

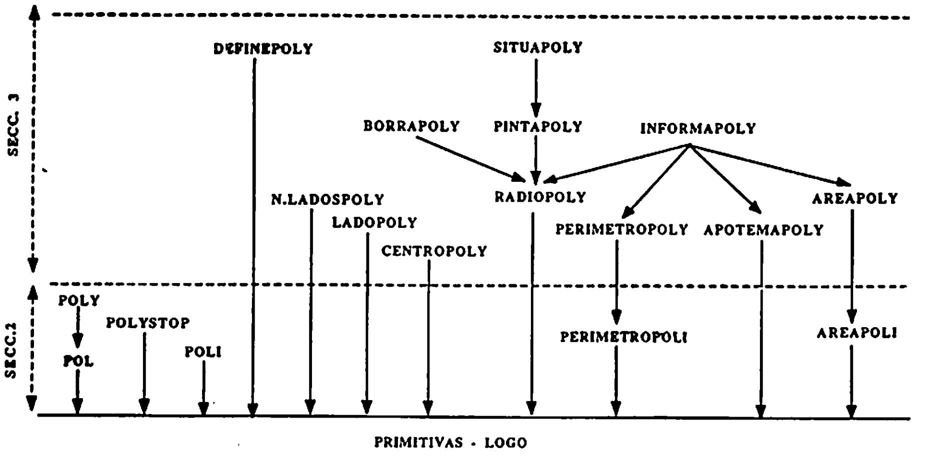
De una manera análoga a las secciones 2 y 3 se han diseñado las secciones 4 y 5 que permiten la generación de circunferencias y arcos; en la sección 4 hemos agrupados las primitivas del micromundo que permiten dibujar circunferencias y arcos a la manera que podemos denominar "clásica, es decir pintando a partir del contorno desde el lugar en que se encuentre la tortuga. En la siguiente sección se tomará una opción distinta más propia del manejo de nuestro micromundo; en todo caso las primitivas de la sección 5 se apoyan en algunas de la sección 4 y por ello la estructura de ambas secciones se presenta conjuntamente en la Figura 11.

En la Sección 4 se agrupan siete procedimientos que perrmiten generar arco (ARCD, ARCODE y ARCOIZ) y circunferencias (CIRC, CIRCU, CIRCULODE y CIRCULOIZ). Los fundamentos del diseño de estos procedimientos ya se explicaron en el punto 3, por lo que nos remitimos a ellos.

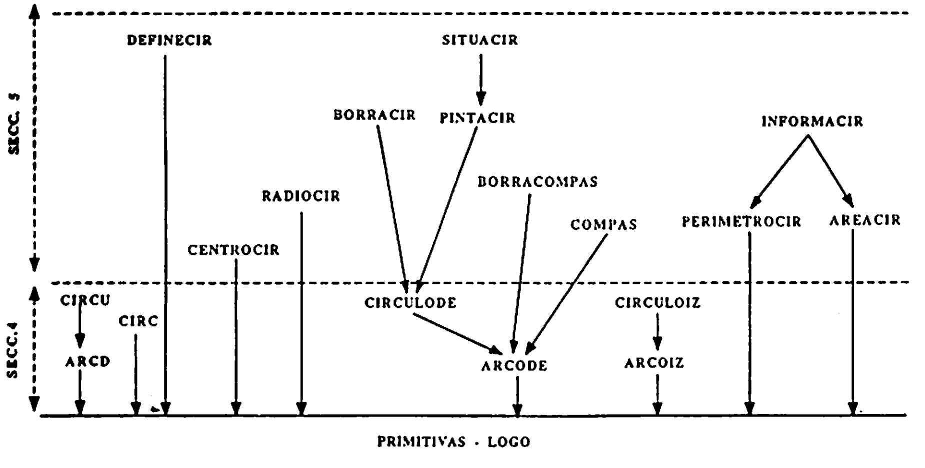
5. DEFINICIONES CIRCUNFERENCIAS Y ARCOS (Opción Centro).

En esta sección se agrupan once primitivas estructuradas tal como se ilustra en la Figura 11. Cada circunferencia,se concibe aquí como un ente mate mático dotado de ciertas propiedades que se asignan mediante su definición (nombre, radio y centro) y que se almacenan en una lista de propiedades y otras características calculables, en cada momento que se necesiten, a partir de las anteriores (perímetros, área).


CONJUNTOS
PRIMITIVAS
CREACION DEL ESPACIO
COMIENZO
CREA.AYUDA
ESCALA
DEFINICIONES POLIGONOS (OPCION VERTICE)
POLY
POL
POLYSTOP
POLI
PERIMETROPOLI
AREAPOLI
DEFINICIONES POLIGONOS (OPCION CENTRO)
DEFINEPOLY
N.LADOSPOLY
LADOPOLY
CENTROPOLY
RADIOPOLY
APOTEMAPOLY
PERIMETROPOLY
AREAPOLY
INFORMAPOLY
PINTAPOLY
SITUAPOLY
DEFINICIONES CIRCUNFERENCIAS Y ARCOS (OPCION CONTORNO)
CIRC
ARCD
CIRCU
ARCODE
ARCOIZ
CIRCULODE
CIRCULOIZ
DEFINICIONES CIRCUNFERENCIAS Y ARCOS (OPCION CENTRO)
DEFINECIR
RADIOCIR
CENTROCIR
INFORMACIR
PERIMETROCIR
AREACIR
PINTACIR
BORRACIR
SITUACIR
COMPAS
BORRACOMPAS
SUBCONJUNTO PROPIO DEL OBJETO GENERADOR
:T.POLI :
FORWARD
BACK
SETPOS
POS
APLICACIONES
ESTRELLA
- OTROS



SECC. 3
SECC. 2
DEFINEPOLY
SITUAPOLY
BORRAPOLY
PINTAPOLY
INFORMAPOLY
N.LADOSPOLY
RADIOPOLY
AREAPOLY
LADOPOLY
PERIMETROPOLY
APOTEMAPOLY
CENTROPOLY
POLY
POLYSTOP
POL
POLI
PERIMETROPOLI
AREAPOLI
PRIMITIVAS - LOGO



SECC. 5
SECC. 4
DEFINECIR
SITUACIR
BORRACIR
PINTACIR
INFORMACIR
BORRACOMPAS
RADIOCIR
COMPAS
PERIMETROCIR
AREACIR
CENTROCIR
CIRCU
CIRC
CIRCULODE
CIRCULOIZ
ARCD
ARCODE
ARCOIZ
PRIMITIVAS - LOGO

6. APLICACIONES.

En esta sección se agruparían los procedimientos generados por el usuario a partir de las primitivas del micromundo. Como ejemplo hemos incluido el procedimiento ESTRELLA que pinta dos polígonos concéntricos girados uno respecto del otro (180 /N) siendo N el nº de lados del polígono.

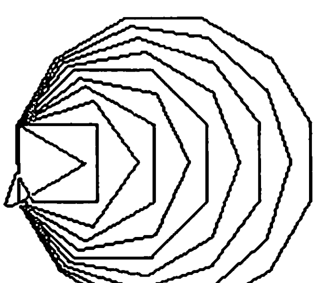
7. DEFINICIONES PROPIAS PARA LA TORTUGA T. POLI.

Es necesario, antes de cargar esta sección que se ejecute el procedimiento COMIENZO que crea la tortuga T.POLI, antes de dotar a la misma de procedimientos propios; por ello al comenzar la sección se incluye en archivo la linea ejecutable RUN [COMIENZO].

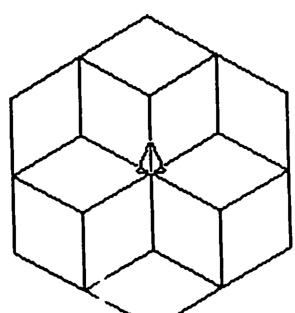
En la Sección 1 (Establecimiento del espacio de trabajo) se diseñó el procedimiento COMIENZO que creó e inicializó una tortuga llamda T.POLI; dicha tortuga es la que nos permite dibujar en el micromundo. Se definió también mediante el procedimiento ESCALA el factor de escala que regiría el tamaño de la unidad de nuestro micromundo en función de pasos de tortuga. En este apartado generamos 4 procedimientos que tienen como misión adaptar las órdenes de movimiento y posición a la escala definida; aprovechando que T.POLI es un objeto al que puedo dotar de procedimientos propios que "tapen" las definiciones homólogas de la tortuga genérica hemos definido los siguientes: FORWARD, BACK, SETPOS y POS.

8. LA PRÁCTICA EN EL MICROMUNDO "POLÍGONOS REGULARES EN EL PLANO". Ejemplos.

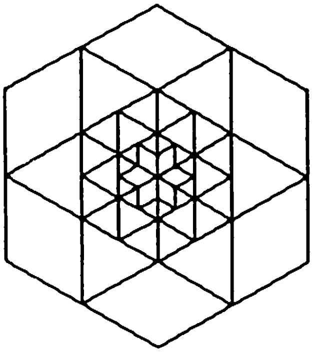
Se incluyen algunas ejecuciones interesantes, cuyos procedimientos generadores se encuentran en anexo.

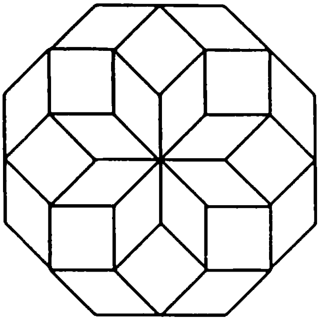


a) MULTIPOL 10 4

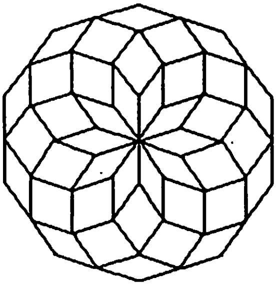


b) TRESCUBOS 4

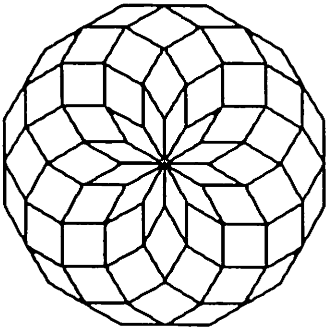




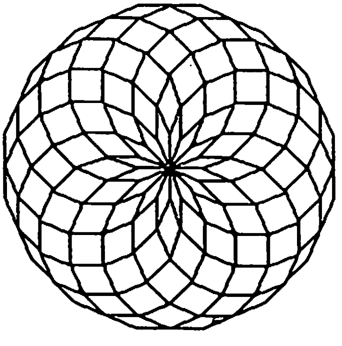
c) TRESCUBOS 1 TRESCUBOS 2 TRESCUBOS 4 d) ROTOPOLI 8 3



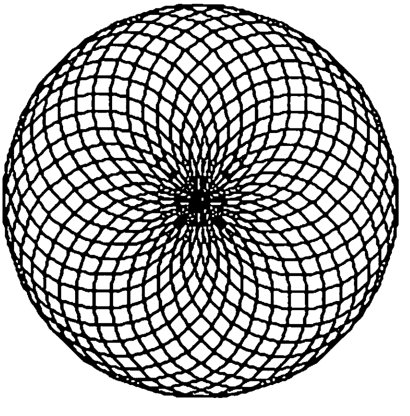
ROTOPOLI 10 2



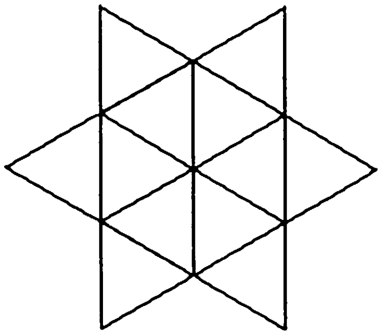
ROTOPOLI 12 2



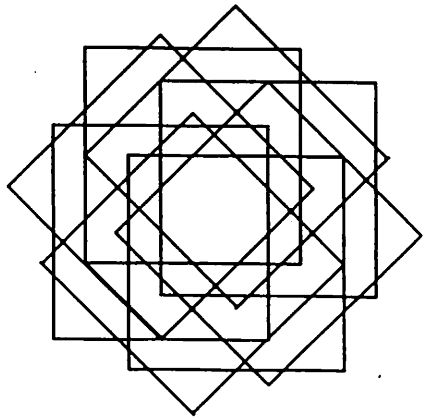
ROTOPOLI 16 1.5



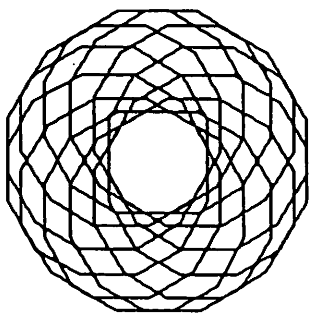
ROTOPOLI 36 0.8



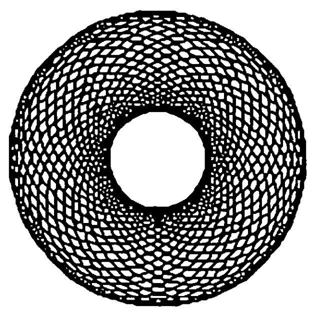
FORMA.1 6 3 10



FORMA.1 8 4 10



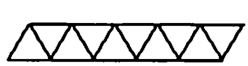
FORMA.1 12 12 2.5



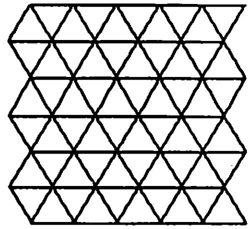
FORMA.1 36 36 0.8

*** Mosaico regular triangular**

a) TIRATRI.1 10 2

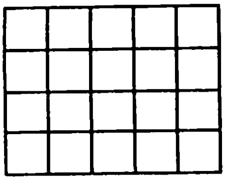


b) TIRATRI.2 10 2

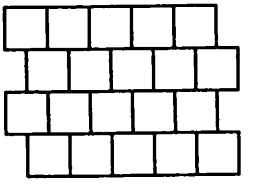


c) BALDOSATRI 6 10 2

*** Mosaíco regular cuadrado**

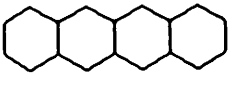


a) BALDOSACUA.1 4 5 2

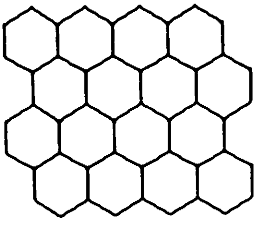


b) BALDOSACUA.2 4 5 2

*** Mosaico regular hexagonal**

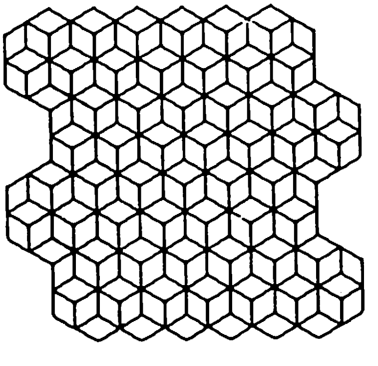


a) TIRAHEXA 4 1.5



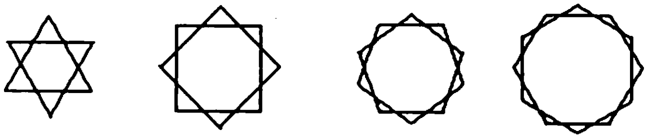
b) BALDOSAHEXA 4 4 1.5

*** Otros mosaicos**

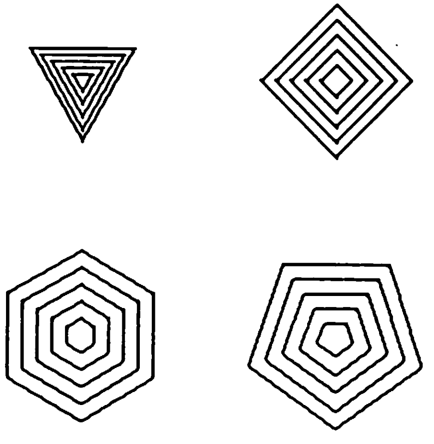


BALDOSACUBOS

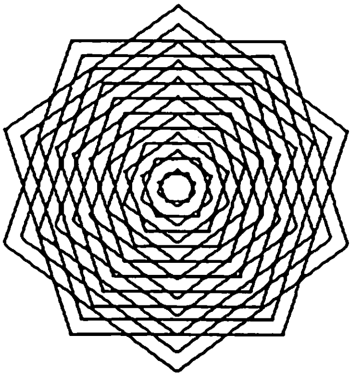
*** Estrellas**



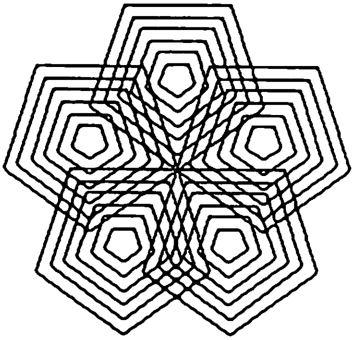
*** Policentricos:**



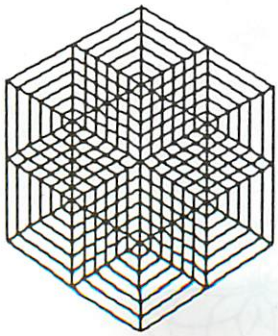
*** MALLACENTRICA Y TELARAÑA**



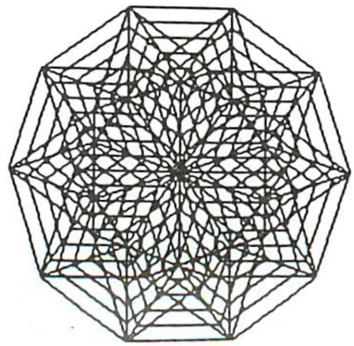
MALLACENTRICA [0 0] 5 1 1 10



TELARANA [0 0] 5 1 0.8 6 5

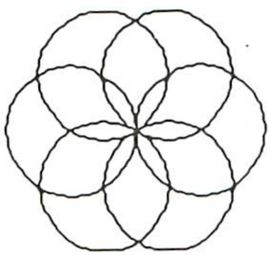


TELARANA.3 [0 6] 6 0.6 0.6 6 6

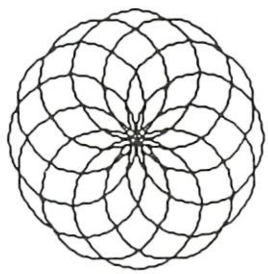


TELARANA.3 [0 6] 5 1 1 5 10

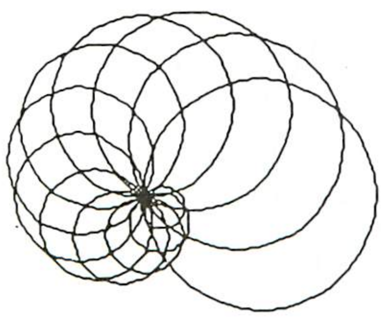
*** Arcos y Círculos:**



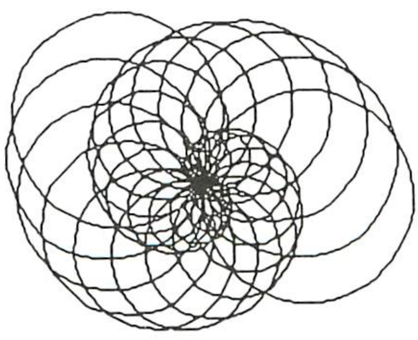
ROTOCIR 3 6



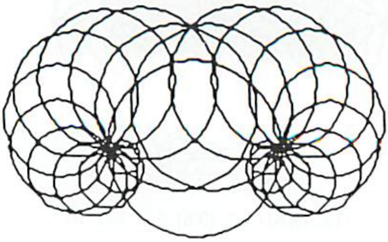
ROTOCIR 3 12



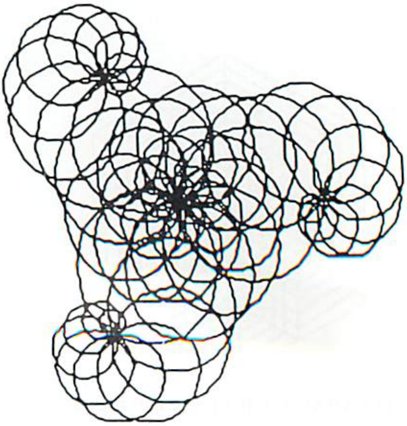
CARACOL 1 12 0.4



OVILLO 0.5 16 0.3 2



MADEJA 0.5 12 0.3

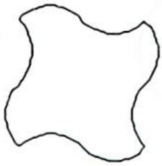


REPEAT 3 [MADEJA 0.3 10 0.3 RT 120]

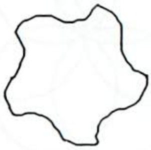
*** Polígonos curvos:**

LADOCURVO 5

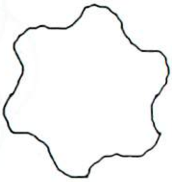
POLICURVO 3 3



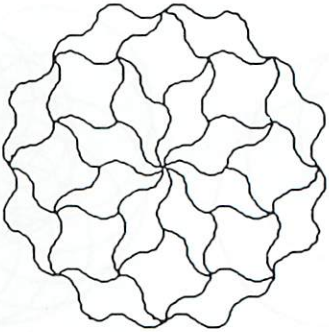
POLICURVO 4 3



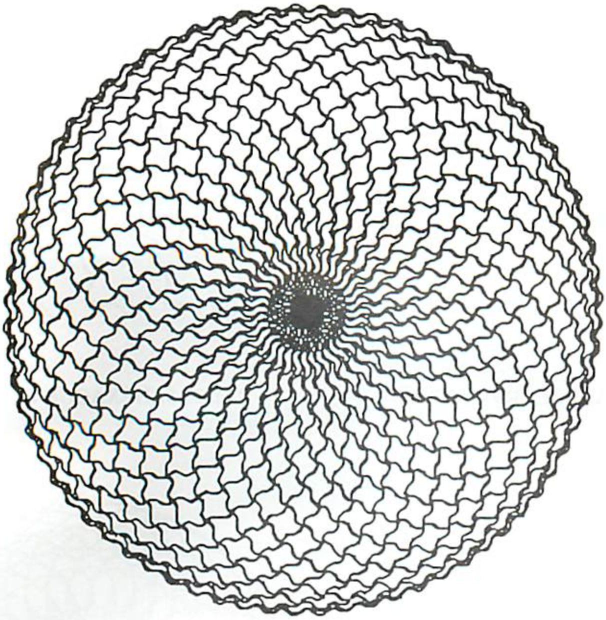
POLICURVO 5 2



POLICURVO 6 2

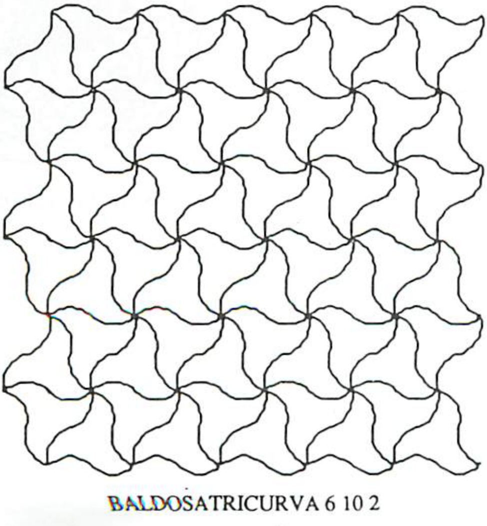


ROTOPOLICURVO 8 1.5

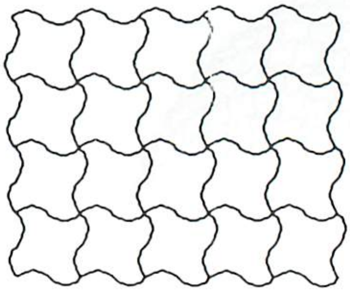


ROTOPOLICURVO 36 0.6

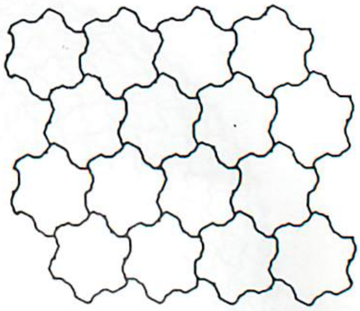
*** Mosaicos con *policurvos*.**



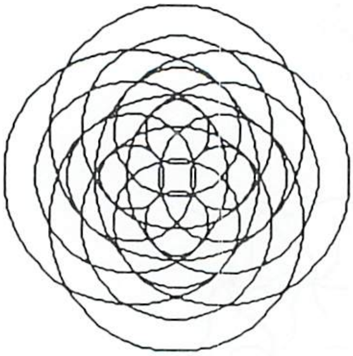
BALDOSATRICURVA 6 10 2



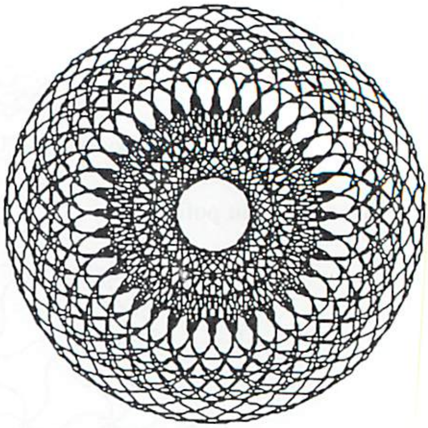
BALDOSACUACURVA 4 5 1.5



BALDOSAHEXACURVA 4 4 1



ROTOCIRCENTRICOS 4 [0 0] 3 1.5 1.5 4



ROTOCIRCENTRICOS 36 [0 0] 1 1 1 4

ANEXO : LISTADO DE PROCEDIMIENTOS:

;..... MICROMUNDO POLIGONOS REGULARES EN EL PLANO ....

1. Establecimiento del espacio de trabajo :

```
TO COMIENZO
MAKE "VENTANA ONEOF TURTLEWINDOW
ASK :VENTANA [SETWTITLE "|POLIGONOS.R/PLANO| SETWSIZE [350 300]
_ SETWPOS [159 39]]
MAKE "TEXTO.POLIG.PLANO ONEOF LISTENER
ASK :TEXTO.POLIG.PLANO [SETWTITLE "TEXTO.POL.PLANO SETWSIZE [156 300]
_ SETWPOS [1 39]]
MAKE "T.POLI ONEOF TURTLE
CREA.AYUDA
MAKE "K 10  MAKE "PI 3.1415926535
SETDIGITS [3 2]   SETDEFAULTTURTLE :T.POLI
BURYALL ERALL RECYCLE
END

TO CREA.AYUDA
MAKE "MENU.P.PLANO ONEOF MENU
ASK :MENU.P.PLANO [INSTALL]
MAKE "NUEVOITEM ONEOF MENUITEM
ASK :NUEVOITEM [SETTITLE "AYUDA SETACTION [EDITFILE
_ "AYUDA.POLI.PLANO]]
ASK :MENU.P.PLANO [SETTITLE "POLIGONOS.PLANO ADDITEM :NUEVOITEM]
END

TO ESCALA :E
PR [SI DECIDE CAMBIAR DE ESCALA PERDERA TODOS SUS DIBUJOS]
PR [DECIDE CAMBIO DE ESCALA: SI (S) / NO (CUALQUIER TECLA)]
LOCAL "RES  MAKE "RES RC
IF :RES= "S [CS MAKE "K :E STOP]
PR [ANULADO EL CAMBIO DE ESCALA]
END
```

2. Definiciones poligonos (Opción vértice)

```
TO POLY :N :LADO
PUBLIC "I MAKE "I 0
POL :N :LADO
END
```

```
TO POLI :N :LADO
REPEAT :N [FD :LADO RT 360/:N]
END
```

```
TO POLYSTOP :N :LADO
LOCAL "GIRO MAKE "GIRO 0
DOUNTIL [FD :LADO RT 360/:N
_MAKE "GIRO :GIRO+(360/:N)] :GIRO=360
END

TO AREAPOLI :N :LADO
LOCAL "ANGULO  MAKE "ANGULO 180/:N
OP :N * :LADO * :LADO / (4* TAN :ANGULO)
END
```

3. Definiciones poligonos (opción Centro)

```
TO DEFINEPOLY :NOMBRE :NUM :LADO
PUTPROP :NOMBRE "N :NUM
PUTPROP :NOMBRE "L :LADO
PUTPROP :NOMBRE "CENTRO [0 0]
PR SE "DEFINIDO.EL.POLIGONO :NOMBRE
END

TO RADIOPOLY :NOMBRE
LOCAL "ANGULO LOCAL "RADIO
MAKE "ANGULO 180/GETPROP :NOMBRE "N
MAKE "RADIO (GETPROP :NOMBRE "L)/(2*SIN ANGULO)
OP :RADIO
END

TO APOTEMAPOLY :NOMBRE
LOCAL "ANGULO LOCAL "APOTEMA
MAKE "ANGULO 180/GETPROP :NOMBRE "N
MAKE "APOTEMA (GETPROP :NOMBRE "L)/
_ (2*TAN :ANGULO)
OP :APOTEMA
END

TO INFORMAPOLY :NOMBRE
PR SE "NUMERO.DE.LADOS: (GETPROP :NOMBRE "N)
PR SE "LADO: (GETPROP :NOMBRE "L)
PR SE "CENTRO: (GETPROP :NOMBRE "CENTRO)
PR SE "RADIO: RADIOPOLY :NOMBRE
PR SE "APOTEMA: APOTEMAPOLY :NOMBRE
PR SE "PERIMETRO: PERIMETROPOLY :NOMBRE
PR SE "AREA: AREAPOLY :NOMBRE
END

TO POL :N :LADO
IF :I=:N [STOP]
MAKE "I :I+1
FD :LADO  RT 360/:N
POL :N :LADO
END

TO PERIMETROPOLI :N :LADO
OUTPUT :N * :LADO
END

TO N.LADOSPOLY :NOMBRE
GETPROP :NOMBRE "N
END

TO CENTROPOLY :NOMBRE
GETPROP :NOMBRE "CENTRO
END

TO LADOPOLY :NOMBRE
GETPROP :NOMBRE "L
END

TO PERIMETROPOLY :NOMBRE
PERIMETROPOLI (GETPROP :NOMBRE "N)
_ (GETPROP :NOMBRE "L)
END

TO AREAPOLY :NOMBRE
AREAPOLI (GETPROP :NOMBRE "N)
_ (GETPROP :NOMBRE "L)
END

TO PINTAPOLY :NOMBRE
LOCAL "ANGULO LOCAL "PLUMA
MAKE "PLUMA PENMODE
MAKE "ANGULO 180/GETPROP :NOMBRE "N
PU BK RADIOPOLY :NOMBRE
LT (90-:ANGULO) PD
POLI (GETPROP :NOMBRE "N) (GETPROP :NOMBRE "L)
RT (90-:ANGULO) PU  FD RADIOPOLY :NOMBRE
SETPENMODE :PLUMA
END
```

```
TO BORRAPOLY :NOMBRE
LOCAL "ANGULO LOCAL "PLUMA
MAKE "PLUMA PENMODE
MAKE "ANGULO 180/GETPROP :NOMBRE "N
PU BK RADIOPOLY :NOMBRE
LT (90-:ANGULO) PE
POLI (GETPROP :NOMBRE "N) (GETPROP :NOMBRE "L)
RT (90-:ANGULO) PU   FD RADIOPOLY :NOMBRE
SETPENMODE :PLUMA
END
```

```
TO SITUAPOLY :NOMBRE :LUGAR
LOCAL "PLUMA MAKE "PLUMA PENMODE
PUTPROP :NOMBRE "CENTRO :LUGAR
PU SETPOS :LUGAR
PINTAPOLY :NOMBRE
SETPENMODE :PLUMA
END
```

4. Definiciones circunferencias y arcos (Opción contorno)

```
TO CIRCU :R
LOCAL "POSI LOCAL "ANGULO
MAKE "POSI POS MAKE "ANGULO HEADING
ARCD :R 360
SETPOS :POSI   SETH :ANGULO
END
```

```
TO ARCD :R :ALFA
LOCAL "ANGULO LOCAL "N
MAKE "N :PI * :R * :ALFA / 180  MAKE "ANGULO 180 / (:PI * :R)
REPEAT :N [FD 1 RT :ANGULO]
PR :N PR :ANGULO   PR :N * :ANGULO
END
```

```
TO ARCODE :R :ALFA
LOCAL "LADO  LOCAL "N
MAKE "LADO 2 * :R * SIN 5 MAKE "N 36 * :ALFA / 360
REPEAT :N [RT 5 FD :LADO RT 5]
END
```

```
TO CIRCULODE :R
ARCODE :R 360
END
```

```
TO CIRC :LADO
POLIG 36 :LADO
END
```

```
TO ARCOIZ :R :ALFA
LOCAL "LADO  LOCAL "N
MAKE "LADO 2 * :R * SIN 5
MAKE "N 36 * :ALFA / 360
REPEAT :N [LT 5 FD :LADO LT 5]
END
```

```
TO CIRCULOIZ :R
ARCOIZ :R 360
END
```

5. Definiciones circunferencias y arcos (Opción centro)

```
TO DEFINECIR :NOMBRE :RADIO
PUTPROP :NOMBRE "R :RADIO
PUTPROP :NOMBRE "CENTRO [0 0]
PR SE "DEFINIDO.EL.CIRCULO :NOMBRE
END
```

```
TO INFORMACIR :NOMBRE
PR SE "RADIO: (GETPROP :NOMBRE "R)
```

```
TO RADIOCIR :NOMBRE
GETPROP :NOMBRE "R
END
```

```
TO CENTROCIR :NOMBRE
GETPROP :NOMBRE "CENTRO
END
```

```
PR SE "CENTRO: (GETPROP :NOMBRE "CENTRO)
PR SE "PERIMETRO: PERIMETROCIR :NOMBRE
PR SE "AREA: AREACIR :NOMBRE
END

TO PERIMETROCIR :NOMBRE
LOCAL "PERIM  MAKE "PERIM 2*:PI*
_ (GETPROP :NOMBRE "R)
OP :PERIM
END

TO PINTACIR :NOMBRE
LOCAL "PLUMA
MAKE "PLUMA PENMODE
PU BK (GETPROP :NOMBRE "R)  LT 90 PD
CIRCULODE (GETPROP :NOMBRE "R)
RT 90 PU  FD (GETPROP :NOMBRE "R)
SETPENMODE :PLUMA
END

TO SITUACIR :NOMBRE :LUGAR
LOCAL "PLUMA  MAKE "PLUMA PENMODE
PUTPROP :NOMBRE "CENTRO :LUGAR
PU SETPOS :LUGAR  PINTACIR :NOMBRE
SETPENMODE :PLUMA
END

TO BORRACOMPAS :CENTRO :INICIO :GRADOS
LOCAL "PLUMA  LOCAL "RADIO
MAKE "PLUMA PENMODE
MAKE "RADIO SQRT (((FIRST :INICIO)
 - (FIRST :CENTRO)) ^ 2 +
_ ((LAST :INICIO) - (LAST :CENTRO)) ^ 2)
PU  SETPOS  :CENTRO  SETHEADING 0
RT  TOWARDS  :INICIO
FD :RADIO RT 90 PE
ARCODE :RADIO :GRADOS
PU LT 90 BK :RADIO
SETPENMODE :PLUMA
END

TO AREACIR :NOMBRE
LOCAL "AREA
MAKE "AREA :PI*(GETPROP
NOMBRE "R)*(GETPROP :NOMBRE "R)
:OP :AREA
END

TO BORRACIR :NOMBRE
LOCAL "PLUMA
MAKE "PLUMA PENMODE
PU  BK (GETPROP :NOMBRE "R)  LT 90 PE
CIRCULODE (GETPROP :NOMBRE "R)
RT 90 PU  FD (GETPROP :NOMBRE "R)
SETPENMODE :PLUMA
END

TO COMPAS :CENTRO :INICIO :GRADOS
LOCAL "PLUMA  LOCAL "RADIO
MAKE "PLUMA PENMODE
MAKE "RADIO SQRT (((FIRST
_:INICIO) - (FIRST :CENTRO)) ^ 2 +
_((LAST :INICIO) - (LAST :CENTRO)) ^ 2)
PU SETPOS :CENTRO  SETHEADING 0
RT TOWARDS :INICIO
FD :RADIO RT 90 PD
ARCODE :RADIO :GRADOS
PU LT 90 BK :RADIO
SETPENMODE :PLUMA
END
```

6. Aplicaciones

```
TO ESTRELLA :NOMBRE
PINTAPOLY :NOMBRE
RT 180 / (N.LADOSPOLY :NOMBRE)
```

```
PINTAPOLY :NOMBRE
LT 180 / (N.LADOSPOLY :NOMBRE)
END
```

7. Definiciones propias para la tortuga T.POLI

```
RUN [COMIENZO]

ASK :T.POLI [TO FORWARD :PASO]
IF (NUMBERP :K) = "FALSE [ESCALA]
USUAL.FORWARD :PASO * :K
END

ASK :T.POLI [TO BACK :PASO]
IF (NUMBERP :K) = "FALSE [ESCALA]
USUAL.BACK :PASO * :K
END

ASK :T.POLI [TO SETPOS :LISTA]
IF (NUMBERP :K) = "FALSE [ESCALA]
USUAL.SETPOS SE (FIRST :LISTA) * :K
_ (LAST :LISTA) * :K
END

ASK :T.POLI [TO POS]
IF (NUMBERP :K) = "FALSE [ESCALA]
OUTPUT SE (FIRST USUAL.POS) /:K
_ (LAST USUAL.POS)/:K
END

RUN [CT PR [BIENVENIDO AL MICROMUNDO "POLIGONOS REGULARES EN EL PLANO"]]
RUN [PR [LISTO PARA COMENZAR]]
```

EJEMPLOS:

* **EJEMPLO 1 : Exploraciones en polígonos regulares (opción vértice)**

```
TO TRESCUBOS :L
REPEAT 6 [POLI 6 :L RT 60]
END

TO FORMA.1 :N :N.LADOS :L
REPEAT :N [FD :L / 2 RT 360 / :N
 POLI :N.LADOS :L]
END

TO MULTIPOL :N.POL :L
LOCAL "I MAKE "I 3
REPEAT :N.POL [POLI :I :L MAKE "I :I + 1]
END

TO ROTOPOLI :N.LADOS :L
REPEAT :N.LADOS [POLI :N.LADOS :L
_ RT 360 / :N.LADOS]
END
```

* **Mosaico regular triangular:**

```
TO TIRATRI.1 :NT :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
PD
REPEAT :NT / 2 [LT 30 POLI 3 :L RT 60
POLI 3 :L RT 60 FD :L LT 90]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

```
TO TIRATRI.2 :NT :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
PD
REPEAT :NT / 2 [RT 30 POLI 3 :L RT 60 FD
 :L LT 120 POLI 3 :L RT 30]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

```
TO BALDOSATRI :NF :NT :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
REPEAT :NF / 2 [TIRATRI.1 :NT L LT 30 FD :L RT 30 TIRATRI.2 :NT L RT 30 FD :L LT 30]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

* **Mosaico regular cuadrado :**

```
TO BALDOSACUA.1 :NF :NC :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
REPEAT :NF [TIRACUA :NC :L FD :L]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

```
TO TIRACUA :NC :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
PD
REPEAT :NC [POLI 4 :L RT 90 FD :L LT 90]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

```
TO BALDOSACUA.2 :NF :NC :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
REPEAT :NF / 2 [TIRACUA :NC :L FD :L LT 90
_ FD :L / 2 RT 90 TIRACUA :NC :L FD :L
_ RT 90 FD :L / 2 LT 90]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

* **Mosaico regular hexagonal :**

```
TO TIRAEXA :NE :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
```

```
PD
REPEAT :NE [POLI 6 :L RT 120 FD :L LT 60 FD :L LT 60]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END

TO BALDOSAEXA :NF :NE :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
REPEAT :NF / 2 [TIRAEXA :NE :L FD :L LT 60 FD :L RT 60
_TIRAEXA :NE :L FD :L RT 60 FD :L LT 60]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

• Otros mosaicos :

```
TO TIRACUBOS :N :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
PD
REPEAT :N [TRESCUBOS :L RT 120 FD :L LT 60 FD :L LT 60]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END

TO BALDOSACUBOS :NF :N :L
LOCAL "LUGAR MAKE "LUGAR POS
LOCAL "PLUMA MAKE "PLUMA PENMODE
REPEAT :NF / 2 [TIRACUBOS :N :L FD :L LT 60 FD :L*2 RT 60 FD :L
_ TIRACUBOS :N :L PU FD :L RT 60 FD :L*2 LT 50 FD :L PD]
PU
SETPOS :LUGAR  SETPENMODE :PLUMA
END
```

BIBLIOGRAFÍA

* ABELSON, H. (1.982): A Beginner's Guide to Logo. Byte.

* ABELSON, H. (1982): *Apple Logo.* McGraw Hill Book Company. New York. USA.

* ABELSON, H.; DISESSA, A. (1976): Student Science Trainer Program in Mathematics Physics and Computer Science, Final Report to the Nacional Science Foundation. *LOGO Memo Nº 29.*

* ABELSON, H.; DISESSA, A. (1986): *Geometría de tortuga. El ordenador como medio de exploración de las Matemáticas.* Anaya (Multimedia). Madrid.

* ABELSON, H.; DISESSA, A; RODOLPH, L. (1974): Velocity Space and the Geometry of Planetary Orbits. *LOGO Memo Nº 15.*

* *ARIAS, J.M. Y BELANGUER, J.E. (1988): Manual de programación en LOGO para la Enseñanza Básica.* Anaya Multimedia. Madrid.

* ARLEGUI DE PABLOS, F.J. (1983): Sentido de la informática en el curriculum escolar. El lenguaje logo y sus aplicaciones educativas. *Bordón (revista de orientación pedagógica): Nº 261 pags. 51-63.*

* *ARLEGUI DE PABLOS, JAVIER. (1988): Logo como lenguaje de representación y simulación en la enseñanza de la dinámica. Actas II Congreso Mundial Vasco. Tomo V / Tecnología y Educación.* Narcea. Madrid.

* CAJARAVILLE PEGITO, J.A. (1989): *Ordenador y educación Matemática. Algunas modalidades de uso.* Síntesis. Madrid.

* LUENGO, R. Y OTROS. (1982-83): *Posibilidades de los pequeños ordenadores en el Ciclo Superior.* I.C.E. de la UNEX. Badajoz.

* LUENGO Y OTROS. (1984): Aprendizaje sintónico de la Geometría a través de ordenador: Las primeras proyecciones de los ejes de referencia. *Actas IV Jornadas de aprendizaje y enseñanza de las Matemáticas.* Tenerife.

* LUENGO GONZÁLEZ, R. Y OTROS. (1985): Un punto de vista para la introducción de la Informática y la Tecnología del ordenador en las escuelas Universitarias de Formación del Profesorado de EGB. *Campo Abierto* Nº 2.

* LUENGO, R. Y OTROS. (1985): *Informática para profesores de EGB.* I.C.E. de la UNEX. Badajoz.

* LUENGO GONZÁLEZ, R. Y OTROS. (1986): El mundo Karel: un posible puente para la introducción al lenguaje Logo. *Campo Abierto* Nº 3, 135-158.

* LUENGO GONZÁLEZ, R. Y OTROS. (1987): Implementación de un micro-mundo Logo tridimensional apartir de una inversión logo en dos dimensiones. *Campo Abierto* Nº4, 13-43.

* LUENGO GONZÁLEZ, R. Y OTROS. (1988): Programación modular en el Micromundo espacio-3D: empleo de algunas utilidades. *Campo Abierto* Nº 5, 111-143.

* LUENGO GONZÁLEZ, R. Y OTROS. (1989): Posibilidades didácticas de Hipertexto: la aplicación Guide. *Campo Abierto* Nº 6, 80-100.

* PAPERT, S. (1984): *Desafío de la Mente: Computadoras y Educación.* Emece Distribuidora. Buenos Aires.

* PAPERT, S. (1985): Different Visions on Logo. *Computers in the Schools,* v2 n 2-3, 3-8.

* REGGINI, H. (1982): *Alas para la mente.* Galápago. Buenos Aires.

* REGGINI, H. (1985): *Ideas y Formas: Explorando el Espacio Tridimensional con Logo.* Galápago. Buenos Aires.

* ROANES MACÍAS, E., ROANES LOZANO, E. (1988): *MACO, Matemáticas con Ordenador.* Síntesis. Madrid.

* RODE, M. Y OTRO (1986): *Metodología y prácticas LOGO.* Ferre Moret, S.A. Barcelona.

* RODRÍGUEZ ROSELLÓ, LUIS. (1985): *De la tortuga a la inteligencia artificial.* Vector. Madrid.

* RODRÍGUEZ ROSELLÓ, LUIS. (1988): Logo y curriculum. *Actas II Congreso Mundial Vasco. Tomo V / Tecnología y Educación.* Narcea. Madrid.