

Anexo.

Inicios con Logo

Las personas sin conocimientos de programación pueden empezar a utilizar LOGO, simplemente dando órdenes a la tortuga. Aprendiendo los gráficos de la tortuga enseñará al usuario algo de geometría (si no lo conoce ya). Y le sorprenderá lo pronto que puede comprender los conceptos de programación una vez que comprende como controlar la tortuga. Veamos unos ejemplos simples:

Dibujar un cuadrado utilizando la tortuga (en este ejemplo los comando están en inglés, pero pueden usarse otros idiomas):

```
FORWARD 100
RIGHT 90
FORWARD 100
RIGHT 90
FORWARD 100
RIGHT 90
FORWARD 100
RIGHT 90
```

Es sencillo, pero hay que teclear mucho, otra forma:

```
REPEAT 4 [FD 100 RT 90]
```

Dibuja el mismo cuadrado. Hemos hecho dos cosas: primero en vez de escribir 4 veces los mismos comandos, le decimos que repita la operación 4 veces, en segundo lugar hemos utilizado abreviaturas de los comandos.

Si en el futuro queremos realizar otro cuadrado podemos instruirle para que cada vez que digamos "SQUARE" dibuje un cuadrado.

```
EDIT "square
<Se abre el editor y tecleamos>
TO SQUARE
REPEAT 4 [FD 100 RT 90]
END
<Salimos del editor salvando>
```

SQUARE

Las palabras TO y END definen un procedimiento (un pequeño programa) para dibujar un cuadrado. Una vez definido el procedimiento, con la orden "SQUARE" se dibuja el cuadrado. Pero sólo dibuja cuadrados de 100x100. Sería mejor poder dibujar cuadrados de cualquier tamaño. Para ello modificamos el procedimiento:

```
EDIT "square
```

```
TO SQUARE :length
REPEAT 4 [FD :length RT 90]
END
```

SQUARE 100
SQUARE 200

Todo lo que hemos hecho ha sido reemplazar 100 por la variable de nombre :length. Ahora al llamar a "SQUARE" debemos decir de que tamaño lo deseamos. En los comandos anteriores hemos pedido a Logo que nos dibuje un cuadrado de 100x100 y otro de 200x200. Notar que ":" delante de la palabra "length" indica a Logo que se trata de una variable.

Sería mejor poder dibujar cualquier tipo de polígono (triángulos, pentágonos, ...). Una opción para hacer esto es crear nuevos procedimientos:

```
TO TRIANGLE :length
REPEAT 3 [FD :length RT 120]
END
```

```
TO PENTAGON :length
REPEAT 5 [FD :length RT 72]
END
```

```
TRIANGLE 100
SQUARE 100
PENTAGON 100
```

Esto requiere teclear bastante y a los programadores no les gusta teclear, porque hay más probabilidades de equivocarse y porque si se necesita cambiar algo, debe cambiarse en muchos sitios. Más pequeño no es siempre mejor pero habitualmente ayuda. Probemos el siguiente procedimiento:

```
TO POLYGON :length :sides
```

```
REPEAT :sides [FD :length RT 360.0/:sides]
END
```

```
POLYGON 100 3
POLYGON 100 4
POLYGON 100 5
```

Los comandos anteriores se comportan como TRIANGLE, SQUARE y PENTAGON. El procedimiento POLYGON dibuja cualquier polígono de lados iguales con una línea de código. Para ello hemos añadido un nuevo parámetro que indica el número de lados. La repetición se realiza este número de veces y el giro de la orden RIGHT se realiza en función de ese número de lados: 360 entre el número de lados. En realidad estamos programando, hemos usado los conceptos básicos de la programación: sentencias, bucles, procedimientos, variables y fórmulas. Sólo queda guardar nuestro trabajo para poderlo utilizar posteriormente. Ahora se encuentra en la memoria de Logo, para pasarlo a disco tecleamos:

```
SAVE "SHAPES.LGO
BYE
```

Para recuperar estas definiciones posteriormente lo recargamos del disco tecleando:
LOAD "SHAPES.LGO