



# Guía rápida de primitivas

## 1. Movimientos de la tortuga

### 1.1. Iniciación

**avanza**, *av número de pasos*

Mueve la tortuga hacia adelante *n* pasos en la dirección que actualmente está mirando.

**retrocede**, *re número de pasos*

Mueve la tortuga hacia atrás *n* pasos en la dirección que actualmente está mirando.

**giraderecha**, *gd ángulo*

Gira hacia la derecha los grados indicados respecto de la dirección que actualmente está mirando.

**giraizquierda**, *gi ángulo*

Gira hacia la izquierda los grados indicados respecto de la dirección que actualmente está mirando.

**centro**

Lleva la tortuga a la posición original, es decir coordenadas [0 0] con rumbo 0.

### 1.2. Avanzando un poco

**borrapantalla**, *bp*

Vacía el área de dibujo, situando a la tortuga en el centro de la pantalla.

**bajalápiz**, *bl*

La tortuga dibujará una línea cuando se mueva.

**subelapiz**, *sl*

La tortuga no dibujará cuando se mueva.

**ocultatortuga**, *ot*

Hace invisible a la tortuga.

**muestratortuga**, *mt*

2

Hace que la tortuga se vea en pantalla.

**poncolorlapiz**, *poncl número, lista o primitiva*

Cambia el color del lápiz

**pon Grosor**, *pongr número*

Define el grosor del trazo del lápiz (en pixels). Por defecto es 1 y la forma es cuadrada.

**rellena**

Rellena con el color activo una zona cerrada.

**rellenazona**

Rellena una zona limitada con el color activo.

**rellenapoligono**

Rellena la forma que se va creando por triangulación.

**goma**, *go*

La tortuga borrará toda traza que encuentre. Para dibujar otra vez, usa **bajalápiz**.

**limpia**

Vacía el área de dibujo, dejando a la tortuga donde estaba tras la ejecución anterior.

## 2. Procedimientos

### 2.1. Escritura

**para**, *palabra variables*

Inicia un procedimiento llamado **palabra** con las variables (si las hay) que se indican.

**fin**

Indica el final de un procedimiento

**#**

Indica que el texto siguiente es un comentario y no debe ser interpretado

### 2.2. Desde la línea de comandos

**edita**, *ed palabra o lista*

Muestra el Editor de Procedimientos con los procedimientos indicados.

**editatodo, editodo**

Muestra el Editor de Procedimientos con todos los procedimientos creados.

**define, def palabra1 lista2 lista3**

Define un nuevo procedimiento llamado **palabra1**, provisto de las variables contenidas en **lista2** y las instrucciones a ejecutar contenidas en **lista3**.

**borra, bo palabra**

Elimina el procedimiento cuyo nombre es **a**.

**borratodo**

Elimina todas las variables y procedimientos actuales. **CUIDADO**: No pide confirmación.

**texto palabra**

Devuelve una lista que contiene toda la información existente del procedimiento **palabra**.

**listaprocs, imts**

Enumera todos los procedimientos actualmente definidos.

**ejecuta lista**

Ejecuta la lista de instrucciones contenida en la lista.

**2.3. Seguimiento y parada****trazado**

Muestra en el Histórico de Comandos los procedimientos que se están ejecutando.

**detienetrazado**

Desactiva el modo **trazado**.

**alto**

Detiene la ejecución de un bucle o de un procedimiento.

**detienetodo**

Interrumpe total y definitivamente todos los procedimientos en ejecución

**devuelve, dev palabra**

Permite salir del procedimiento "llevándose" el resultado guardado en la variable.

**3. Variables****haz palabra b**

Si la variable **palabra** existe, se le asigna el valor **b**. Si no, se crea la variable **palabra** y se le asigna el valor **b**.

**cosa palabra**

Reenvía el valor de **a.cosa** "a" y **:a** son notaciones equivalentes

**objeto palabra**

Reenvía el valor de **palabra.objeto** "a" y **:a** son notaciones equivalentes

**listavars, imvars**

Enumera todas las variables actualmente definidas.

**borravariable, borv palabra**

Elimina la variable indicada.

**local palabra**

Creo una variable llamada **palabra**. Atención: la variable no es inicializada. Para asignarle un valor, hay que usar **haz**.

**hazlocal palabra b**

Creo una nueva variable llamada **palabra** y le asigna el valor **b**, para usar únicamente en el procedimiento activo.

**4. Operaciones****4.1. Operaciones binarias****4.1.1. Con números****suma, + número\_1 número\_2**

Devuelve el resultado de sumar **n\_1** y **n\_2**.

**diferencia, - número\_1 número\_2**

Devuelve el resultado de restar **n\_2** de **n\_1**.

**producto, \* número\_1 número\_2**

Devuelve el resultado de multiplicar **n\_1** por **n\_2**

**division, div / número\_1 número\_2**  
Devuelve el resultado de dividir **n\_1** entre **n\_2**

**cociente entero\_1 entero\_2**  
Devuelve el resultado de la división entera de **n\_1** entre **n\_2**

**resto número\_1 número\_2**  
Devuelve el resto de la división de **n\_1** por **n\_2**

**potencia número\_1 número\_2**  
Devuelve **n\_1** elevado a la potencia **n\_2**

#### 4.1.2. Con listas

**frase, fr a b**  
Devuelve una lista compuesta de **a** y **b**. Si **a** o **b** son una lista, entonces cada uno de los componentes de **a** y **b** se convierten en elementos de la lista creada. (los corchetes son suprimidos).

**lista a b**  
Devuelve una lista compuesta de **a** y **b**. Si **a** o **b** son una lista, se convierte/nm en una sublista de la lista creada.

## 4.2. Operaciones unitarias

### 4.2.1. Cálculo elemental

**cambiasigno, cs número**  
Devuelve el opuesto del número.

**absoluto, abs número**  
Devuelve el valor absoluto (distinto de cero) del número.

**redondea número**  
Devuelve el entero más próximo al número **n**

**trunca número**  
Devuelve el entero inmediatamente anterior al número **n**

**raizcuadrada, rc número**  
Devuelve la raíz cuadrada del número.

**aleatorio**  
Devuelve un número al azar comprendido entre 0 y 1.

**azar número**  
Devuelve un número al azar mayor o igual que 0 y menor que **n**.

### 4.2.2. Cálculo superior

**pi**  
Devuelve el número  $\pi$  (3.141592653589793)

**exp número**  
Devuelve  $e$  ( $e = 2,71828183\dots$ ) elevado a **n**

**log10, log, logaritmo número**  
Devuelve el logaritmo decimal de **n**.

**logneperiano, ln número**  
Devuelve el logaritmo neperiano de **n**.

**coseno, cos número**  
Devuelve el coseno del número **n**.

**seno, sen número**  
Devuelve el seno del número **n**.

**tangente, tg, tan número**  
Devuelve la tangente del número **n**.

**arcoseno, arccos, acosos, arccos número**  
Devuelve el ángulo, en grados, cuyo coseno vale **n**.

**arcoseno, arcsen, asen número**  
Devuelve el ángulo, en grados, cuyo seno vale **n**.

**arcotangente, arctg, atan número**  
Devuelve el ángulo, en grados, cuya tangente vale **n**.

**decimales, digitos**  
Devuelve el número de dígitos (precisión) con que está trabajando XLogo.

**pondecimales, pondigitos número**

Cambia la precisión con la que trabajará xLOGO. Por defecto, 16 dígitos.

## 5. Coordenadas

### 5.1. Cuadrícula y ejes

**cuadrícula número\_1 número\_2**  
Dibuja una cuadrícula en el **Área de dibujo** de dimensiones **n\_1** x **n\_2**

**borracuadrícula, detienecuadrícula**  
Quita la cuadrícula del **Área de dibujo**

**poncolorcuadrícula, pcc número, primitiva o lista [r v a]**  
Cambia el color de la cuadrícula

**colorcuadrícula**  
Devuelve el color actual de la cuadrícula.

**ejes número**  
Dibuja los ejes cartesianos (X e Y) de escala (separación entre marcas) **n**, con las etiquetas correspondientes.

**ejej número**  
Dibuja el eje de abscisas (eje X) de escala (separación entre marcas) **n**, con las etiquetas correspondientes.

**ejej número**  
Dibuja el eje de ordenadas (eje Y) de escala (separación entre marcas) **n**, con las etiquetas correspondientes.

**borraejjes, detieneejjes**  
Quita los ejes del **Área de dibujo**

**poncolorejes, pcc número, primitiva o lista [r v a]**  
Cambia el color de los ejes cartesianos

**colorejes**  
Devuelve el color actual de los ejes cartesianos

### 5.2. Coordenadas

**posicion, pos**  
Devuelve las coordenadas [X Y] en el plano correspondientes a la posición actual de la tortuga.

**ponposicion, ponpos [x y/]**  
Mueve la tortuga a las coordenadas especificadas por los números en la lista (**x** especifica el eje **x**, **y** el eje **y**)

**ponx número**  
Mueve la tortuga hasta el punto de abscisa **n** horizontalmente.

**pony número**  
Mueve la tortuga hasta el punto de ordenada **n** verticalmente.

**ponxy x y**  
Idéntico a **ponpos [x y]**

**coordenadax, coordx no**  
Devuelve el valor de la abscisa (coordenada **x**) de la posición que ocupada la tortuga. Equivalente a **primero posicion**.

**coordenaday, coordy**  
Devuelve el valor de la ordenada (coordenada **y**) de la posición que ocupada la tortuga. Equivalente a **ultimo posicion** (en el plano).

**punto [x y/]**  
El punto definido por las coordenadas de la lista se resaltarán con el color del lápiz.

### 5.3. Rumbo

**ponrumbo, ponr número**  
Orienta la tortuga en la dirección especificada. **0** corresponde a mirar hacia arriba verticalmente.

**rumbo**  
Devuelve el rumbo o el ángulo de la tortuga.

**centro**  
Lleva la tortuga a la posición original, es decir coordenadas [0 0] con rumbo **0**.

**hacia** *lista*

La lista debe contener dos números que representen coordenadas. Devuelve el rumbo que la tortuga deberá seguir hacia el punto definido por las coordenadas.

**distancia** *lista*

La lista debe contener dos números que representen coordenadas. Devuelve el número de pasos desde la actual posición y el punto definido por las coordenadas.

## 6. Condicionales y operaciones lógicas

### 6.1. Condicional

**si** *expr\_lógica lista\_1 lista\_2*

Ejecuta las órdenes guardadas en las listas *l\_1* y *l\_2* según la expresión lógica sea **cierto** o **falso**

**sisino** *expr\_lógica variable\_1 variable\_2*

Ejecuta las órdenes guardadas en las variables según la expresión lógica sea **cierto** o **falso**

### 6.2. Operaciones lógicas

**y**, **&** *booleano\_1 booleano\_2*

Devuelve cierto si los dos booleanos son ciertos, si no, devuelve **falso**

**no** *Booleano*

Devuelve la negación del booleano. Si es **cierto**, devuelve **falso** y viceversa.

**o**, **|** *booleano\_1 booleano\_2*

Devuelve cierto si *b\_1* o *b\_2* son ciertos, si no, devuelve **falso**

**cierto**

Devuelve "cierto"

**falso**

Devuelve "falso"

### 6.3. Booleanos

**mayor?**, **>** *número\_1 número\_2*

Devuelve cierto si *n\_1* es mayor que *n\_2*, falso si no.

**mayoroinigual?**, **>=** *número\_1 número\_2*

Devuelve cierto si *n\_1* es mayor o igual que *n\_2*, falso si no.

**menor?**, **<** *número\_1 número\_2*

Devuelve cierto si *n\_1* es menor que *n\_2*, falso si no.

**menoroinigual?**, **<=** *número\_1 número\_2*

Devuelve cierto si *n\_1* es menor o igual que *n\_2*, falso si no.

**entero?** *número*

Devuelve cierto si *n* es un número entero, falso si no.

**lista?** *palabra*

Devuelve cierto si *palabra* es una lista, falso si no.

**numero?** *palabra*

Devuelve cierto si *palabra* es un número, falso si no.

**palabra?** *palabra*

Devuelve cierto si *palabra* es una palabra, falso si no.

**vacio?** *lista o palabra*

*lista* Devuelve cierto si es una lista o una palabra vacía, falso si no.

**iguales?** *a b*

Devuelve cierto si *a* y *b* son iguales, falso si no.

**antes?**, **anterior?** *palabra1 palabra2*

Devuelve cierto si *palabra1* está antes que *palabra2* siguiendo el orden alfabético, falso si no.

**miembro?** *a b*

Si *b* es una lista, determina si *a* es un elemento de *b*. Si *b* es una palabra, determina si *a* es un caracter de *b*.

**cuadrícula?**

Devuelve cierto si la cuadrícula está activa, **falso** si no.

**ejex?**

Devuelve cierto si está activo el eje de abscisas (eje X), **falso** si no.

**ejey?**

Devuelve cierto si está activo el eje de ordenadas (eje Y), **falso** si no.

**bajalapiz?, bl?**

Devuelve cierto si el lápiz está abajo, **falso** si no.

**visible?**

Devuelve cierto si la tortuga está visible, **falso** si no.

**primitiva?, prim? palabra**

Devuelve cierto si la palabra es una primitiva de xLOGO, **falso** si no.

**procedimiento?, proc? palabra**

Devuelve cierto si la palabra es un procedimiento definido por el usuario, **falso** si no.

**variable?, var? palabra**

Devuelve cierto si la palabra es una variable ya definida, **falso** si no.

## 7. Listas y palabras

### 7.1. Manejo de listas

**primero, pr palabra o lista**

Si el argumento es una lista, devuelve el primer elemento de la lista. Si es una palabra, devuelve la primera letra de la palabra.

**ultimo palabra o lista**

Si el argumento es una lista, devuelve el elemento de la lista. Si es una palabra, devuelve la última letra de la palabra.

**elemento número, lista o palabra**

Si el segundo argumento es una palabra, devuelve la letra que ocupa la posición **n** (1 señala la primera letra). Si es una lista, devuelve el elemento número **n** de la lista.

**elige palabra o lista**

Si el argumento es una palabra, devuelve una de las letras que la forman al azar. Si es una lista, devuelve uno de sus elementos al azar.

**cuenta palabra o lista**

Si el argumento es una palabra, devuelve el número de letras de la misma. Si es una lista, devuelve el número de elementos que contiene.

**miembro a b**

Investiga a en b

### 7.2. Modificando listas

**menosprimero, mp palabra o lista**

Si el argumento es una lista, devuelve toda la lista menos el primer elemento. Si es una palabra, devuelve la palabra sin la primera letra.

**menosultimo, mu palabra o lista**

Si el argumento es una lista, devuelve toda la lista menos el último elemento. Si es una palabra, devuelve la palabra sin la última letra.

**quita a lista**

Elimina el elemento **a** de la lista, si aparece dentro.

**ponprimero, pp a lista**

Inserta **a** en la primera posición de la lista.

**ponultimo, pu a lista**

Inserta **a** en la última posición de la lista.

**agrega LI número L2**

Dada la primera lista (o palabra) **L1**, inserta en la posición del número **n** la palabra o lista **L2**.

**reemplaza lista\_1 número L2**

Dada la lista **lista\_1**, reemplaza el elemento **n** por la palabra o lista **lista\_2**.

**invierte lista**

Invierte el orden de los elementos de la lista.

### 7.3. Combinando listas

**frase, fr a b**

Devuelve una lista compuesta de **a** y **b**. Si **a** o **b** son una lista, entonces cada uno de los componentes de **a** y **b** se convierten en elementos de la lista creada. (los corchetes son suprimidos).

**lista a b**

Devuelve una lista compuesta de **a** y **b**. Si **a** o **b** son una lista, se convierte/**nn** en una sublista de la lista creada.

**palabra palabra\_1 palabra\_2**

Concatena las dos palabras.

### 7.4. Listas de propiedades

**propriedad, noprop nombre\_lista clave palabra**

Añade la propiedad **palabra** a la clave de lista indicada.

**leopropiedad, leeprop nombre\_lista clave**

Devuelve el valor asociado a la **clave** de la lista.

**borrapropiedad, boprop nombre\_lista palabra**

Borra el elemento **palabra** de la lista indicada.

**listapropiedad, listasprop nombre**

Muestra todos los elementos de la lista especificada.

**listaspropiedades, listasprop**

Enumera en una lista todas las listas de propiedades definidas.

## 8. Bucles

**repite número lista**

Ejecuta **n** veces las órdenes contenidas en la lista.

**repitepara lista\_1 lista\_2**

Ejecuta las órdenes contenidas en la lista **lista\_2** según las condiciones descritas en **lista\_1**.

**paracada lista o palabra lista**

Ejecuta las órdenes contenidas en la segunda lista tantas veces como elementos o caracteres contiene la primera lista o palabra.

**repitehasta lista\_1 lista\_2**

Ejecuta las órdenes contenidas en la lista **lista\_1** hasta que las condiciones descritas en **lista\_2** sean ciertas.

**mientras lista\_1 lista\_2**

La primera lista contiene un contador y sus valores inicial, final y el paso. La segunda es la serie de órdenes que deben ejecutarse el número de veces que determine la lista 1.

**repite mientras lista\_1 lista\_2**

Ejecuta las órdenes contenidas en la lista **lista\_1** mientras las condiciones descritas en **lista\_2** sean ciertas.

**repite siempre lista**

Ejecuta las órdenes contenidas en la lista hasta que se haga *click* en el botón *alto* o alguna orden de la lista se lo indique.

**alto**

Detiene la ejecución de un bucle o de un procedimiento.

**detienetodo**

Interrumpe total y definitivamente todos los procedimientos en ejecución

**cuentalrepite, contador**

En un bucle *repite*, determina el número de la iteración en curso.

## 9. Comunicación con el usuario

**escribe, es número, palabra o lista**

Muestra en el **Histórico de Comandos** el argumento indicado.

**tipa número, palabra o lista**

Idéntico a *escribe*, pero el cursor queda en la línea donde se mostró el contenido del argumento.

**rotula palabra o lista**

Dibuja la palabra o lista especificada, en la posición actual, y en la dirección que está mirando.

**largoetiqueta lista**

Devuelve, en píxels, la longitud en pantalla de la lista.

**mensaje, msj lista**

Muestra una caja de diálogo con el mensaje que está en la lista. El programa se detiene hasta que el usuario hace un *click* en el botón “Aceptar”.

## 9.1. Propiedades del Histórico de Comandos

**borratexto, bt**

Borra el Área de comandos, y el área del Histórico de comandos.

**ponfuentetexto, pft número**

Define el tamaño de la tipografía del área del Histórico de comandos. Sólo disponible para ser usada por la primitiva **escribe**.

**poncolortexto, pctexto número, primitiva o lista**

Define el color de la tipografía del área del Histórico de comandos. Sólo disponible para ser usada por la primitiva **escribe**.

**ponnombrefuentetexto, pnft número**

Selecciona la tipografía número **n** para escribir en el área del Histórico de comandos con la primitiva **escribe**. Puedes encontrar la relación entre fuente y número en el menú **Herramientas** → **Preferencias** → **Fuente**.

**ponestilo, pest lista o palabra**

Define los efectos de fuente para el Histórico de comandos, que pueden ser:

**ninguno**

Elimina todos los efectos de fuente en la primitiva **ponestilo**

**negrita**

**subíndice**

**subrayado**

**superíndice**

**tachado**

**cursiva**

**fuentetexto, ftexto**

Devuelve el tamaño de la tipografía usada por la primitiva **escribe**.

**colortexto**

Devuelve el color de la tipografía usada por la primitiva **escribe** en el área del Histórico de comandos.

**nombrefuentetexto, nft**

Devuelve una lista con dos elementos. El primero es un número correspondiente a la fuente utilizada para escribir en el área del Histórico de comandos con la primitiva **escribe**. El segundo elemento es una lista que contiene el nombre de la fuente.

**estilo, est**

Devuelve una lista que contiene todos los efectos de fuente utilizados por las primitivas **escribe** y **tipaea**.

## 9.2. Escritura en pantalla

**ponnombrefuente, pnf número**

Selecciona la tipografía número **n** para escribir en el Área de Dibujo con la primitiva **rotula**. Puedes encontrar la relación entre fuente y número en el menú **Herramientas** → **Preferencias** → **Fuente**.

**nombrefuente, nf**

Devuelve una lista con dos elementos. El primero es un número correspondiente a la fuente utilizada para escribir en el Área de Dibujo con la primitiva **rotula**. El segundo elemento es una lista que contiene el nombre de la fuente.

**ponfuente, pf número**

Cuando se escribe en pantalla con la primitiva **rotula**, es posible modificar el tamaño de la tipografía con **ponfuente**. Por defecto, el tamaño es 12.

**fuentes**

Devuelve el tamaño de la tipografía cuando se escribe en pantalla con la primitiva **rotula**.



`ponjustificadofuente lista`

Indica cómo alinea el texto alrededor de la tortuga. La lista debe contener dos números: el alineamiento horizontal y el vertical.

`justificadofuente`

Devuelve la alineación del texto en pantalla.

### 9.3. Interacción con el teclado

`tecla?`

Devuelve `cierto` o `falso` según se haya pulsado o no alguna tecla desde el inicio del programa.

`leecar, leetecla`

Espera a que el usuario pulse una tecla y devuelve la tecla pulsada.

`leeteclado, leelista lista palabra`

Muestra una ventana titulada `lista` y recoge la respuesta del usuario en la variable `palabra`.

### 9.4. Interacción con el ratón

`leeraton`

Espera a que el usuario haga “algo” con el ratón, y devuelve la acción realizada.

`posraton`

Devuelve las coordenadas X e Y correspondientes a la posición actual del ratón.

`raton?`

Devuelve `cierto` o `falso` según se mueva o no el ratón.

### 9.5. Componentes gráficos

`botonigu nombre palabra`

Crear un botón (componente gráfico) llamado `nombre` con la leyenda `palabra`.

`menuigu nombre lista`

Crear un menú (componente gráfico) llamado `nombre` que abre un *desplegable* con las opciones incluidas en la lista.

`posicionigu nombre lista`

Determina la posición en pantalla del componente gráfico `nombre`, asignándole las coordenadas de la `lista`.

`eliminaigu nombre`

Elimina el componente gráfico `nombre`

`accionigu nombre lista`

Asocia las acciones contenidas en `lista` al componente gráfico `nombre`

`dibujaiigu nombre`

Dibuja en la pantalla el componente gráfico indicado

## 10. Técnicas avanzadas de dibujo

### 10.1. Opciones para la *tortuga*

`muestratortuga, mt`

Hace que la *tortuga* se vea en pantalla.

`ocultatortuga, ot`

Hace invisible a la *tortuga*.

`pongrosor número`

Define el grosor del trazo del lápiz (en pixels). Por defecto es 1 y la forma es cuadrada.

`ponformalapiz, pf1 0 ó 1`

Fija la forma del lápiz: `pf1 0`: cuadrada; `pf1 1`: ovalada.

`grosorlapiz, gl`

Devuelve el grosor del lápiz.

`formalapiz, fl`

Devuelve la forma del lápiz.

### 10.2. Control del color

`poncolorlapiz, poncl número, lista o primitiva`

Cambia el color del lápiz

**poncolorpapel**, **poncp número**, **primitiva o lista [r v a]**  
 Cambia el color del papel (fondo)

**inviertelapiz**, **ila**

Pone la tortuga en "modo inverso", y lápiz abajo.

**ponlapiz**, **pla**

Pone la tortuga en el modo normal de dibujo y lápiz abajo.

**colorlapiz**, **cl**

Devuelve el color actual del lápiz.

**colorpapel**

Devuelve el color actual del "papel" (fondo, área de dibujo).

**encuentracolor**, **ec lista**

Devuelve el color del punto definido por las coordenadas de **lista**.

**goma**, **go**

La tortuga borrará toda traza que encuentre. Para dibujar otra vez, usa **bajalapiz**.

### 10.3. Colores

**negro**

Color número 0, codificación [RGB] = [0 0 0].

**rojo**

Color número 1, codificación [RGB] = [255 0 0].

**verde**

Color número 2, codificación [RGB] = [0 255 0].

**amarillo**

Color número 3, codificación [RGB] = [255 255 0].

**azul**

Color número 4, codificación [RGB] = [0 0 255].

**magenta**

Color número 5, codificación [RGB] = [255 0 255].

**cyan**

Color número 6, codificación [RGB] = [0 255 255].

**bianco**

Color número 7, codificación [RGB] = [255 255 255].

**gris**

Color número 8, codificación [RGB] = [128 128 128].

**grisclaro**

Color número 9, codificación [RGB] = [192 192 192].

**rojoscuro**

Color número 10, codificación [RGB] = [128 0 0].

**verdeoscuro**

Color número 11, codificación [RGB] = [0 128 0].

**azuloscuro**

Color número 12, codificación [RGB] = [0 0 128].

**naranja**

Color número 13, codificación [RGB] = [255 200 0].

**rosa**

Color número 14, codificación [RGB] = [255 175 175].

**violeta**

Color número 15, codificación [RGB] = [128 0 255].

**marron**

Color número 16, codificación [RGB] = [153 102 0].

## 10.4. El Área de Dibujo

### 10.4.1. Control del dibujo

**lupa**, **zoom**

Devuelve el valor del **zoom**, es decir, es el factor de escala del Área de dibujo respecto a la imagen original.

**ponlupa**, **ponzoom número**

Acerca o aleja el Área de dibujo. En concreto, el valor de **n** es el factor de escala respec-

to a la imagen original: ( $n > 1$ ) acerca el Área de dibujo; ( $0 < n < 1$ ) aleja el Área de dibujo.

**poncalidaddibujo**, *pcd 0, 1 ó 2*

Fija la calidad del dibujo: *pcd 0*: normal; *pcd 1*: alta; *pcd 2*: baja;

**calidaddibujo**, *cdib*

Devuelve la calidad del dibujo.

**modojaula**

La tortuga queda confinada al área de dibujo. Si intenta salir, aparecerá un mensaje de error avisando cuántos pasos faltan para el punto de salida.

**modoventana**

La tortuga puede salir del área de dibujo (pero no dibujará nada).

**modovuelta**

Si la tortuga sale del área de dibujo, vuelve a aparecer en el lado opuesto

#### 10.4.2. Dimensiones

**tamaño pantalla**, *tpant*

Devuelve una lista que contiene el tamaño de la pantalla

**tamaño ventana**, *tv*, **esquinasventana**

Devuelve una lista con cuatro elementos, las coordenadas de la esquina superior izquierda y de la esquina inferior derecha.

**separacion**

Devuelve el valor de la proporción de pantalla ocupada por el **Área de Dibujo** y el **Histórico de Comandos**.

**ponseparacion**, *ponsep número (comprendido entre 0 y 1)*

Determina la proporción de pantalla ocupada por el **Área de Dibujo** y el **Histórico de Comandos**. Si *n* vale 1, el **Área de Dibujo** ocupará toda la pantalla. Si *n* vale 0, será el **Histórico** quien la ocupe.

**pontamaño pantalla**, *ptp lista*

Fija el tamaño de la pantalla.

## 11. Multitortuga y animación

### 11.1. Modo multitortuga

**ponforma**, *pforma número*

Puedes elegir tu tortuga preferida en la segunda etiqueta del menú **Herramientas** → **Preferencias**, pero también es posible con **ponforma**. El número *n* puede ir de 0 a 6. (0 es la forma triangular del LOGO tradicional).

**forma**

Devuelve un número que representa la forma actual de la tortuga.

**pontortuga**, *ptortuga número*

La tortuga número *n* es ahora la tortuga activa. Por defecto, cuando XLOGO comienza, está activa la tortuga número 0.

**tortuga**

Da el número de la tortuga activa.

**tortugas**

Da una lista que contiene todos los números de tortuga actualmente en pantalla.

**eliminar tortuga número**

Elimina la tortuga número *n*

**ponmaximastortugas**, *pmt número*

Fija el máximo número de tortugas

**maximastortugas**, *maxt*

Devuelve el máximo número de tortugas

### 11.2. Animación

**animacion**

Se accede al modo de animación.

**detieneanimacion**

Detiene el modo animación, retornando al modo *normal*.

**refresca**, **refrescar**

En modo de animación, ejecuta las órdenes y actualiza la imagen

## 12. Manejo de archivos

### 12.1. Navegación por el sistema de archivos

**catalogo, cat**

Lista el contenido del directorio actual. (Equivalente al comando `ls` de Linux, `dir` de DOS)

**pondirectorio, pondir *lista***

Especifica el directorio actual. La ruta debe ser absoluta. El directorio debe especificarse dentro de una lista, y la ruta no debe contener espacios.

**cambiadirectorio, cd *palabra o lista***

Cambia el directorio de trabajo desde el directorio actual (ruta relativa). Puede utilizarse `..` para referirse a la ruta del directorio superior.

**directorio, dir**

Da el directorio actual. Por defecto, es `/home/tu_nombre` en Linux, `C:\WINDOWS` en Windows.

### 12.2. Carga y guardado de procedimientos

**carga *palabra***

Abre y lee el archivo indicado por *palabra*.

**guarda *palabra lista***

Guarda en el archivo *palabra* los procedimientos especificados en *lista*, en el directorio actual.

**guardatodo *palabra***

Guarda en el archivo *palabra* todos los procedimientos definidos, en el directorio actual.

### 12.3. Modificando archivos de texto

**abreflujo *número nombre\_fichero***

Para poder leer o escribir en un fichero, es necesario crear un flujo hacia él. El argumento `nombre_fichero` debe ser su nombre, que se refiere al directorio de trabajo. El argumento `n` es el número que identifica a ese flujo.

**cierraflujo *número***

Cierra el flujo `n`.

**listaflujos *lista***

Carga una lista con los flujos abiertos indicando su identificador.

**leeflujo *número***

Abre el flujo cuyo identificador es `n`, y lee una línea del fichero

**leecarflujo *número***

Abre el flujo cuyo identificador es `n`, después lee un carácter del fichero. Esta primitiva devuelve el número correspondiente al carácter unicode

**agregalineaflujo *número lista***

Escribe la línea de texto indicada en *lista* al final del fichero indicado por el flujo `n`.

**escribelineaflujo *número lista***

Escribe la línea de texto indicada en *lista* al principio del fichero indicado por el flujo `n`.

**finflujo? *número***

Devuelve cierto si se ha llegado al final del fichero, y falso en caso contrario.

### 12.4. Ejecutando programas externos

**comandoexterno**

lista de listas Ejecuta un comando externo (del Sistema operativo correspondiente), conteniendo sub-listas con el comando deseado y las opciones de dicho comando.

## 13. Geometría de la tortuga en 3-D

**perspectiva**

Inicia el modo de dibujo en 3 Dimensiones

**modojaula, modoventana, modovuelta**

Vuelven al modo de dibujo en 2 Dimensiones

### 13.1. Movimiento en 3D

**avanza, av *número de pasos***

Mueve la tortuga hacia adelante `n` pasos en la dirección que actualmente está mirando.

**retrocede, re número de pasos**

Mueve la tortuga hacia atrás  $n$  pasos en la dirección que actualmente está mirando.

**giraderecha, gd ángulo**

Gira hacia la derecha los grados indicados respecto de la dirección que actualmente está mirando.

**giraizquierda, gi ángulo**

Gira hacia la izquierda los grados indicados respecto de la dirección que actualmente está mirando.

**subenariz, sn, cabeceaarriba ángulo**

Sube el "morro" de la tortuga los grados indicados

**bajanariz, bn, cabeceabajó ángulo**

Baja el "morro" de la tortuga los grados indicados

**balanceaderecha, bd ángulo**

Cambia el ángulo de balanceo de la tortuga. Sube el "ala" izquierda y baja la derecha  $n$  grados.

**balanceaizquierda, bi ángulo**

Sube el "ala" derecha y baja la izquierda  $n$  grados.

**balanceo**

Devuelve el ángulo de balanceo (inclinación de las alas).

**cabeceo**

Devuelve el ángulo de cabeceo (inclinación del "morro").

**rumbo**

Devuelve el rumbo (giro horizontal) de la tortuga.

**ponbalanceo ángulo**

Pone el balanceo con el ángulo especificado. 0 corresponde a "timón" hacia arriba.

**poncabeceo ángulo**

Orienta el "morro" de la tortuga con el ángulo especificado. 0 corresponde a "morro" horizontal.

**ponrumbo, ponr número**

Orienta la tortuga en la dirección especificada. 0 corresponde a mirar hacia "nosotros" ho-

rizontalmente.

**ponorientacion [balanceo cabeceo rumbo]**

Orienta la tortuga según las direcciones especificadas.

**orientacion**

Devuelve una lista con los tres valores: [balanceo cabeceo rumbo] de orientación de la tortuga.

**posicion, pos**

Devuelve las coordenadas [X Y Z] correspondientes a la posición actual de la tortuga.

**ponposicion, ponpos [x y z]**

Mueve la tortuga a las coordenadas especificadas por los números en la lista ( $x$  especifica el eje  $x$ ,  $y$  el eje  $y$  y  $z$  el eje  $z$ ).

**ponxyz x y z: coordenadas x, y, z**

Idéntico a **ponpos** [ $x$  y  $z$ ]

**ponx número**

Mueve la tortuga hasta el punto de abscisa  $n$ , en el plano horizontal.

**pony número**

Mueve la tortuga hasta el punto de ordenada  $n$ , en el plano horizontal.

**ponz número**

Mueve la tortuga hasta el punto de altura  $n$ , en el espacio.

**coordenadax, coordx**

Devuelve el valor de la abscisa (coordenada  $x$ ) de la posición que ocupada la tortuga. Equivalente a **primero posicion**.

**coordenaday, coordy**

Devuelve el valor de la ordenada (coordenada  $y$ ) de la posición que ocupada la tortuga.

**coordenadz, coordz**

Devuelve el valor de la altura (coordenada  $z$ ) de la posición que ocupada la tortuga. Equivalente a **ultimo posicion**.

Otras primitivas disponibles en 3D son:

`subelapiz`, `bajalapiz`, `inviertelapiz`, `ponlapiz`, ...

Sin argumentos

`circulo`, `arco`, `centro`, `rotula`, `largoetiqueta`, `poncolorlapiz`, `pongrosorlapiz`, ...

Con los mismos argumentos que en el plano.

`hacia`, `distancia`, `punto`

Con tres argumentos (como `posicion`)

### 13.2. El visor 3D

`empiezalinea`, `definelinea`

Inicia el modo de definición de líneas.

`empiezapoligono`, `definepoligono`

Inicia el modo de definición de polígonos.

`empiezapunto`, `definepunto`

Inicia el modo de definición de puntos.

`empiezatexto`, `definetexto`

Inicia el modo de definición de texto.

En las primitivas anteriores, las órdenes intermedias se guardan para crear el conjunto definido hasta la aparición de las siguientes:

`finlinea`

Termina el modo de definición de líneas.

`finpoligono`

Termina el modo de definición de polígonos.

`finpunto`

Termina el modo de definición de puntos.

`fintexto`

Termina el modo de definición de texto.

`vista3D`, `vistapoligono`

Inicia el visor 3D, todos los objetos guardados se dibujan en una nueva ventana.

## 14. Tocar música

`poninstrumento`, `pinstr número`

Queda seleccionado el instrumento número `n`. Puedes ver la lista de instrumentos disponibles en el menú **Herramientas** → **Preferencias** → **Sonido**.

`instrumento`, `instr`

Da el número que corresponde al instrumento actualmente seleccionado.

`secuencia`, `sec lista`

Carga en memoria la secuencia incluida en la lista. Siguiendo a esta tabla, se indica cómo escribir una secuencia de notas musicales.

`ponindicesecuencia`, `pinindicesecuencia`, `pinindicesecuencia número`

Pone el puntero en la posición `a` de la secuencia corriente.

`indicesecuencia`, `indsec`

Da la posición del puntero en la secuencia corriente.

`borrasecuencia`, `bos`

Elimina de memoria la secuencia corriente.

`tocamusica`

Toca la secuencia de notas en memoria.

`escuchamp3 palabra`

Inicia la reproducción del archivo `mp3` indicado.

`detienemp3`

Detiene la reproducción del archivo `mp3` que está sonando.

## 15. Gestión de tiempos

`espera número`

Hace una pausa en el programa, la tortuga espera (`n/60`) segundos.

**cronometro, cromo número**

Inicia un conteo descendiente de **n** segundos. Para saber que la cuenta ha finalizado, disponemos de la primitiva **fincronometro?**

**fincronometro?, fincromo?**

Devuelve "cierto" si no hay ningún conteo activo. Devuelve "falso" si el conteo no ha terminado.

**fecha**

Devuelve una lista compuesta de 3 números enteros que representan la fecha del sistema. El primero indica el día, el segundo el mes y el último el año. [día mes año]

**hora**

Devuelve una lista compuesta de 3 números enteros que representan la hora del sistema. El primero representa las horas, el segundo los minutos y el último los segundos. [horas minutos segundos]

**tiempo**

Devuelve el tiempo, en segundos, transcurrido desde el inicio de XLOGO.

## 16. La red y robótica

**chattcp palabra lista**

Permite *chatear* con el PC **palabra**, enviando el mensaje contenido en **lista**

**ejecutatcp palabra lista**

Ejecuta las instrucciones contenida en la lista en el PC indicado por **palabra**.

**enviatcp palabra lista**

Envía las instrucciones contenida en la lista al PC indicado por **palabra**.

**escuchatcp**

Pone al PC "en espera" de órdenes desde otro ordenador.

## 17. Otras primitivas

**adios**

Termina la ejecución de XLOGO

**arco radio ángulo\_1 ángulo\_2**

Dibuja un arco de circunferencia de radio **r** alrededor de la tortuga, comprendido entre los ángulos **a\_1** y **a\_2**, midiendo desde el rumbo de la tortuga.

**caracter, car número**

Devuelve el carácter unicode que corresponde al número **n**

**cargaimagen, ci palabra**

Carga el archivo de imagen indicado por la palabra.

**circulo radio**

Dibuja una circunferencia de radio **r** alrededor de la tortuga

**inicializa, reponetodo**

Reinicia la interfaz XLOGO a los valores por defecto: Color del lápiz: negro; color del papel: blanco; modo animación: falso; Tamaño de fuente para las zonas de texto y de dibujo: 12 puntos; Forma del lápiz: cuadrado; calidad del dibujo: normal; Número máximo de tortugas: 16

**unicode palabra**

Devuelve el número unicode que corresponde al carácter.

**\\**

Permite obtener el carácter *barra invertida* (\) en la línea de comandos.

**\n**

Permite generar saltos de línea al escribir en el Histórico de comandos

**\\_**

Permite generar espacios en blanco al escribir en el Histórico de comandos



Disfruta con XLOGO

