

GEOMETRÍA DE LA TORTUGA **CON SCRATCH 2.0**

Eugenio Roanes Lozano¹ y Eugenio Roanes Macías²

Unidad Docente de Álgebra³
Facultad de Educación
Universidad Complutense de Madrid

26 Diciembre 2015
(versión 7)

¹ eroanes@mat.ucm.es

² roanes@mat.ucm.es

³ www.ucm.es/info/secdealg/

TRABAJAR ONLINE




Si se tiene conexión a Internet no es preciso instalar **Scratch 2.0** en el ordenador, pudiéndose trabajar online. Para ello basta hacer click en:



que se puede encontrar en: <https://scratch.mit.edu/>

TRABAJAR OFFLINE

Se puede también instalar **Scratch 2.0** en el ordenador. Para ello hay que instalar primero AdobeAIR. Ambos se pueden descargar de <https://scratch.mit.edu/scratch2download/> , donde se ofrecen:

Adobe AIR	Editor Offline de Scratch	Materiales de apoyo
		
If you don't already have it, download and install the latest Adobe AIR	Next download and install the Scratch 2.0 Offline Editor	Need some help getting started? Here are some helpful resources.
Mac OS X - Download ⬇ Mac OS 10.5 & Older - Download ⬇ Windows - Download ⬇ Linux - Download ⬇	Mac OS X - Download ⬇ Mac OS 10.5 & Older - Download ⬇ Windows - Download ⬇ Linux - Download ⬇	Starter Projects - Download ⬇ Getting Started Guide - Download ⬇ Scratch Cards - Download ⬇

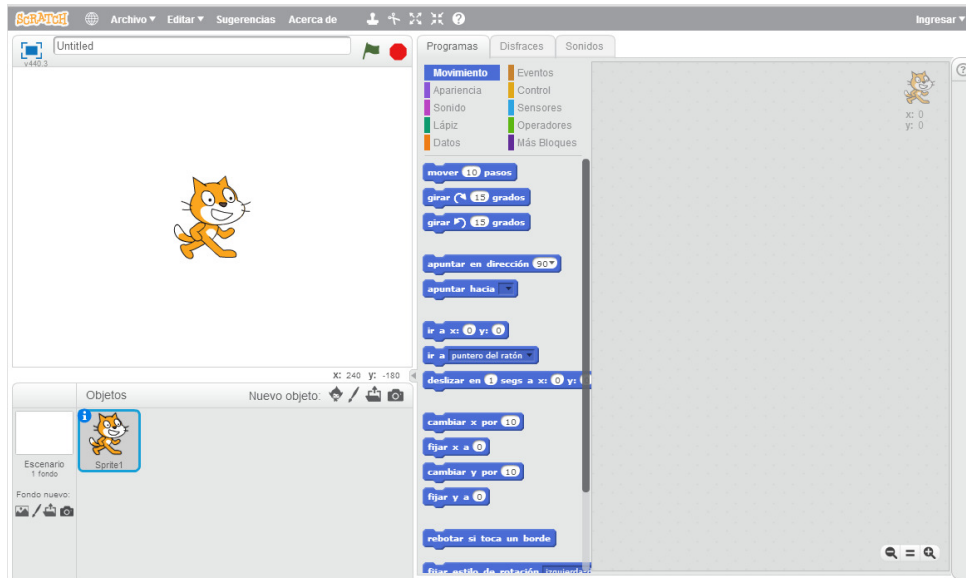
Una vez instalados, tendremos en el escritorio el icono:



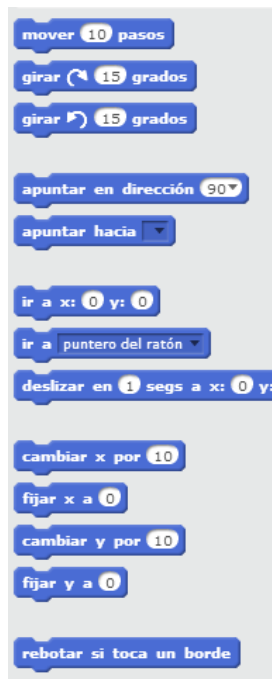
con el que se puede arrancar **Scratch 2.0** del modo usual.

PRIMERAS ÓRDENES

Al arrancar aparecerá algo como:



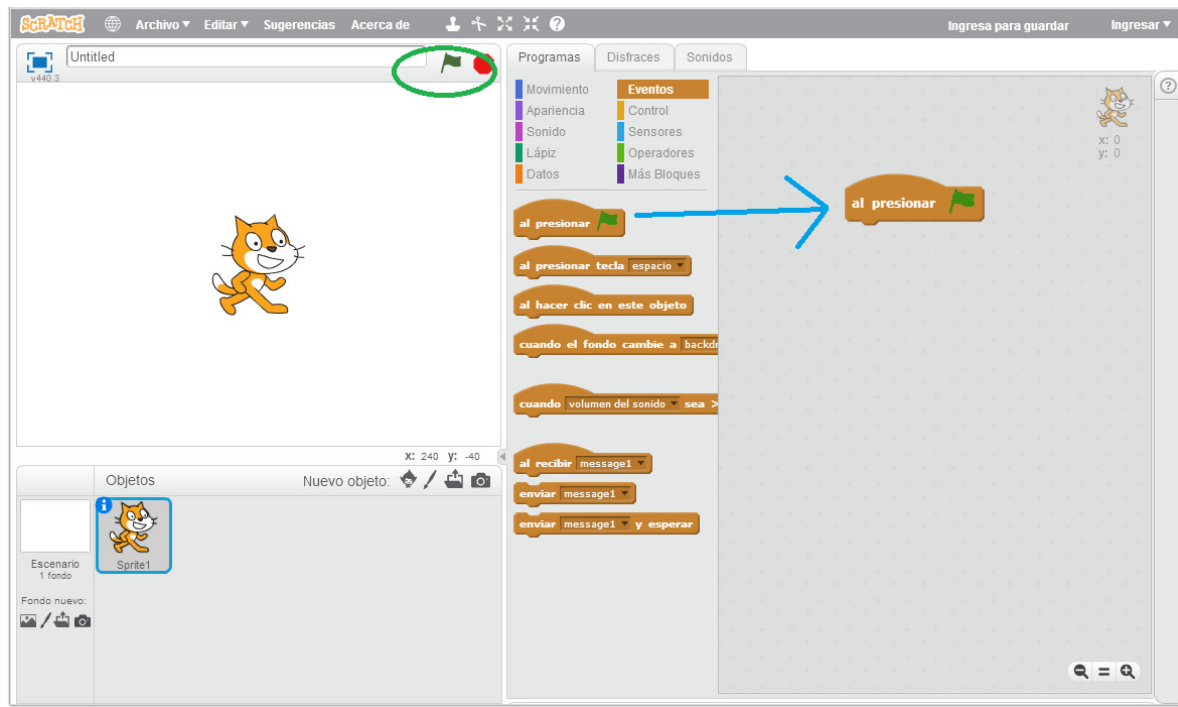
Los comandos (bloques)



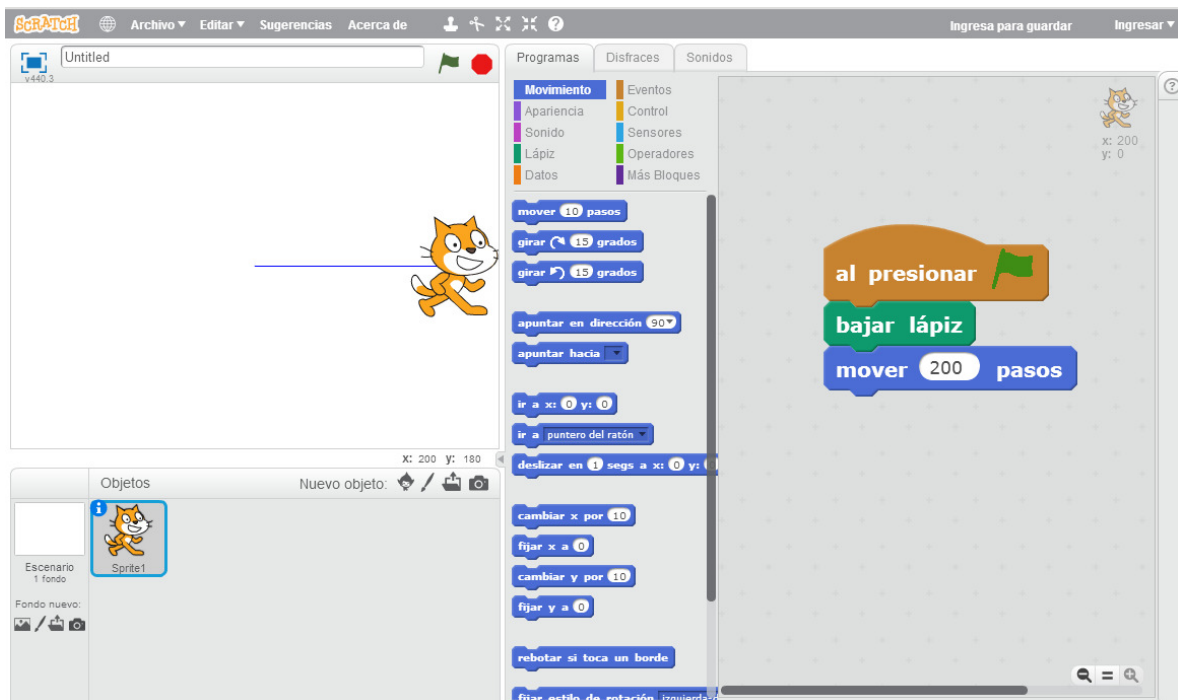
están agrupados por “categorías” (distinguibles por sus colores), y se encajan como piezas de Lego:



Comenzaremos seleccionando la “categoría” **Eventos**. De los bloques que se ofrecen, arrastraremos el bloque “**al presionar**

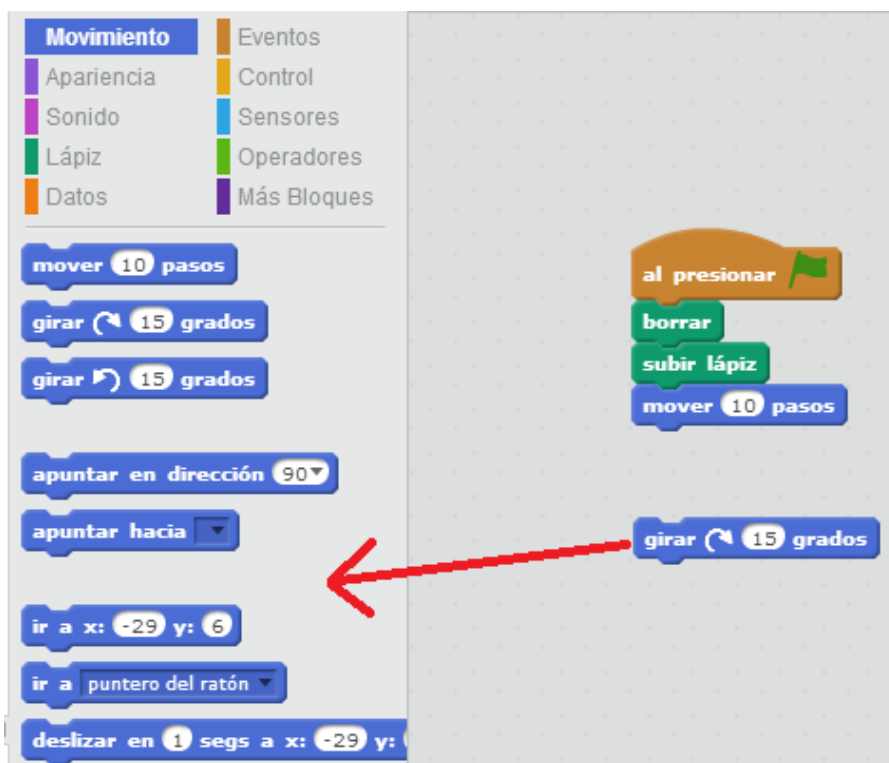
” a la ventana de la derecha (**área de programas**). Cualquier programa (sucesión finita de bloques) que encajemos debajo se ejecutará al “hacer click” en la bandera encima de la ventana izquierda (**escenario**).


Entonces el cursor gráfico (gato, que se corresponde con la “tortuga” clásica de Logo) ejecutará los movimientos correspondientes. Por ejemplo⁴:



⁴ El tamaño de los bloques en la ventana de la derecha se pueden cambiar con las lupas (abajo, a la izquierda).

Para reformar un programa (también denominado procedimiento), podemos intercalar bloques en él o arrastrar un(os) bloque(s) a borrar a la zona central para eliminarlo(s):



Guardar un archivo en disco: Archivo / Guardar a tu computadora

Cargar un archivo desde disco: Archivo / Subir de tu computadora


Se guardan con la extensión: SB2

Se puede guardar como archivo PNG la ventana *escenario* (ventana gráfica) haciendo click con el botón derecho del ratón en esa zona.

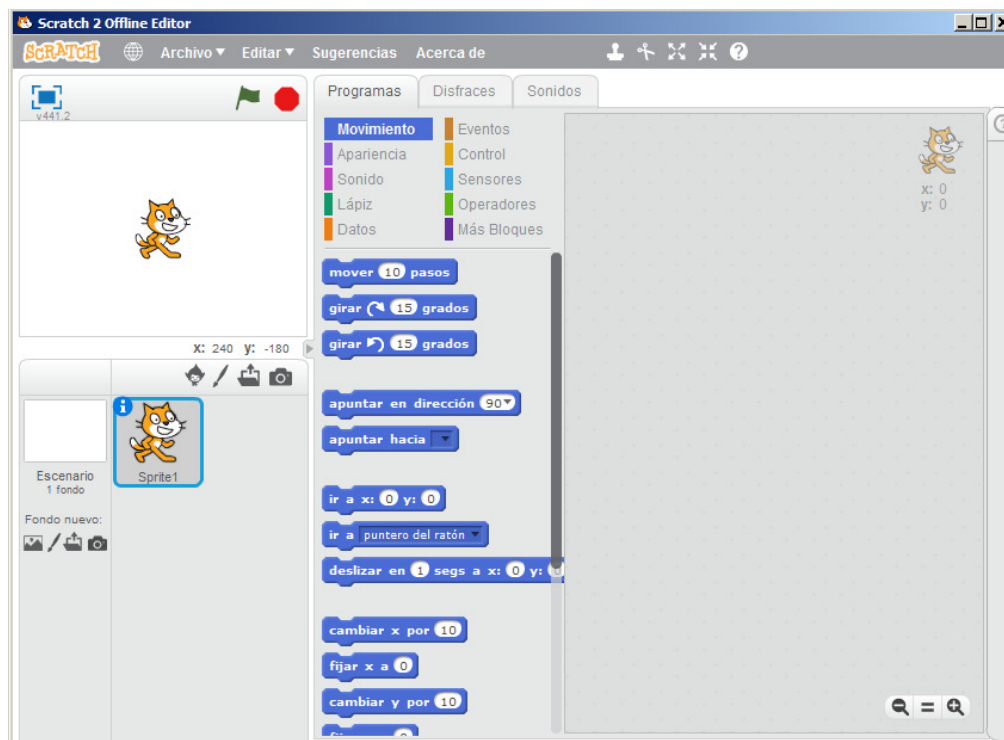
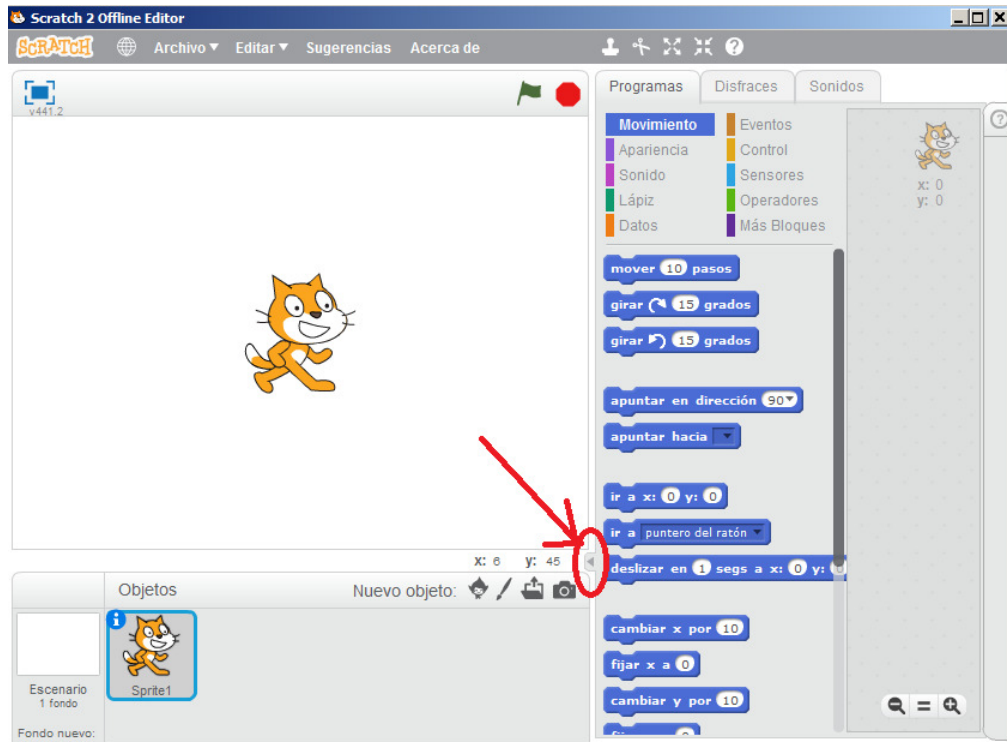
En la ventana *área de programas* se puede borrar todo o introducir un comentario (haciendo click con el botón derecho del ratón en esa zona).

Para duplicar un bloque situado en esa área se puede hacer click con el botón derecho del ratón sobre él.

Si Scratch está activado en otro idioma se “traducen” automáticamente los bloques de un programa que se cargue desde disco.

El idioma por defecto es el configurado por Windows. Se puede cambiar en: 

Se puede cambiar la distribución de espacios a la izquierda y derecha de la columna de bloques (comandos) haciendo click en el pequeño triángulo en la esquina inferior derecha de la ventana *escenario*.



Hoja 1: INICIACION A LA GEOMETRIA DE LA TORTUGA

Por defecto el gato “mira” hacia la derecha y el “lápiz” que porta está “subido” (no como en Logo, en el que la tortuga suele empezar orientada hacia arriba y el lápiz bajado).

Categorías: **Movimiento** y **Lápiz**. Descripción de los bloques:

Avanzar 10 pasos (10 pixels aproximadamente)

mover 10 pasos

Girar a la derecha 15 grados (sentido horario)

girar 15 grados

Girar a la izquierda 15 grados (sentido antihorario)

girar 15 grados

(Las entradas pueden ser negativas; así, por ejemplo, se puede retroceder con un avance negativo).

Borrar Pantalla (no mueve la tortuga)

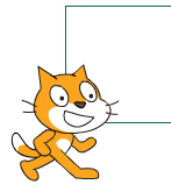
borrar

Tortuga al centro:

ir a x: 0 y: 0

Ejemplo: Dibujar un cuadrado de lado 100.

```
al presionar
  borrar
  bajar lápiz
  mover 100 pasos
  girar 90 grados
  mover 100 pasos
  girar 90 grados
  mover 100 pasos
  girar 90 grados
  mover 100 pasos
  girar 90 grados
```



Subir Lápiz (no dejar traza al moverse)

subir lápiz

Bajar Lápiz (dejar traza al moverse)

bajar lápiz

Ocultar la Tortuga

esconder

Mostrar la Tortuga

mostrar

Cambiar el color del rastro (valores admisibles: 0 a 200):

fijar color de lápiz a 0

o bien:

fijar color de lápiz a

y clicks en:

(junto a)

(hasta llegar al color deseado)

o bien:

cambiar color del lápiz por 10

(para ir incrementado el valor del color).

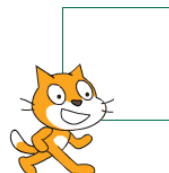
Cambiar el grosor del trazo:

fijar tamaño de lápiz a 1

(No hay GOMA) (No hay color negro o blanco) (No hay RELLENA)

Hoja 1: INICIACION A LA GEOMETRIA DE LA TORTUGA (Continuación)

Ejemplo: Dibujar un cuadrado de lado 100, haciendo uso de **REPETIR**.

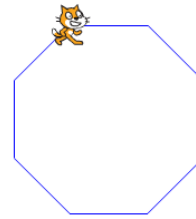


EJERCICIOS:

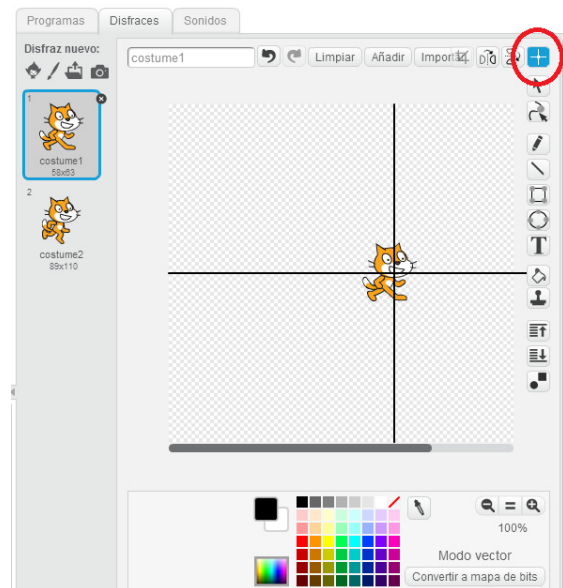
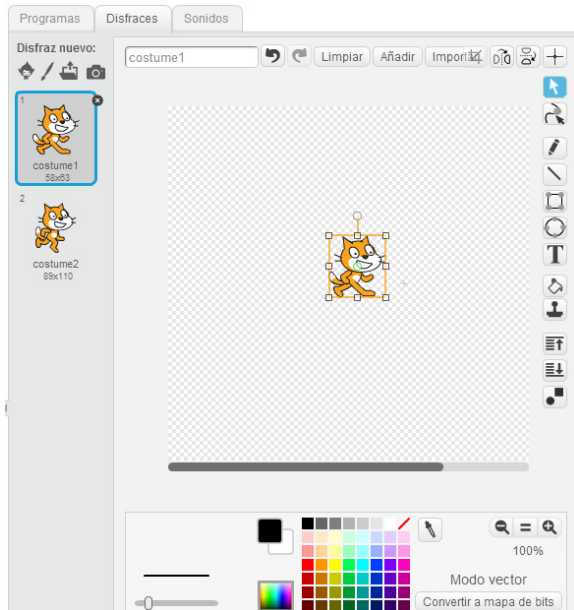
1. Dibujar el borde de un cuadrado en sentido antihorario.
 2. Dibujar el borde de un rectángulo en sentido horario.
 3. Dibujar el borde de un triángulo equilátero en sentido antihorario.
 4. Dibujar el borde de un cuadrado de centro el centro de pantalla.
 5. Dibujar el borde de un hexágono regular.
 6. Dibujar el borde de un cuadrado, haciendo uso de **REPETIR**.
 7. Dibujar el borde de un triángulo equilátero, haciendo uso de **REPETIR**.
 8. Dibujar el borde de un hexágono regular, haciendo uso de **REPETIR**.
 9. Dibujar un rectángulo haciendo uso de **REPETIR**.
 10. Dibujar los radios de una rueda con 24 radios.
 11. *⁵ Dibujar el borde de un hexágono regular y las diagonales de extremos cada dos vértices opuestos del hexágono (hay, al menos, tres formas distintas de hacerlo).
-

⁵ Los ejercicios marcados con un asterisco tienen cierta dificultad. Los marcados con dos asteriscos son más complejos.

Hoja 1: INICIACION A LA GEOMETRIA DE LA TORTUGA (Continuación)

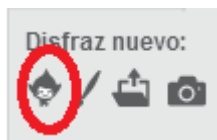
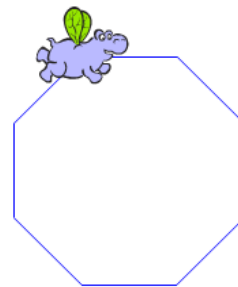


Se puede cambiar el tamaño del gato (cursor gráfico) haciendo click en “Disfraces” (en la barra gris), seleccionando el gato y cambiando el tamaño del recuadro que lo contiene. Luego comprobar haciendo que se vean los ejes que el gato está centrado en el origen de coordenadas (arrastrar los ejes un poco si no lo está):

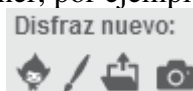


(para volver a la ventana usual basta hacer click en “Programas” en la barra gris).

En “Disfraces” se puede cambiar el cursos gráfico por otro. Para elegir uno predefinido, hacer click



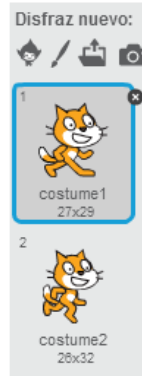
en y elegirlo. Así podemos obtener, por ejemplo:



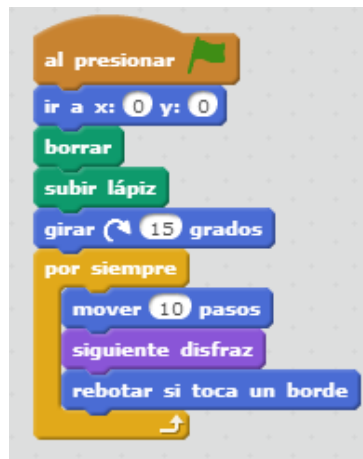
Los ya usados pueden elegirse de la columna bajo

Hoja 1: INICIACION A LA GEOMETRIA DE LA TORTUGA (Continuación)

Si tenemos dos disfraces muy similares, podemos conseguir una animación usando **siguiente disfraz**. Por ejemplo, si tenemos:



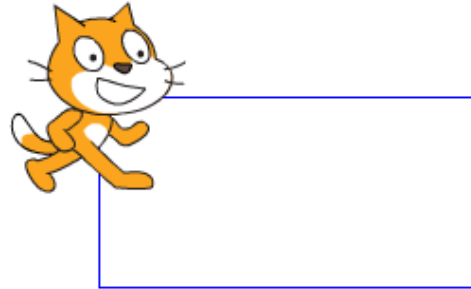
con:



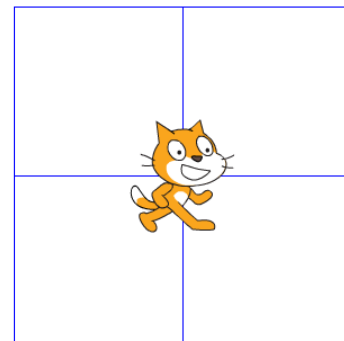
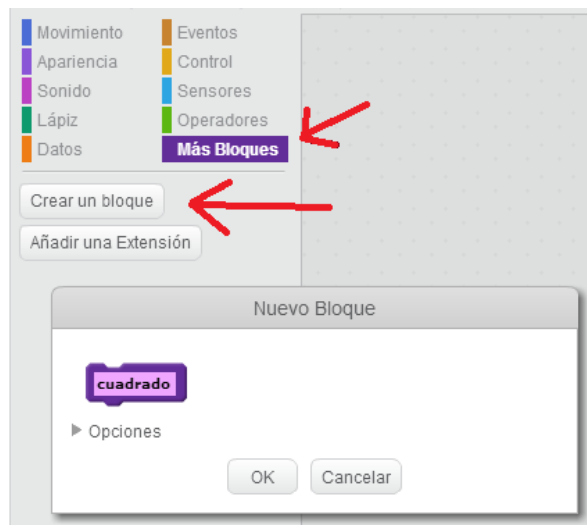
conseguimos que parezca que el gato va “andando” (y rebotando en los bordes de la ventana gráfica).

Hoja 2: PROCEDIMIENTOS Y SUBPROCEDIMIENTOS

Procedimiento titulado RECTANG, que dibuje un rectángulo de lados fijos:

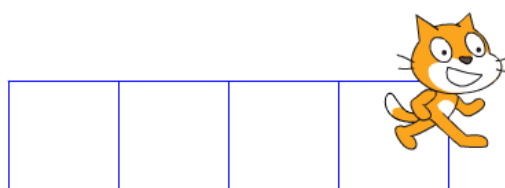
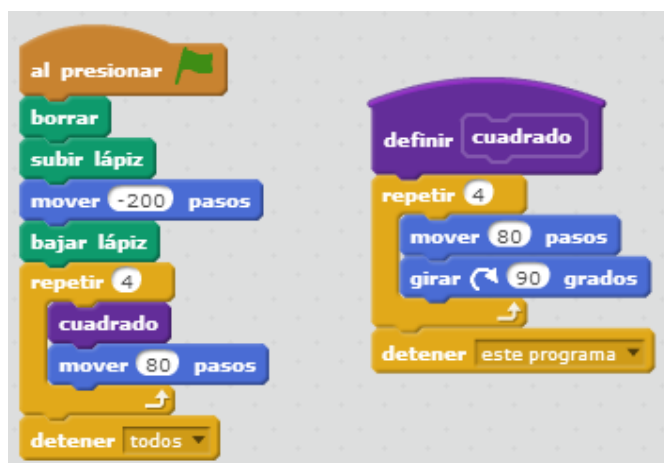




Programa que dibuje una colección de cuadrados con un vértice común, cada uno de ellos girado 36 grados respecto del anterior (llamaremos COLECUAD al procedimiento y CUADRADO al subprocedimiento):



Hoja 2: PROCEDIMIENTOS Y SUBPROCEDIMIENTOS (Continuación)

Procedimiento que dibuja una fila horizontal de cinco cuadrados adyacentes de lado 80, llamando a un subprocedimiento CUADRADO (comenzar 200 pasos a la izquierda de la posición en que empieza el gato):



Nota: haciendo doble click sobre  se ejecuta el procedimiento y al hacerlo sobre  se ejecuta el subprocedimiento.

EJERCICIOS

1. Programa que dibuje una colección de hexágonos regulares con un vértice común, cada uno de ellos girado 60 grados respecto del anterior, anidando un **REPETIR** dentro de otro **REPETIR**⁶.
 2. Hacer lo mismo llamando a un subprocedimiento **HEXAGONO** en lugar de anidando dos **REPETIR**.
 3. * Programa que dibuje una fila horizontal de cinco triángulos equiláteros, cuyas bases están contenidas en la misma recta (llamando a un subprocedimiento).
 4. *. En el procedimiento anterior, si consideramos un triángulo y los adyacentes a él, pueden compartir un vértice, cortarse o están separados. Pensar cuando se da cada caso.
-

⁶ Si el cursor gráfico se sale mucho de los límites de pantalla, no siempre “regresa” dentro de los límites de pantalla correctamente. Fijar en este ejercicio como longitud de los lados de los hexágonos 90 pasos o menos.

Hoja 3: PROCEDIMIENTOS CON ENTRADAS (VARIABLES)

Un procedimiento que dibuja un cuadrado cuyo lado mide el número que se introduce como variable L (100 en el ejemplo más abajo):

Movimiento

Apariencia

Sonido

Lápiz

Datos

Eventos

Control

Sensores

Operadores

Más Bloques

Crear una variable

Crear una lista

Variable nueva

Nombre de la variable:

☒ Para todos los objetos ☐ Sólo para éste objeto

OK

Cancelar

Movimiento

Apariencia

Sonido

Lápiz

Datos

Eventos

Control

Sensores

Operadores

Más Bloques

Crear una variable

☒ L

fijar L a 0

cambiar L por 1

mostrar variable L

esconder variable L

Crear una lista

al presionar

fijar L a 100

ir a x: 0 y: 0

borrar

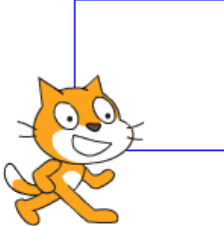
bajar lápiz

repetir 4

mover L pasos

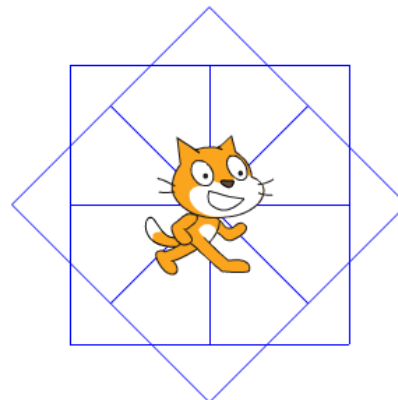
girar 90 grados

detener todos

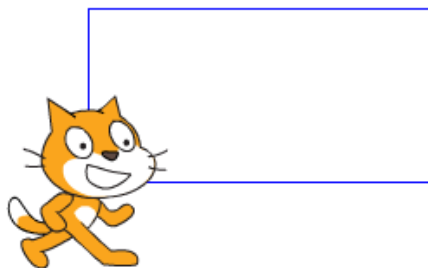
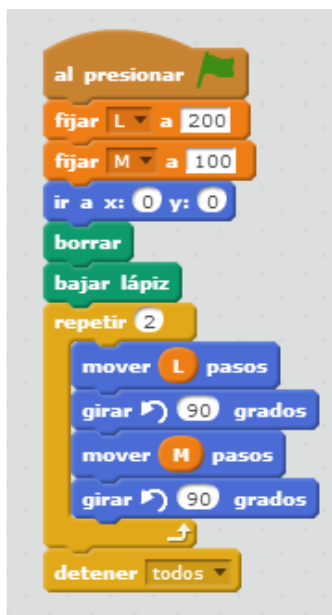


Hoja 3: PROCEDIMIENTOS CON ENTRADAS (VARIABLES) (Continuación)

Un procedimiento con entradas "transfiere" los valores de las variables a un subprocedimiento suyo.



Procedimiento que dibuje un rectángulo cuyos lados midan los números almacenados en las variables L y M:

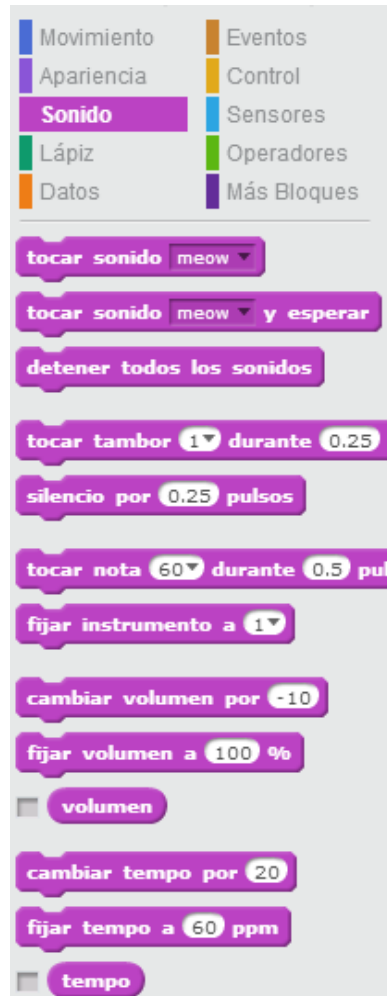


EJERCICIOS

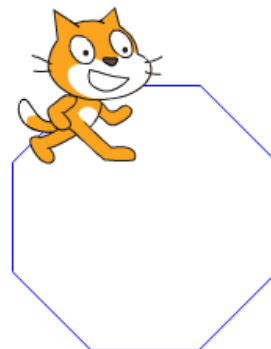
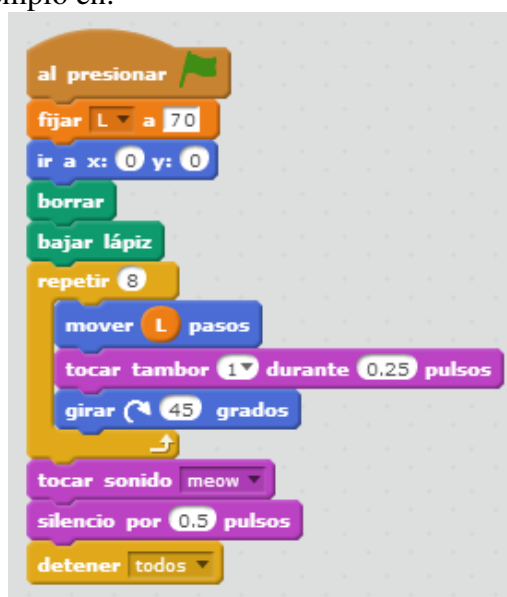
1. Procedimiento que dibuje un triángulo equilátero de lado L.
2. Procedimiento que dibuje los 36 radios de longitud L de una rueda.
3. Procedimiento que dibuje una bandera consistente en un mástil de longitud L y cuya tela sea rectangular de lados A y B.
4. Programa que dibuje una fila horizontal de M baldosas cuadradas (no separadas ni solapadas) de lado L
5. * Programa que dibuje una fila de N ruedas (de 36 radios de longitud L), de modo que la distancia entre los centros de dos contiguas sea D.

Hoja 3: PROCEDIMIENTOS CON ENTRADAS (VARIABLES) (Continuación)

Se pueden incluir sonidos:



Como por ejemplo en:



(toca el tambor tras dibujar cada segmento y dice “miau” al terminar)

Hoja 4: OPERACIONES

Las operaciones se encuentran en la categoría **Operadores**

Operaciones binarias:



Sumar, restar, multiplicar y dividir: . Ejemplo:

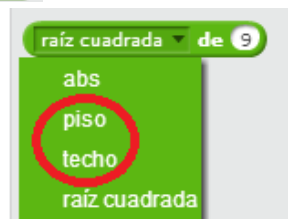


Operaciones unarias:

Para calcular la raíz cuadrada de 64:



Para redondear un número:

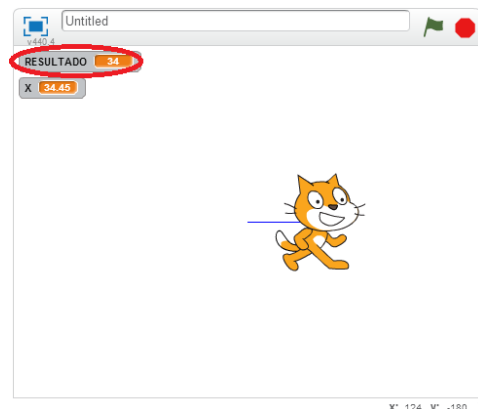
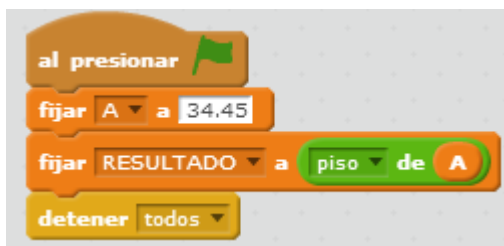


Parte entera por defecto y por exceso:

Para número natural aleatorio entre 1 y 10 (ambos incluidos):

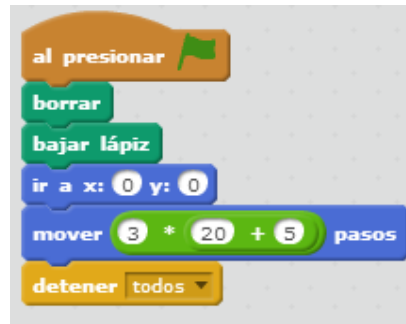


Se puede realizar un cálculo aritmético y en la ventana de la izquierda se muestra el resultado (si le hemos llamado así a la variable donde se almacena el resultado final:



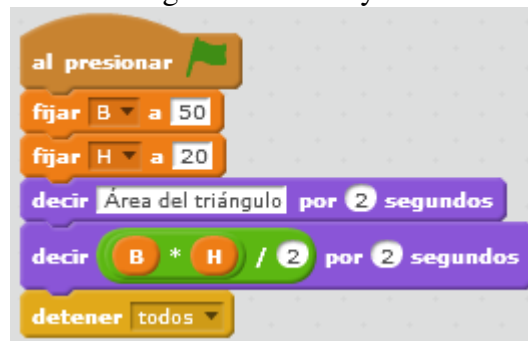
Hoja 4: OPERACIONES (Continuación)

Es posible incluir una operación como una de las entradas de otra operación:



donde se pide avanzar $3 * (20+5)$.

Procedimiento que calcula el área de un triángulo de base B y altura H:



Hoja 4: OPERACIONES (Continuación)

Procedimiento que calcula el cociente y el resto de la división entera del número A entre el B:



The image shows a Scratch script in the 'Scripts' area. The script starts with 'al presionar' (when clicked) followed by 'fijar A a 36' (set A to 36), 'fijar B a 5' (set B to 5), 'fijar COCIENTE a piso de A / B' (set COCIENTE to floor of A / B), 'fijar RESTO a A - B * COCIENTE' (set RESTO to A - B * COCIENTE), and finally 'detener todos' (stop all). To the right of the script, there are four monitors: 'A' with value 36, 'B' with value 5, 'COCIENTE' with value 7, and 'RESTO' with value 1. The Scratch cat is visible in the center of the stage.

Procedimientos que simulan la tirada de un dado:



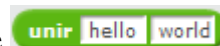
The image shows a Scratch script in the 'Scripts' area. The script starts with 'al presionar' (when clicked) followed by 'fijar DADO a número al azar entre 1 y 6' (set DADO to random number between 1 and 6), and finally 'detener todos' (stop all).

Para simular la tirada de una moneda con un 5 en una cara y un 0 en la otra (respectivamente) podríamos sustituirla operación en el programa anterior por:



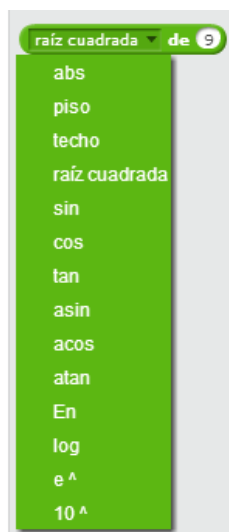
The image shows a Scratch script in the 'Scripts' area. The script starts with 'al presionar' (when clicked) followed by 'fijar DADO a 5 * número al azar entre 0 y 1' (set DADO to 5 * random number between 0 and 1), and finally 'detener todos' (stop all).

Si se desea que el gato “diga” o “piense” más de una cosa, puede usarse




The image shows a Scratch script in the 'Scripts' area. The script starts with 'al presionar' (when clicked) followed by 'decir hello world por 2 segundos' (say hello world for 2 seconds).

Hoja 4: OPERACIONES (Continuación)



Otras operaciones unarias:

Obsérvese que En es “logaritmo neperiano” (habitualmente representado con “ln”).

No existe una operación “resto”, pero puede usarse en su lugar: 

EJERCICIOS

- 1 Reformar el procedimiento que calcula la división entera de modo que el gato indique que ha calculado el cociente y el resto de la división entera.
- 2 Procedimiento que calcule la medida de la hipotenusa de un triángulo rectángulo cuyos catetos midan X e Y.
- 3 Procedimiento que dibuje los N radios de longitud L de una rueda (sólo los radios).
- 4 Procedimiento que dibuje un polígono regular de N lados de longitud L (cada lado).
- 5 * Procedimiento que saque al azar uno de los números siguientes: 20, 25, 30, 35, 40, 45, 50.
- 6 * Procedimiento que simule la suma de la tirada de dos dados.
- 7 ** Procedimiento que calcule x^n (supuesto que n es un número natural mayor que 1), sin usar el bloque 10^{\wedge} .
Nota: Usar **REPETIR** y una variable para ir guardando los resultados parciales.
- 8 ** Hacer algo análogo para obtener el producto por un número natural no nulo a partir de la operación suma.
- 9 ** Hacer algo análogo para obtener la operación suma a partir de la operación elemental “sumar 1”.
Nota: Los ejercicios 7,8,9 corresponden con la idea de la axiomática de Peano.
- 10 ** Procedimiento que calcule la factorial de n.
Nota: Usar **REPITE**, una variable "contador" y otra variable para ir guardando los resultados parciales.
- 11 ** Procedimiento con una entrada, N, que calcule $2+4+8+\dots+2^n$ usando el comando **POTENCIA**.

Nota: Implementaremos de otro modo más elegante los ejercicios 7, 8 y 9 en la hoja 7.

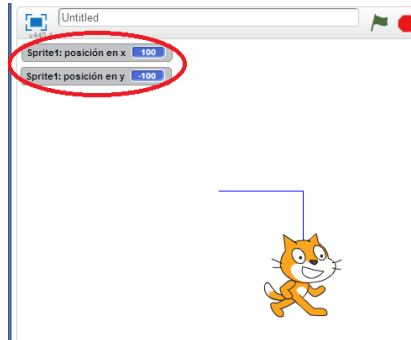
Hoja 5: COORDENADAS Y RUMBO

Se puede pedir la posición (abscisa u ordenada) del cursor gráfico (gato, por defecto), siendo el origen de coordenadas el centro de ventana gráfica y la unidad de longitud un “paso” del gato:

para mostrar la abscisa, marcar:



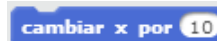
para mostrar su ordenada, marcar:



Para que la tortuga se traslade (en línea recta) al punto (100,50):



Para que se traslade al punto de abscisa 10 (sin cambiar la ordenada):



Para que se traslade al punto de ordenada 10 (sin cambiar la abscisa):



(estas primitivas no alteran la orientación de la tortuga).

Para mostrar el rumbo de la tortuga, marcar:



El rumbo se mide en grados, respecto de la vertical hacia arriba y en sentido horario, es decir, respecto del semieje y+ (como lo haría el timonel de un barco).

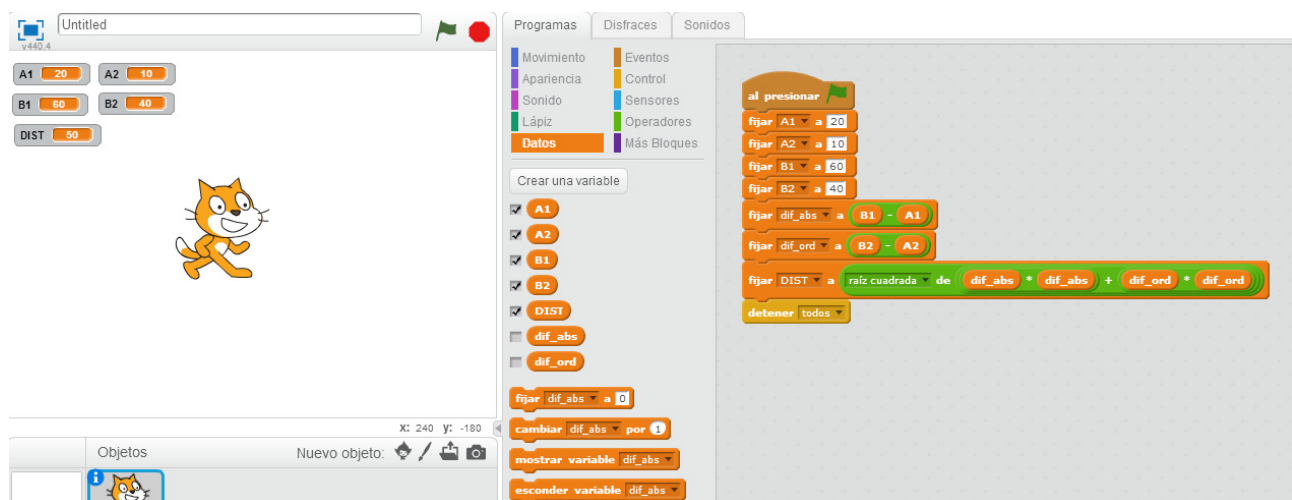
Para orientar la tortuga con rumbo 45 grados:



(esta primitiva no altera la posición de la tortuga -sus coordenadas-, sólo su orientación).

(No existe PONRUMBO HACIA en el sentido de Logo)

Programa con 4 entradas (las coordenadas de 2 puntos) que calcula y muestra la distancia entre ellos



Hoja 5: COORDENADAS Y RUMBO (Continuación)

EJERCICIOS

- 1 Dibujar un rectángulo usando sólo las primitivas **fijar x a** y **fijar y a**.
 - 2 Dibujar un triángulo rectángulo isósceles usando sólo **ir a x: ... y:**
 - 3 Dibujar un triángulo rectángulo de catetos de longitud 100 y tal que su hipotenusa está contenida en el eje x (usando para mover al gato solamente los comandos de este tema).
 - 4 Construir un procedimiento cuyas 4 entradas sean las coordenadas de 2 puntos y que dibuje el segmento de extremos dichos puntos.
 - 5 Definir un procedimiento cuyas 8 entradas sean las coordenadas de 4 puntos y dibuje el cuadrilátero de vértices dichos puntos.
 - 6 Definir unos subprocedimientos N, S, E, W, NE, NW, SE, SW, que pongan a la tortuga rumbo a estos puntos cardinales.
 - 7 ** Definir un procedimiento cuyas 6 entradas sean las coordenadas de 3 puntos y tal que el gato acabe dentro del interior del triángulo.
 - 8 ** Definir un procedimiento con cuatro entradas (A1, A2, B1, B2) que mida, usando comandos de la Hoja 4, el ángulo que forman la recta que pasa por (0,0) y (A1,A2) con la que pasa por (0,0) y (B1,B2). Se supone que los dos puntos están en el primer cuadrante. Sugerencia: determinar lo primero el ángulo que forma cada una de las rectas con la horizontal.
-

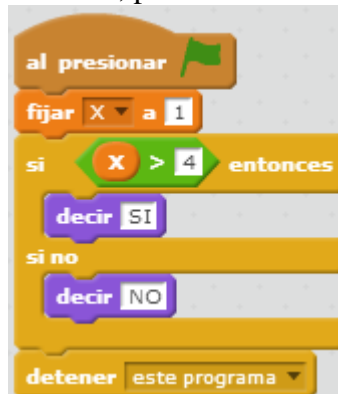
Hoja 6: CONDICIONALES Y OPERACIONES LÓGICAS

Procedimiento que compara un número dado con 4 y, si el número es mayor que 4, contesta *MAYOR*:



Nota importante: no terminar este programa con **detener todos** o no se verá el mensaje “que dice” del gato.

Procedimiento que compara un número con 4, para ver si es mayor que 4 o no lo es:

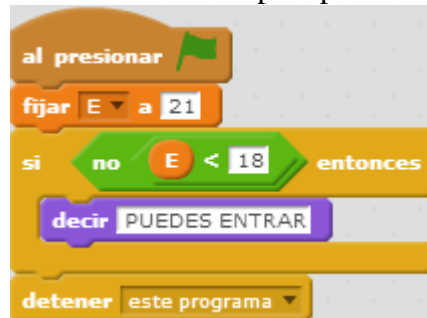


Procedimiento que decide si un número es cero, o no lo es:



Hoja 6: CONDICIONALES Y OPERACIONES LÓGICAS (Continuación)

Procedimiento que decide si la edad no es suficiente para poder entrar (usando negación):



Procedimiento que, si se verifican las dos condiciones $T > 18$ y $T < 25$, indica que la temperatura es agradable:



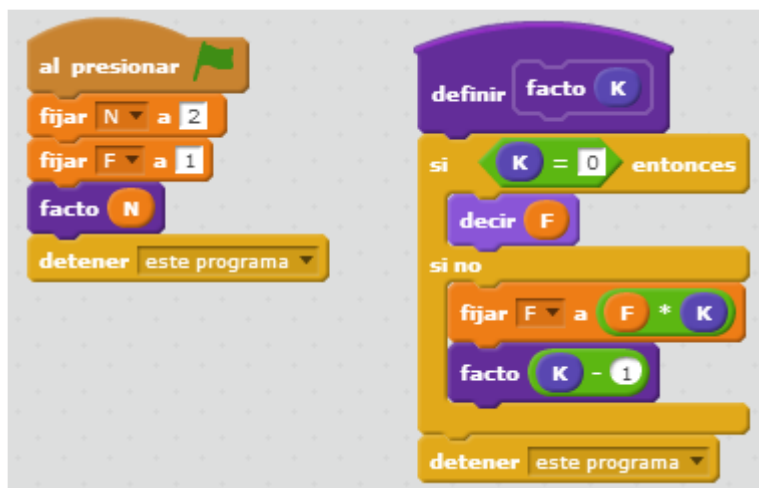
Procedimiento que, si se verifica una (al menos) de las dos condiciones $E > 70$ o $E < 6$, informa que la edad permite asistir gratis a cierto espectáculo:



Hoja 6: CONDICIONALES Y OPERACIONES LÓGICAS (Continuación)

Un procedimiento que se llama a sí mismo (esto es, que es subprocedimiento de si mismo) se denomina *procedimiento recursivo*. Una definición *recursiva* típica en Matemáticas es la de factorial. En lugar de definir: $n! = n \cdot (n-1) \cdot \dots \cdot 2 \cdot 1$, se puede definir:

$$n! = \begin{cases} 1, & \text{si } n=0 \\ n \cdot ((n-1)!), & \text{si } n>0 \end{cases}$$



Nota: Para que el subprocedimiento tenga una entrada, al definirlo elegir “Añadir entrada numérica”.

EJERCICIOS

1. Procedimiento que decida si, dados dos números, el primero es mayor o igual al segundo, respondiendo “sí” en caso afirmativo.
2. Procedimiento que compruebe si la edad, e, de una persona verifica la condición $17 < e < 65$, respondiendo “sí” en caso afirmativo.
3. Procedimiento que compruebe si la edad, e, de una persona verifica la condición $17 \leq e < 65$, respondiendo “sí” en caso afirmativo.
4. Procedimiento, que verifique si el número A es múltiplo del B (usando negación), respondiendo “sí” en caso afirmativo y “no” en caso negativo.
Nota: Se puede hacer usando chequeando el resto o comparando el cociente con el resultado de la división con decimales (/).
5. Procedimiento que decida si dos números no nulos son del mismo signo (comparando su producto con 0).
6. Procedimiento con dos entradas que aplique a la primera entrada (precio) el % de descuento indicado por la segunda entrada.
7. * Comprobar que añadir IVA y hacer un descuento permutan. Pensar la razón.
8. Procedimiento que decida si un número es menor que 100, igual a 100 o mayor que 100.

Hoja 6: CONDICIONALES Y OPERACIONES LÓGICAS (Continuación)

9. * Procedimiento que califique la nota x de acuerdo con el baremo usual: suspenso si $x < 5$, aprobado si $5 \leq x < 6.5$, notable si $6.5 \leq x < 8.5$, sobresaliente si $8.5 \leq x < 9.5$ y matrícula de honor si $x \geq 9.5$.
 10. Programa que calcule la hipotenusa de un triángulo rectángulo, dados sus dos catetos, usando un subprocedimiento que calcule el cuadrado de un número dado. Sugerencia: en el subprocedimiento usar una variable (global) “resultado” a la que accede el programa principal.
 11. ** Programa recursivo que calcule potencias de exponente natural positivo.
** Programa recursivo que calcule potencias de exponente natural.
 12. ** Programa recursivo que calcule el producto de dos números naturales a partir de la operación +.
 13. ** Programa recursivo que calcule la suma de dos números naturales a partir de la operación “sumar 1”.
 14. ** Programa recursivo que calcule el término n -ésimo de la sucesión de Fibonacci. Esta sucesión es la: 1,1,2,3,5,8,13,21,34,55,... (comienza: 1,1,... y cada término se obtiene sumando los dos anteriores).
 15. ** Programa recursivo que devuelva la suma de los n primeros pares (excluido el cero), esto es: $2+4+6+8+\dots+2 \cdot n$ (Nota: aplicar la fórmula de la suma de términos de una progresión aritmética no es lo que se pide).
 16. ** Programa recursivo que devuelva la suma de los n primeros términos de la progresión aritmética de primer término 5 y razón 3, esto es: 5, 8, 11, 14, 17,... (Nota: aplicar la fórmula de la suma de términos de una progresión aritmética no es lo que se pide).
 17. ** Programa con una entrada, L , que, cuando se ejecute con 15 como entrada, dibuje cuadrados de lados 15, 25, 35, 45, 55, 65 y 75 (estos cuadrados deben tener todos un vértice común y sus lados paralelos). Este programa debe ser recursivo y debe ir enviando a un subprocedimiento cada vez incrementando la entrada del subprocedimiento en 10. Debe usar un condicional para parar (al sobrepasarse el valor 75).
 18. ** Programa que tenga una entrada, N , y escriba los números naturales que son menores que 1000 y cuadrados de un natural. (Idea: no ir comprobando qué números son cuadrados de natural, sino generar con un programa recursivo los cuadrados de los sucesivos naturales y que no pare mientras estos cuadrados se mantengan menores que 1000).
-