

Fundamentos para programación y robótica
Módulo 2 – Programación y robótica
Capítulo 2 – Programar en TortuBots

Agenda

- Las expresiones y los tipos de datos.
- Comandos y sensado.
- Variables en TortuBots.
- Estructuras de control.
 - Estructuras de selección.
 - Estructuras de iteración.

Programar en TortuBots

Objetivos

- Repasar conceptos de programación.
- Continuar familiarizándose con TortuBots.
- Aprender a programar acciones de la tortuga.
- Aprender a programar estructuras de control en TortuBots.
- Resolver problemas sencillos con la tortuga.

Programar en TortuBots

Requisitos para este capítulo

- Los conocimientos del Capítulo 0.
- Haber repasado los conocimientos de programación del Módulo 1.

Programar en TortuBots

Un breve repaso del módulo 1...
Con algunas cosas de TortuBots.

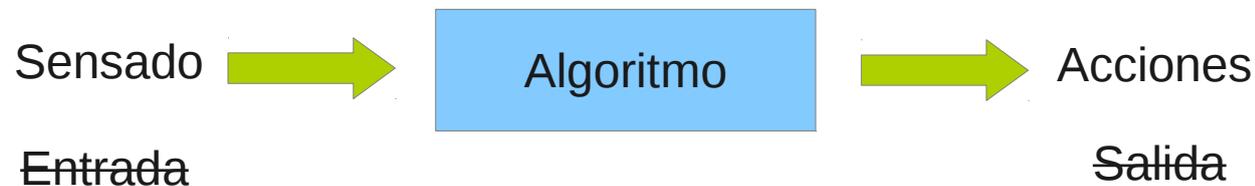
Programar en TortuBots

- Un programa es una secuencia de instrucciones que una computadora puede interpretar y ejecutar.
- Escrito en un lenguaje de programación.
- Para ciertos datos de entrada, el programa aplica un algoritmo y genera una salida.



Programar en TortuBots

- Un robot también tiene un programa. **Lo mismo aplica para la tortuga!**
- También escrito en un lenguaje de programación.
- Pero cambia el esquema...



Programar en TortuBots

- En TortuBots, el lenguaje **son los bloques**.
- Los tipos son los mismos que en Python. Se distinguen según su encastre.

Programar en TortuBots

¿Qué bloques veremos?

- Veremos bloques para construir el algoritmo.
- Bloques de sensado. Es posible realizar mediciones con la PC (Magallanes, XO, etc..) y con el kit Lego.
- Bloques de comandos para la Tortuga y Lego.

Programar en TortuBots

Tipos en TortuBots:

- Numérico. En Python, estaban los enteros (int) y los reales(float). En TortuBots hay bloques que aceptan float e int y otros solamente int.
- String o cadena de caracteres.
- Booleano. Valores de *Verdadero* y *Falso*.

Programar en TortuBots

- Recordar además que el resultado de cualquier *expresión* que devuelve un valor(o sea, una *operación*), tiene un cierto tipo.
- Ej: 0.1 (float) + 2 (int) = 2.1 (float)
- Se distinguen dos tipos de retorno:
 - String o número. Encastre tipo T. Puede traer errores!
 - Booleano. Encastre redondo.



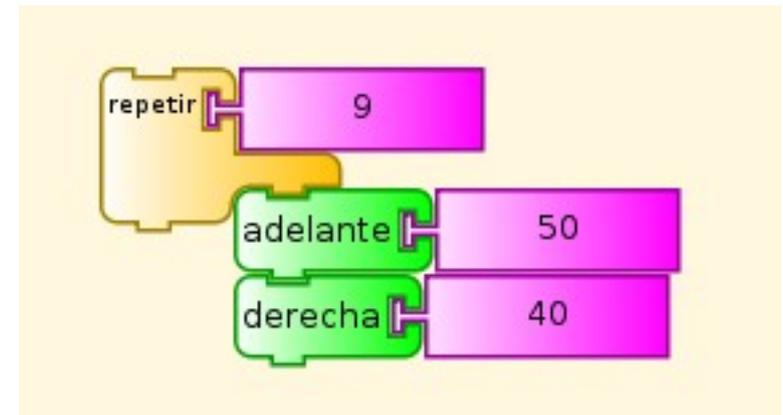
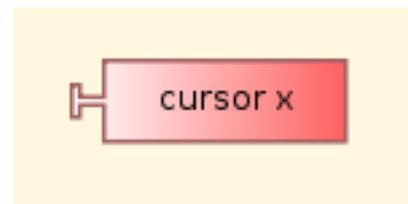
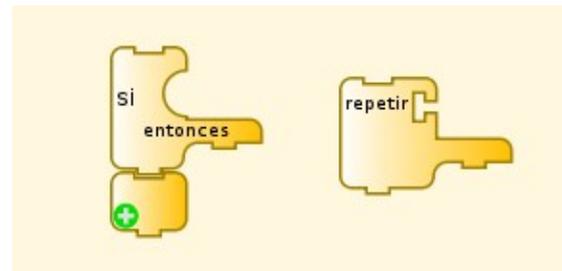
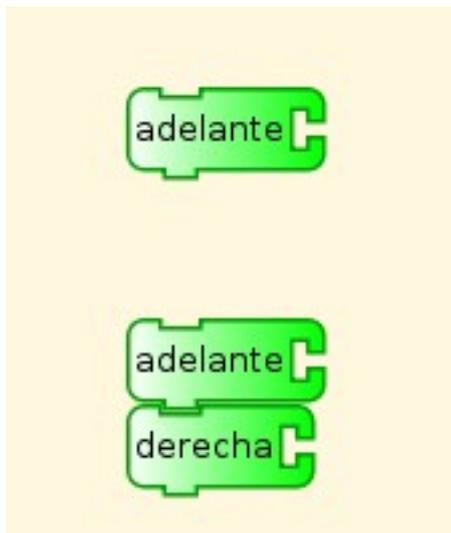
Instrucciones en python:

- Asignación.
 - `mi_variable = 8`
- Entrada / Salida
 - `print mi_variable`
- Estructuras de control.
 - `if mi_variable < 10 :`
 `print 'Es menor que diez.'`

Programar en TortuBots

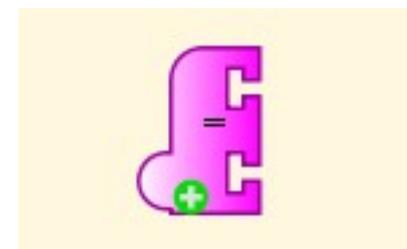
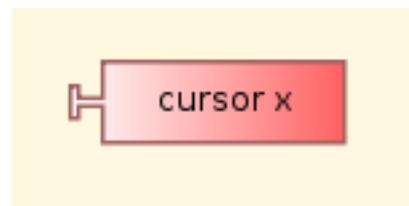
Instrucciones en TortuBots:

- Los **bloques**! Orientados a operar sobre la tortuga.



Programar en TortuBots

- Algunos bloques aceptan *parámetros*. Los parámetros van en los encastrados del lado derecho del bloque.
- Los bloques pueden *valores*. Son encastrados del lado izquierdo.
- Si son instrucciones, tienen conectores arriba y abajo. Sino, son operaciones o valores.



Programar en TortuBots

Veamos formalmente los bloques.

Programar en TortuBots

Comandos de movimiento de la Tortuga.

- La primer paleta a ver en detalle.
- Agrupa parte de las acciones que toma la tortuga.



Comandos de movimiento de la Tortuga.

- Los bloques en la paleta de Tortuga.



Mueve la tortuga hacia delante la distancia dada. Acepta números.



Mueve la tortuga hacia atrás la distancia dada. Acepta números.



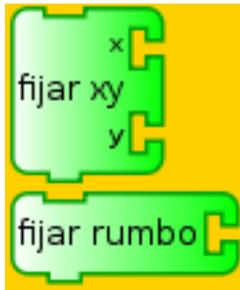
Gira en sentido anti-horario la tortuga la cantidad de grados dada. Acepta números.



Gira en sentido horario la tortuga la cantidad de grados dada. Acepta números.

Programar en TortuBots

Comandos de movimiento de la Tortuga.



fijar xy lleva la tortuga a la posición dada por los dos parámetros (ambos numéricos).

fijar rumbo hace apuntar a la tortuga en la posición dada en grados. Acepta un parámetro numérico.

Programar en TortuBots

Primer ejercicio: Dibujar una letra P

Programar en TortuBots

Solución posible en Python

- Supongamos que la P tiene altura 100.

```
adelante(100)
```

```
derecha(90)
```

```
adelante(50)
```

```
derecha(90)
```

```
adelante(50)
```

```
derecha(90)
```

```
adelante(50)
```

Solución posible



The screenshot displays the TortuBots programming environment. The top toolbar includes icons for a turtle, a pencil, a color palette, a calculator (showing 123), a refresh button, a stack icon, a list icon, an eye, a gear, a monitor, a trash can, and a camera. The workspace contains a sequence of blocks:

- empezar** (yellow start block)
- adelante 100** (green movement block)
- derecha 90** (green rotation block)
- adelante 50** (green movement block)
- derecha 90** (green rotation block)
- adelante 50** (green movement block)
- derecha 90** (green rotation block)
- adelante 50** (green movement block)

The block palette on the left includes:

- Movement: adelante, atrás, limpiar
- Rotation: izquierda, derecha
- Angle/Radius: ángulo, arco, radio
- Coordinates: fijar xy, x, y, coorx, coory
- Direction: fijar rumbo, rumbo

A mouse cursor is positioned over the 'adelante 100' block. A small green turtle icon is visible in the workspace.

Programar en TortuBots

Solución posible

The screenshot displays the TortuBots programming environment. At the top, a toolbar contains various icons for editing and execution. Below the toolbar is a palette of blocks categorized into movement (adelante, atrás, izquierda, derecha, limpiar), rotation (ángulo, arco, radio), and coordinate setting (fijar xy, fijar rumbo). The main workspace shows a sequence of blocks starting with 'empezar', followed by a series of movement and rotation blocks: 'adelante 100', 'derecha 90', 'adelante 50', 'derecha 90', 'adelante 50', 'derecha 90', and 'adelante 50'. A red square with a green turtle icon is positioned on the workspace, indicating the robot's current position and orientation. A mouse cursor is visible over the workspace.

Programar en TortuBots

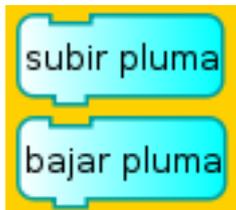
Solución posible

The screenshot displays the TortuBots programming environment. The top toolbar includes icons for a turtle, a pencil, a color palette, a calculator (showing '123 +/-'), a refresh button, a 3D cube, a book, an eye, a gear, a screen, a trash can, and a camera. The left sidebar contains a palette of movement blocks: 'adelante', 'izquierda', 'ángulo arco radio', 'fijar xy', 'coorx', 'coory', 'rumbo', 'atrás', 'derecha', and 'limpiar'. The main workspace shows a sequence of blocks: 'empezar', 'adelante 100', 'derecha 90', 'adelante 50', 'derecha 90', 'adelante 50', 'derecha 90', and 'adelante 50'. On the right, a red line drawing shows a path that starts with a vertical line, followed by a horizontal line to the right, and then a vertical line down, forming an 'L' shape. A small turtle icon is positioned at the start of the path.

Programar en TortuBots

Instrucciones del trazo.

- Los bloques en la paleta de Lápiz.



Sube la pluma (el lápiz) con la que realiza trazos la tortuga.
Luego de ejecutado el comando, la tortuga deja de marcar el fondo.

Realiza lo opuesto a subir pluma. Usualmente utilizado luego de subir pluma para volver a escribir.

Programar en TortuBots

Segundo ejercicio: Dibujar una letra F en base al código del ejercicio anterior

Programar en TortuBots



Solución posible

The screenshot shows the TortuBots programming environment. The workspace contains a sequence of blocks for drawing a square:

- empezar (yellow)
- adelante 100 (green)
- derecha 90 (green)
- adelante 50 (green)
- derecha 90 (green)
- subir pluma (cyan)
- adelante 50 (green)
- bajar pluma (cyan)
- derecha 90 (green)
- adelante 50 (green)

The left sidebar shows a palette of blocks:

- subir pluma
- bajar pluma
- fijar tamaño
- rellenar pantalla (with color and tono sub-blocks)
- comenzar a rellenar
- finalizar rellenado
- tamaño

A mouse cursor is pointing at the 'rellenar pantalla' block in the palette.

Programar en TortuBots

Solución posible

The screenshot displays the TortuBots programming environment. The interface includes a toolbar at the top with icons for a robot, a pencil, a color palette, a calculator (showing '123 +-='), a refresh button, a 3D cube, a document, an eye, a gear, a screen, a trash can, and a camera. A palette on the left contains blocks for 'subir pluma', 'bajar pluma', 'fijar tamaño', 'rellenar pantalla' (with 'color' and 'tono' sub-blocks), 'comenzar a rellenar', and 'finalizar rellenado'. A central workspace shows a red square with a green turtle icon at its bottom-left corner. On the right, a sequence of blocks is arranged vertically: 'empezar' (yellow), 'adelante' (100) (green), 'derecha' (90) (green), 'adelante' (50) (green), 'derecha' (90) (green), 'subir pluma' (cyan), 'adelante' (50) (green), 'bajar pluma' (cyan), 'derecha' (90) (green), and 'adelante' (50) (green). A mouse cursor is pointing at the 'empezar' block.

Programar en TortuBots

Solución posible

The screenshot displays the TortuBots programming environment. The interface includes a toolbar at the top with icons for a turtle, drawing tools, color selection, a numeric keypad (showing '123 +-='), undo, redo, a folder, a camera, a trash can, and a play button. The main workspace is a yellow canvas where a green turtle icon is positioned. To the right of the turtle, a red letter 'F' has been drawn. The programming area on the left contains a sequence of blocks: 'subir pluma', 'bajar pluma', 'fijar tamaño', 'rellenar pantalla' (with 'color' and 'tono' sub-blocks), 'comenzar a rellenar', and 'finalizar rellenado'. On the right side, a vertical stack of blocks defines the drawing sequence: 'empezar', 'adelante' (100), 'derecha' (90), 'adelante' (50), 'derecha' (90), 'subir pluma', 'adelante' (50), 'bajar pluma', 'derecha' (90), and 'adelante' (50).

Programar en TortuBots

El bloque Esperar

Programar en TortuBots

- Las instrucciones en un programa y los bloques en TortuBots se ejecutan en un tiempo imperceptible.
- Usualmente, es lo que se desea.
- Sin embargo, existen casos donde interesa que un programa espere un tiempo determinado sin hacer nada.
- Ej: Un semáforo que mantiene las luces encendidas en un color dado un cierto tiempo.

Programar en TortuBots

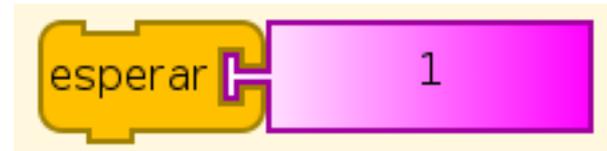
- Para esto se cuenta con un bloque especial: el bloque esperar.



- Se encuentra en la barra de herramientas de estructuras de control.



- El bloque esperar recibe un número que representa la cantidad de segundos a esperar.



- Cuando se ejecuta, el programa queda “trancado” en dicho bloque la cantidad de segundos especificado.

Programar en TortuBots

Veamos un ejemplo, al dibujar la letra P (el primer ejercicio).

Primero, lo hacemos con una estructura repetir.
¿Qué es lo que se repite?

Programar en TortuBots



Bloque esperar - Ejemplo

The screenshot displays the TortuBots programming environment. At the top, a toolbar contains icons for a turtle, a pencil, a palette, a calculator (showing '123 +-='), a refresh button, a cube, a folder, an eye, a gear, a screen, a trash can, and a camera. Below the toolbar is a yellow palette of blocks including 'esperar', 'repetir', 'si entonces', 'si entonces sino', 'parar acción', 'mientras', and 'hasta'. The main workspace shows a sequence of blocks: a yellow 'empezar' block, a green 'adelante' block with a value of 100, a green 'derecha' block with a value of 90, an orange 'repetir' block with a value of 3, and a sub-block containing a green 'adelante' block with a value of 50 and a green 'derecha' block with a value of 90. A mouse cursor is positioned over the 'repetir' block. A small green turtle icon is visible in the workspace.

Programar en TortuBots

Bloque esperar - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons including a turtle, a pencil, a color palette, a calculator (showing '123 +-='), a refresh button, a folder, a book, an eye, a gear, a screen, a trash can, and a camera. Below the toolbar is a yellow palette containing several control blocks: 'esperar', 'por siempre', 'repetir', 'si entonces', 'si entonces sino', 'parar acción', 'mientras', and 'hasta'. The main workspace contains a script starting with an 'empezar' block, followed by 'adelante' (100), 'derecha' (90), a 'repetir' block with a sub-block 'adelante' (50) and 'derecha' (90), and a '3' block. A mouse cursor is pointing at the 'esperar' block in the palette. To the right of the script, a red square outline is drawn around a green turtle icon, indicating the robot's current position and orientation.

Programar en TortuBots

Ahora, insertamos un esperar.

Programar en TortuBots

Bloque esperar - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons including a turtle, a pencil, a color palette, a calculator (123 +-=), a refresh button, a folder, a book, an eye, a gear, a screen, a trash can, and a play button. Below the toolbar is a palette of programming blocks: 'esperar', 'repetir', 'si entonces', 'si entonces sino', 'parar acción', 'mientras', and 'hasta'. The main workspace contains a sequence of blocks: 'empezar' (yellow), 'adelante' (green) with value 100, 'derecha' (green) with value 90, 'repetir' (orange) with value 3, 'esperar' (orange) with value 1, 'adelante' (green) with value 50, and 'derecha' (green) with value 90. A mouse cursor is pointing at the 'esperar' block. A small green turtle icon is visible on the right side of the workspace.

Programar en TortuBots

Bloque esperar - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons including a turtle, a pencil, a color palette, a calculator (showing '123 +--='), a refresh button, a folder, a document, an eye, a gear, a screen, a trash can, and a play button. Below the toolbar is a yellow palette containing several control blocks: 'esperar', 'repetir', 'si entonces', 'si entonces sino', 'parar acción', 'mientras', and 'hasta'. The main workspace is a light yellow area where a sequence of blocks is assembled. The sequence starts with an 'empezar' block, followed by 'adelante' (100), 'derecha' (90), a 'repetir' block with a value of 3, and then a sub-sequence of 'esperar' (1), 'adelante' (50), and 'derecha' (90). A mouse cursor is pointing at the 'esperar' block. To the right of the workspace, there is a small green turtle icon.

Programar en TortuBots

Ejecutamos...

Programar en TortuBots

Bloque esperar - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons including a turtle, a pencil, a color palette, a calculator (showing '123 +-='), a refresh button, a cube, a book, an eye, a gear, a screen, a trash can, and a camera. Below the toolbar is a yellow palette containing several control blocks: 'esperar', 'repetir', 'si entonces', 'si entonces sino', 'parar acción', 'mientras', and 'hasta'. The main workspace contains a sequence of blocks starting with 'empezar', followed by 'adelante 100', 'derecha 90', 'repetir 3', 'esperar 1', 'adelante 50', and 'derecha 90'. A mouse cursor is pointing at the 'empezar' block. To the right of the blocks, a red square path is drawn on the workspace, with a green turtle icon at the bottom center of the square.

Programar en TortuBots

- El bloque esperar no cambia la tarea que termina realizando la tortuga.
- Cambia el comportamiento y espera.
- Esto será importante al leer sensores.

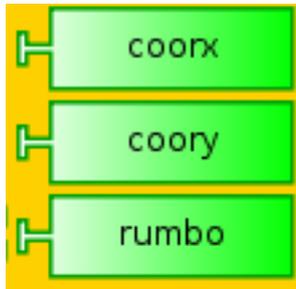


Comandos de sensado básico.

- Se mide una magnitud como puede ser la cantidad de luz, el volumen del sonido ambiente o el tiempo transcurrido.
- Haciendo clic sobre el bloque sin que esté encastrado podemos ver el valor que mide.

Programar en TortuBots

Comandos de sensado básico de la Tortuga.
Disponibles en la paleta de Tortuga.



Los bloques `coorx` y `coory` devuelven la posición `x` e `y` de la tortuga respectivamente.

El bloque `rumbo` devuelve la orientación de la tortuga en grados.

Comandos de sensado básico de la PC.



volumen

Devuelve el volumen del sonido escuchado por el micrófono del PC (Magallanes o XO)



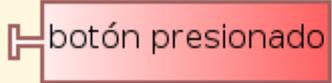
la tortuga ve

Indica el color que “ve” la tortuga debajo de ella en el lienzo.



tiempo

Indica el tiempo transcurrido en segundos desde el principio de la ejecución del programa. (No Tortugarte)



botón presionado

Indica con un valor diferente a 0 si el botón del mouse fue apretado. 0 si no.

Variables en TortuBots

Programar en TortuBots

- Una variable es un espacio de memoria donde se pueden almacenar y consultar valores.
- Las variables en general tienen 4 atributos:
 - Nombre.
 - Tipo.
 - Valor.
 - Alcance.

Programar en TortuBots

- Las variables en TortuBots, se llaman “cajas”.
- Tienen 3 atributos:
 - Nombre.
 - Tipo.
 - Valor.

Programar en TortuBots

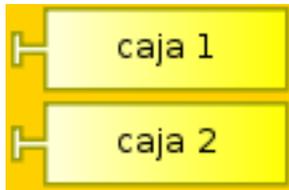
- Las variables en TortuBots se ven como *cajas*.
- Existen siempre dos variables fácilmente utilizables, las cajas 1 y 2.
- Existe la posibilidad de definir nuevas cajas con cualquier nombre. Por ahora, nos mantendremos con la 1 y 2.
- *Cuidado!* Es una mala práctica de programación tener variables con nombres genéricos!

Programar en TortuBots

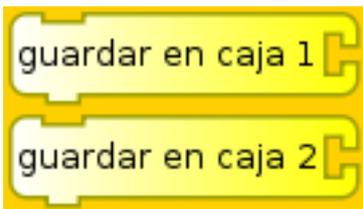
- Se acceden mediante la paleta de variables.
- Interesa leer y escribir dichas variables. Dos operaciones separadas para esto.
- **Recordar la inicialización!**



Instrucciones de variables.



Cada uno de estos bloques permite acceder al valor contenido en su respectiva caja.



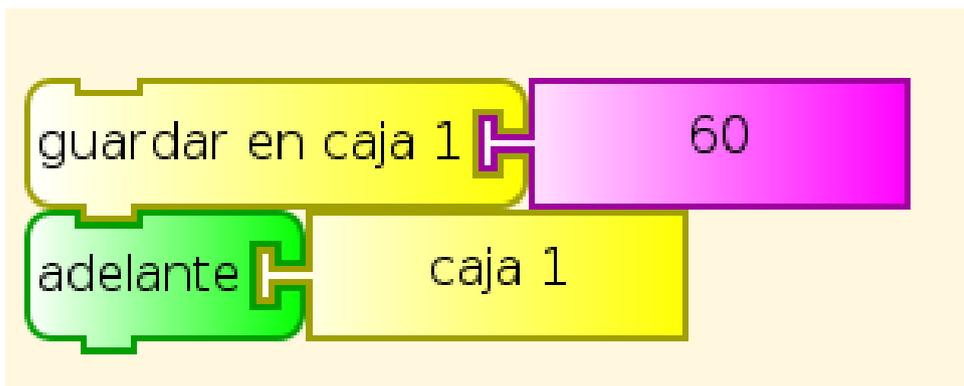
Estos bloques permiten guardar en la caja respectiva, es el equivalente de la **asignación**. El valor que guardan es aquel que se engancha en la parte derecha del bloque. Acepta números o string. Son **instrucciones**.

Programar en TortuBots



Instrucciones de variables – Ejemplo

- El siguiente ejemplo guarda en la caja 1 el valor 60. *Le asigna el valor 60.*
- Luego, avanza el contenido de la caja 1 (que vale 60).



Es el equivalente del siguiente código Python:

```
caja_1 = 60  
adelante(caja_1)
```

Programar en TortuBots



Tercer ejercicio: Dibujar un cuadrado cuyo lado sea el valor de la caja 2. El valor de caja 2 debe establecerse al principio del programa.

Programar en TortuBots

Solución posible – Valor 70 para probar.

The image shows the TortuBots programming environment. At the top is a toolbar with icons for a turtle, a pencil, a palette, a calculator (showing '123 +-='), a refresh button, a folder, a camera, a gear, a monitor, a trash can, and a camera. Below the toolbar is a workspace with a yellow background. On the left, there is a palette of blocks: 'empezar' (yellow), 'texto' (green), 'caja' (yellow), 'acción' (yellow), 'guardar en caja 1' (yellow), 'caja 1' (yellow), 'caja guardar en valor' (yellow), 'acción 1' (yellow), 'guardar en caja 2' (yellow), 'caja 2' (yellow), and 'acción 2' (yellow). A small red minus sign is at the bottom right of the palette. In the center of the workspace, a black arrow points towards the right. On the right side, a sequence of blocks is assembled: 'empezar' (yellow), 'guardar en caja 2' (yellow) with a purple block containing the number '70', 'adelante' (green) with a yellow block containing 'caja 2', 'derecha' (green) with a purple block containing '90', 'adelante' (green) with a yellow block containing 'caja 2', 'derecha' (green) with a purple block containing '90', 'adelante' (green) with a yellow block containing 'caja 2', 'derecha' (green) with a purple block containing '90', and 'adelante' (green) with a yellow block containing 'caja 2'. A small green turtle icon is visible in the bottom left corner of the workspace.

Programar en TortuBots

Solución posible - Ejecutamos

The image shows the TortuBots programming environment. At the top is a toolbar with icons for starting, editing, palettes, undo, redo, zoom, settings, presentation, trash, and help. Below the toolbar is a palette of blocks. The main workspace contains a script of blocks and a 2D simulation of a robot.

Block Palette:

- empezar (Start)
- texto (Text)
- caja (Box)
- acción (Action)
- guardar en caja 1 (Save to box 1)
- caja 1 (Box 1)
- caja guardar en valor (Box save value)
- acción 1 (Action 1)
- guardar en caja 2 (Save to box 2)
- caja 2 (Box 2)
- acción 2 (Action 2)

Script:

- empezar
- guardar en caja 2 (with value 70)
- adelante (forward) (with box 2)
- derecha (right) (with value 90)
- adelante (forward) (with box 2)
- derecha (right) (with value 90)
- adelante (forward) (with box 2)
- derecha (right) (with value 90)
- adelante (forward) (with box 2)

Simulation:

A 2D simulation of a robot (TortuBot) is shown in the bottom left corner. The robot is a green turtle with a red shell, positioned at the bottom left of a red square path.

Programar en TortuBots

Cuarto ejercicio: ¿Qué sucede si duplicamos el valor inicial de la variable?
¿Queda igual?
¿Cambia el tamaño?

Programar en TortuBots

Solución posible – Ahora con 140.

The screenshot displays the TortuBots programming environment. At the top, a toolbar contains icons for a robot, a pencil, a color palette, a calculator (showing '123 +-='), a refresh button, a 3D cube, a document, an eye, a gear, a screen, a trash can, and a camera. The main workspace is divided into a palette on the left and a workspace on the right. The palette contains several blocks: 'empezar' (yellow), 'texto' (green), 'caja' (yellow), 'acción' (yellow), 'guardar en caja 1' (yellow), 'caja 1' (yellow), 'caja guardar en valor' (yellow), 'acción 1' (yellow), 'guardar en caja 2' (yellow), 'caja 2' (yellow), 'acción 2' (yellow), and 'acción 2' (yellow). A mouse cursor is pointing at the 'acción' block in the palette. The workspace on the right shows a sequence of blocks: 'empezar' (yellow), 'guardar en caja 2' (yellow) with a value of '140' (purple), 'adelante' (green) with 'caja 2' (yellow), 'derecha' (green) with '90' (purple), 'adelante' (green) with 'caja 2' (yellow), 'derecha' (green) with '90' (purple), 'adelante' (green) with 'caja 2' (yellow), 'derecha' (green) with '90' (purple), and 'adelante' (green) with 'caja 2' (yellow). A small green turtle icon is positioned in the center of the workspace.

Programar en TortuBots

Solución posible - Ejecutamos

The screenshot displays the TortuBots programming environment. At the top, a toolbar contains icons for a turtle, a pencil, a palette, a calculator (showing '123 +--='), a refresh button, a 3D box icon, a document, an eye, a gear, a screen, a trash can, and a camera. Below the toolbar, a yellow palette contains various blocks: 'empezar' (start), 'texto' (text), 'caja' (box), 'guardar en caja 1' (save to box 1), 'guardar en caja 2' (save to box 2), 'caja guardar en valor' (box save value), 'acción' (action), 'acción 1', and 'acción 2'. A red minus sign is visible at the bottom right of the palette.

The main workspace shows a simulation area on the left with a red square and a green turtle icon at the bottom-left corner. A black arrow points from the simulation area towards the script on the right.

The script on the right consists of the following blocks in sequence:

- empezar
- guardar en caja 2 (with a value of 140)
- adelante (to caja 2)
- derecha (90)
- adelante (to caja 2)
- derecha (90)
- adelante (to caja 2)
- derecha (90)
- adelante (to caja 2)

Programar en TortuBots

Veremos más conceptos de variables durante el resto del capítulo.

Bloques de estructuras de control

Programar en TortuBots

Estructuras de control

- Las estructuras de control son aquellas que nos permiten modificar el flujo de ejecución de un programa. *Ej: Repetir una sección de código.*
- En el caso de TortuBots, son los bloques que nos permiten modificar el flujo de ejecución de una lista de bloques. *Ej: Repetir la ejecución de un conjunto de bloques.*
- Son **instrucciones**.

Programar en TortuBots

Estructuras de control

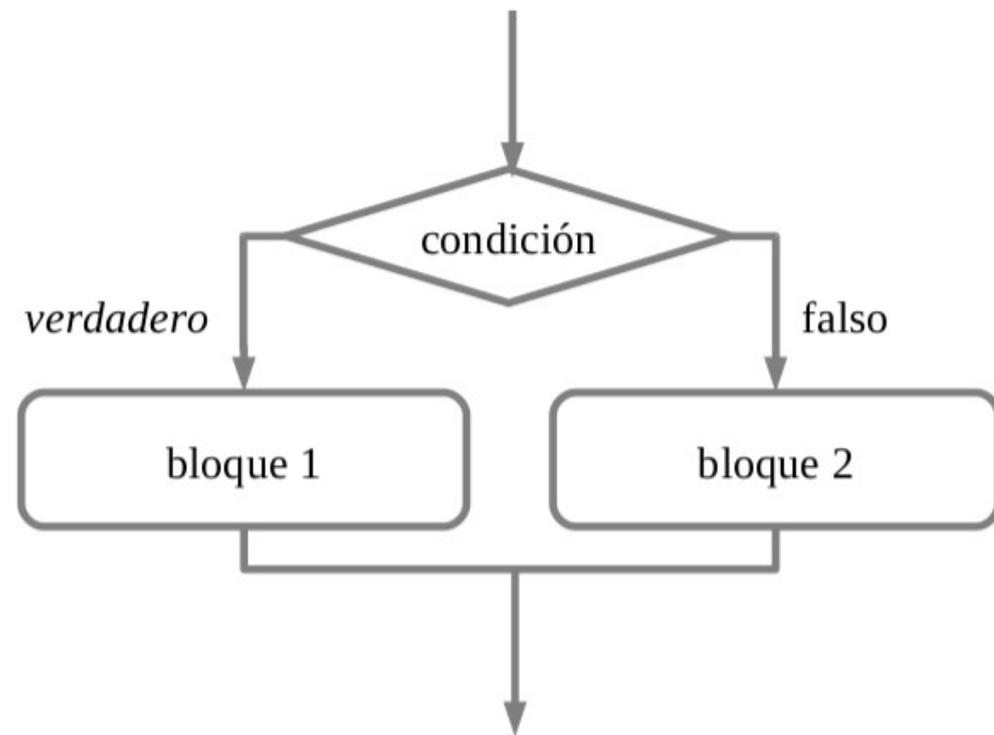
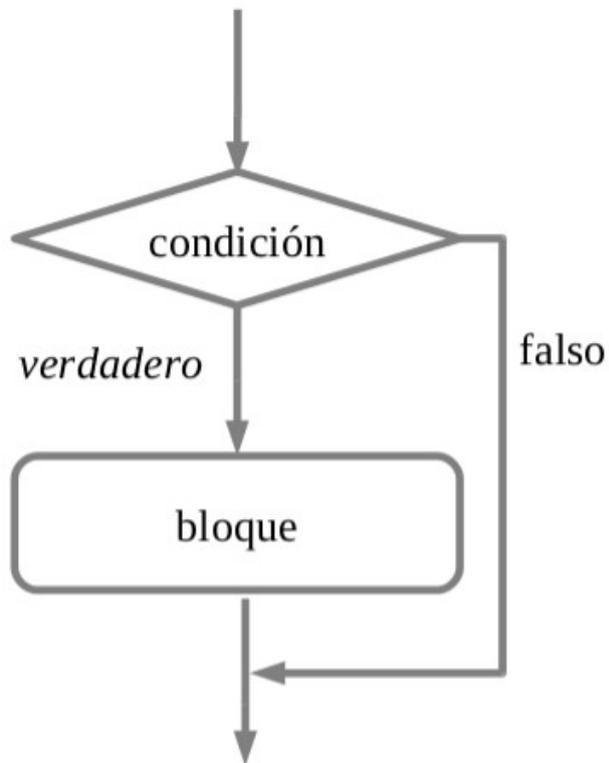
- Hay algunas que son estructuras **condicionales**. Es decir, dependen del resultado de una **expresión** para elegir entre el flujo a seguir.
- Otras dependen de un cierto valor (Ej: Repetir 3 veces.)

Estructuras de control

- Existen de dos tipos:
 - Selección: Permiten elegir entre diferentes flujos del programa. *Ej: if-elif-else*
 - Iteración: Permiten repetir un conjunto de instrucciones. Puede ser una cantidad dada o variable según el caso. *Ej: while*

Estructuras de selección

- En base a una expresión, eligen uno u otro camino.



Estructuras de selección

- En Python

```
if cond1:  
    bloque1  
bloque2
```

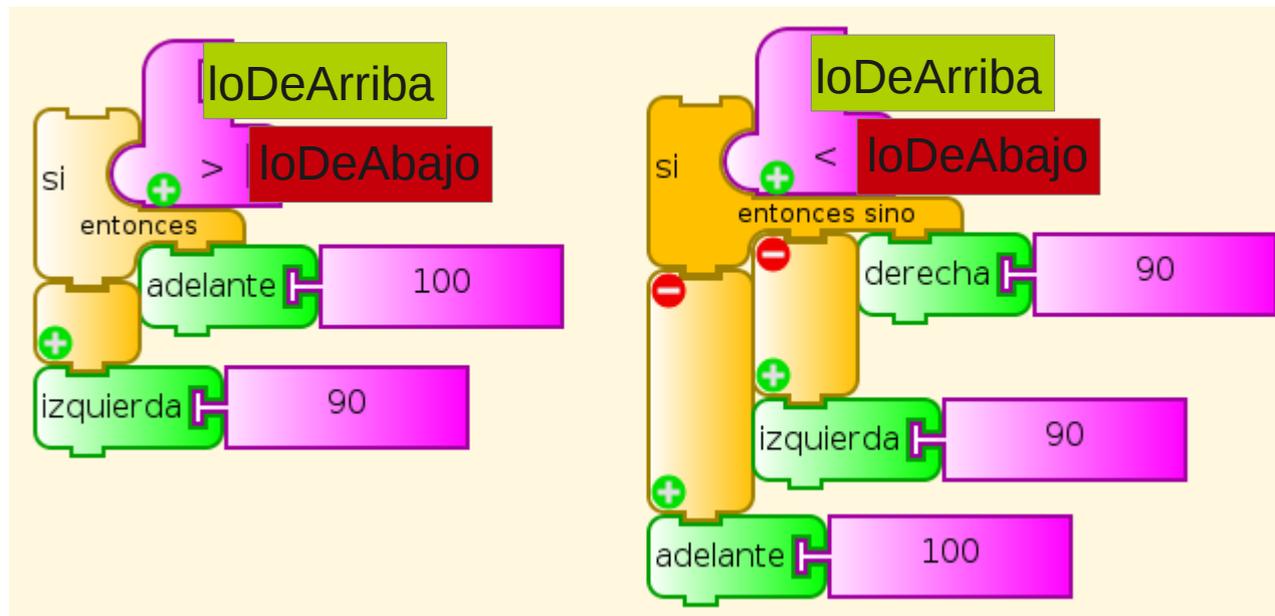
```
if cond1:  
    bloque1  
else:  
    bloque2  
bloque3
```

```
if cond1:  
    bloque1  
elif cond2:  
    bloque2  
elif cond3:  
    bloque3  
else:  
    bloque4  
bloque5
```

Programar en TortuBots

Estructuras de selección

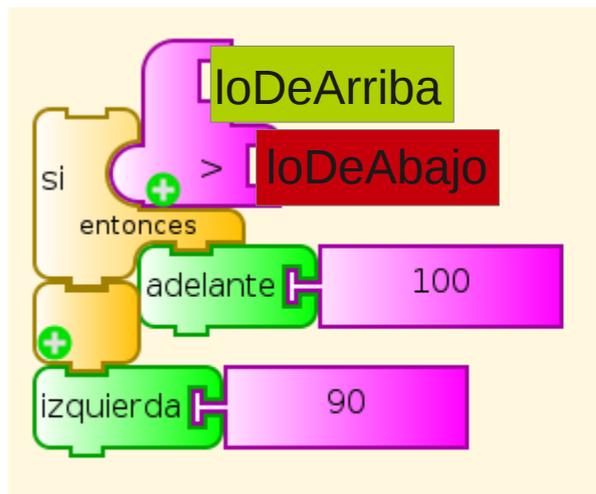
- En TortuBots (*la condición es de prueba y las instrucciones representativas*)





Estructuras de selección

- En TortuBots (*la condición es de prueba y las instrucciones representativas*)



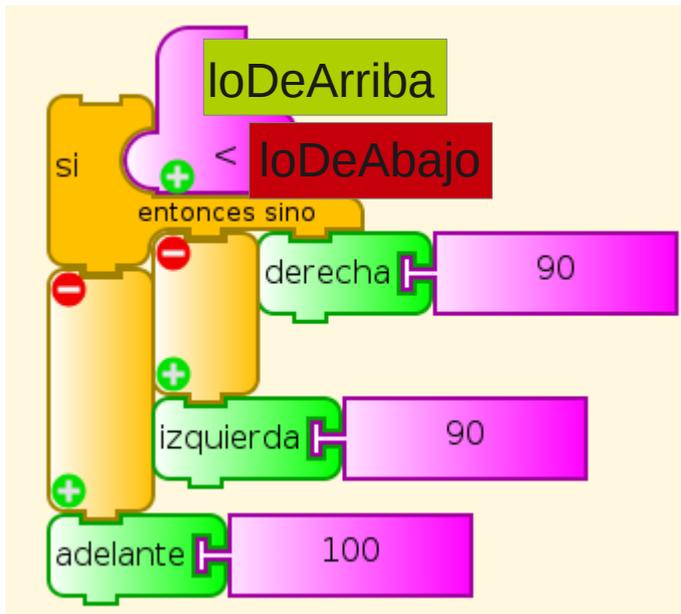
```
if cond1:
    bloque1
bloque2
```

```
if loDeArriba > loDeAbajo:
    adelante(100)
    izquierda(90)
```



Estructuras de selección

- En TortuBots (*la condición es de prueba y las instrucciones representativas*)



```
if cond1:  
    bloque1  
else:  
    bloque2  
bloque3
```

```
if loDeArriba < loDeAbajo:  
    izquierda(90)  
else:  
    derecha(90)  
adelante(100)
```

Programar en TortuBots

Estructuras de selección - Ejemplo



- Hacer avanzar la tortuga 100 hacia el este o el oeste según cual sea la orientación de la tortuga.
- Si apunta más hacia el este que hacia el oeste, ir hacia el este. Sino, al oeste.
- Notar que si está entre 0 y 180 apunta más hacia el este.
- El este coincide con la orientación 90 mientras que el oeste es 270

Estructuras de selección - Ejemplo

- El código Python podría ser:

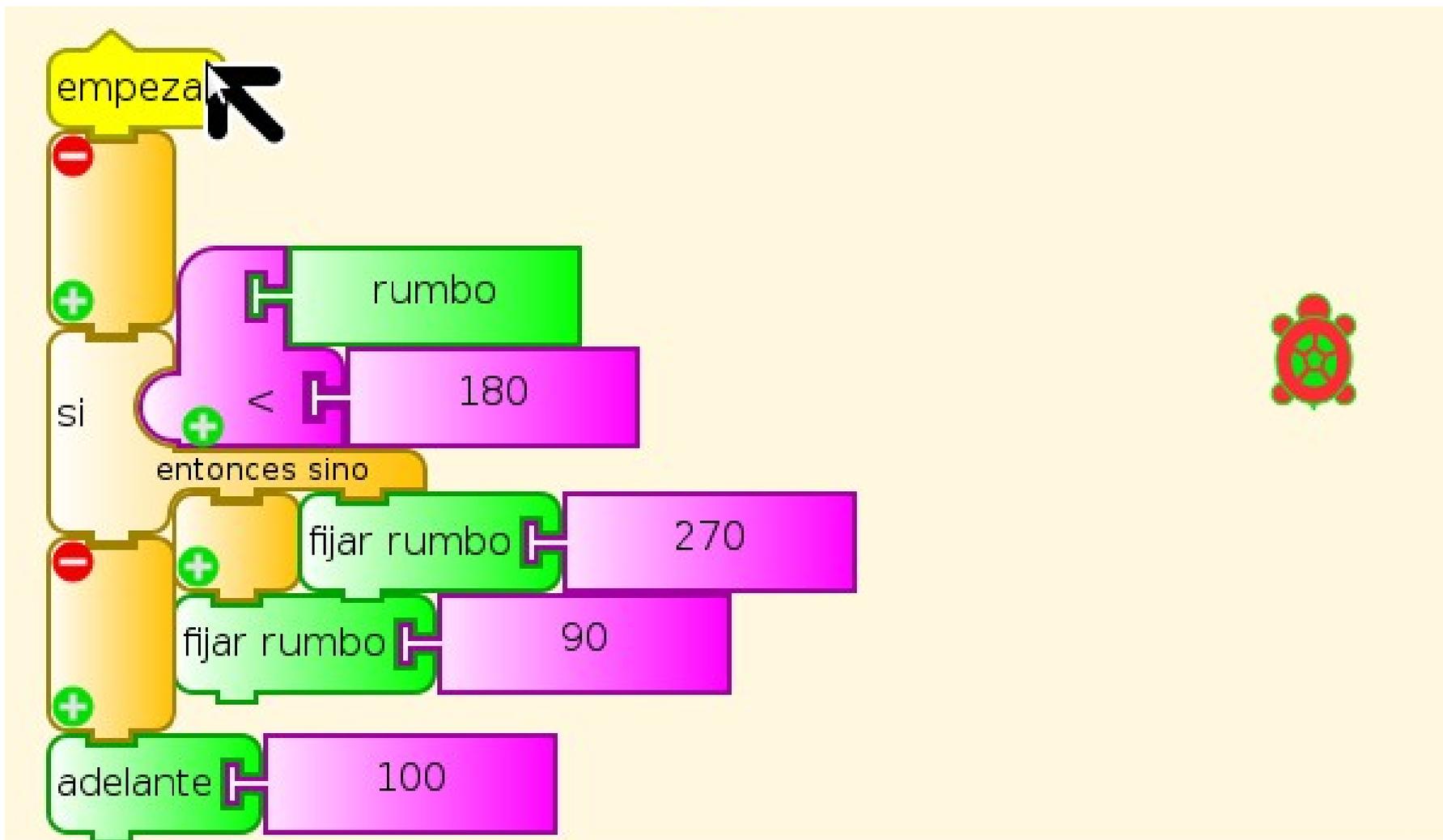
```
if rumbo < 180 :  
    fijar_rumbo(90)  
else:  
    fijar_rumbo(270)  
adelante(100)
```

Estructuras de selección - Ejemplo

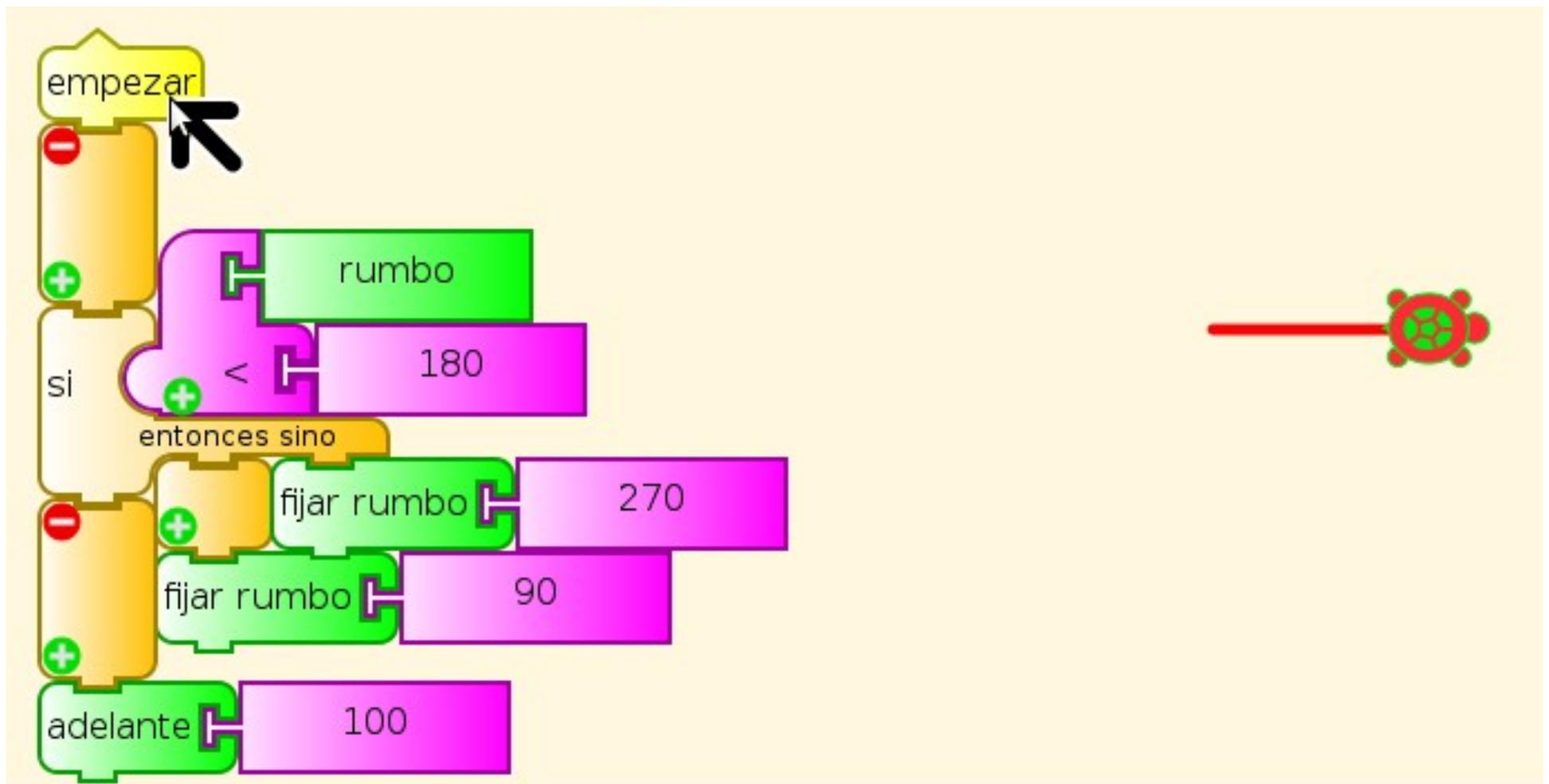


The image shows the TortuBots programming environment. At the top is a toolbar with various icons for navigation and editing. Below the toolbar is a palette of blocks categorized into movement (adelante, atrás, izquierda, derecha, limpiar), rotation (ángulo, arco, radio), coordinate setting (fijar xy, fijar rumbo), and coordinate variables (coorx, coory, rumbo). The main workspace contains a script starting with an 'empezar' block. This is followed by a 'si' (if) block with a condition 'rumbo < 180'. The 'entonces' (then) branch contains a 'fijar rumbo' block set to 270. The 'sino' (else) branch contains a 'fijar rumbo' block set to 90. Finally, the script ends with an 'adelante' block set to 100. A small turtle icon is visible in the workspace.

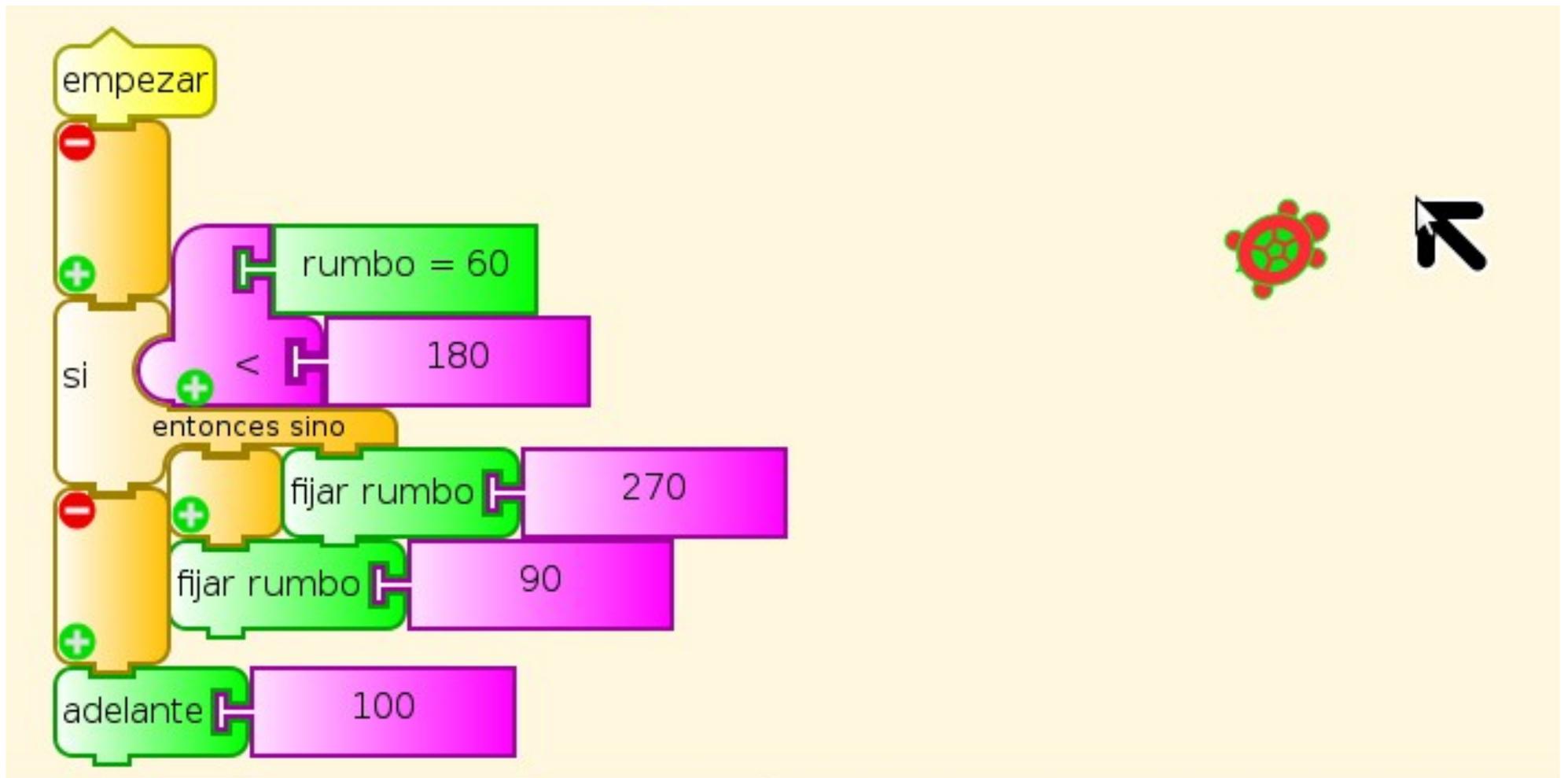
Estructuras de selección - Ejemplo



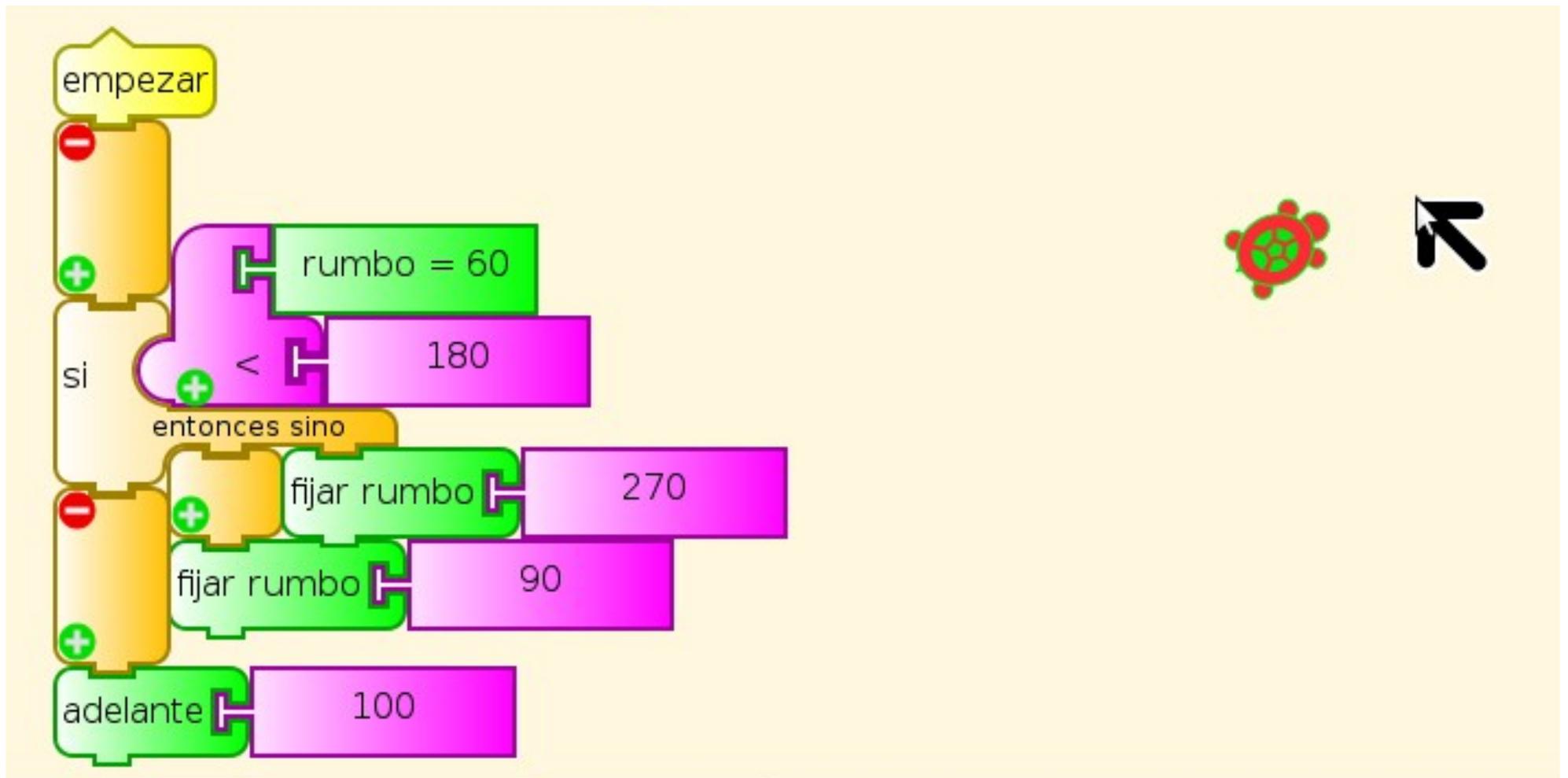
Estructuras de selección - Ejemplo



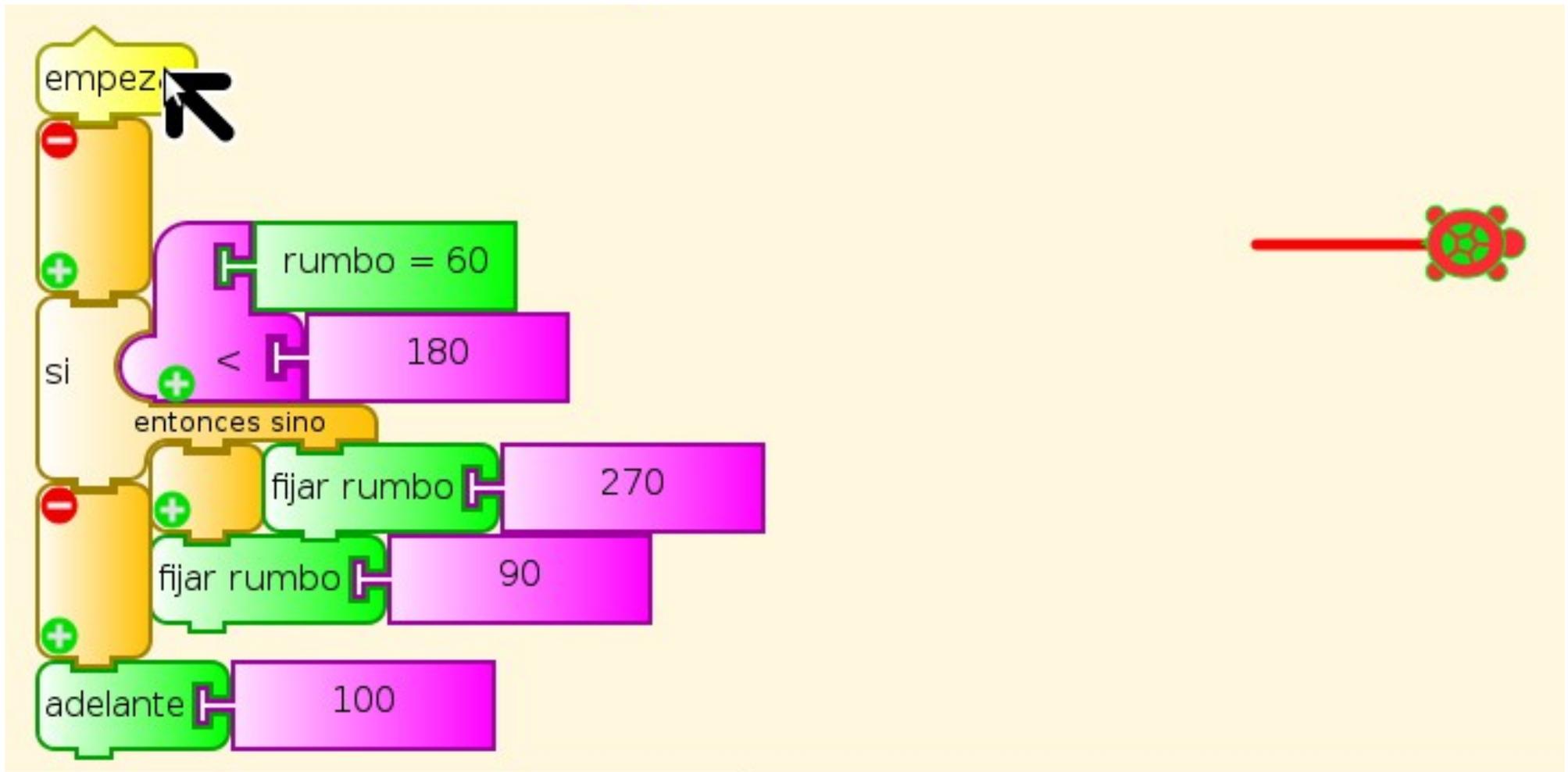
Estructuras de selección - Ejemplo



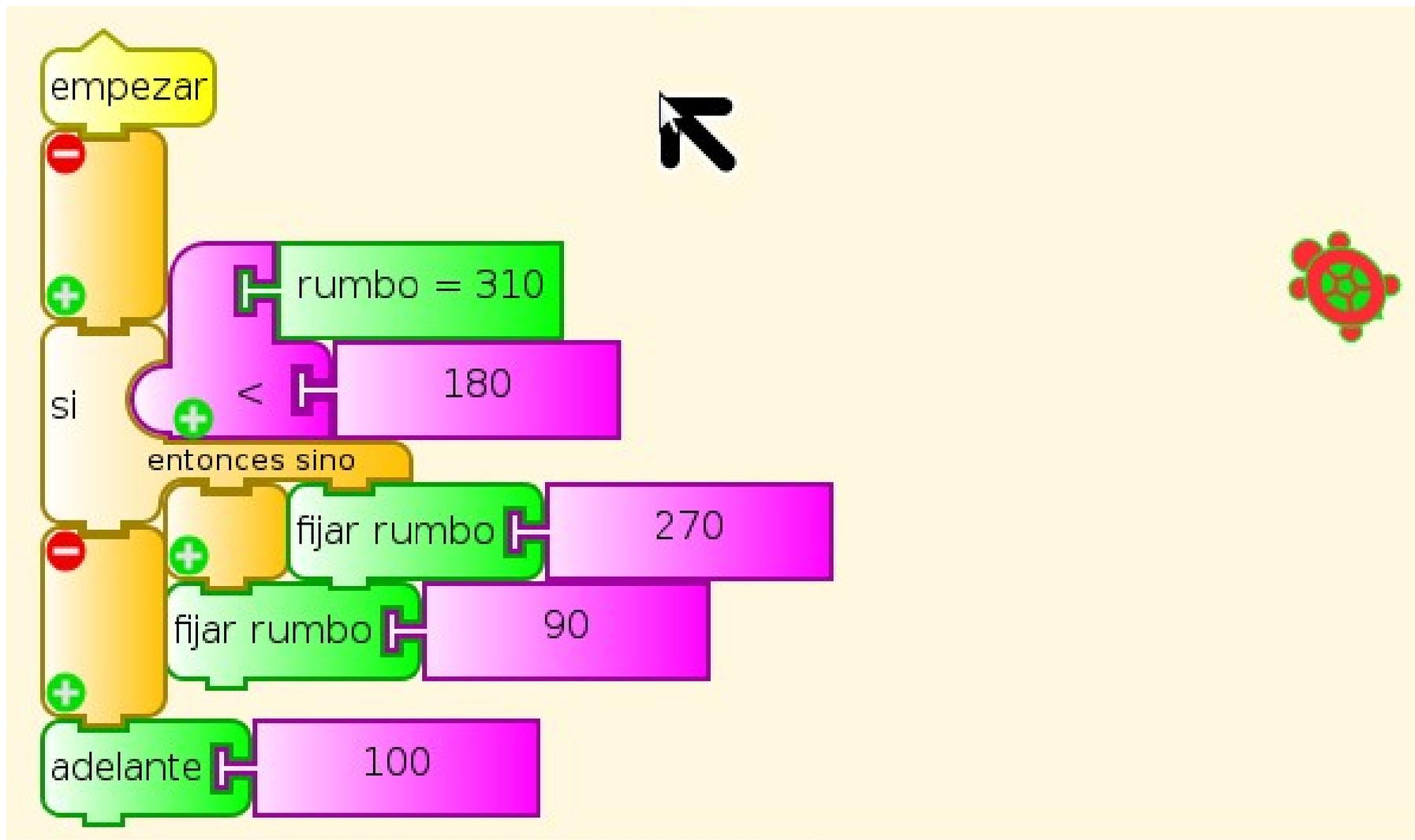
Estructuras de selección - Ejemplo



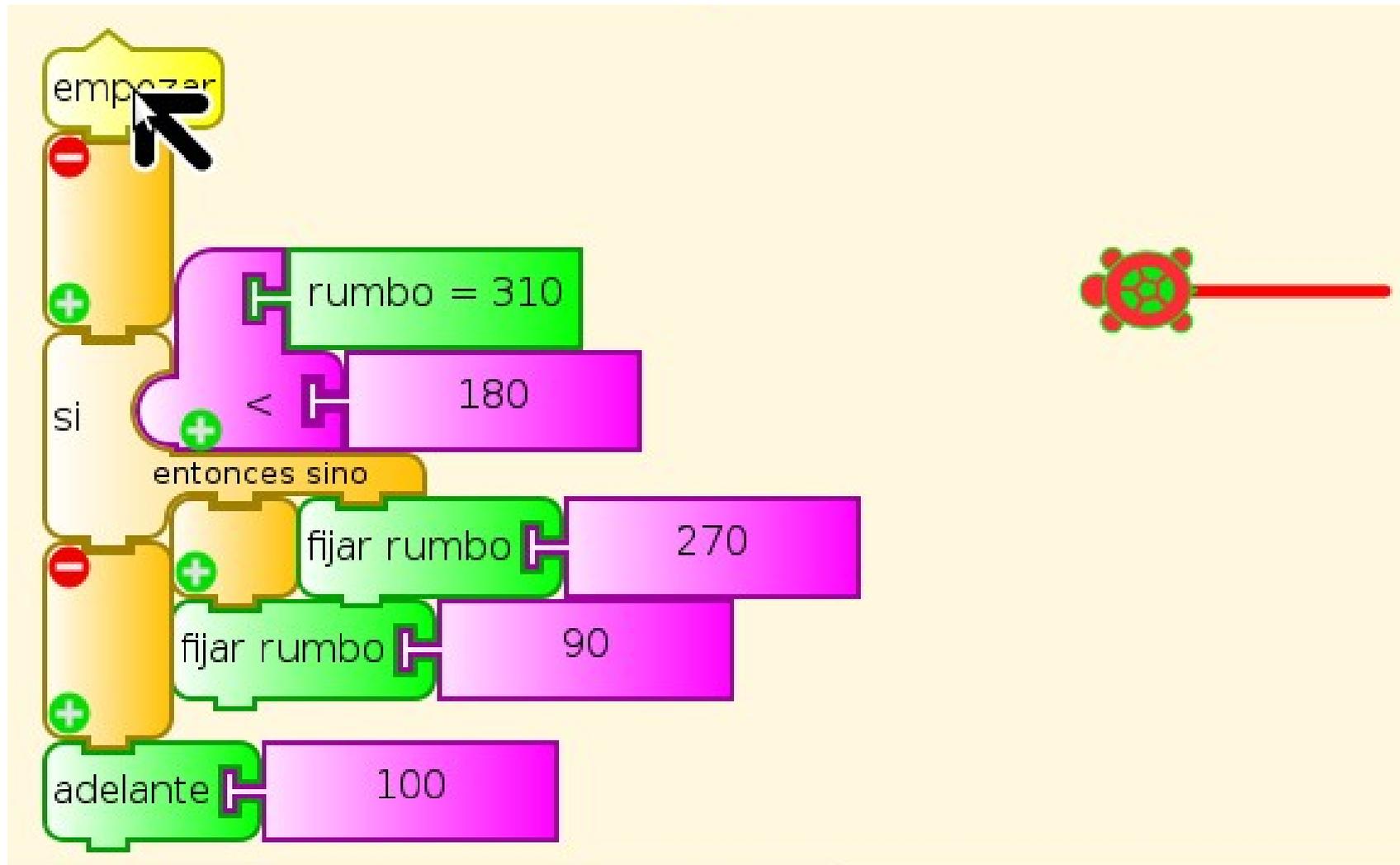
Estructuras de selección - Ejemplo



Estructuras de selección - Ejemplo

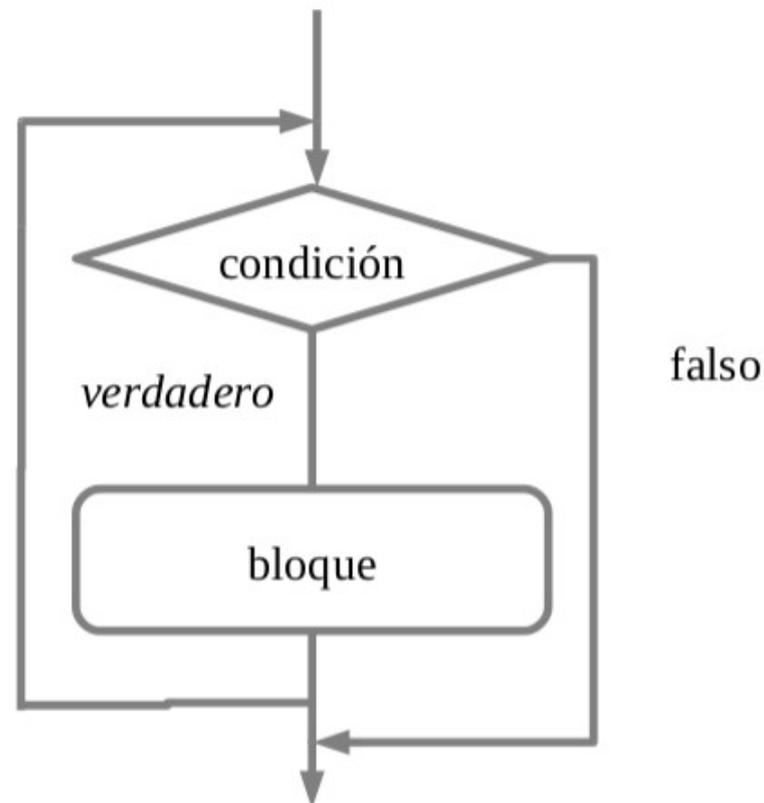


Estructuras de selección - Ejemplo



Estructuras de **iteración**

- Repiten un bloque de instrucciones un cierto número de veces.



Programar en TortuBots

Estructuras de iteración

- Las hay de dos tipos:
 - Repiten una cantidad fija de veces.
 - Repiten mientras se cumple alguna condición.

Estructuras de iteración

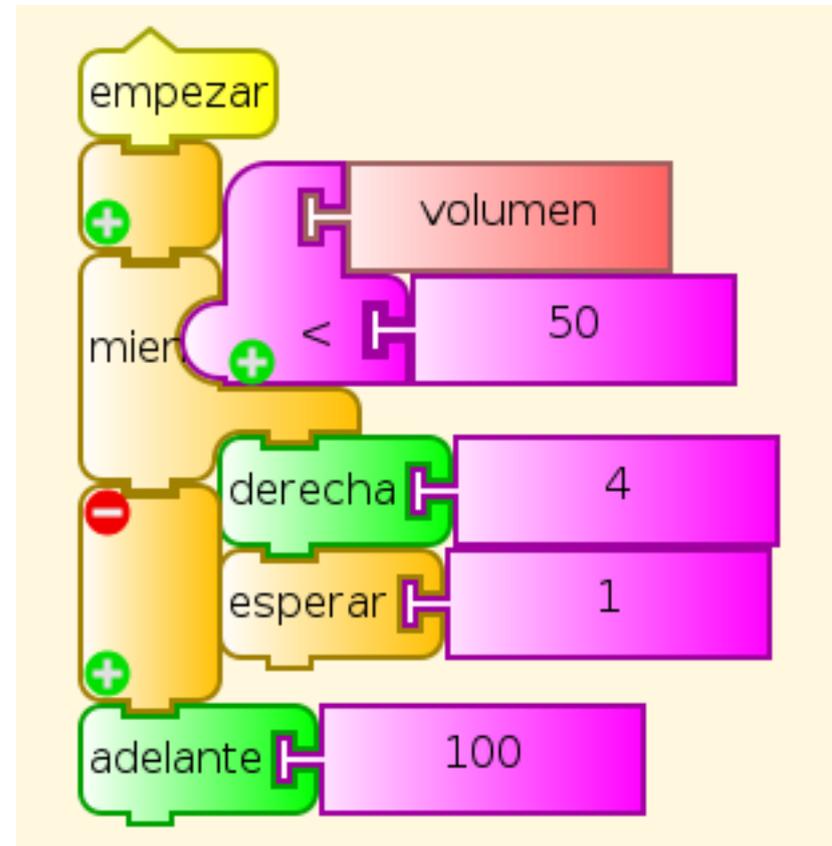
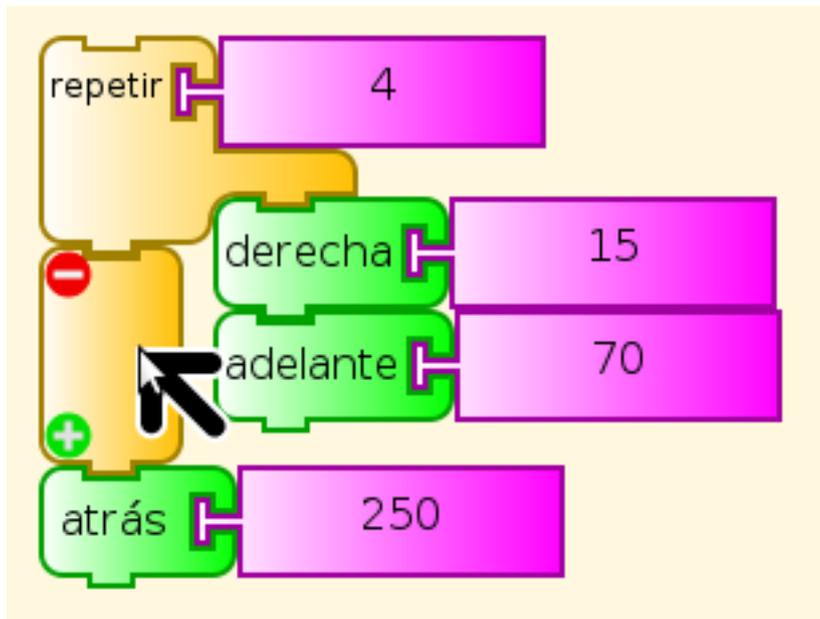
- En Python

```
for i in range(0, cant):  
    bloque1  
bloque2
```

```
while cond:  
    bloque1  
bloque2
```

Estructuras de iteración

- En TortuBots





Estructuras de iteración

- ¿Qué hacen los ejemplos dados?

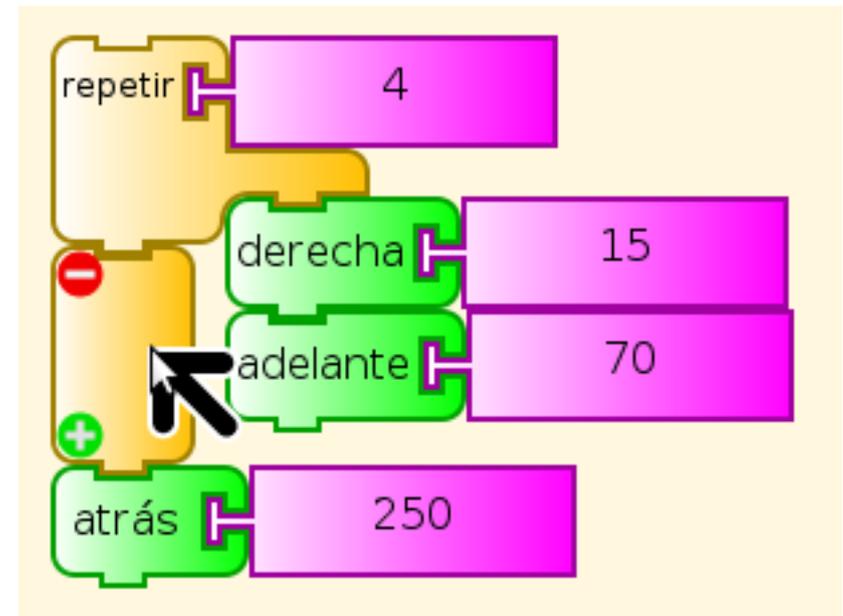
En python:

```
for i in range(0,4):
```

```
    derecha(15)
```

```
    adelante(70)
```

```
atrás(250)
```



Estructuras de iteración - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons for navigation and editing. Below the toolbar is a yellow palette containing several green blocks: 'adelante', 'izquierda', 'ángulo arco radio', 'fijar xy', 'coorx', 'coory', 'rumbo', 'atrás', 'derecha', and 'limpiar'. The main workspace is a light yellow area where a program is being built. The program starts with a 'repetir' block set to 4 iterations. Inside the loop, there are four blocks: 'derecha' (15 degrees), 'adelante' (70 units), and 'atrás' (250 units). A mouse cursor is pointing at the 'adelante' block. To the right of the workspace, there is a small green turtle icon.

Estructuras de iteración - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons including a turtle, a pencil, a color palette, a calculator (displaying 123), a refresh button, a cube, a folder, an eye, a gear, a monitor, a trash can, and a camera. Below the toolbar is a palette of blocks. The blocks are organized into several groups: movement (adelante, atrás, izquierda, derecha, limpiar), rotation (ángulo, arco, radio), coordinate setting (fijar xy, fijar rumbo), and coordinate variables (coorx, coory, rumbo). The main workspace is a light yellow area where a green turtle icon is positioned. A red line traces a path starting from the turtle, moving up and to the right, then curving down and to the left, and finally moving up and to the right again. A mouse cursor is pointing at the turtle. In the bottom-left corner of the workspace, there is a loop structure consisting of a 'repetir' block with the number '4' inside. The loop contains three blocks: 'derecha' with the value '15', 'adelante' with the value '70', and 'atrás' with the value '250'.

Quinto ejercicio: Utilizar el bloque repetir para dibujar el cuadrado del ejercicio 4.

Programar en TortuBots

Ejercicio – Solución posible Ej. 4

The screenshot displays the TortuBots programming environment. The top toolbar contains various icons for navigation, editing, and execution. The main workspace shows a script for a robot to follow a path and store items in boxes.

Script Structure:

- empezar** (Start)
- guardar en caja 1** (Store in box 1) - **caja 1** (box 1)
- guardar en caja 2** (Store in box 2) - **caja 2** (box 2)
- caja guardar en valor** (Store value in box)
- acción 1** (Action 1)
- acción 2** (Action 2)

Execution Log:

- empezar
- guardar en caja 2 (140)
- adelante (caja 2)
- derecha (90)
- adelante (caja 2)
- derecha (90)
- adelante (caja 2)
- derecha (90)
- adelante (caja 2)

Programar en TortuBots

Usamos el bloque repetir para dibujar el lado y girar, las 4 veces.

Programar en TortuBots

Ejercicio – Solución posible

The image shows the TortuBots programming environment. At the top is a toolbar with icons for: a turtle, a pencil, a color palette, a calculator (123 +/-), a refresh button, a 3D box icon, a folder, an eye, a gear, a screen, a trash can, and a camera. Below the toolbar is a block palette with the following blocks: 'empezar' (yellow), 'texto' (green), 'caja' (yellow), 'acción' (yellow), 'guardar en caja 1' (yellow), 'caja 1' (yellow), 'caja guardar en valor' (yellow), 'acción 1' (yellow), 'guardar en caja 2' (yellow), 'caja 2' (yellow), 'acción 2' (yellow). A mouse cursor is pointing at the 'acción' block in the palette. The main workspace contains a script with the following blocks: 'empezar' (yellow), 'guardar en caja 2' (yellow) with the value '140' (purple), a 'repetir' (orange) loop containing 'adelante' (green) with 'caja 2' (yellow) and 'derecha' (green) with '90' (purple). A green turtle icon is positioned to the right of the script.

Programar en TortuBots

Ejercicio – Solución posible

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons for navigation and editing. Below the toolbar is a script editor with a yellow background. The script starts with an 'empezar' block, followed by a 'texto' block containing 'caja'. This is followed by a 'guardar en caja 1' block with 'caja 1' and a 'guardar en caja 2' block with 'caja 2'. These are followed by a 'guardar en valor' block. The script then branches into two parallel paths: one with 'acción 1' and 'acción 1', and another with 'acción 2' and 'acción 2'. A mouse cursor is pointing at the top-right corner of the script editor. Below the script editor is a stage view with a light yellow background. The stage contains a green turtle icon. The script on the stage starts with 'empezar', followed by 'guardar en caja 2' with the value '140'. This is followed by a 'repetir' block with the value '4'. Inside the repeat loop, there are two blocks: 'adelante' with 'caja 2' and 'derecha' with '90'.

Programar en TortuBots

Ejercicio – Solución posible

The image shows the TortuBots programming environment. At the top is a toolbar with icons for a turtle, a pencil, a palette, a calculator (displaying 123), a refresh button, a 3D box, a folder, an eye, a gear, a monitor, a trash can, and a camera. Below the toolbar is a block palette with the following categories and blocks:

- Inicio:** empezar
- Texto:** texto
- Caja:** caja
- Acción:** acción
- Guardar:** guardar en caja 1, guardar en caja 2
- Guardar en:** guardar en (with sub-blocks for 'caja' and 'valor')
- Acción 1/2:** acción 1, acción 2

The main workspace contains a script with the following blocks:

- empezar
- guardar en caja 2 (with value 140)
- repetir (with value 4)
- adelante (with value 'caja 2')
- derecha (with value 90)

To the right of the script, a red square outline represents the robot's path, starting from a green turtle icon at the bottom-left corner of the square.

Programar en TortuBots

¿Y si cambiamos el valor de la variable y volvemos a ejecutar?

Programar en TortuBots

Ejercicio – Solución posible – Ahora con 160



The screenshot displays the TortuBots programming environment. At the top, a toolbar contains various icons for navigation and editing. A notification bar on the right states "No se puede conectar con el servidor." (Cannot connect to the server).

The main workspace is divided into two sections. The top section is a yellow palette containing various code blocks: "empezar" (start), "texto" (text), "caja" (box), "acción" (action), "guardar en caja 1" (save to box 1), "caja 1" (box 1), "caja guardar en valor" (box save value), "acción 1" (action 1), "guardar en caja 2" (save to box 2), "caja 2" (box 2), "acción 2" (action 2), "acción 2" (action 2).

The bottom section shows a script on the left and a robot on a grid on the right. The script consists of the following blocks:

- "empezar" (start)
- "guardar en caja 2" (save to box 2)
- "repetir" (repeat) block with a count of 4, containing:
 - "adelante" (forward) block with "caja 2" (box 2) selected
 - "derecha" (right) block with "90" (90 degrees) selected

The robot is positioned at the bottom-left corner of a red square grid. The grid is empty, suggesting the robot's path is yet to be fully visualized or the environment is a simple starting point.

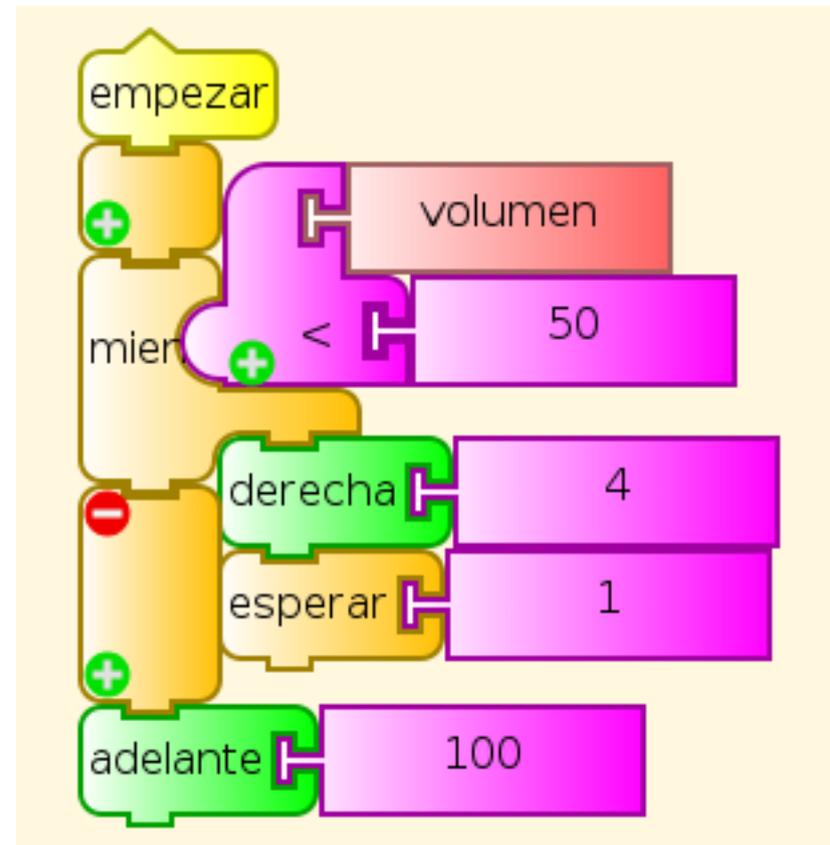
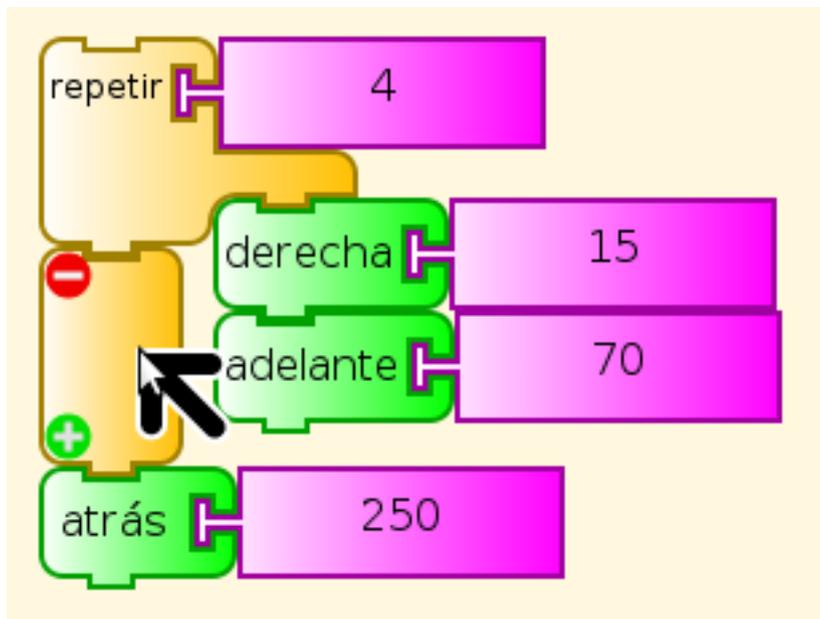
Programar en TortuBots

Ejercicio – Solución posible – Ahora con 160

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons for navigation and editing. Below the toolbar, a yellow palette contains several code blocks: 'empezar', 'texto', 'caja', 'acción', 'guardar en caja 1', 'caja 1', 'caja 2', 'guardar en caja 2', 'caja guardar en valor', 'acción 1', and 'acción 2'. The main workspace is light yellow and contains a sequence of code blocks: 'empezar', 'guardar en caja 2' with the value '160', a 'repetir' block with the value '4', and two movement blocks: 'adelante' with 'caja 2' and 'derecha' with '90'. A black mouse cursor is pointing at the 'empezar' block. To the right of the code, a green turtle icon is positioned at the bottom-left corner of a red square path. The path consists of two concentric squares, with the inner square being smaller than the outer one, representing a square path.

Estructuras de iteración

- En TortuBots



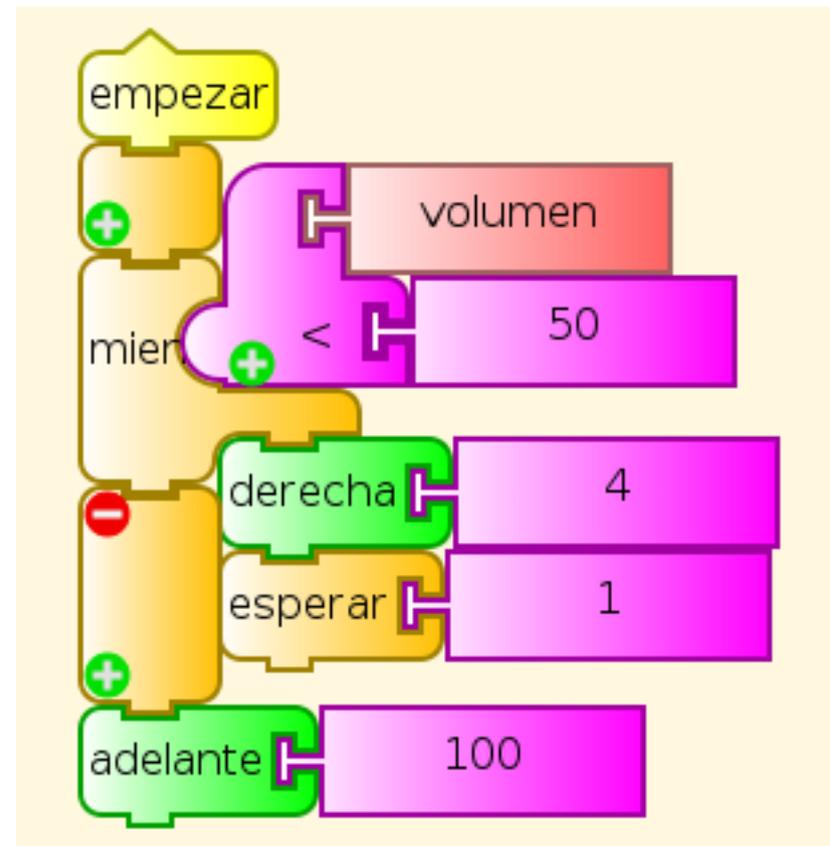
Programar en TortuBots

Estructuras de iteración

- ¿Qué hacen los ejemplos dados?

En python:

```
while volumen < 50:  
    derecha(4)  
    esperar(1)  
adelante(100)
```



Estructuras de iteración - Ejemplo

The image shows the TortuBots programming environment. At the top, there is a toolbar with various icons for editing and execution. Below the toolbar is a palette of programming blocks. The main workspace contains a script starting with an 'empezar' block, followed by a 'mientras' loop. The loop's condition is 'volumen < 50'. Inside the loop, there are three blocks: 'derecha 4', 'esperar 1', and 'adelante 100'. A small green turtle icon is visible on the right side of the workspace.

Toolbar icons: Turtle, Eraser, Color palette, Variable (123 +-=), Loop, Stack, List, Eye, Gear, Screen, Stop, Home, Circle, Dots, Trash, Arrow.

Block palette: esperar, repetir, si entonces, si entonces sino, parar acción, mientras, hasta, por siempre.

Script structure:

```
empezar
+
mientras (volumen < 50)
-
  derecha 4
+
  esperar 1
+
  adelante 100
```

Estructuras de iteración - Ejemplo

The image shows the TortuBots programming environment. At the top, a toolbar contains various icons for navigation and editing. Below the toolbar, a yellow palette displays several control blocks: 'esperar' (wait), 'por siempre' (forever loop), 'repetir' (repeat), 'si' (if), 'entonces' (then), 'si entonces sino' (if-then-else), 'parar acción' (stop action), 'mientras' (while loop), and 'hasta' (until loop). The main workspace features a black arrow cursor pointing to a script. The script begins with an 'empezar' (start) block, followed by a 'mientras' (while) loop block with a '+' sign. Inside the loop, there are four blocks: a 'volumen' (volume) block, a '<' (less than) comparison block with the value '50', a 'derecha' (right) block with the value '4', and an 'esperar' (wait) block with the value '1'. Below the loop is an 'adelante' (forward) block with the value '100'. A red line with a green turtle icon is positioned to the right of the script.

Programar en TortuBots

Sexto ejercicio: Leer el valor del volumen.
Mientras sea menor a 70, dibujar cuadrados cuyo
lado sea el valor leído.

Programar en TortuBots

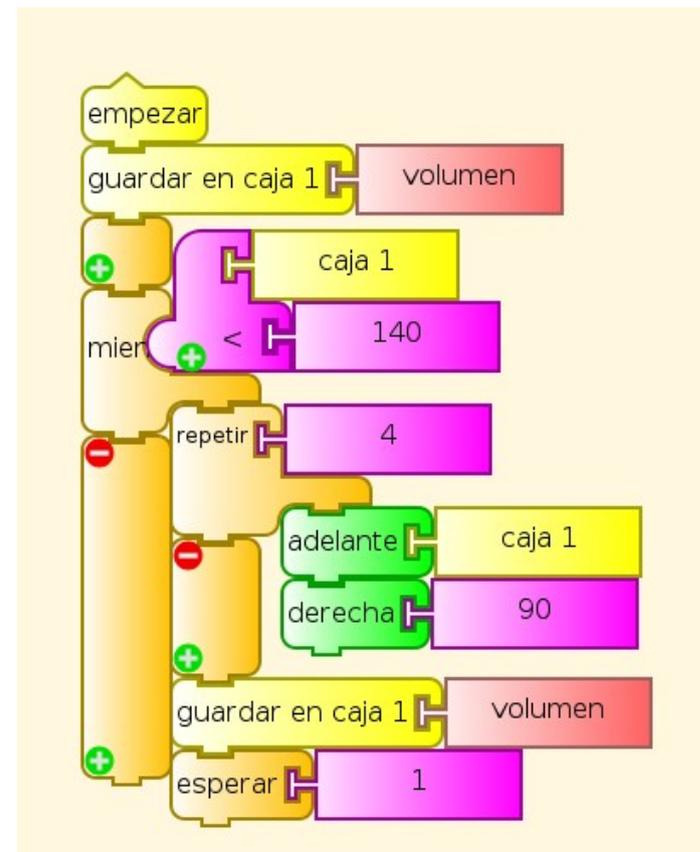
Sexto ejercicio – Ideas:

- Leer el sensor y guardarlo en una variable.
- Dibujar el cuadrado con la estructura repetir.
- Leer dentro de la estructura mientras el sensor

Programar en TortuBots

Una solución posible:

```
caja1 = volumen
while (caja1 < 140) :
    for i in range(0,4) :
        adelante(caja1)
        derecha(90)
    caja1 = volumen
```



Una solución posible:



The image shows a screenshot of the TortuBots programming environment. The interface features a toolbar at the top with various icons for editing and execution. The main workspace contains a sequence of programming blocks:

- empezar** (Start)
- guardar en caja 1** (Save to box 1) with the value **volumen**.
- mientras** (While) loop containing:
 - caja 1** (Box 1)
 - <** (Less than) condition with the value **140**.
 - repetir** (Repeat) loop with the value **4**, containing:
 - adelante** (Forward) with the value **caja 1**.
 - derecha** (Right) with the value **90**.
 - guardar en caja 1** (Save to box 1) with the value **volumen**.
 - esperar** (Wait) with the value **1**.

A small turtle icon is visible on the right side of the workspace.

Una solución posible:

The image shows a screenshot of the TortuBots programming environment. At the top, there is a toolbar with various icons for editing and execution. The main workspace contains a script with the following blocks:

- empezar** (Start)
- guardar en caja 1** (Save to box 1) with the value **volumen** (volume).
- +** (Add) block with **caja 1** (box 1).
- mientras** (While) loop with a **<** (less than) condition and the value **140**.
- repetir** (Repeat) block with the value **4**.
- adelante** (Forward) block with **caja 1** (box 1).
- derecha** (Right) block with the value **90**.
- guardar en caja 1** (Save to box 1) with the value **volumen** (volume).

To the right of the script, there is a red square path diagram with a small green turtle icon at the start, illustrating the path the robot would take. A mouse cursor is pointing at the diagram.

Programar en TortuBots

Para ver estructuras condicionales más avanzadas, debemos aprender un poco más sobre las **expresiones** en TortuBots.

Programar en TortuBots

Bloques de operaciones.

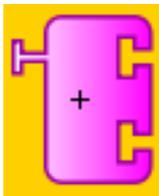
Programar en TortuBots

- Las operaciones en TortuBots son de color violeta.
- Se encuentran en la paleta de operaciones.
- Existen operaciones aritméticas y lógicas.

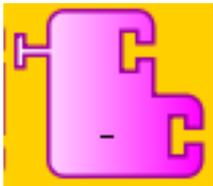


Operaciones aritméticas

- Operan entre dos números y devuelven un número.



Suma los dos parámetros y devuelve el resultado.



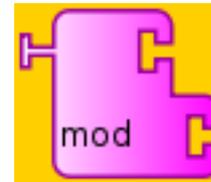
Resta el parámetro de abajo al parámetro de arriba y devuelve el resultado.



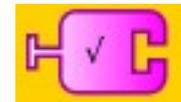
Multiplica los dos parámetros y devuelve el resultado.



Devuelve el resultado de la división del parámetro de arriba sobre el parámetro de abajo.



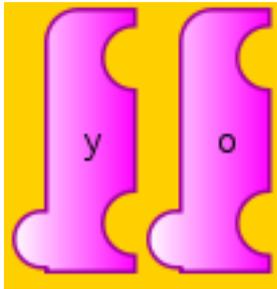
Devuelve el resto de la división del parámetro de arriba sobre el parámetro de abajo.



Devuelve la raíz cuadrada del parámetro recibido (recibe un único parámetro que debe ser positivo).

Operaciones lógicas

- Operan entre dos Booleanos y devuelven un Booleano.



Realizan el AND o el OR entre los dos parámetros recibidos y devuelve el resultado.



Realiza el NOT del parámetro recibido y devuelve el resultado. Esto es, devuelve el opuesto del parámetro recibido.

Programar en TortuBots

Evaluaciones

- Operan entre dos **números** y devuelven un **Booleano**.

Los bloques de evaluación comparan dos valores numéricos y devuelven el resultado.



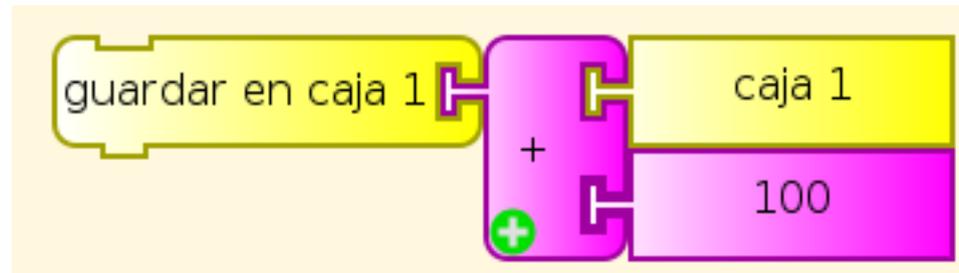
Para mayor/menor: Compara si el parámetro de arriba es mayor/menor que el parámetro de abajo y devuelve el resultado.

Para igual: Compara si los dos parámetros recibidos son iguales.



Ejemplos

- Utilizamos las operaciones en diferentes ejemplos.

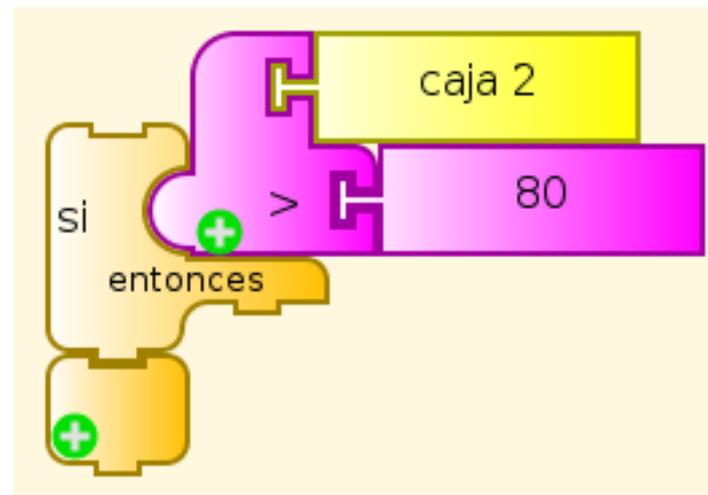


Obtiene el valor de la caja 1 (una variable), le suma 100 y el resultado se vuelve a guardar en la caja 1, sobrescribiendo el valor que estaba antes.

Programar en TortuBots

Ejemplos

- Utilizamos las operaciones en diferentes ejemplos.



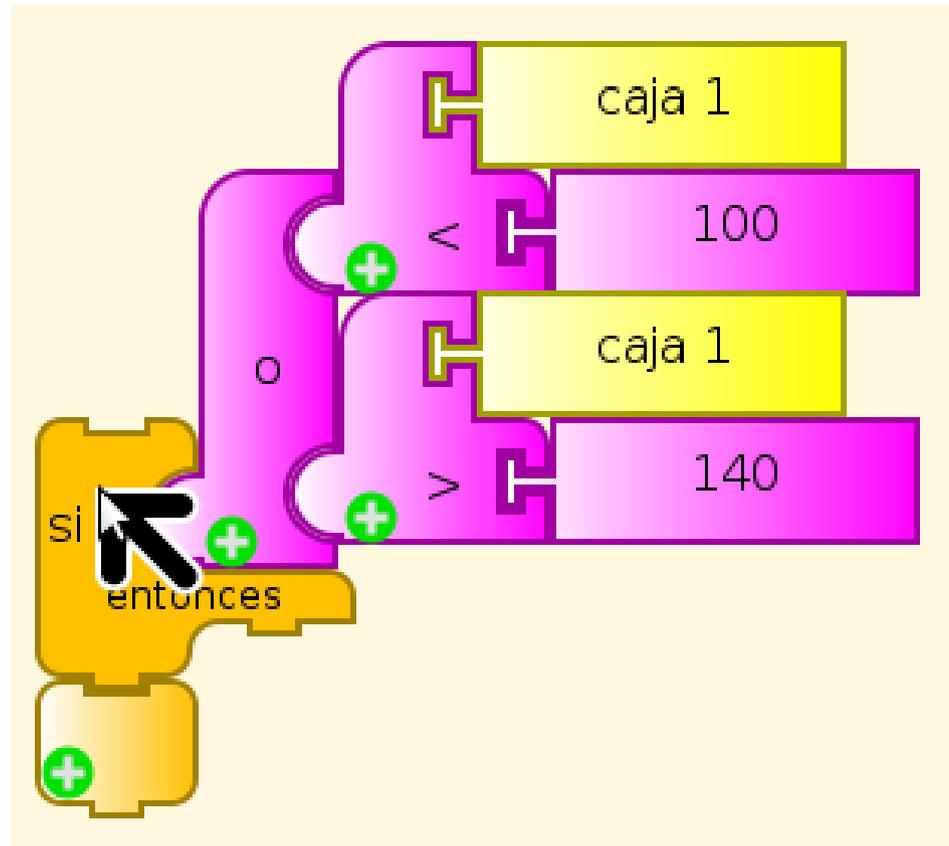
Se evalúa si el valor contenido en la caja 2 es mayor a 80.

Si es mayor, se ejecuta el conjunto de bloques que se ubique en el conector “entonces”.

Programar en TortuBots

Ejemplos

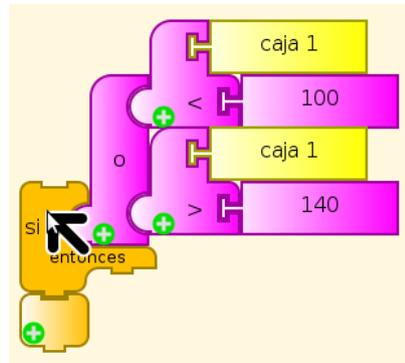
- Utilizamos las operaciones en diferentes ejemplos.



Programar en TortuBots

Ejemplos

- Utilizamos las operaciones en diferentes ejemplos.



Este ejemplo muestra dos evaluaciones y un operador lógico. Se evalúa si caja 1 es menor a 100 o si caja 1 es mayor a 140. Como la condición es o si alguna de las dos comparaciones es verdadera, se ejecuta el conjunto de bloques que se conecta en “entonces”. Esto sucede si caja 1 es menor a 100 o mayor a 140, o sea, si caja 1 NO está entre 100 y 140.

Programar en TortuBots

Con todo esto es posible realizar varios ejercicios.

¿Cómo podríamos programar algo que reaccione frente a una acción del usuario como apretar el botón del mouse?

Programar en TortuBots

Para resolver dichos ejercicios, es necesario saber cómo realizar esperas por una medida que tome cierto valor o un **evento**.

Programar en TortuBots

Esperar valor:

- Leer el sensor y guardarlo en una variable.
- Revisar el valor y si no está en el rango deseado, esperar un tiempo a ver si cambia.
- Cuando cambia, continuar ejecutando un programa.

Programar en TortuBots

Esperar valor:

- A veces, los cambios de valores pueden entenderse como un *evento*. Ej: Cuando el usuario aprieta el botón del mouse (evento) este cambia de valor.
- Usualmente evaluaremos si el valor está en un cierto rango, **no** si toma un cierto valor.

Séptimo ejercicio: Dibujar un cuadrado de lado 75 cuando el usuario haga clic con el mouse.

Programar en TortuBots

Séptimo ejercicio – Ideas:

- Tenemos que conocer qué valor toma el bloque “botón presionado” cuando hace clic y cuando no.
- Mientras el valor sea el de “no clic”, esperamos una fracción de tiempo.
- El código del cuadrado es conocido.

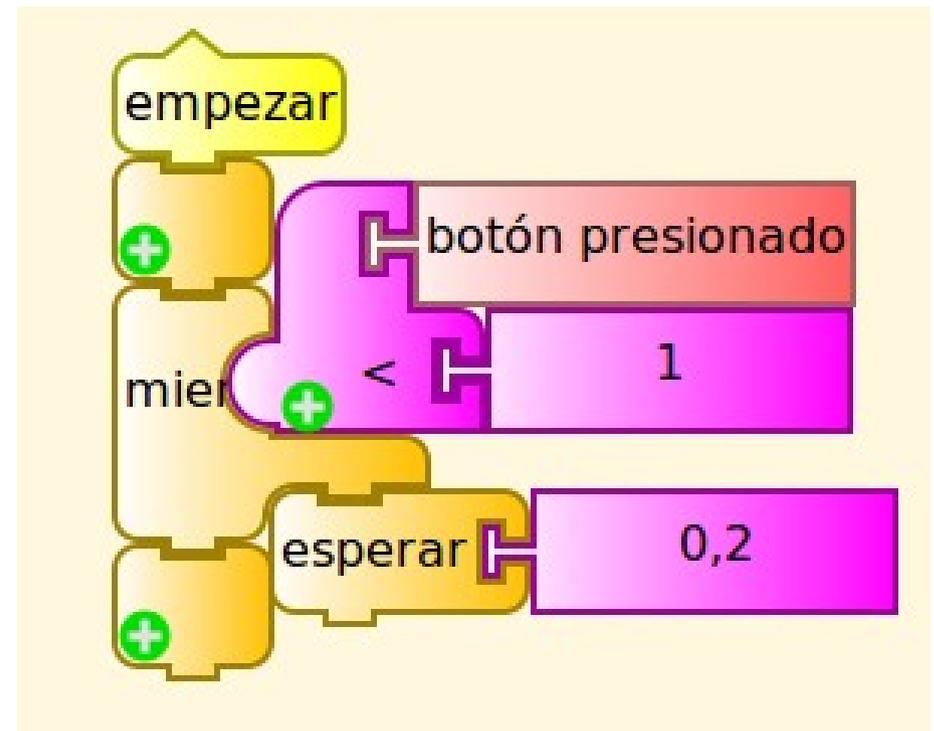
Programar en TortuBots

Séptimo ejercicio – Posible solución:



En Python:

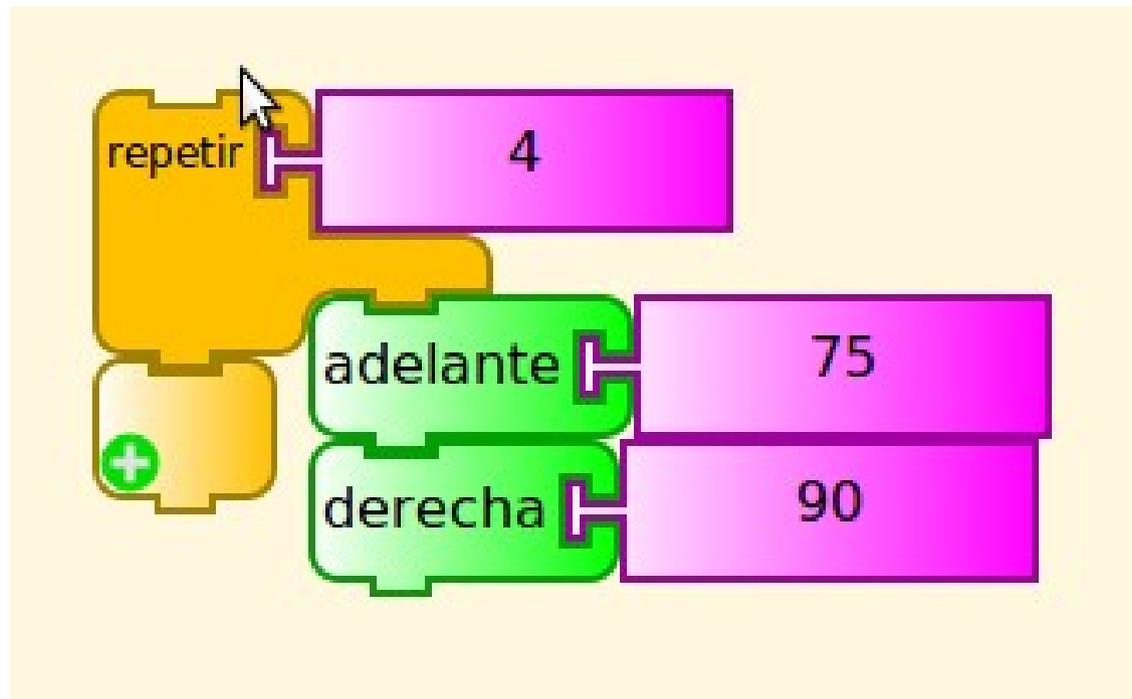
```
while valorBoton < 1 :  
    esperar(0.2)
```



Programar en TortuBots

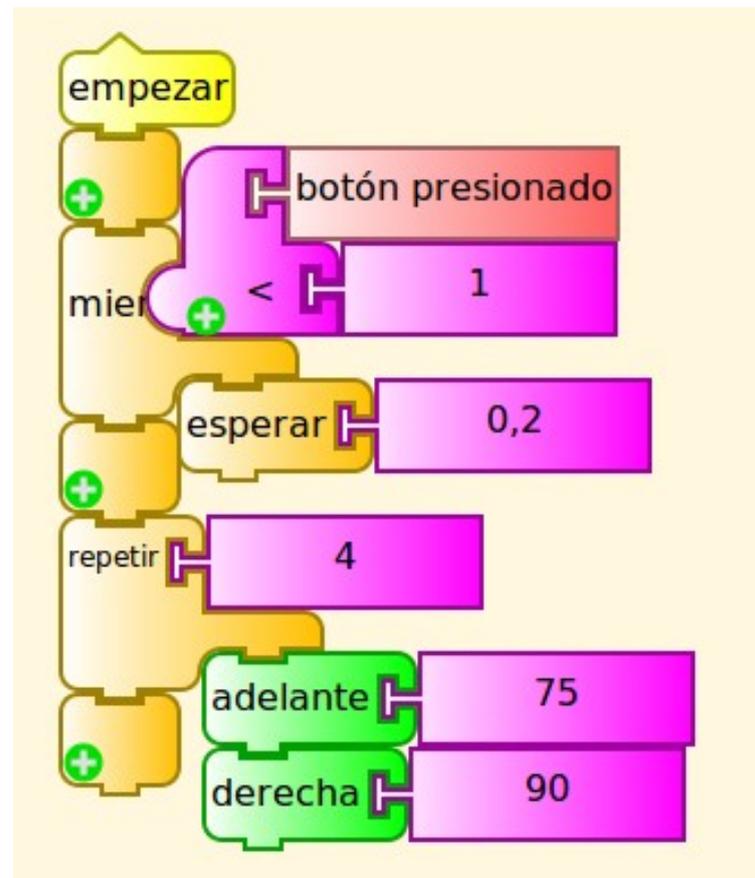
Séptimo ejercicio – Posible solución:

- Ya hemos visto cómo construir el cuadrado



Programar en TortuBots

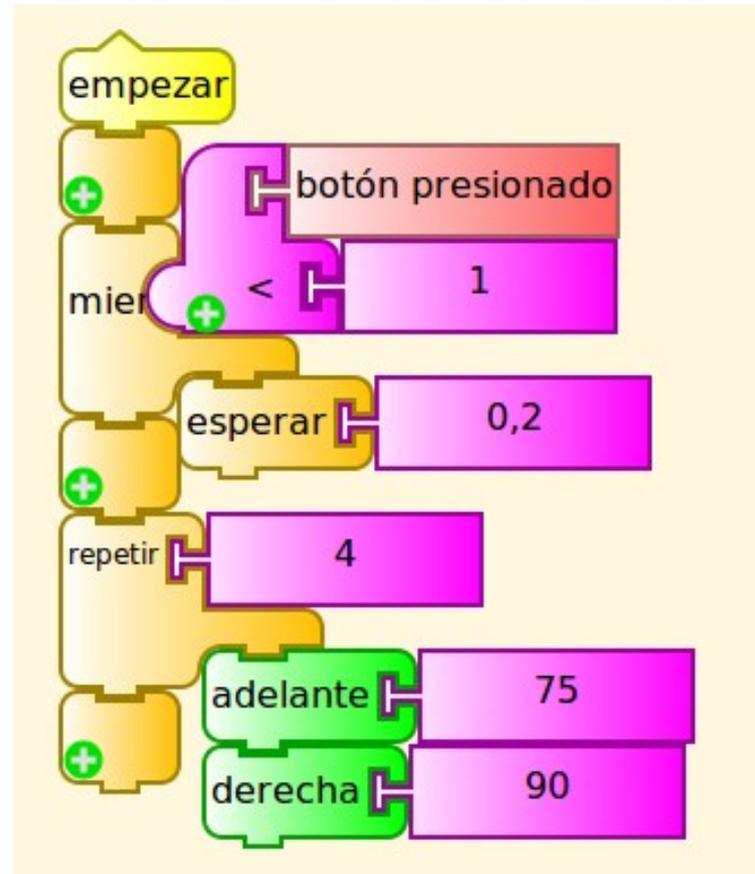
Séptimo ejercicio – Posible solución:



Programar en TortuBots

Séptimo ejercicio – Posible solución:

Probarla! 😊



Programar en TortuBots

Octavo ejercicio: Hacer un programa que diga “Usted ha hecho clic” cada vez que el usuario haga clic.

Programar en TortuBots

Octavo ejercicio – Ideas:

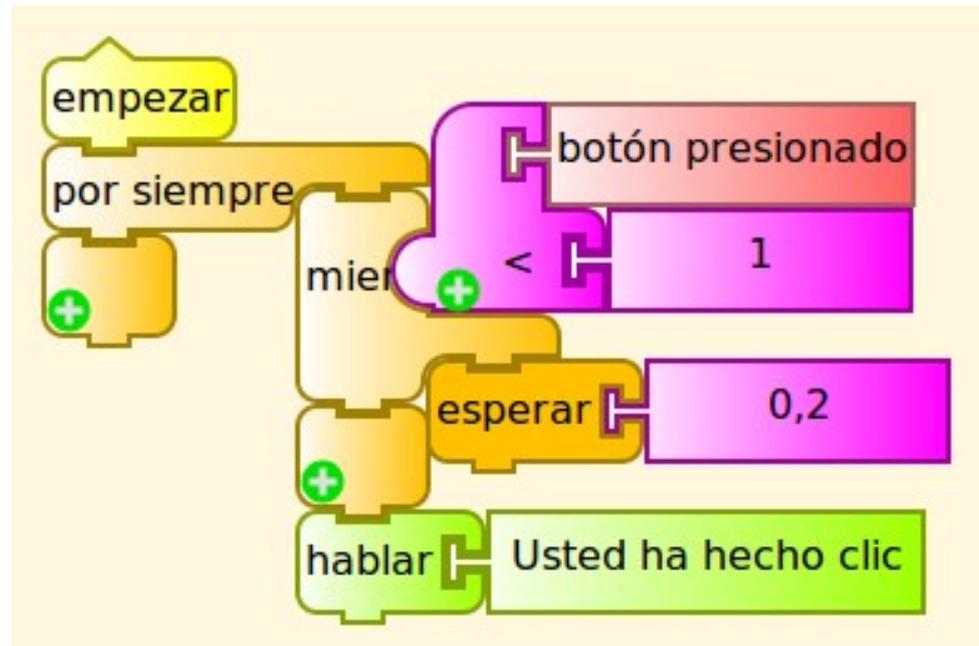
- Ya sabemos leer el botón del mouse y esperar por el clic.
- Este programa se ejecutará “para siempre”.
- Utilizar el bloque *hablar* en la paleta de medios.

Programar en TortuBots

Octavo ejercicio – Posible solución:



Probarla! 😊



Programar en TortuBots

Objetivos

- Repasar conceptos de programación.
- Continuar familiarizándose con TortuBots.
- Aprender a programar acciones de la tortuga.
- Aprender a programar estructuras de control en TortuBots.
- Resolver problemas sencillos con la tortuga.

Fin del Capítulo 2 – Programar en TortuBots