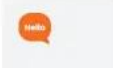


matatalab

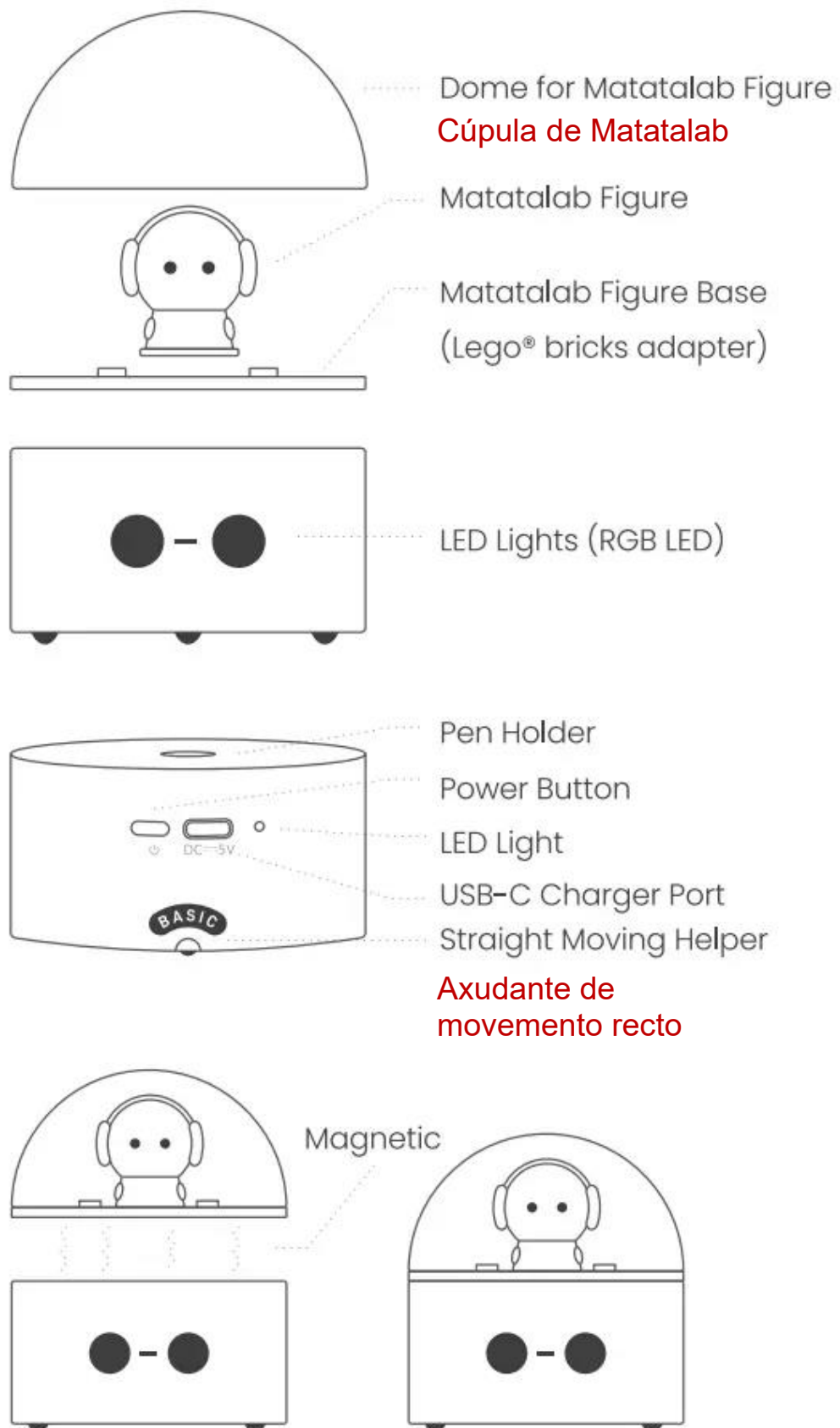
Así de fácil. Guía de uso

Benvi@d@s a este Así de fácil ! Aquí aprenderemos a traballar e programar con Matatalab. Prepárate para explorar o mundo da programación de maneira divertida e sinxela!

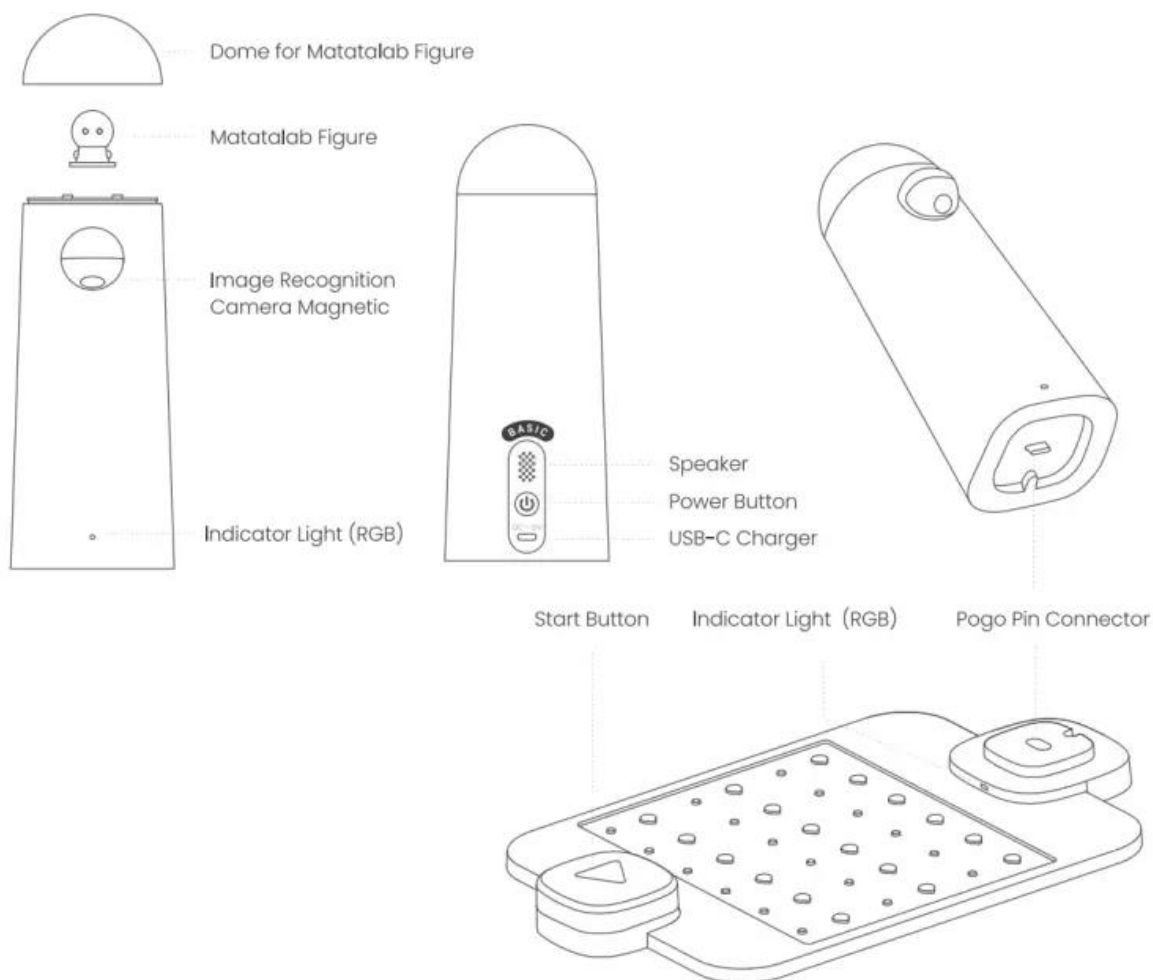
Elementos da caixa

MatataBot BASIC x1	Control Board x1		
			Loop Ends x2
Command Tower BASIC x1			Preset Movement x1
			Preset Music x1
			Preset Dancing x1
			Alto Clef Music Block x7
			Treble Clef Music Block x1
	Move Forward x4		Numbers x2
	Move Backward x4		Angles x1
	Left Turn 90° x4		Flags x3
	Right Turn 90° x4		Obstacles x8
	Loop Begins x2		USB-C Cable x1
	Accessory Pack x1		

MatataBot



Torre de mando e taboeiro de control



Prender e apagar

Torre de mando

Prendido: Preme o botón de prendido durante 1 segundo ata que vexas unha luz azul e se reproduza a música de inicio.

Apagado: Preme o botón de prendido durante 1 segundo ata que se reproduza o son de apagado.

MatataBot

O proceso é o mesmo que seguimos tanto para o prendido coma o apagado da **torre de mando**.

Emparellamento

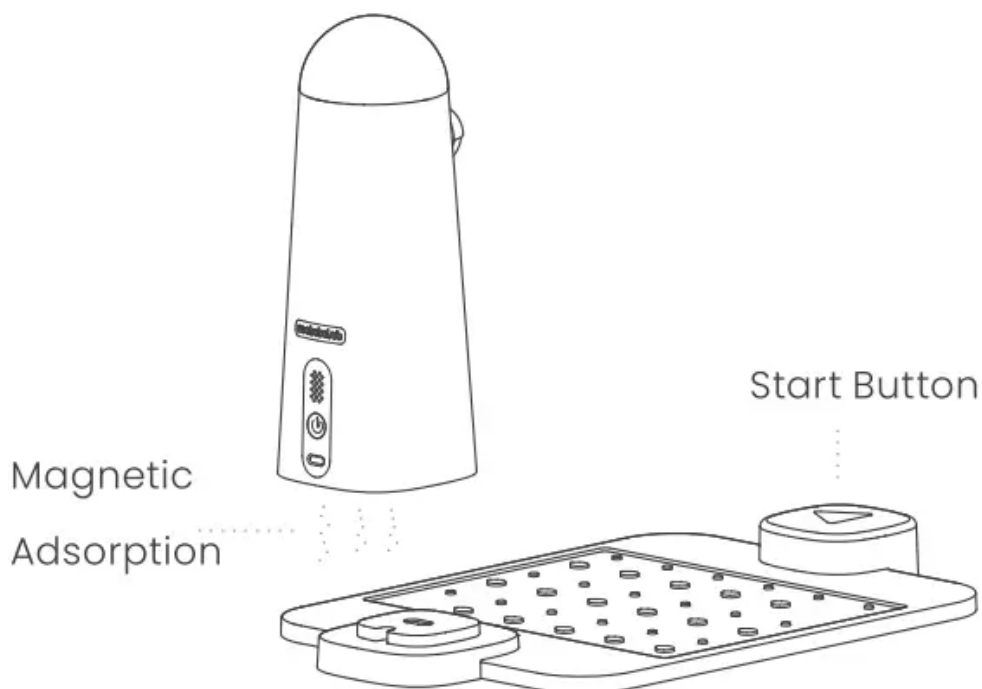


Ao prender tanto a **torre de mando** como **MatataBot**, verás que as luces de ámbolos dous dispositivos comezan a pestanexar, indicando que están intentando conectarse. Cando o emparellamento sexa exitoso, escoitarás un sonido de 'Ding Ding' e as luces azuis quedarán fixas.

Se os leds azuis non se quedan fixos, significa que non se completou o emparellamento. Neste caso, o mellor é apagar ámbolos dous dispositivos e volver intentalo.

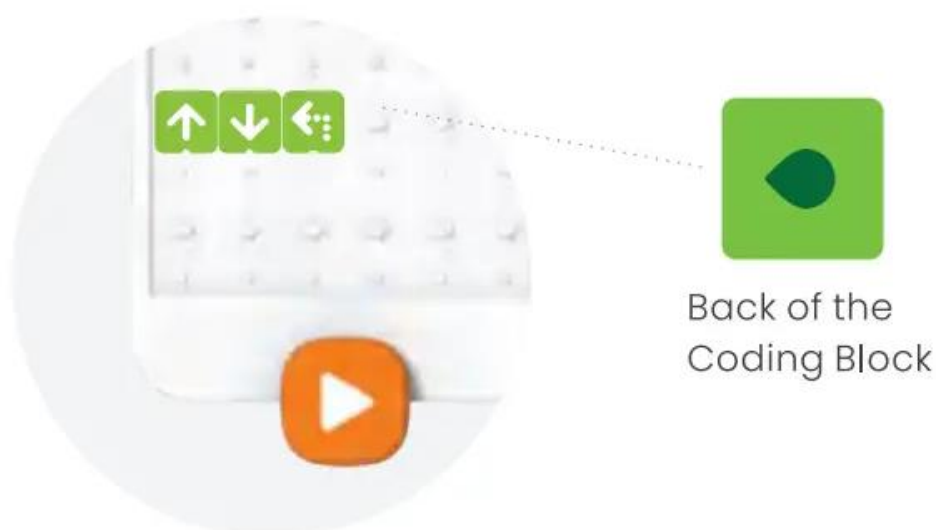
Instrucións de programación

Colocaremos o **taboeiro de control** nunha superficie horizontal e conectarémoslle a **torre de mando** como se amosa na seguinte imaxe.



Para programar iremos engadindo no **taboeiro** os bloques necesarios para crear secuencias de comandos. Ao premear o **botón de inicio** a **torre de mando** lerá coa súa lente a secuencia e enviará as instrucións a **MatataBot** para que as execute.

Os bloques encaixan de forma segura no **taboeiro**, desta forma evítanse erros ao programar. Asegúrate de que os bloques con fendas estean colocados coa mesma cara abaixo.



Execución do programa

Durante a execución do programa, a luz laranxa do **taboeiro** pestanexará ata que o programa remate. Se desexas deter o programa en calquera momento, preme o **botón de inicio** do taboeiro.

Cando o programa finalizase ou non estea en execución, a luz laranxa quedarase fixa, indicando que podes colocar novos bloques para un próximo programa. Se hai un erro na secuencia de comandos, a luz laranxa pestanexará e a **torre de mando** emitirá un son de alarma 'Ding Ding'.

Regras para a programación

Ao colocar os bloques no **taboeiro**, a **torre de mando** leeraos en orde de esquerda a dereita e de arriba a abaixo. Cando completes unha liña, podes continuar colocando bloques na seguinte.

Esta mesma regra aplícase tamén cando hai separacións nas programacións, como é o caso das funcións, permitindo crear secuencias máis complexas sen perder a orde.

Instrucións de carga

Cargaremos a **torre de mando** e **MatataBot** cun dispositivo de alimentación V2A cando os seus respectivos leds se iluminen en azul e pestanexen rapidamente, isto significa que o dispositivo non ten enerxía dabondo e deben cargarse, durante a carga, o indicador ilumínase en vermello e vólvese verde cando a batería está completamente cargada.

Bloques de movemento



Adiante



Atrás



Xiro esquerda



Xiro dereita

Os bloques **Adiante** e **Atrás** fan que o robot avance ou retroceda 10 cm nesa dirección.

Cos bloques de **xiro**, o MatataBot xirará 90° cara a dereita ou á esquerda, segundo o bloque que elixas.

Bloques de números



Os bloques **numéricos** poden colocarse debaixo dos bloques **Adiante**, **Atrás** e **Bucle** para indicar cantas veces o robot debe realizar esa acción.

Por exemplo, ao colocar o bloque numérico 2 debaixo de **Adiante**, o robot avanzará 2 pasos (20 cm).



Se colocas el **bloque de dado**, a acción repetirase un número aleatorio de veces entre 1 e 6.

Bloques divertidos



Música



Baile



Movemento

Dispoñemos de tres bloques para que o robot interprete aa 6 pezas de música, 6 secuencias de baile e 6 conxuntos de movementos divertidos. Se colocas un **bloque numérico** debaixo destes bloques, o robot reproducirá a música, baile ou acción correspondente ao número elixido. Se non colocas ningún número, executarase a primeira opción de cada tipo.



Por exemplo, ao colocar un **3** debaixo del bloque de **baile**, o robot realizará o baile preestablecido número 3.

Bloques de ángulos



Estes bloques están incluídos no kit de debuxo, colócanse debaixo dos bloques de **xiro** e serven para indicar cantos graos xirará o robot cara a dereita ou esquerda.



Por exemplo, se colocas un **60** debaixo do bloque **xirar á dereita**, o robot xirará 60° nesa dirección.

Bloques de bucles



Inicio
de bucle



Fin
de bucle

En programación, un **bucle** é unha secuencia de instrucións que se repite varias veces. Coloca o bloque de **Inicio del bucle** antes da primeira instrución que queres repetir e o bloque de **Fin de bucle** ao final.

Se colocas un bloque **numérico** debaixo de **Inicio do bucle**, indicarás cantas veces debe repetirse a secuencia; se non hai un número, executarase só unha vez. Por exemplo, colocando un **2** debaixo de **Inicio del bucle**, o robot repetirá as instrucións entre Inicio e Fin de bucle dúas veces.



Bloques de función



Definir
función

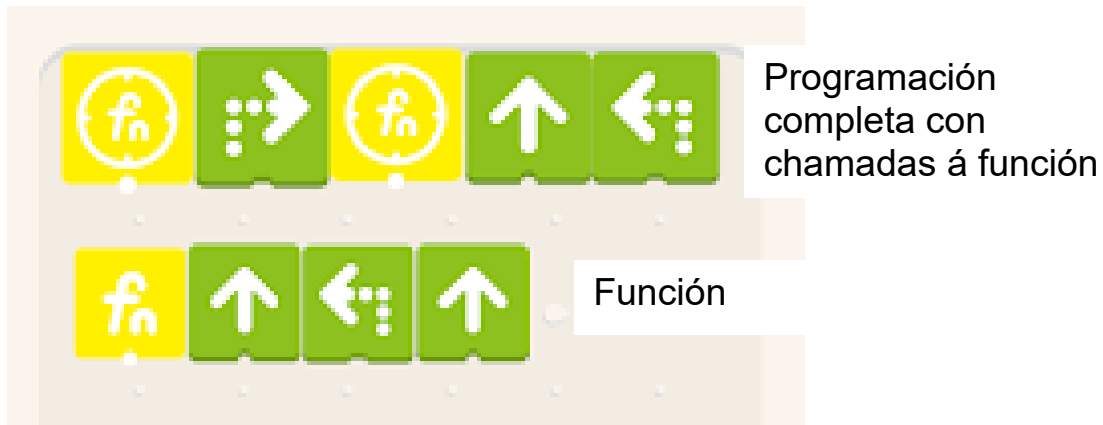


Executar
función

Unha **función** é un conxunto de instrucións que realiza unha tarefa específica (subprograma) e que pode usarse varias veces no programa. As funcións son esenciais na programación porque aforran tempo e fan que o código sexa máis craro.

Existen 4 bloques para funcións: 1 bloque para definir a función e 3 bloques para executala.

Como podes ver na seguinte imaxe, a función defínese na parte inferior do **taboeiro** usando o bloque **fn**. Na parte superior do **taboeiro** irá o programa. Colocase o bloque **fn** cun círculo cada vez que quieras executar a función.



Exemplo 2



Exemplo 3



Taboeiro de programación

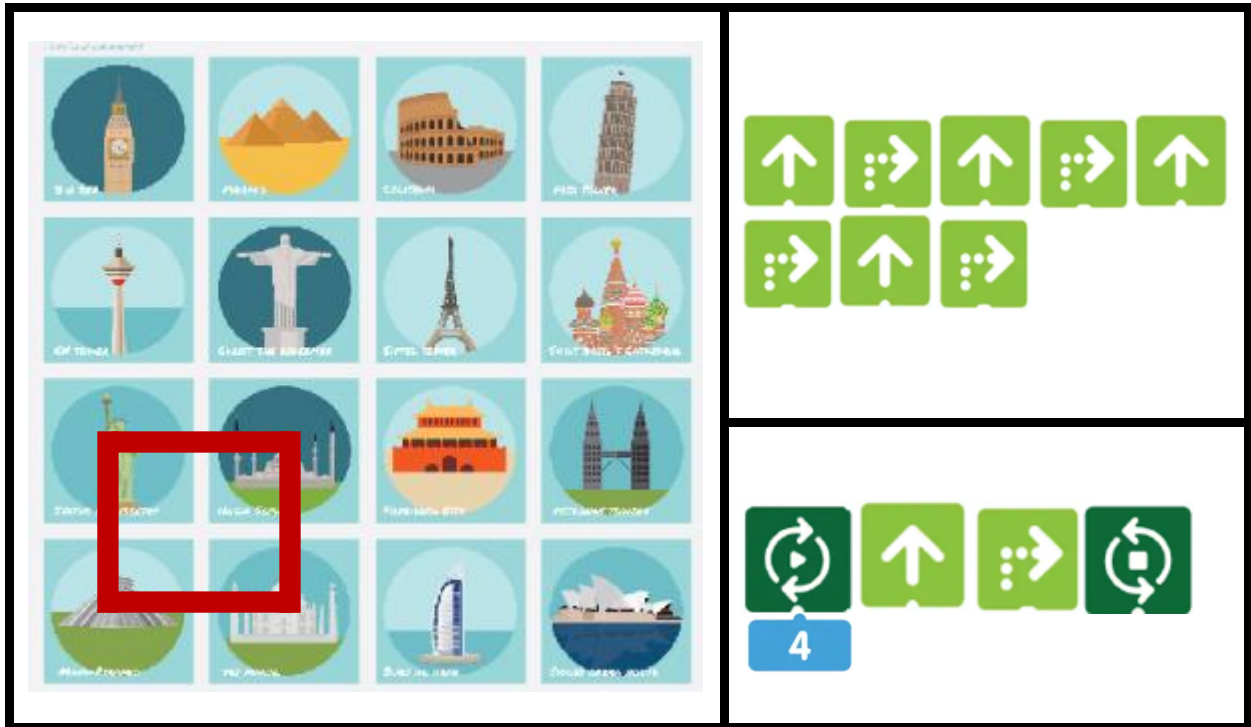
Taboeiro de programación

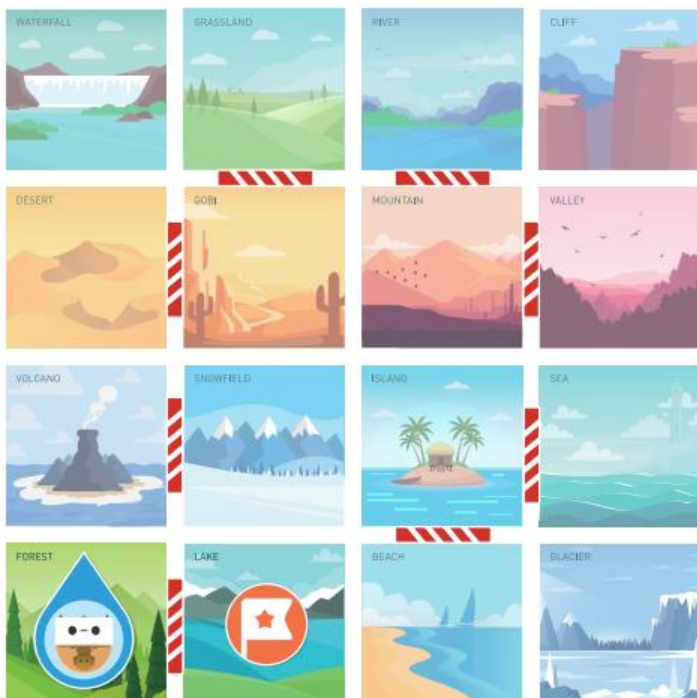
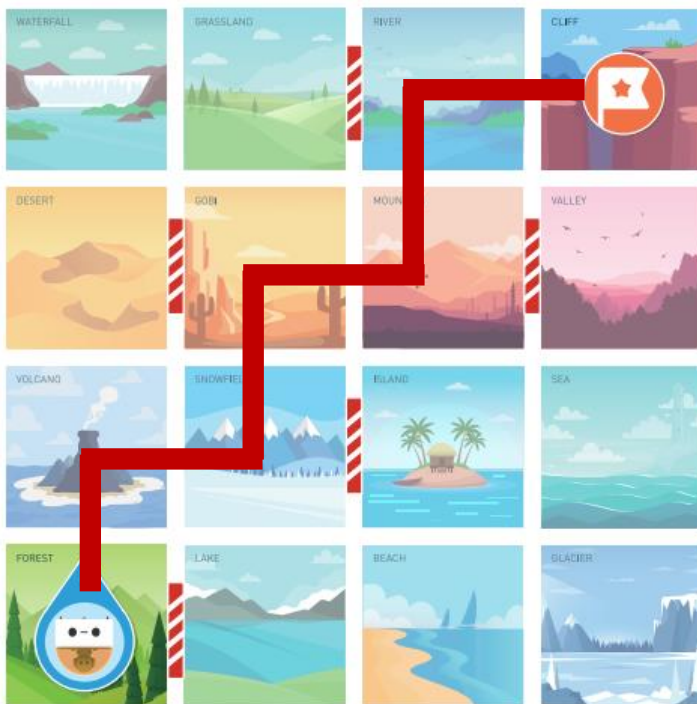
Taboeiro de programación

Taboeiro de programación

Exemplos de bucles

Dúas formas de percorrer un cadrado





Funcións

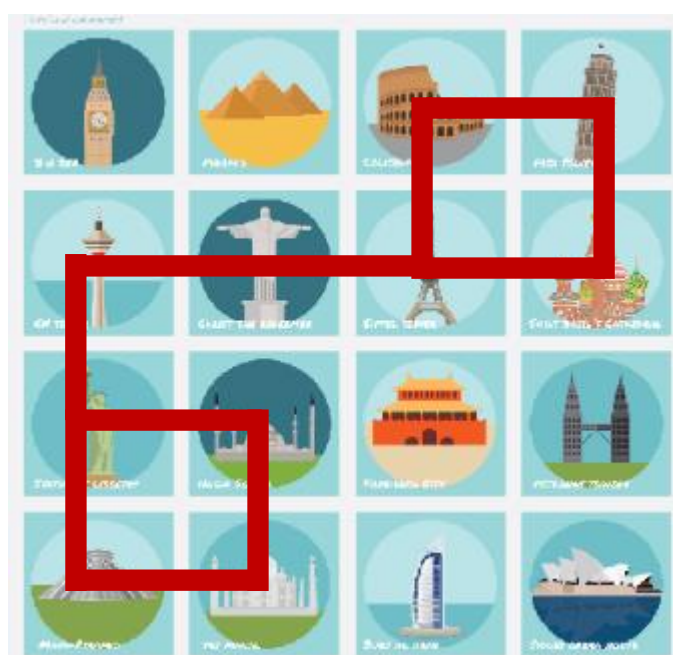
Tres formas de percorrer un cadrado



Forma 1: Un botón verde con una flecha circular y un botón verde con una flecha hacia arriba. Debajo del botón de la flecha hacia arriba hay un botón azul con el número 4.

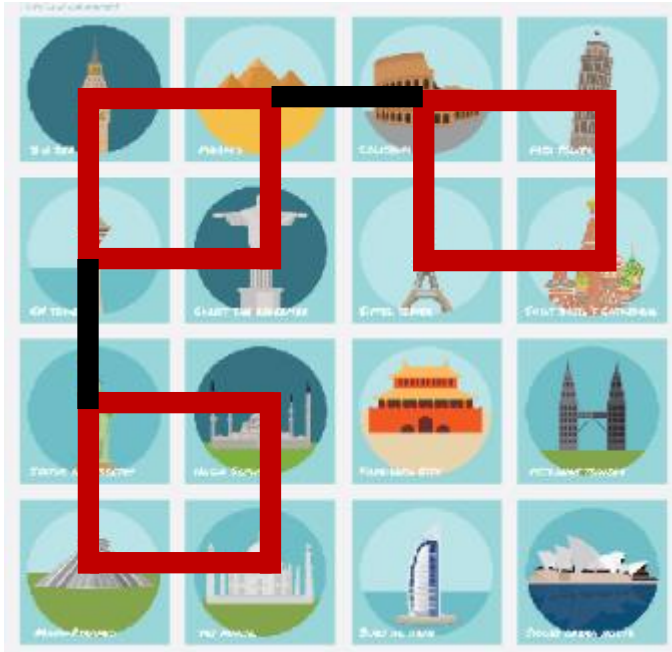
Forma 2: Tres botones amarillos con el símbolo f_n , seguidos de un botón verde con una flecha hacia arriba y un botón verde con tres puntos y una flecha hacia la derecha. Debajo del botón de la flecha hacia arriba hay un botón azul con el número 4.

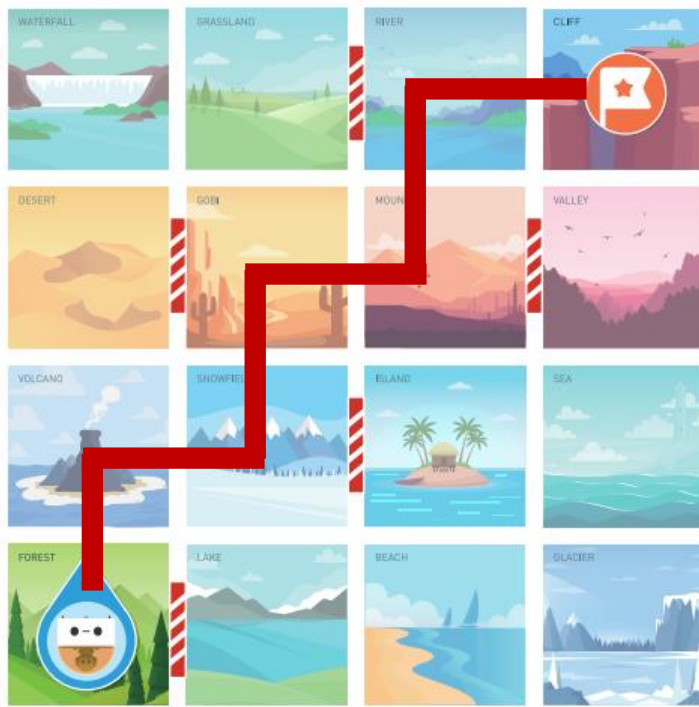
Forma 3: Un botón amarillo con el símbolo f_n , un botón verde con una flecha circular, un botón verde con una flecha hacia arriba, un botón verde con tres puntos y una flecha hacia la derecha, y un botón verde con una flecha circular. Debajo del botón de la flecha hacia arriba hay un botón azul con el número 4.



Forma 4: Un botón amarillo con el símbolo f_n , un botón verde con una flecha hacia arriba, un botón verde con tres puntos y una flecha hacia la derecha, un botón verde con una flecha hacia arriba, un botón verde con tres puntos y una flecha hacia la izquierda, y un botón amarillo con el símbolo f_n . Debajo del primer botón de la flecha hacia arriba hay un botón azul con el número 2, y debajo del segundo botón de la flecha hacia arriba hay un botón azul con el número 2.

Forma 5: Un botón amarillo con el símbolo f_n , un botón verde con una flecha circular, un botón verde con una flecha hacia arriba, un botón verde con tres puntos y una flecha hacia la derecha, y un botón verde con una flecha circular. Debajo del botón de la flecha hacia arriba hay un botón azul con el número 4.

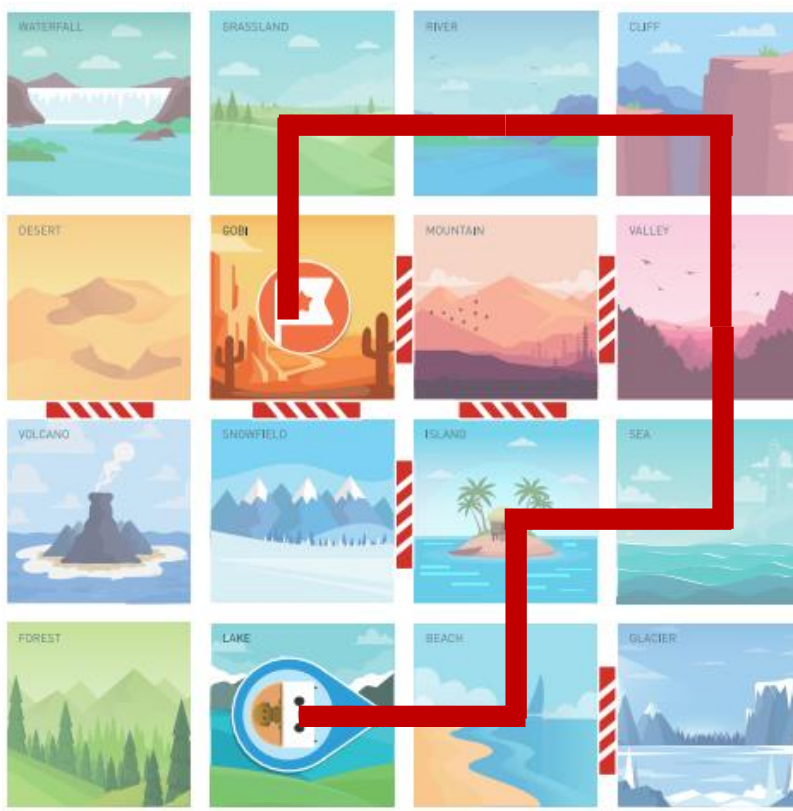
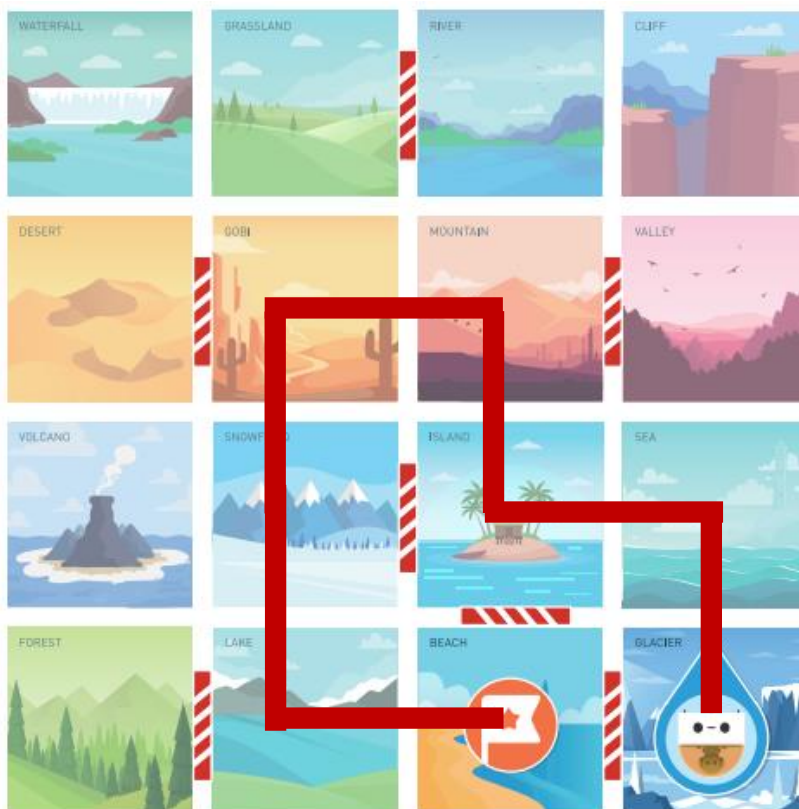




20 chanzos



Dá as instrucións para crear o número máximo de chanzos



Disanedu:

Diego Guerrero
Santiago Ortiz

**Co fin de seguir mellorando podes enviarnos propostas,
suxerencias, aportacións e críticas a:**

diegoguerrero@educarex.es

santiagoortiz@educarex.es