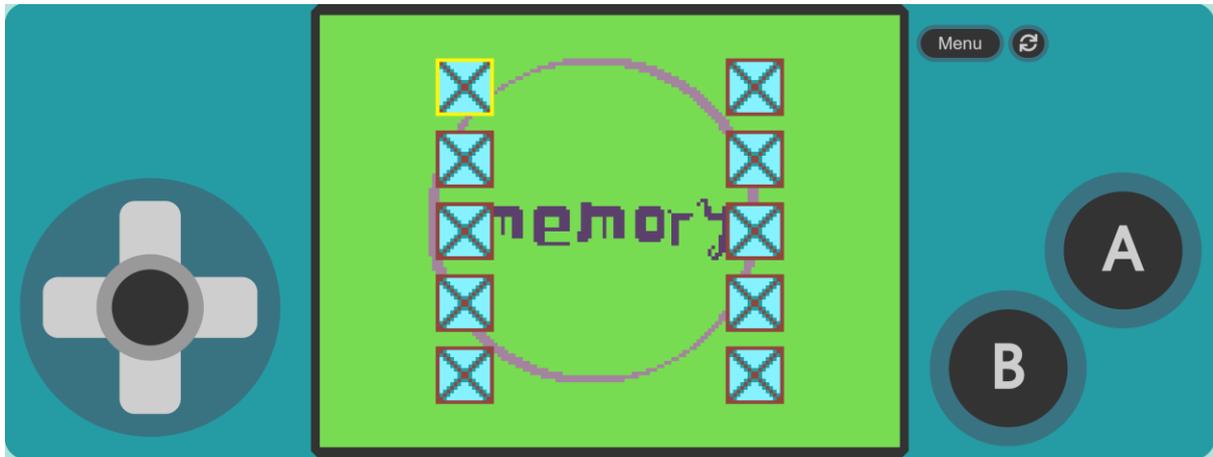


Tercer Ejercicio Arrays



Descripción.

En este proyecto crearemos una inspiración del juego clásico “Memory” donde el jugador tiene una mesa con cartas y tiene que escoger dos veces seguidas cartas con la misma imagen.

Se usará muchos de los elementos aprendidos aunque mayormente se trabaja el apartado “Array” donde se guardará en listas las posiciones de cada carta, las imágenes que están relacionadas, etc.

Para ello accederemos a [MakeCode Arcade](#) y realizaremos las operaciones necesarias.

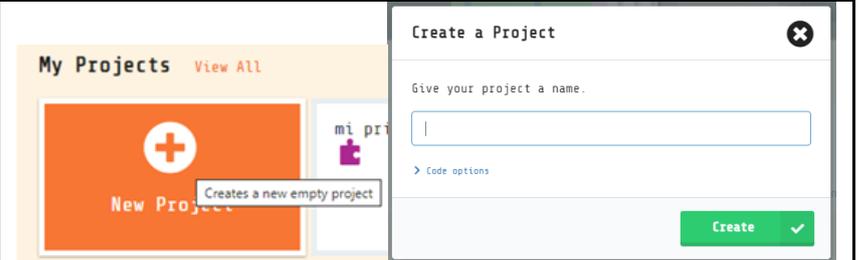
Objetivos

- Crearemos un sprite que sirva como de cursor para navegar entre distintas opciones.
- Navegaremos entre array para conseguir el resultado requerido, tanto para poner contenido aleatorio como imágenes asociadas a cierta posición.

Guía de programación

NEW PROJECT

We start creating a project, we should establish the name, for example "Picking up food" and then press "create" button.



PROGRAMACIÓN PRINCIPAL

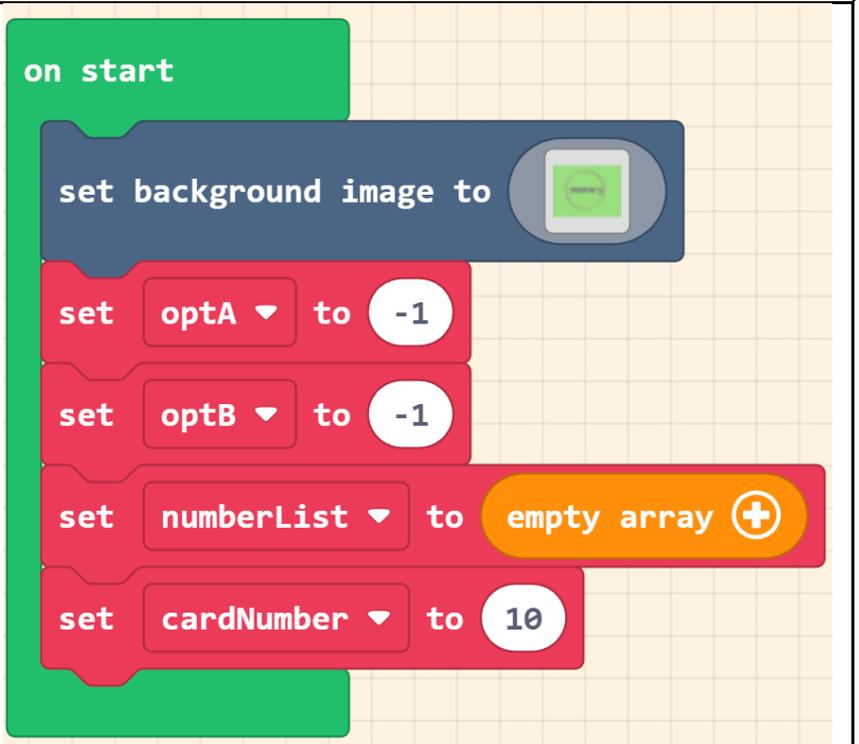
CREACIÓN INICIO DEL JUEGO

Empezaremos programando el fondo del juego y estableciendo valores iniciales.

Las variables "optA" y "optB" servirán para comparar las elecciones realizadas por el jugador y determinar si ha acertado o no en el emparejamiento. Se establecerán a -1 porque es el valor que tendrá si no se ha escogido ninguna opción

La variable "numberList" la declaramos de tipo array, establecerá la posición de cada una.

La variable "cardNumber" establecerá la cantidad de cartas que habrá en el juego



INICIAR ARRAY NUMBERLIST

Creamos la función con el parámetro numérico. El parámetro que introduciremos más adelante es el "cardNumber"

Dentro de esta función usaremos un bucle "for" que se repita tantas veces como la mitad de la cantidad de cartas que se usen.

En el array que hemos declarado antes "numberList" se introducirá dos veces seguidas el valor de "index" teniendo duplicado el valor. Para este caso los valores de "numberList" una vez terminados serán

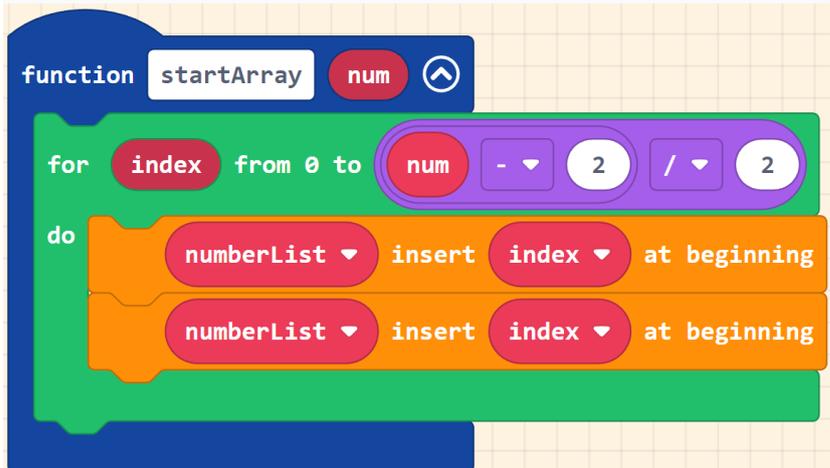
```
numberList = [0,0,1,1,2,2,3,3,4,4]
```

Edit Function

Add a parameter



Done



COLOCACIÓN DE LAS CARTAS

Volveremos al bloque on start e introducimos al final la llamada a la función “startArray” introduciendo como variable a “cardNumber”.

Programaremos a continuación un bucle “while” que creará y ubicará las cartas que tendremos en el juego. Usaremos una formula para ir desplazando las cartas en el eje y, en el eje x dependiendo de si es par o no, tendrán la misma distancia entre ellas.

Aumentaremos en 1 el valor de “index” para que se desplace en el array en cada iteración del bucle.

Creamos un Sprite “pointer” para la elección de la carta.

```

call startArray cardNumber
while index < length of array numberList
do
  set card to sprite of kind hideCard
  if remainder of index / 2 = 0 then
    set card position to x 40 y 20 + 10 x index
  else
    set card position to x 120 y 10 + 10 x index
  change index by 1
set pointer to sprite of kind Player
set pointerPos to 0
  
```

GENERACIÓN ALEATORIA DE POSICIONES DE LAS CARTAS

Crearemos la función "RandomiceArray". Esta función programaremos como se colocan cada carta de manera aleatoria.

La variable n tendrá de valor el tamaño de la cantidad de cartas que usemos.

Usaremos un bucle "for" donde irá aumentando el valor de index empezando desde 0 hasta la cantidad de cartas menos 1, recordad que de 0 a 9 son 10 veces.

Declararemos el array "imaList" que colocará aleatoriamente los valores de "numberList".

Para no tener conflicto con los valores introducidos, borramos los valores usados.

Llamaremos la función en "RandomiceArray" para activar.

```

function RandomiceArray
  set n to length of array numberList
  for index from 0 to n - 1
  do
    set random to pick random 0 to length of array numberList - 1
    imaList insert numberList get value at random at beginning
    numberList remove value at random
  
```

```

change index by 1
set pointer to sprite of kind Player
set pointerPos to 0
call RandomiceArray
  
```

Con esta programación tendremos establecido la cantidad de las cartas del juego, su posición, lo que escoden cada una de ellas.