





Tercer Ejercicio Sprites



Descripción

En este proyecto realizaremos, lo que en videojuegos se denomina, cinemática. Esto consiste en hacer una secuencia de video donde el jugador no tiene el control y se utiliza para avanzar en la historia del juego o para reforzar la trama del personaje principal, presentar personajes o enemigos, atmosfera, etc. Para ello, usaremos al famoso fontanero Mario en el que acaba de vencer a Bowser y ha llegado al rescate de la princesa Peach.

Para ello accederemos a MakeCode Arcade y realizaremos las operaciones necesarias.

Objetivos de programación y diseño

- Dibujar un mapa representativo con distintos elementos para determinar distintos escenarios.
- Trabajar animaciones de distintas formas.
- Crear una animación para nuestro personaje.
- Realizar una introducción a la cinemática.
- Realizar la cinemática con los distintos elementos de programación.
- Poner en contexto al jugador mediante elementos expresivos del juego o mensajes mostrados.







Programación del juego

Aquí os dejamos un enlace con parte de los Assets hechos y de la programación inicial:

https://makecode.com/ ApoXXtEx2JrX

A continuación, os dejamos un resumen de los Assets hechos y de la creación del escenario.

CREACIÓN DE ASSETS

CREACIÓN SPRITE PRINCIPAL

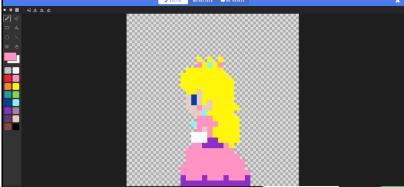
Te recomendamos utilizar una matriz de 16x16 px para el Sprite de "MarioLeft".



CREACIÓN SPRITES ADICIONALES

Crearemos el Sprite de Bowser en una matriz de 40x40 px.

Crearemos el Sprite de Peach en una matriz de 32x32px.









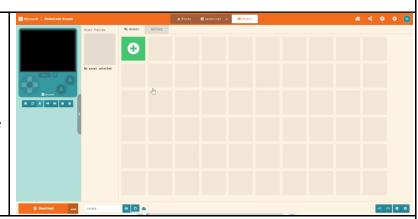
Crearemos el <mark>Sprite</mark> de <mark>Mario</mark> en una matriz de 16x16px. Aquí Mario mira a la derecha y será donde adjuntaréis la animación más

adelante.



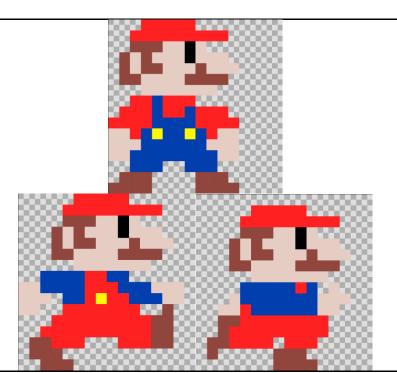
CREACIÓN TILEMAP

En Assets crearemos el escenario con los distintos elementos que puedes encontrar en el propio editor. Éste tiene unas dimensiones de 40x8 px.



CREACIÓN SPRITE DE ANIMACIÓN

Con nuestro Sprite de Mario, haremos los distintos Sprites que compondrán las animaciones. Todos de 16x16 px.









PROGRAMACIÓN PRINCIPAL

CREACIÓN INICIO DEL JUEGO

```
Para comenzar, crearemos un bloque
"on start" y colocaremos en su interior
"set tile map to", Sprite de "leftMario",
lo declaramos como Player y lo
colocaremos en la posición "set
"leftMario" position to 'x' 130 'y' 75".

Repetimos con "Bowser", lo declaras
Enemy y "set 'Bowser' position to 'x' 80
'y' 70".
```

Y por último, hacemos lo mismo con Peach. Lo declaramos Player y lo colocaremos en la posición "set 'Peach position to 'x' 570 'y' 65".

```
on start

set tilemap to 
set leftMario ▼ to sprite from of kind Player ▼

set leftMario ▼ position to x 130 y 75

set Bowser ▼ to sprite from of kind Enemy ▼

set Bowser ▼ position to x 80 y 70

set Peach ▼ to sprite from of kind Player ▼

set Peach ▼ position to x 570 y 65
```

INCORPORACIÓN DE TRAMA

Vamos a crear una pequeña atmosfera para saber que está ocurriendo.

Es el final del juego y Mario ha derrotado a Bowser y ha salvado a la princesa. Mario va a reunirse con Peach y ésta le dará las gracias por salvarla a ella y a su reino.

Primero pondremos un texto para entrar en situación.

Para ello, Game y show long text "escribimos el contexto que queramos" top. Lo colocamos antes de la creación de Peach.

```
set tilemp to set leftMario v to sprite of kind Player v

set leftMario v to sprite of kind Player v

set Bouser v to sprite of kind Enemy v

set Bouser v position to x 80 y 70

show long test Mario acaba de coger el martillo y es el final de Bouser. Ahora irá a rescatar a la princesa Peach top v

set Peach v to sprite of kind Player v

set Peach v position to x 570 y 65
```







Después pondremos una pausa de 1seg para darle más suspense.

Acto seguido en Images cogemos el bloque flip '(Bowser) image' vertically. Con esto haremos que el Sprite se gire verticalmente.

Le damos una aceleración (para que caiga) set (Bowser) 'ay (acceleration y) to 30.

Por último, ponemos otra pausa de 1seg para que le dé tiempo a caer.

Todo lo colocamos entre el texto anterior y Peach.

```
on start

set leftHario v to aprite of kind Player v

set leftHario v to aprite of kind Player v

set Bowser v to aprite of kind Enomy v

set Bowser v to aprite of kind Enomy v

set Bowser v position to x 30 y 70

show long text Mario acaba de coger al martillo y as al final de Bowser. Ahora iná a rescatar a la princesa Peach top v

pause 1000 v ms

flip Bowser v image vertically

set Bowser v ay (acceleration y) v to 30

pause 1000 v ms

set Peach v to aprite of kind Player v

set Peach v position to x 570 y 65
```

MECÁNICA DE CINEMÁTICA

Vamos a empezar con la cinemática.

Para ello empezamos reemplazando el Sprite leftMario. Esto se consigue destruyendo el Sprite existente y poniendo en el mismo lugar el nuevo. Sprites, destroy (leftMario). Ahora colocamos set 'Mario' to (Sprite 'Mario' of Kind Player). Lo colocamos en su posición, le damos velocidad y hacemos que le siga la cámara:

set (Mario) position to 'x' 130 'y' 75 set (Mario) velocity to 'vx' 50 'vy' 0 camera follow sprite (Mario).

Todo lo colocamos entre la última pausa anterior y Peach.

```
set tilemap to sprite of kind player v

set leftMario v to sprite of kind player v

set leftMario v to sprite of kind fnewy v

set Bouser v to sprite of kind fnewy v

set Bouser v position to x 80 y 70

show long text (Nario acaba de coger el martillo y es el final de Bowser. Ahora irá a rescatar a la princesa Peach top v

panse 1000 v as

flip Bouser v image vertically

set Pario v image vertically

set Pario v v (acceleration y) v to 30

set Pario v velocity to vx 50 vy 6

set Peach v to sprite of kind Player v

set Peach v to sprite of kind Player v

set Peach v to sprite of kind Player v
```







CREAR ANIMACIÓN

Vamos a crear una función que se encargará de controlar la animación de Mario cuando ande o este quieto.

Functions walkMario

Empezamos por la <mark>animación</mark> de reposo (Idle).

Animation

set anim to créate animation of 'Idle' with Interval 100 ms

Añadimos el frame de Mario en reposo.

Y lo asignamos al Sprite Mario.

A continuación, seguimos con la animación andando (Walking).
Exactamente igual, pero incorporando los frames de movimiento.

Animation

set anim to créate animation of 'Walking' with Interval 100 ms

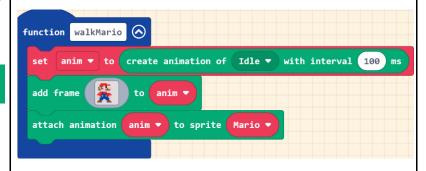
Añadimos los <mark>frames</mark> de Mario andando.

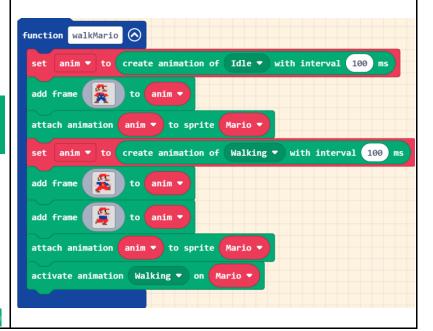
Asignamos la animación al Sprite de Mario.

Por último

activate animation 'Walking' on (Mario)













Por último, llamamos a la función en el on start para iniciar el movimiento de Mario mientras anda.

```
set tilemap to
   leftMario ▼ to sprite 🙀 of kind Player ▼
   leftMario → position to x 130 y 75
       er 🔻 to sprite 🎇 of kind Enemy 🕶
            position to x 80 y 70
  w long text <sup>(*</sup>Mario acaba de coger el martillo y es el final de Bowser. Ahora irá a rescatar a la princesa Peach <sup>(*</sup>
   e 1000 ms
flip Bowser ▼ image verticall
set Bowser ▼ ay (acceleration y) ▼ to 30
   se 1000 ▼ ms
  t Mario ▼ to sprite 🎇 of kind Player ▼
camera follow sprite Mario ▼
set Mario ▼ position to x 130 y 75
 et Mario • velocity to vx 50 vy θ
   set Peach ▼ position to x 570 y 65
```

CREACIÓN FINAL DE LA CINEMÁTICA

Ahora haremos el final de la cinemática.

Para ello crearemos un Loop forever.

Dentro colocaremos

Logic if 'Mario $x \ge 550$ ' then

Con esto le decimos que: si la posición de x de Mario es mayor o igual a 550 que realice la siguiente programación.

Set (Mario) velocity to vx 0 vy 0 -> que se pare.

Peach say "Lo que queramos" activate animation 'Idle' on (Mario)







DETALLES ADICIONALES

Para que le demos el toque final. Añadir unos efectos de <mark>sonido</mark> y <mark>música</mark> de fondo.

```
Siley from at 120 of 120 * best
play from at 120 of 120 * best
```







CREACIÓN MELODÍAS forever play tone at Low G for Para que la música de fondo suene durante toda la cinemática, crearemos un Loops forever y dentro meteremos toda la música. Middle F# Para las melodías, vamos a "Music" y play tone at (Middle G después cogemos el bloque "play tone at (Middle C) for ½ beat ". Aquí play tone at (High C) fo colocaremos la nota musical que play tone at (High E) necesitéis y le asignaremos el tiempo de reproducción. Introduciremos tantas play tone at (High E notas necesitemos para vuestra play tone at (High D melodía. Aquí tenemos un ejemplo de melodía de Mario. play tone at (High F play tone at (Middle B play tone at (High D play tone at (High C Por último, colocaremos un efecto de sonido cuando Bowser sea eliminado. Para ello Music y play sound 'magic wand'. Lo colocamos después de la última pausa para darle ese efecto de caída.

Gracias a esta programación hemos aprendido a realizar animaciones y cinemática. Esto hace que nuestros juegos sean visualmente más vistosos e interesantes de jugar.







Glosario

Funciones: Es un subprograma que recoge un conjunto de instrucciones y pueden ejecutarse desde el programa principal haciendo una llamada a él.

Muros: Objetos o espacios donde los distintos elementos del juego no pueden atravesar.

Cámara: Objeto dentro de un escenario que sirve como vista del jugador respecto al juego.

Narrativa: Parte de un videojuego que sirve para construir una historia.

Efecto: Algo que se aplica sobre el escenario, objeto, personaje y más elementos para transmitir realismo o una sensación dentro del juego.

Música: Combinación de sonidos y silencios que componen un ritmo.

Imagen: Elemento que se visualiza en una pantalla y muestra algo (un paisaje, personas, etc).

Aceleración: Es la variación de velocidad por unidad de tiempo.

Velocidad: Es una magnitud física que relaciona la posición con el incremento de tiempo.