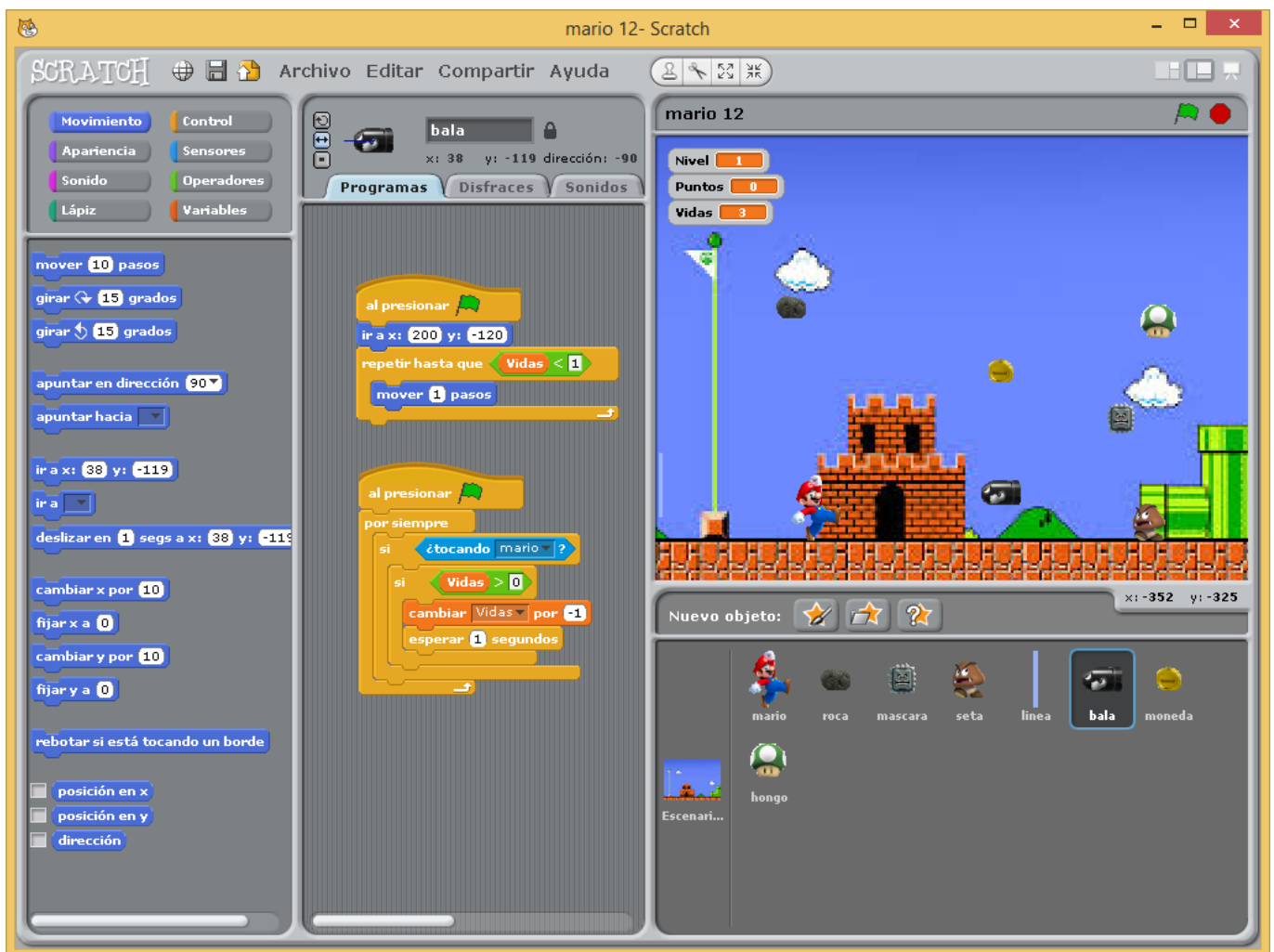


Curso Scratch



NOMBRE DEL ALUMNO/A:

- _____
- _____

Mario Bros

PRÁCTICA 1

En esta primera práctica crearemos el **Objeto Mario** y le pondremos **2 disfraces** (andando1 y andando2) para que tenga más realismo. Mario se moverá de forma automática por la pantalla y cuando llegue al final rebotará.

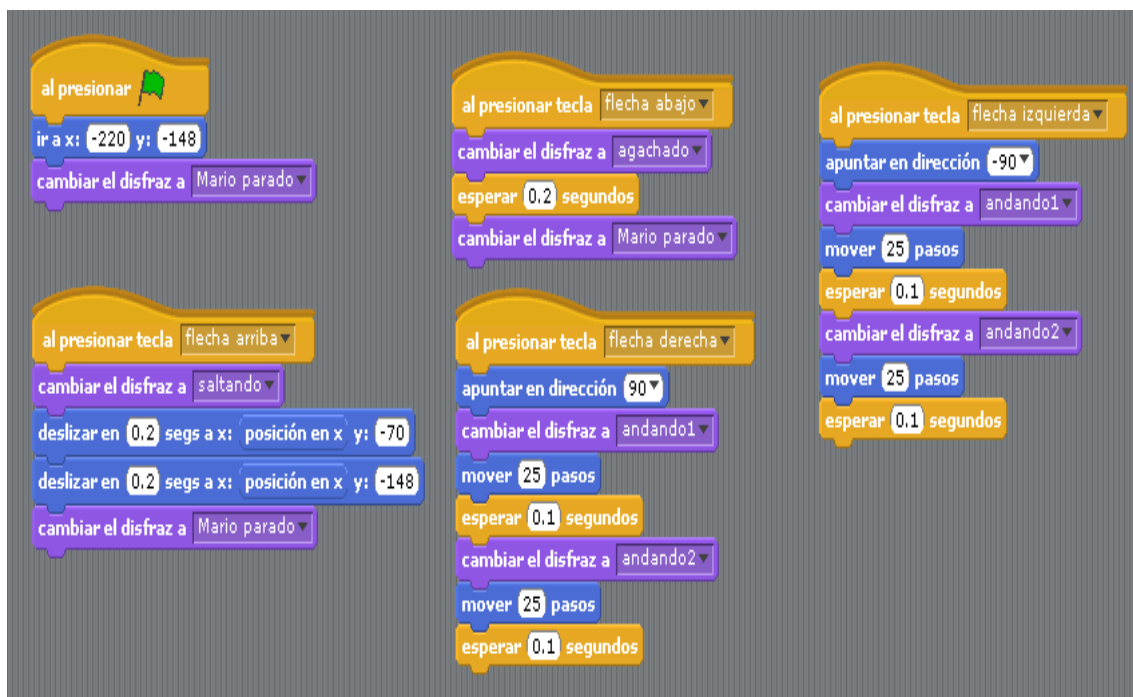
- 1) Primero borrar el Objeto 1 que sale por defecto y elegir un nuevo objeto que será el archivo (Mario andando1).
- 2) Después ir a disfraces e importar el archivo Mario andando2.
- 3) Borrar el nombre Objeto1 y poner Mario.
- 4) Ajustar el tamaño de Mario (achicar objeto).
- 5) Por último ir a programas y realizar la siguiente secuencia.



PRÁCTICA 2

En esta segunda práctica Mario ya se manejará con las **teclas de dirección** e irá cambiando de disfraz según la tecla que activemos.

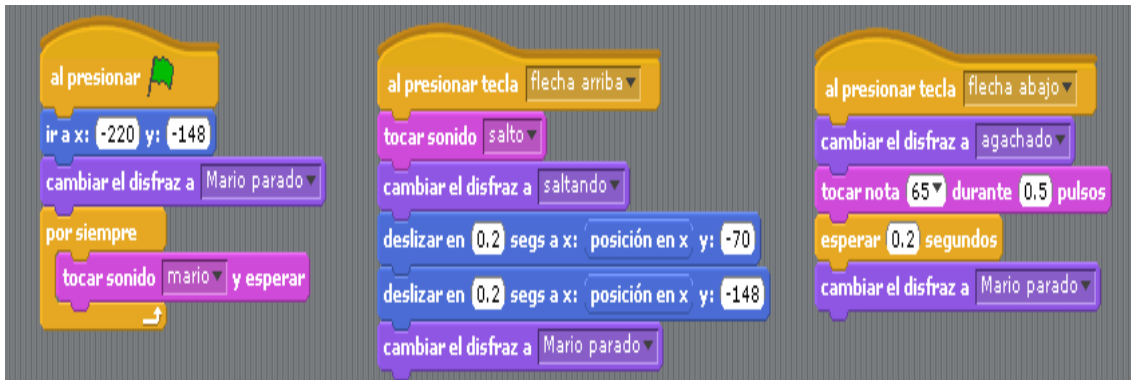
- 1) Importar tres disfraces nuevos (Mario parado, agachado y saltando) como en la práctica anterior.
- 2) Al disfraz saltando hay que editarlo para poder quitarle el fondo blanco (con la goma).
- 3) Programar la secuencia de abajo y ejecutar.
- 4) Si vemos que los disfraces no coinciden al moverse hay que ajustar los tamaños (editar - achicar).
- 5) Por último poner el centro de los disfraces en los pies (editar - Seleccionar el centro del disfraz).



PRÁCTICA 3

En esta práctica se agregarán algunos sonidos y tonos e insertaremos un **escenario** nuevo.

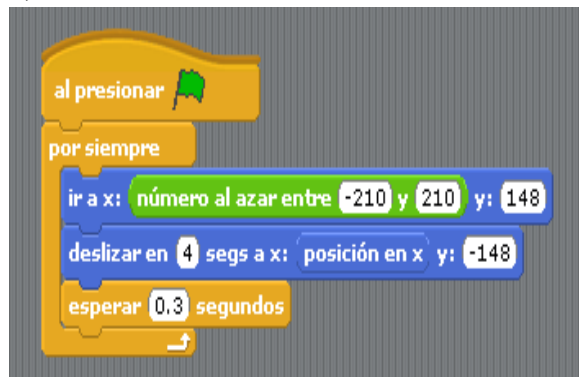
- 1) Seleccionar el objeto Mario e importar el archivo Mario.mp3 suministrado por el profesor. Esa música empezará a sonar cuando empieza la partida.
- 2) Importar un sonido de los existentes en el programa (sonido - importar - sonido) y llamarle salto.
- 3) Volver a programar a Mario modificando las acciones de algunas teclas.
- 4) Por último pinchar en escenario (cartulina blanca) e importar el archivo fondo1.



PRÁCTICA 4

En esta cuarta práctica empezaremos a agregar a los enemigos de Mario que se mueven de arriba a abajo.

- 1) Importar el objeto **mascara** y en editar quitarle el fondo blanco.
- 2) El objeto saldrá al **azar** desde cualquier punto de arriba de la pantalla (eje y) y se **deslizará** de forma vertical (mantener la posición x) hasta la situación de Mario.



PRÁCTICA 5

En esta quinta práctica le pondremos otro enemigo que se moverá igual al anterior pero al llegar al suelo cambiará de disfraz.

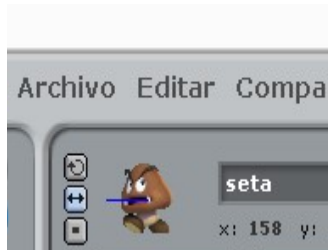
- 1) Importar el objeto **roca** y después importar un segundo disfraz con el mismo fichero.
- 2) Hay que editar el segundo disfraz (roca 1) y borrar trozos para que parezca rota.
- 3) Saldrá al azar como el objeto mascara por lo que para que no coincidan hay que ponerle diferente velocidad de deslizamiento y de tiempo de espera.



PRÁCTICA 6

En esta sexta práctica se agregará un tercer enemigo que se **moverá horizontalmente** de derecha a izquierda.

- 1) Importar el objeto **seta** y programar para que salga desde la derecha y a la misma altura que Mario por lo que hay que poner el centro del objeto en los pies.
- 2) Además el objeto rebotará cuando toque el borde y volverá hacia la derecha.
- 3) Si al ejecutar el objeto mira hacia el lado contrario al que se mueve hay que editar el disfraz y girar horizontalmente.
- 4) Si al rebotar el objeto gira poniéndose boca abajo hay que pulsar en “sólo mirar izquierda-derecha”.



PRÁCTICA 7

En esta séptima práctica se agregará un segundo enemigo que se moverá también horizontalmente de derecha a izquierda.

- 1) Importar el objeto **bala** y que surja también desde la derecha pero a la altura apropiada para que pase por encima de Mario agachado.
- 2) Deberá desaparecer cuando llegue a la izquierda y volver a aparecer en la derecha.
- 3) Para que esto ocurra hay que crear un objeto nuevo (llamarlo **linea**) que se hará pintando y que se ajustará en la parte izquierda del escenario donde choque con la bala.
- 4) Hay que programar para que siempre que la bala choque con la línea la bala desaparezca.
- 5) Hay que ajustar los números de bala para que no salga a la vez que la seta.
- 6) Por último ajustar bien los parámetros de Mario para que pueda caer más despacio (flecha arriba) y así no chocar con la seta cuando salte y también para que se quede agachado siempre (flecha abajo) y no choque con la bala.
- 7) Si hay algún problema con la dirección o el sentido de la bala mirar el paso 3 de la práctica anterior.



PRÁCTICA 8

En esta práctica se programará los últimos movimientos de Mario. En este caso queremos que pueda **saltar** y la vez que **avanzar** hacia la derecha o hacia la izquierda.

- 1) Para que no haya rutinas duplicadas y el programa reaccione de manera errónea es mejor utilizar una tecla que no hayamos utilizado todavía (tecla espaciadora) y combinarla con las teclas derecha e izquierda.
- 2) La orden que se utiliza es que se deslice hacia arriba pero que caiga en una posición más avanzada ($x + 25$) si se utiliza la tecla derecha y en una posición más retrasada ($x - 25$) si se utiliza la tecla izquierda.



PRÁCTICA 9

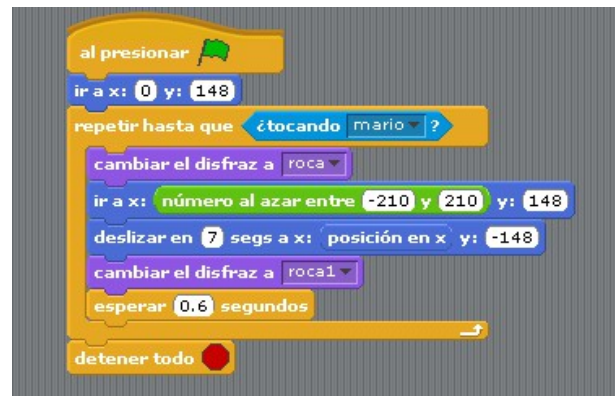
En esta práctica se programará para que la **partida termine** en el momento en que alguno de los objetos **toque a Mario**.

- 1) Primeramente hay que poner una orden en los objetos de movimiento vertical (mascara y roca) para que empiecen en un determinado punto de la pantalla, ya que si tocan a Mario antes de que este se mueva no se podría volver a jugar. Los objetos seta y bala ya tienen la orden puesta.
- 2) Por último hay que poner la orden para que los objetos se muevan como siempre a no ser que estén tocando a Mario (repetir hasta tocando a Mario). En el caso que esto ocurra se acaba la partida (detener todo).

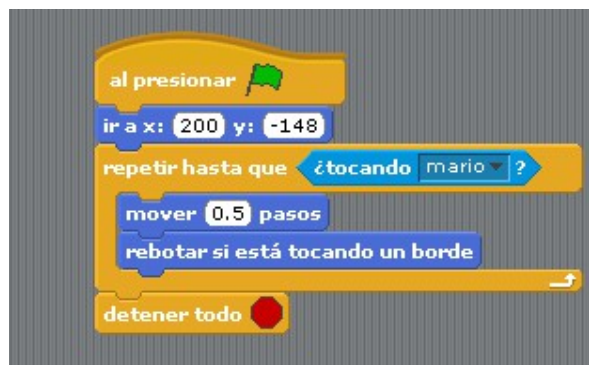
Máscara



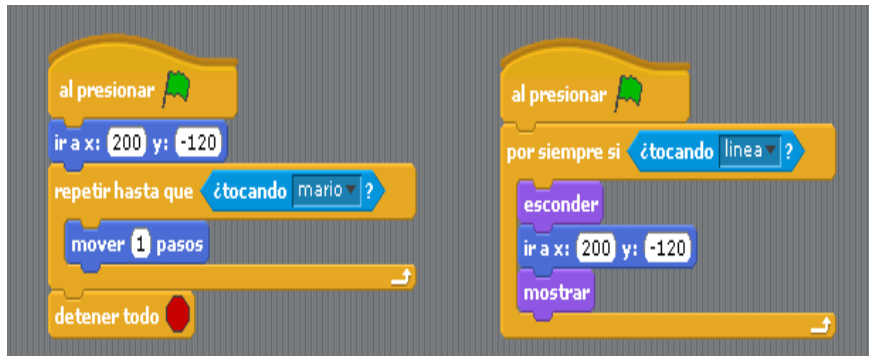
Roca



Seta



Bala



PRÁCTICA 10

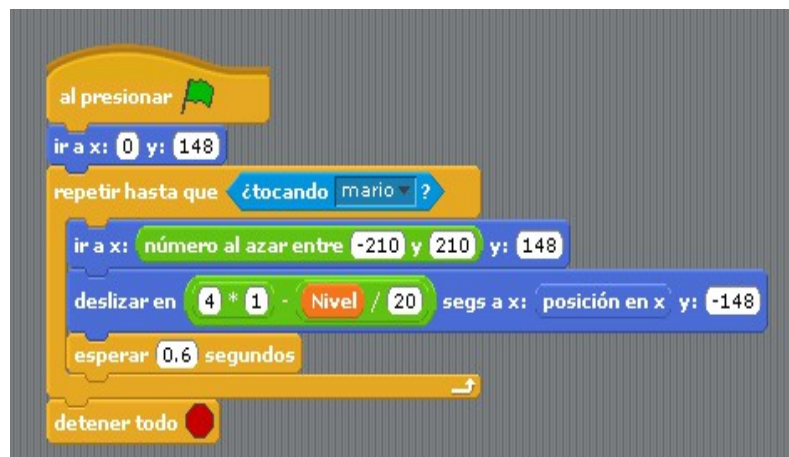
En esta práctica se empezará a utilizar las **variables**. En este caso se creará la variable **Nivel** y se utilizará para que los enemigos de Mario vayan teniendo una velocidad mayor a medida que pase el tiempo.

- 1) Pinchamos en el escenario y creamos una **variable** (variables - nueva variable) y la llamamos **Nivel**. Inmediatamente aparecen los comandos fijar, cambiar, mostrar y esconder la variable. Además esta variable aparecerá en todos los objetos en su carpeta variable.
- 2) Hay que pinchar en nivel para que se vea en el escenario.
- 3) Programamos para que el nivel sea 1 y para que mediante un cronómetro el nivel suba su valor en una unidad cada 5 segundos.
- 3) Ahora programamos para que la velocidad de los enemigos aumente a medida que aumenta el valor de la variable nivel. Seleccionamos el objeto máscara y en la orden de deslizar le ponemos una función matemática donde intervenga la variable.
- 4) Ahora cada 5 segundos irá cambiando el nivel y el tiempo de deslizamiento será cada vez más pequeño.
- 5) Se puede cambiar la velocidad de los otros enemigos con la misma fórmula o con otra diferente.

Escenario



Máscara



PRÁCTICA 11

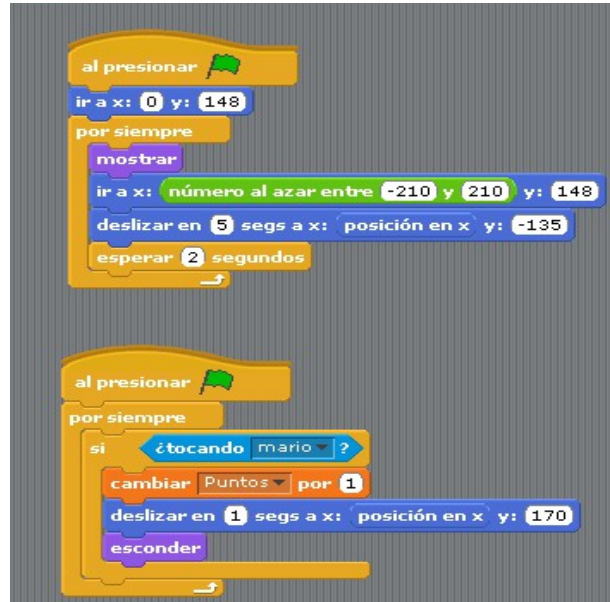
En esta práctica crearemos un objeto llamado **moneda** que cuando Mario lo toque **sumará puntos**.

- 1) Seleccionamos el escenario y creamos una nueva **variable** (variables - nueva variable) y la llamamos **Puntos**. Pinchamos en ella para que se vea en el escenario.
- 2) Programamos el escenario para que los puntos empiecen en cero.
- 3) Ahora vamos a crear un objeto nuevo, para ello importamos el objeto **moneda** y lo editamos para quitarle el fondo blanco.
- 4) Programa moneda para que se deslice desde la parte superior y que se espere un tiempo en la posición de Mario.
- 5) Por último hay que programar para que cuando Mario la toque se deslice hacia arriba, se esconda y que marque 1 punto.
- 6) En este caso aunque Mario esté en el aire lo reconocerá y marcará puntos.

Escenario



Moneda

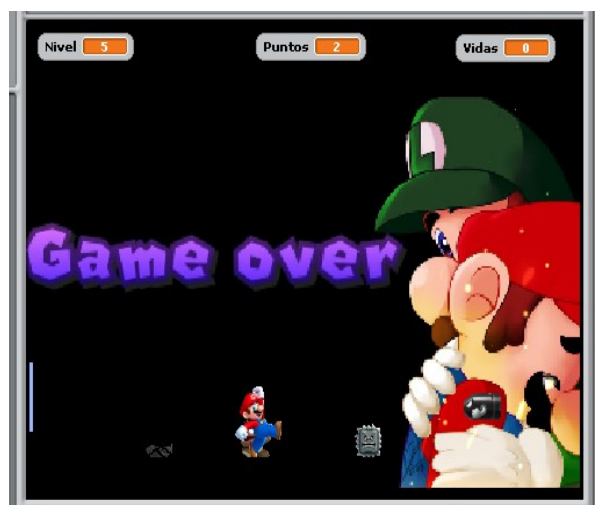


PRÁCTICA 12

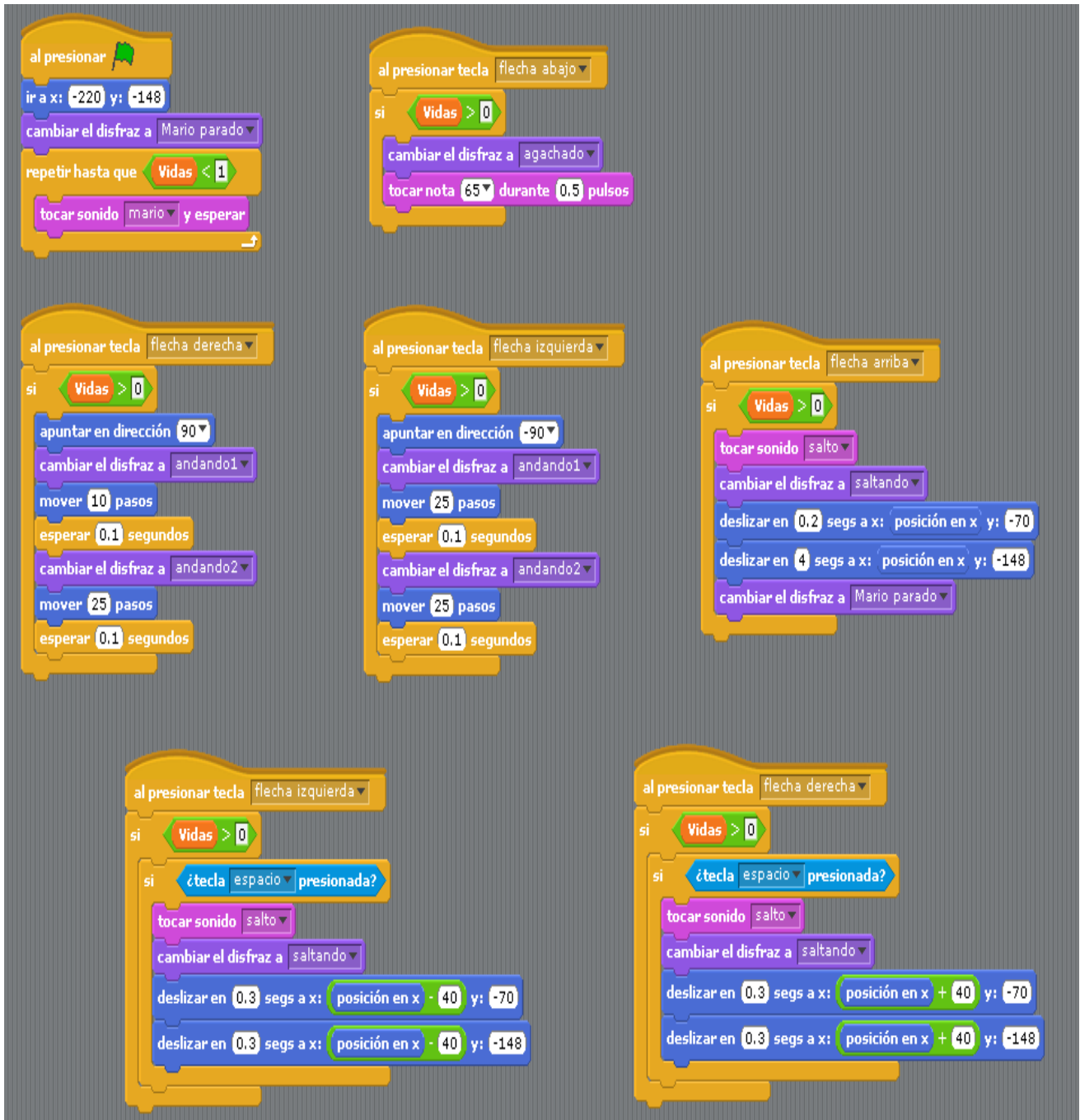
En esta práctica le pondremos **vidas a Mario**, de tal manera que los enemigos le vayan restando una vida cada vez que lo toquen. Al acabarse las vidas saldrá un escenario que indicará que la partida se ha perdido y todo se parará.

- 1) Seleccionamos el escenario y creamos una nueva **variable** (variables - nueva variable) y la llamamos **Vidas**. Pinchamos en ella para que se vea en la pantalla.
- 2) Aquí programamos para que las vidas de Mario sean 3 al comienzo de la partida.
- 3) Además importamos el archivo **perdiste** y lo programamos para que aparezca de fondo cuando se pierde la partida.
- 4) Ahora hay que programar a Mario para que sólo se mueva mientras le queden vidas ($\text{vidas} > 0$), en caso contrario Mario dejará de moverse.
- 5) Programamos también al objeto moneda para que repita sus movimientos sólo mientras Mario tenga vidas (repetir hasta que $\text{vidas} < 1$) se parará.
- 6) De la misma manera volver a programar a todos los enemigos para que repitan sus movimientos sólo mientras Mario tenga vidas (repetir hasta que $\text{vidas} < 1$), sino es así se pararán.
- 7) Por último en la programación de los enemigos tiene que aparecer una orden para que cada vez que toquen a Mario le resten una vida.

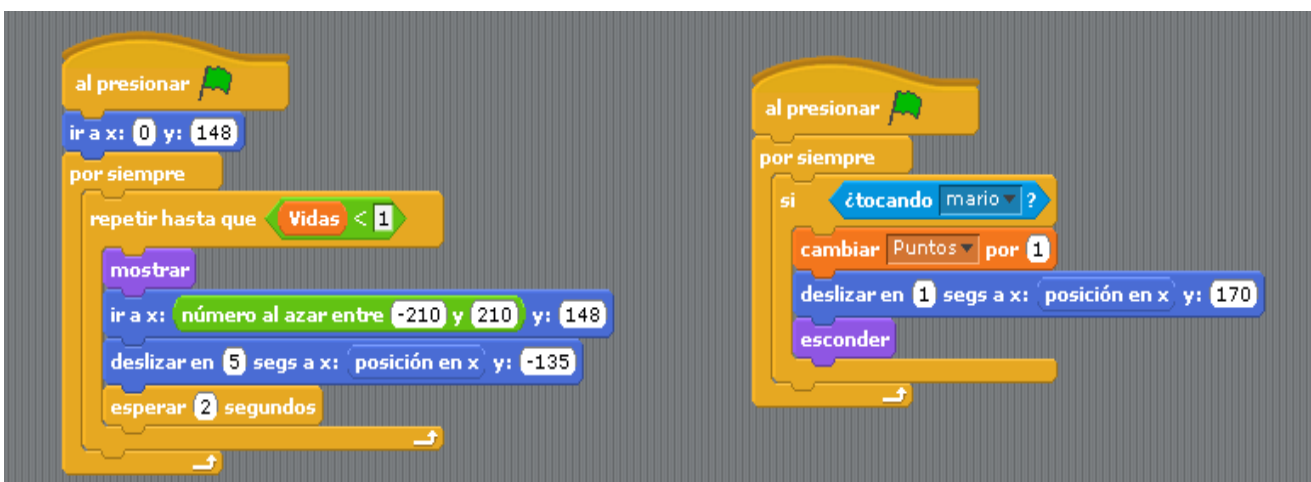
Escenario



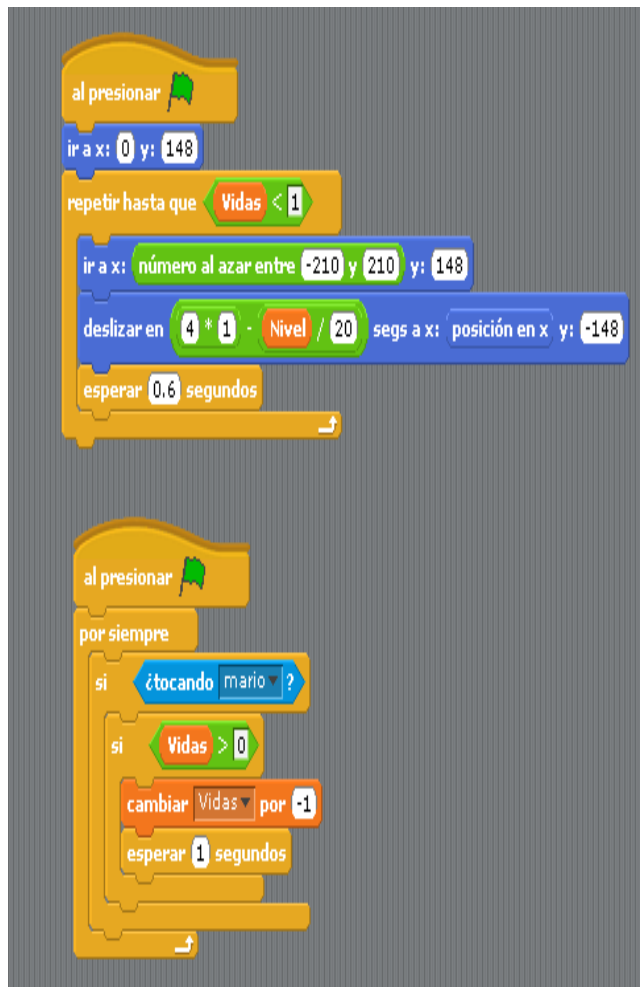
Mario



Moneda



Máscara



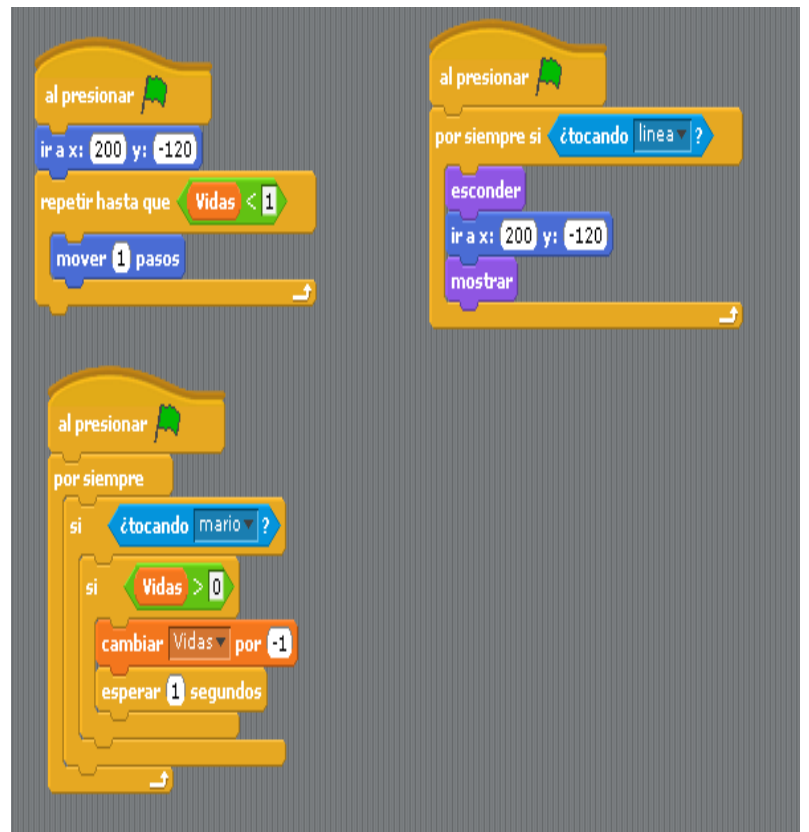
Roca



Seta



Bala

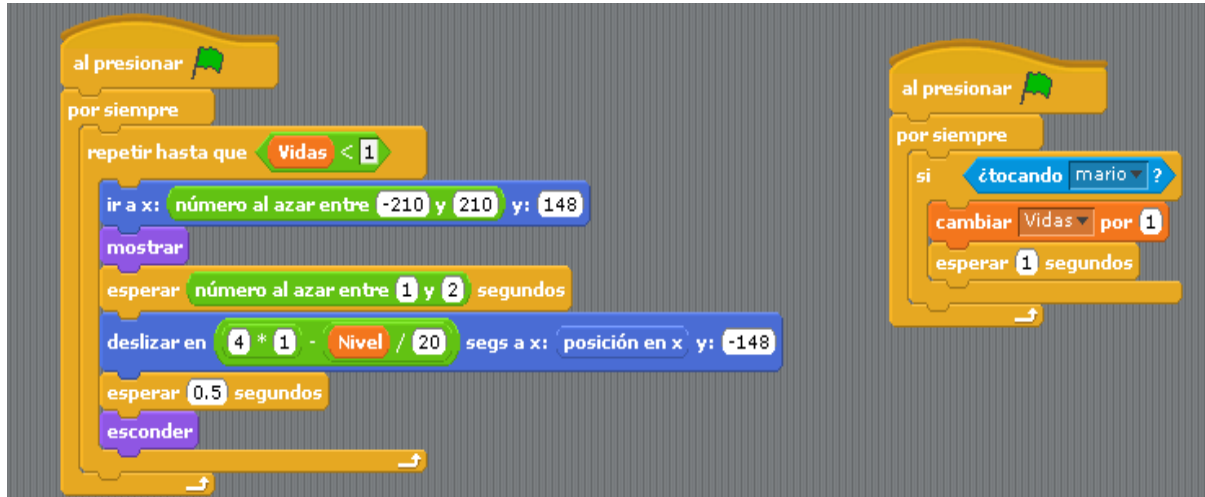


PRÁCTICA 13

En esta práctica crearemos un nuevo objeto que le **sume** una vida a Mario cada vez que se toquen.

- 1) Empezamos importando el objeto **hongo verde** y lo editamos para quitarle el fondo blanco.
- 5) Programa el objeto para que se deslice desde la parte superior y que se espere un tiempo en la posición de Mario. esto se repetirá mientras Mario tenga vidas.
- 6) Por último hacer que cuando toque a Mario le suma una vida.

Hongo verde

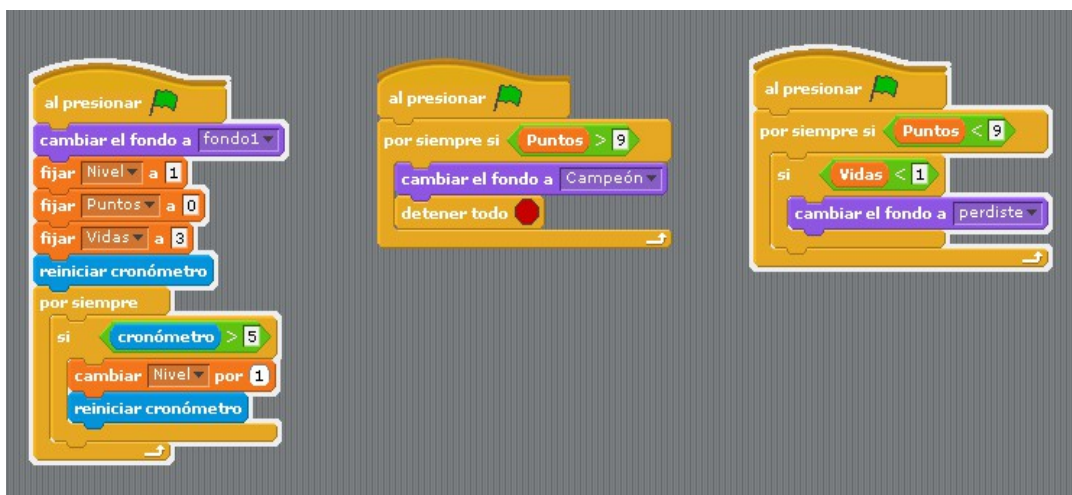


PRÁCTICA 14

En esta práctica vamos programar para que también podamos **ganar la partida**. Queremos que **al llegar a 10 puntos** la partida se acabe y salga un escenario que diga **campeón**. Si antes de llegar a los diez puntos se acaban las vidas la partida se perderá.

- 1) Primeramente seleccionamos el escenario e importamos el archivo Campeón que será el que saldrá al llegar a los 10 puntos.
- 2) Programar también para que salga el escenario perdiste si se acaban las vidas.

Escenario



PRÁCTICA 15

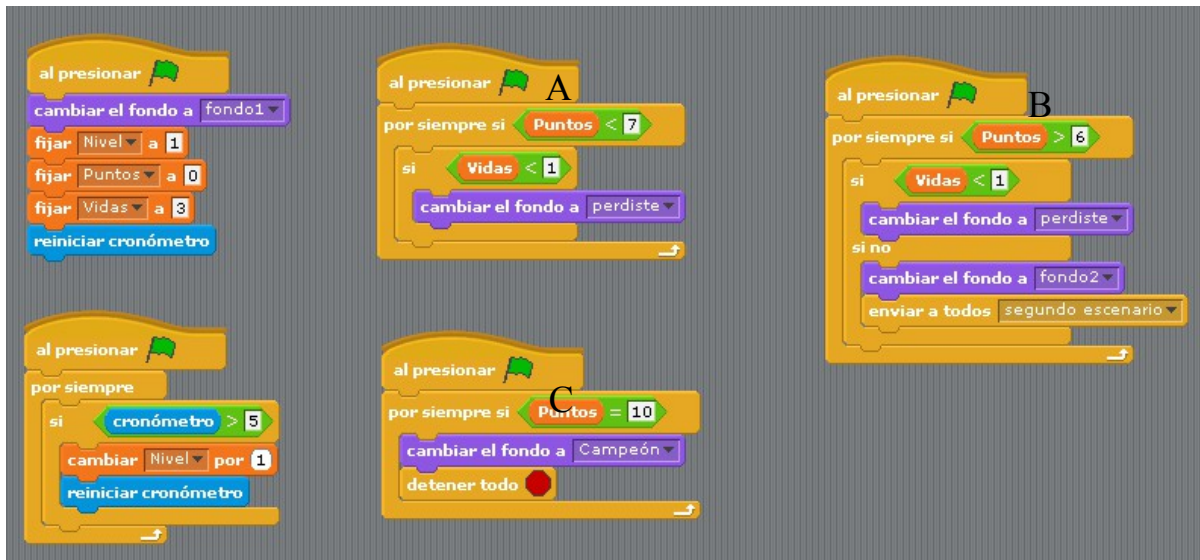
Como es difícil ganar la partida vamos a facilitar un poco las cosas. Queremos que haya 2 fondos diferentes y que en el primero sea más fácil conseguir los puntos y eso lo haremos haciendo que dos de los enemigos no salgan al principio. Al llegar a 7 puntos cambiará el escenario y aparecerán los dos enemigos.

- 1) Para que en el fondo 1 no salgan ni la seta ni la roca se les pone la orden **esconder**.
- 2) Ahora vamos a utilizar dos ordenes nuevas, que son, **enviar a todos** y **al recibir**.

- 3) Nos metemos en el escenario y programamos para que al llegar a los 7 puntos salga el segundo escenario y que **envíe** un mensaje a **todos** los objetos llamado **segundo escenario**.
- 4) Después se programa los objetos roca y seta para que **al recibir** el mensaje **segundo escenario** se muestren.
- 5) Lo último que queda es programar para que no haya dos ordenes contradictorias:
Lo que queremos es:

- A) Empezar con el fondo1 y seguir hasta llegar a 7 puntos. Si se acaban las vidas sale el fondo perdiste.
- B) Si llegamos a 7 puntos sale el fondo2 pero si se acaban las vidas sale el fondo perdiste.
- C) Si los puntos llegan a 10 saldrá el fondo campeón.

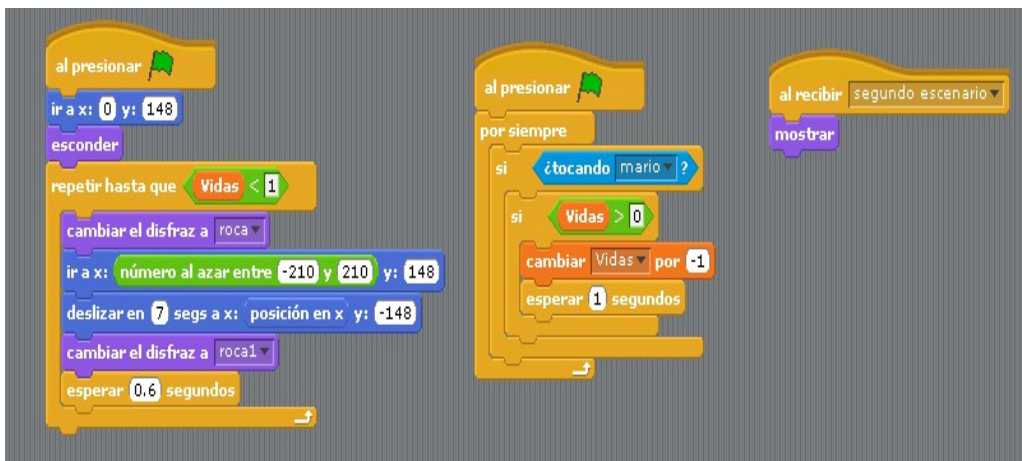
Escenario



Seta



Roca



PRÁCTICA 16

El programa ya está casi terminado, sólo hace falta preparar el comienzo de la partida. Pondremos una portada que salga al presionar la bandera verde, que dure unos segundos y que aparezca una cuenta atrás. Además mandaremos un mensaje a todos los objetos para que empiecen a moverse.

1) Primero insertamos el archivo **Portada** como un nuevo objeto y los editamos para que ocupe todo el escenario. Después lo volvemos a editar y escribimos la información. Por último lo programamos para que se muestre durante unos segundos y que mande el **mensaje empezar**.

2) Creamos (pintar un objeto nuevo) un objeto y dibujamos el número 3 y lo llamamos **Tiempo**. Vamos a los disfraces y lo copiamos dos veces más. Los modificamos con los números 2 y 1. Después lo programamos para que empiece la cuenta atrás.

3) Los dos objetos tienen que desaparecer después de la cuenta atrás.

Portada



Tiempo



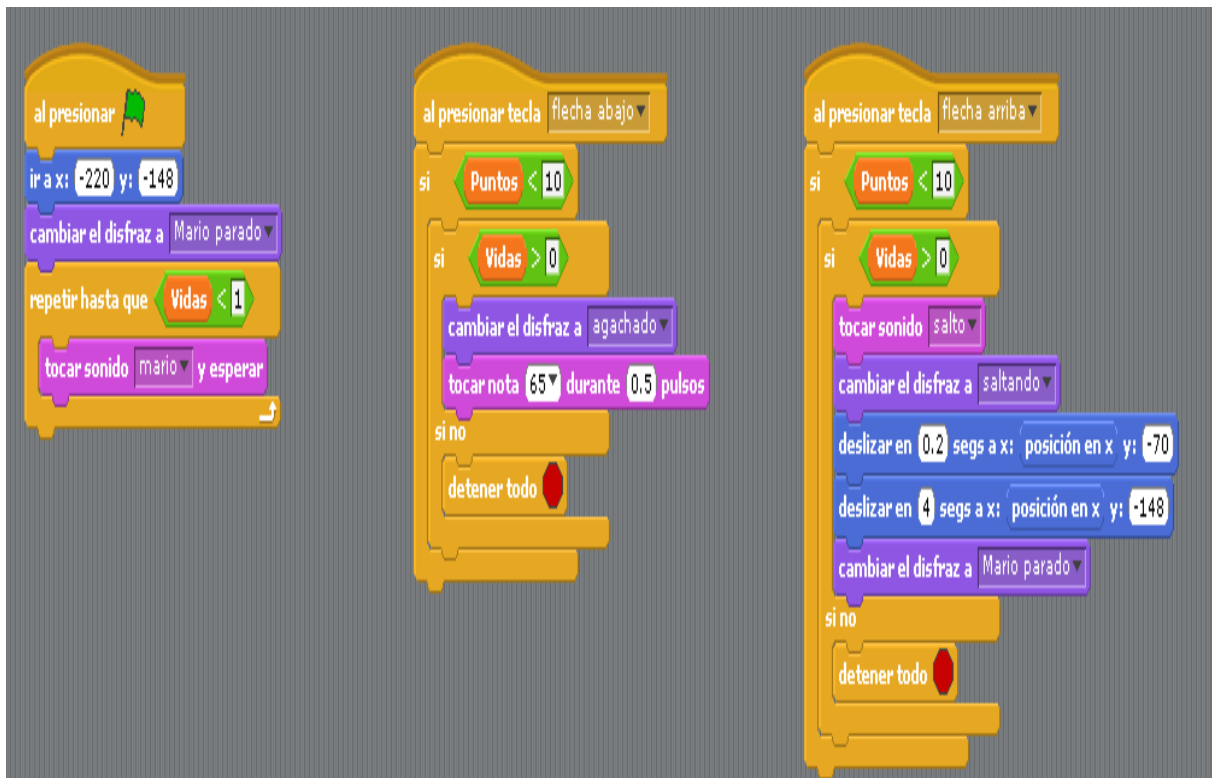
3) Para que la partida empiece de forma automática hay que quitarles a todos los objetos la orden de que empiecen a moverse al presionar la bandera verde y ponerles la orden de que **empiecen** cuando **reciban** el mensaje **empezar**.



PRÁCTICA 17

Si has practicado un poco verás que cuando se gana la partida Mario sigue moviéndose, esto es debido a que en la práctica 12 programamos para que se moviera siempre que tuviera vidas (si $\text{vidas} > 0$) y cuando se llega a 10 puntos y se gana la partida siempre hay vidas. Cuando se pierde no hay problemas porque no le quedan vidas y por tanto no se mueve.

En esta última práctica programaremos para que **Mario deje de moverse al ganar**.




```

al presionar tecla flecha derecha
si Puntos < 10
si Vidas > 0
apuntar en dirección 90
cambiar el disfraz a andando1
mover 10 pasos
esperar 0.1 segundos
cambiar el disfraz a andando2
mover 25 pasos
esperar 0.1 segundos
si no
detener todo

```

```

al presionar tecla flecha izquierda
si Puntos < 10
si Vidas > 0
apuntar en dirección -90
cambiar el disfraz a andando1
mover 25 pasos
esperar 0.1 segundos
cambiar el disfraz a andando2
mover 25 pasos
esperar 0.1 segundos
si no
detener todo

```

```

al presionar tecla flecha izquierda
si ¿tecla espacio presionada?
si Puntos < 10
si Vidas > 0
tocar sonido salto
cambiar el disfraz a saltando
deslizar en 0.3 segs a x: posición en x - 40 y: -70
deslizar en 0.3 segs a x: posición en x - 40 y: -148
si no
detener todo

```

```

al presionar tecla flecha derecha
si ¿tecla espacio presionada?
si Puntos < 10
si Vidas > 0
tocar sonido salto
cambiar el disfraz a saltando
deslizar en 0.3 segs a x: posición en x + 40 y: -70
deslizar en 0.3 segs a x: posición en x + 40 y: -148
si no
detener todo

```