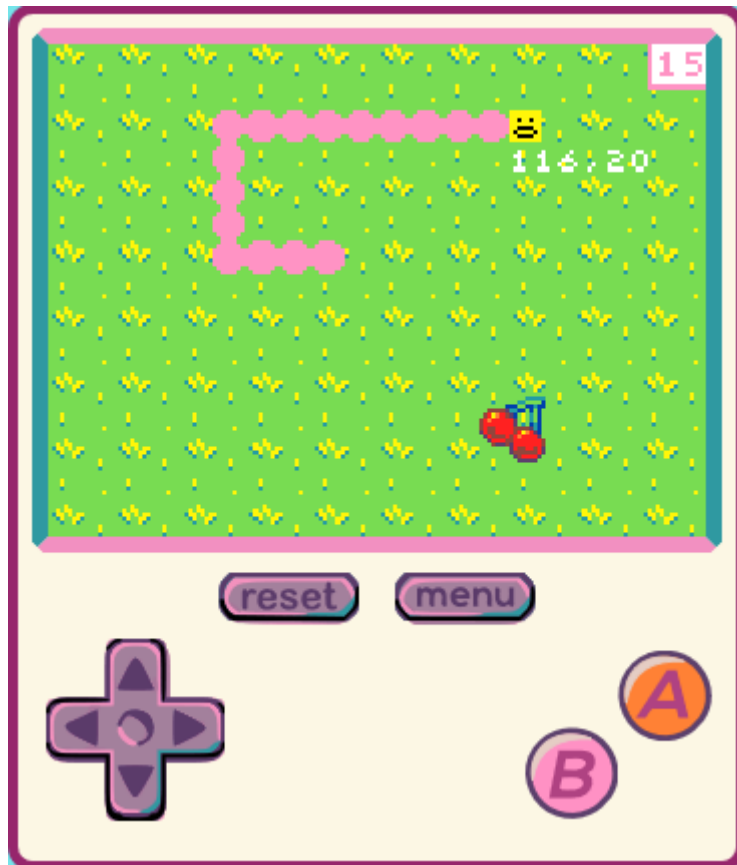


## Worm - Blocks – Microsoft Arcade

The following instructions will take you through the steps of creating a game where you control a worm trying to eat fruit to grow as long as possible. Each time the worm eats a piece of fruit it grows longer. The game ends when head of the worm hits one of the sides of the screen or crashes into its own body.



### Controls

To control the worm, use the direction keys on the console. Alternatively, it may be easier to use the keyboard mappings for the keys as follows

	<b>Up</b> (↑ or W)			
<b>Left</b> (← or A)		<b>Right</b> (→ or D)	<b>B</b> (E, X or Enter)	<b>A</b> (Q, Z or Space)
	<b>Down</b> (↓ or S)			

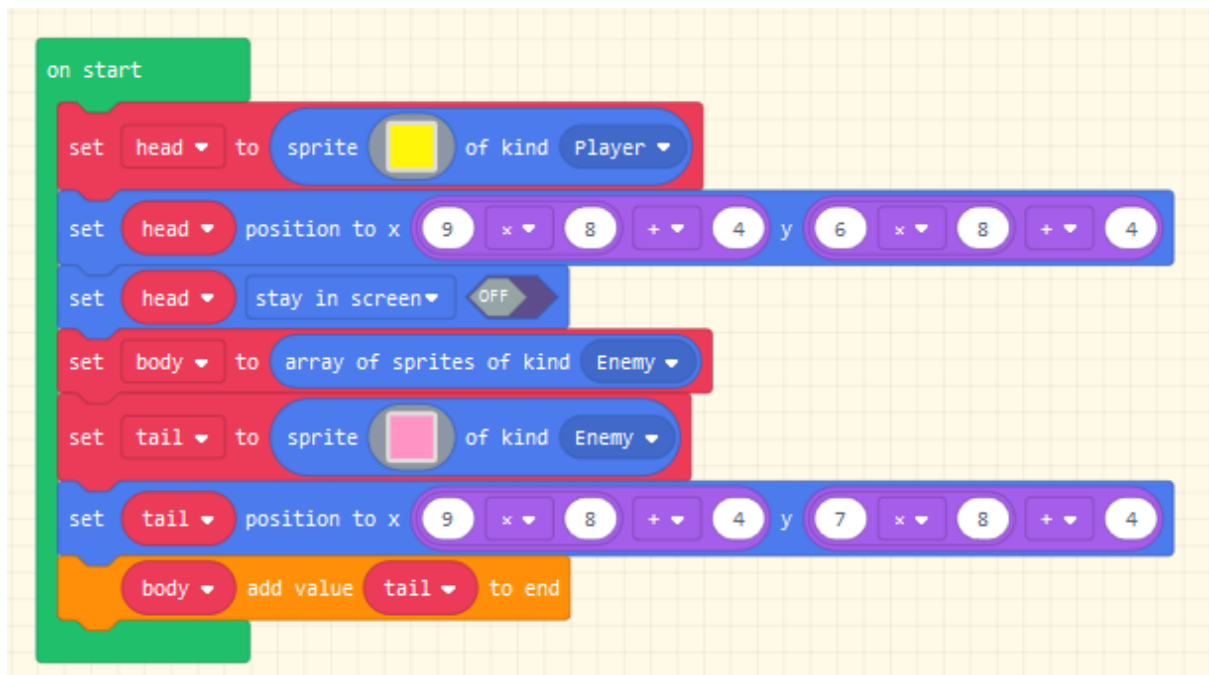
## Worm - Blocks – Microsoft Arcade

### Step 1 – Create the worm

The first step is to create the worm that will move around the screen. The worm will consist of two part, the head and the body. The body will be an array which is like a list that allows us to grow the length of the body by adding new body segments to it. For this first step we will need 3 variables:

- Head sprite
- Body array
- Tail variable

The sprites for the head and tail should be of size 8x8 pixels. Don't worry about what they look like at this stage as we will animate them later.



---

### **Experiment**

*Run you game and you should see the worm in the middle of the screen.*

---

## Worm - Blocks – Microsoft Arcade

### Step 2 – Moving the worm

The movement of the worm will be controlled by the buttons that make up the direction keys. Because we will be manually controlling the movement of the worm, we will need to keep track of what direction the worm is moving in. For this we will need to create a new variable called direction and use the following numeric values to indicate the direction of movement.

0. Up
1. Right
2. Down
3. Left

Add the variables initial value to the “on start” code.

```
on start
  set head to sprite of kind Player
  set head position to x 9 x 8 + 4 y 6 x 8 + 4
  set head stay in screen OFF
  set body to array of sprites of kind Enemy
  set tail to sprite of kind Enemy
  set tail position to x 9 x 8 + 4 y 7 x 8 + 4
  body add value tail to end
  set direction to 0
```

New variable goes here

---

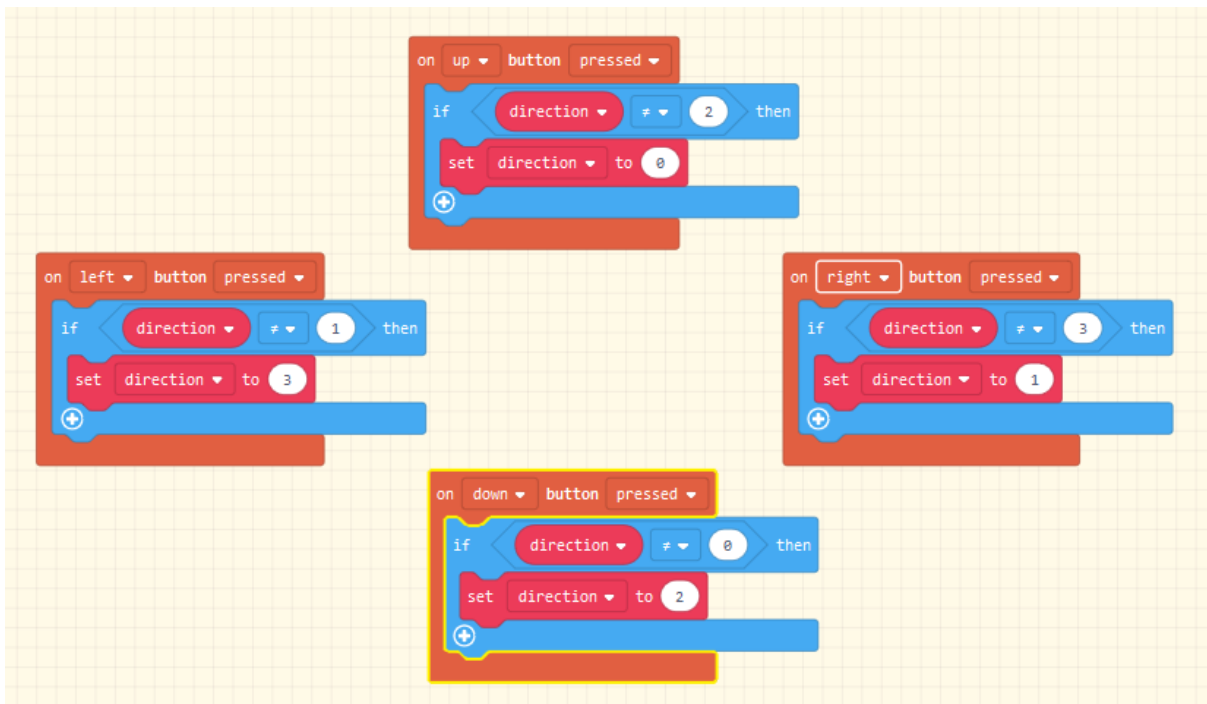
### Experiment

Run your game after you've added all the movement code to make sure that it works. Try going off the edge of the screen; what happens? Can you think of a way to fix it?

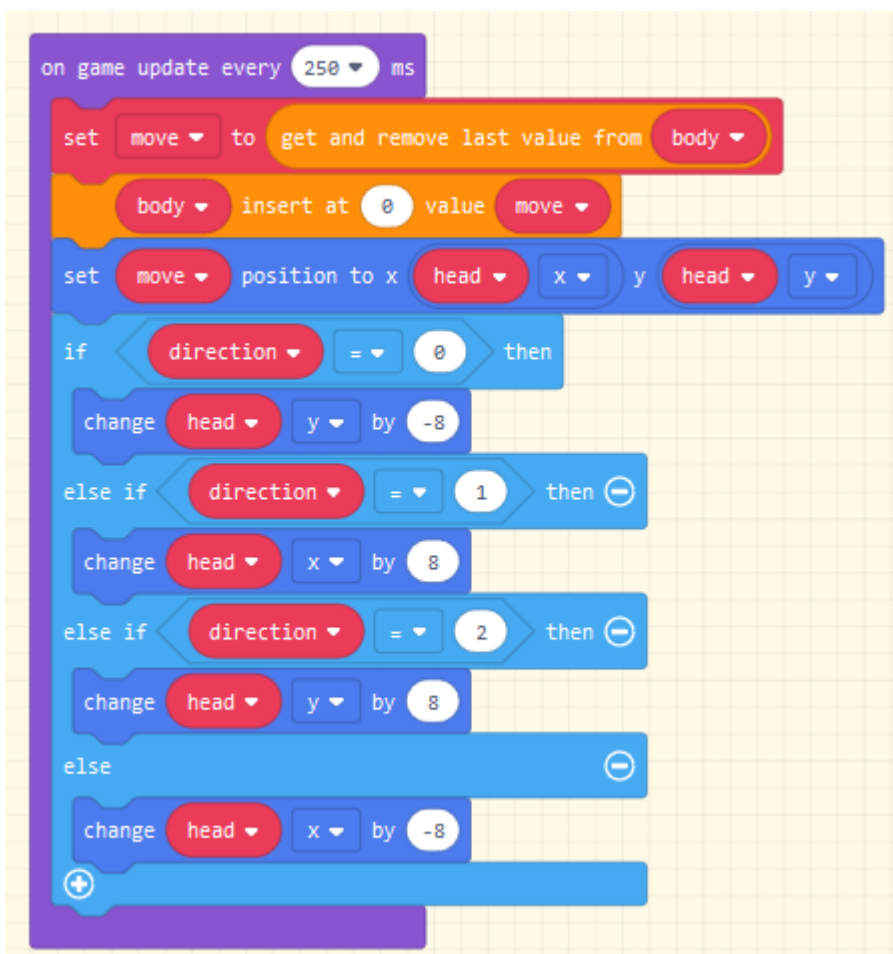
---

## Worm - Blocks – Microsoft Arcade

Then add the following code to control the direction of the worm.



Add the following code which moves the body of the worm. A new variable move will be needed.



## Worm - Blocks – Microsoft Arcade

### Step 3 – Eating food and growing

For the worm to grow, it will need food to eat. We randomly place on the screen an item of fruit which when the worm eats, it will grow. We start by creating a new food sprite as shown below.

```
on start
  set head to sprite of kind Player
  set head position to x 9 * 8 + 4 y 6 * 8 + 4
  set head stay in screen OFF
  set body to array of sprites of kind Enemy
  set tail to sprite of kind Enemy
  set tail position to x 9 * 8 + 4 y 7 * 8 + 4
  body add value tail to end
  set direction to 0
  set food to sprite of kind Food
  set food position to x pick random 8 to 156 y pick random 8 to 116
```

The following code grows the worm when it eats the food. A new variable copy will need to be created which is used to place the new body part in the worm.

```
on sprite of kind Player overlaps othersprite of kind Food
  change score by 1
  set othersprite position to x pick random 8 to 156 y pick random 8 to 116
  set copy to body get value at length of array body - 1
  set tail to sprite of kind Enemy
  set tail position to x copy x y copy y
  body add value tail to end
```

## Worm - Blocks – Microsoft Arcade

### Step 4 – Hitting the edges

When you run the game, you will notice that the worm can go off the screen. We need to add the code in the red box below to the game update loop. This checks the position of the worm and ends the game if the head goes out of bounds.

```
on game update every 250 ms
  set move to get and remove last value from body
  body insert at 0 value move
  set move position to x head x y head y
  if direction = 0 then
    change head y by -8
  else if direction = 1 then
    change head x by 8
  else if direction = 2 then
    change head y by 8
  else
    change head x by -8
  if head x < 4 then
    set life to 0
  if head x > 156 then
    set life to 0
  if head y < 4 then
    set life to 0
  if head y > 116 then
    set life to 0
```

## Worm - Blocks – Microsoft Arcade

### Step 5 – Hitting your body

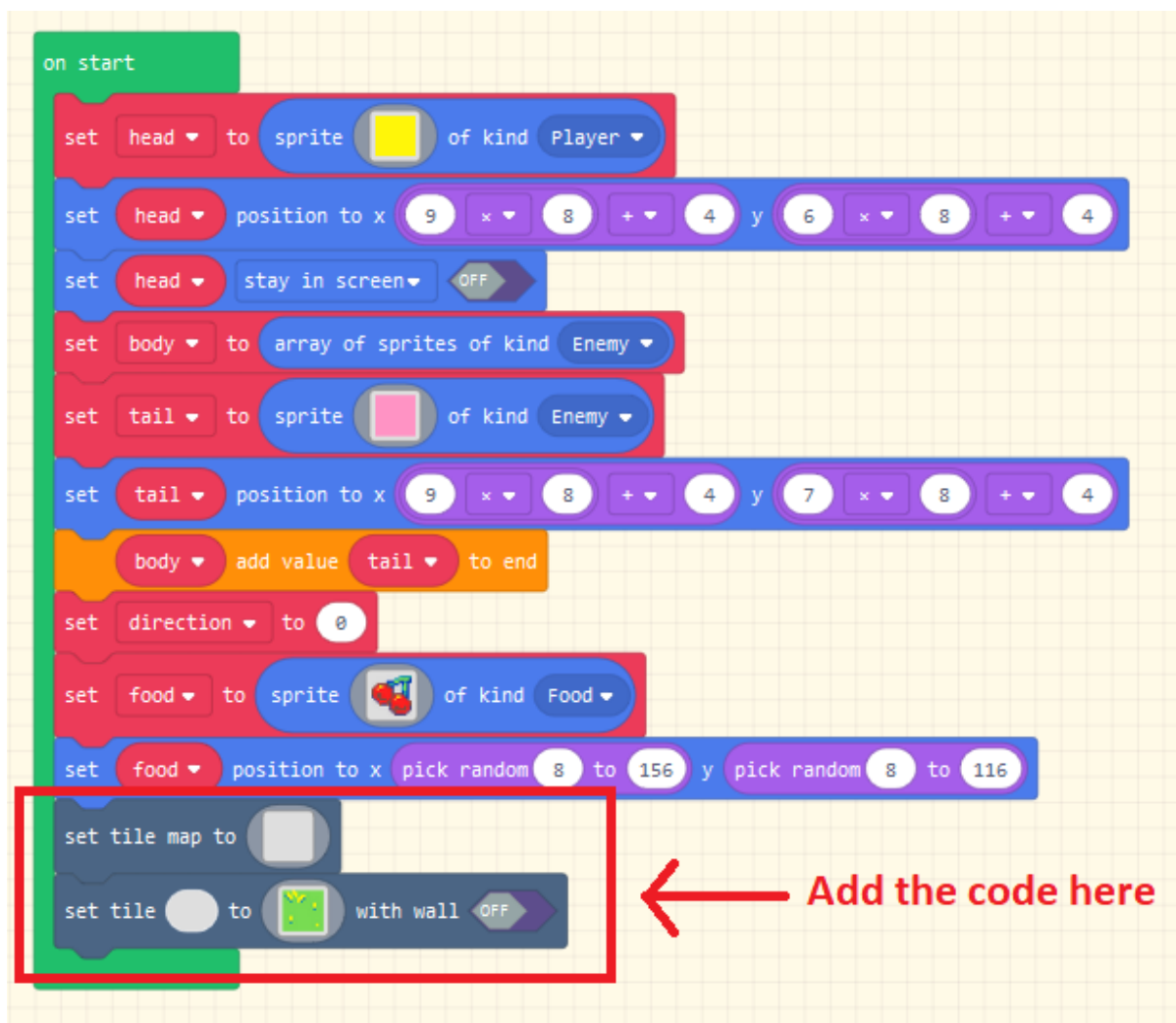
We also want the game to end if the worms head crashes into the body. Add the following code to our game to do this.



```
on sprite of kind Player overlaps otherSprite of kind Enemy
  set life to 0
```

### Step 6 – Setting the background

The game will currently look a little bland so use the tile map to add some colour and interest.



```
on start
  set head to sprite of kind Player
  set head position to x 9 * 8 + 4 y 6 * 8 + 4
  set head stay in screen OFF
  set body to array of sprites of kind Enemy
  set tail to sprite of kind Enemy
  set tail position to x 9 * 8 + 4 y 7 * 8 + 4
  body add value tail to end
  set direction to 0
  set food to sprite of kind Food
  set food position to x pick random 8 to 156 y pick random 8 to 116
  set tile map to
  set tile to with wall OFF
```

← Add the code here

---

### Experiment

Try using more than one tile in your tile map to create complex pictures.

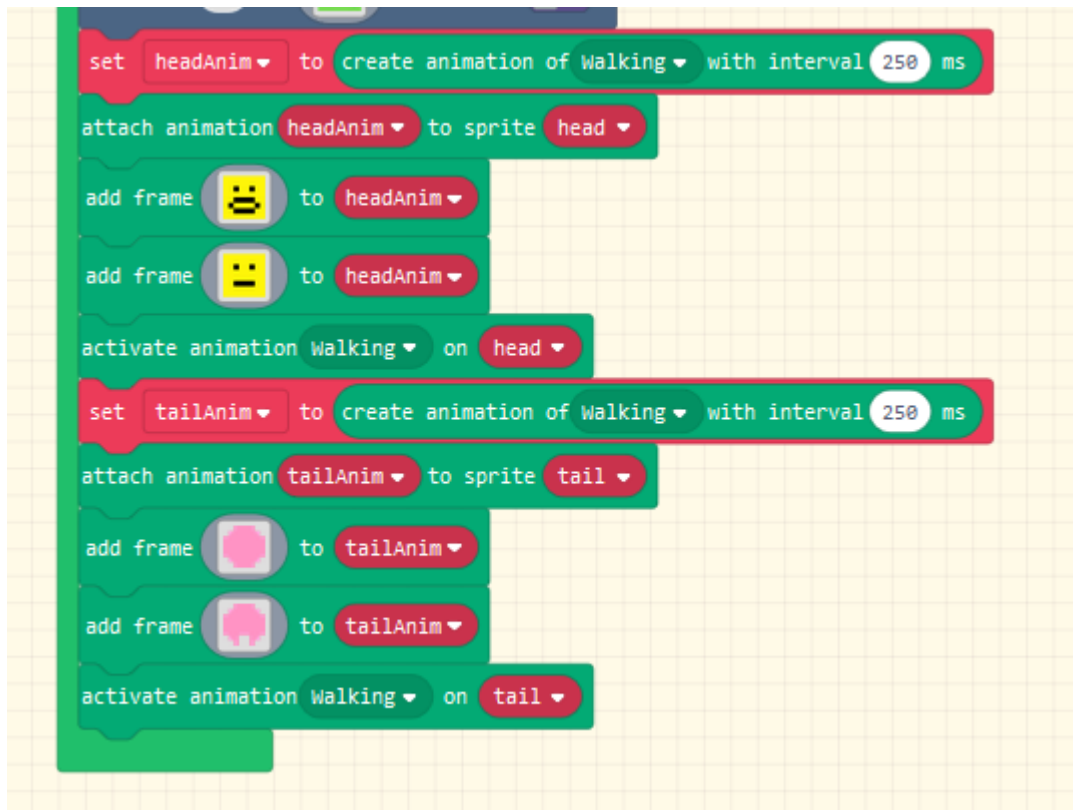
---

## Worm - Blocks – Microsoft Arcade

### Step 6 – Animating the head and body

To make a professional looking game we will want to animate the worms head and body. To do this we will need to add the Animation package to our game.

Once the Animation package has been added, add the following code to the “on start” section of the code. Two new variables will need to be created, headAnim and tailAnim. One animates the head and the other animates the tail and body.



---

### **Experiment**

Run you game. What happens after the worm eats a piece of fruit?

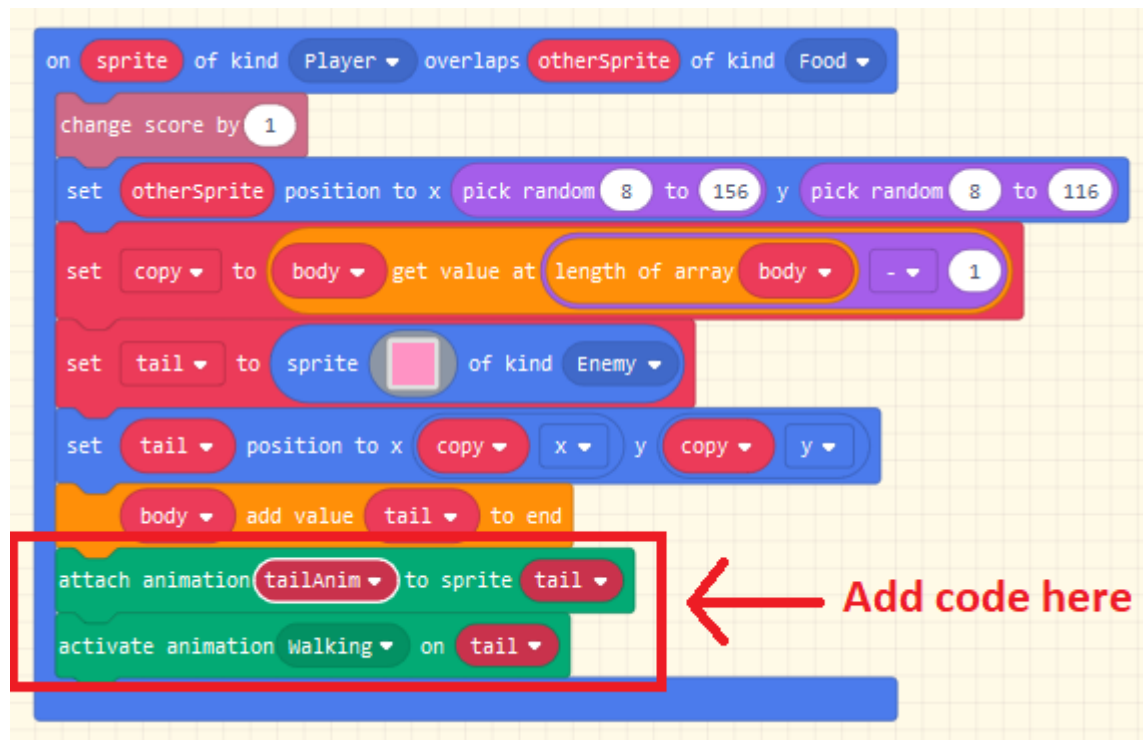
Why do you think this happens?

---



## Worm - Blocks – Microsoft Arcade

As the worm grows, the newly added body parts don't automatically animate. We need to attach the animation to the newly created sprites. We do this by adding the code below to the code that adds the body parts when the worm eats some food.



---

### Experiment

Run you game and you should now see the whole worm animated even after it eats food.

---

### Extending the game

There are many ways that this game can be extended. Just a few ideas are given below.

- Make the worm get faster as the players score increases.
- Increase the number of food items on the screen.
- Randomly position obstacles on the screen to avoid
- Add in extra sound effects to the game. Maybe play a melody on the home screen or something when the players game ends.