

# **End of Program and Recap**

## **Coding & STEAM 2019**

**Mr Daniel Hickmott & Dr Elena Prieto-Rodriguez**

**Week 8: Coding and Mathematics Part 2**

**19th September 2019**

## End of Program 🙄

- This is the last week of the **Coding & STEAM** program
- Before we start today, I am going to go over a few things about the research part of the program
- Like Week 1, we will ask you to complete a survey
- After this presentation and the survey, we will continue with some activities
- I can stay around for 30-60 mins at the end

# Research

- This Professional Learning (PL) program is part of a research project
- The purpose is to help understand how teachers can be supported in learning and teaching Coding and Computational Thinking
- We hope this research helps other PL providers implement PL that is useful and relevant to teachers
- Please take 10-15 minutes to complete the End of Program Survey linked to on the session page

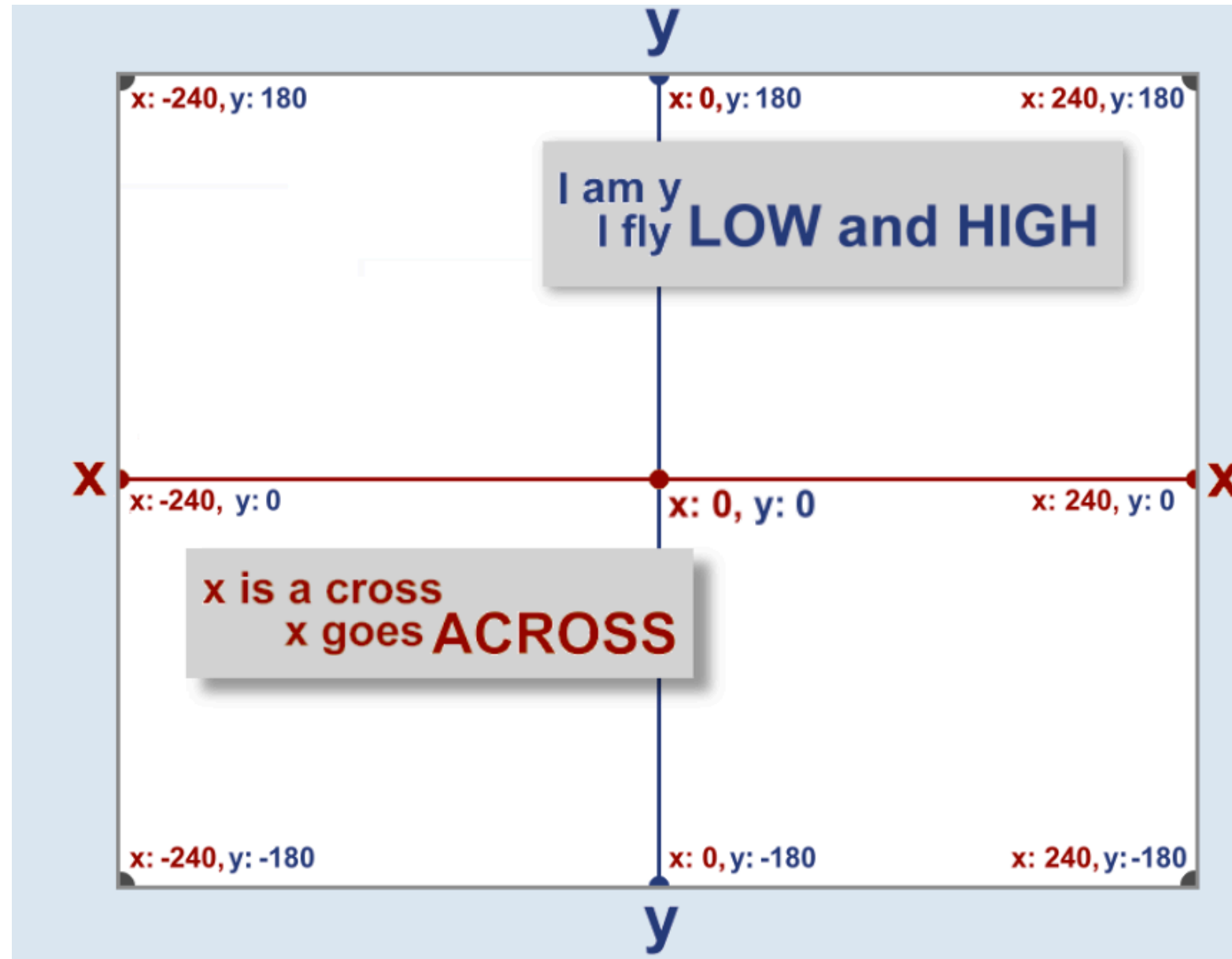
# Homework & Accreditation

- If you have completed all of the tasks and attended/caught up all of the sessions, I will ask the office staff to process your accreditation hours asap
- I would like to finalise the hours and accreditation by the end of the school holidays (October 11th)
- We will send out certificates around October 11th as well, please let me know if you need the certificate before then

# Coding and Mathematics

- Last week we started the **Games** unit of the Creative Computing Guide
- Also looked at some ways that **Game Design** can be linked to the teaching of Mathematics
- Coding, the design of games and the teaching of Mathematics have been intertwined since the first educational Coding languages (for example, LOGO)
- What **mathematical concepts** and **computational concepts** were in the Week 7 activities?

# Motion (Coordinates & Geometry)

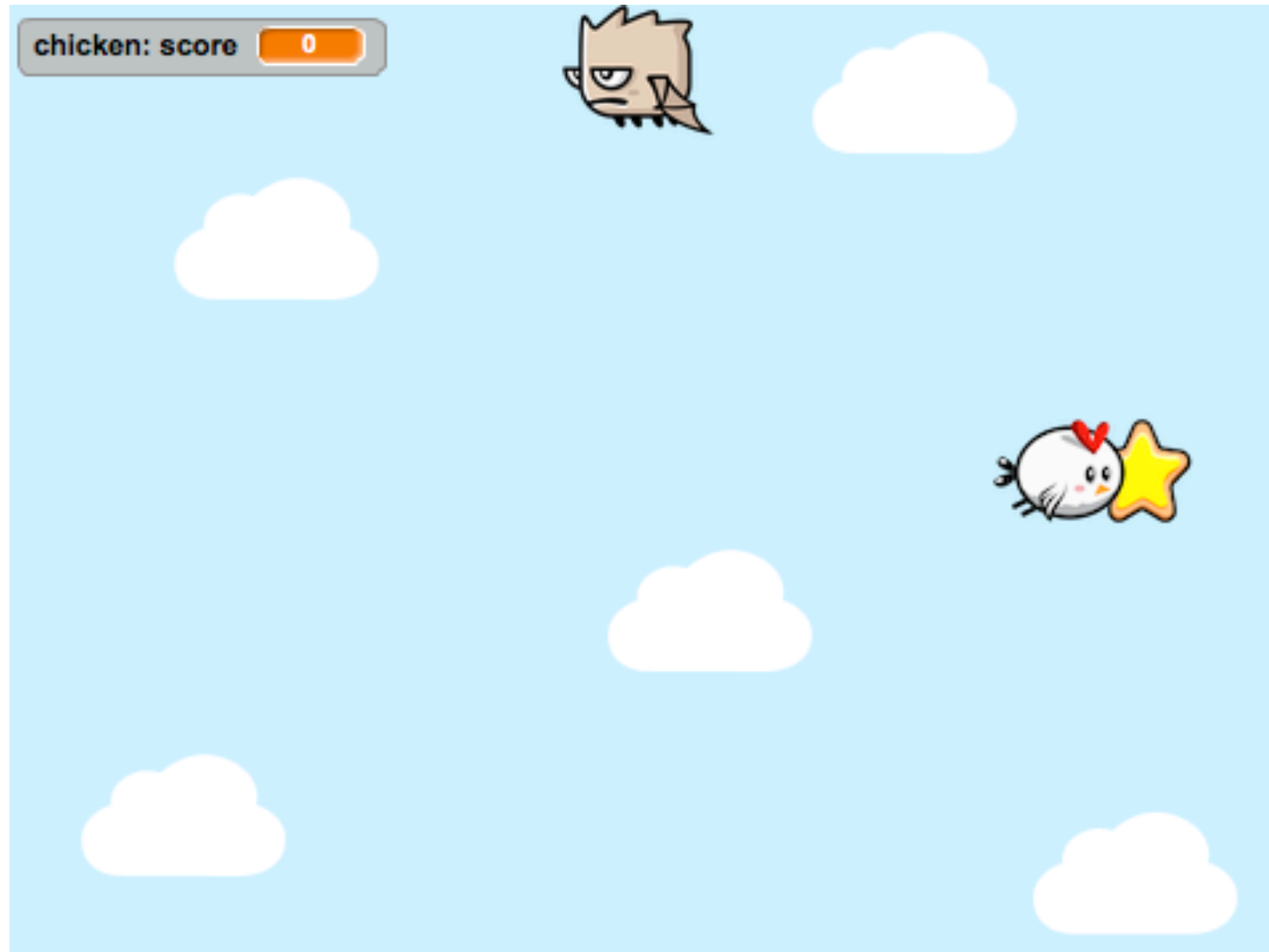


# Motion (Coordinates & Geometry): Maze

The image displays a Scratch project interface and its corresponding code blocks. On the left, the 'Sprite' panel shows a 'Ball' sprite with a size of 100 and a 'Goal' sprite. The code blocks are organized as follows:

- When clicked:** A yellow 'when clicked' block is followed by a blue 'go to x: -205 y: 147' block.
- When up arrow key pressed:** A yellow 'when up arrow key pressed' block is followed by a blue 'point in direction 0' block and a blue 'move 10 steps' block.
- When left arrow key pressed:** A yellow 'when left arrow key pressed' block is followed by a blue 'point in direction -90' block and a blue 'move 10 steps' block.
- When down arrow key pressed:** A yellow 'when down arrow key pressed' block is followed by a blue 'point in direction 180' block and a blue 'move 10 steps' block.
- When right arrow key pressed:** A yellow 'when right arrow key pressed' block is followed by a blue 'point in direction 90' block and a blue 'move 10 steps' block.

# Scores (Formulas & Variables)





# Scores (Formulas & Variables): Fish Chomp

The image displays Scratch code blocks for a game titled "Fish Chomp". The code is organized into three main sections:

- Score Initialization:** A "when clicked" event block triggers a "set score to 0" block.
- Score Update:** A "when I receive got-me" event block triggers a "start sound chomp" block, followed by a "change score by 1" block.
- Costume Animation:** A "repeat 2" loop contains:
  - "switch costume to fish - closed"
  - "wait 0.3 seconds"
  - "switch costume to fish - open"

The visual interface shows two fish costumes: "hungry fish" (a yellow fish) and "little fish" (a purple fish). The "little fish" costume has a trash can icon, indicating it is a removable object.

The movement logic (shown in a separate block) includes:

- "when clicked" event
- "show" block
- "point in direction 90" block
- "forever" loop:
  - "move 2 steps" block
  - "turn pick random -20 to 20 degrees" block
  - "if on edge, bounce" block
  - "if color purple is touching yellow?" block, which triggers a "broadcast got-me" block when true.
  - "hide" block
  - "wait 3 seconds" block
  - "go to x: -200 y: pick random -200 to 200" block
  - "show" block

# Randomness (Probability & Statistics)



```
when clicked
  forever
    glide 2 secs to x: pick random -240 to 240 y: pick random -180 to 180
```

# Randomness (Probability & Statistics)

The image displays a Scratch script and a sprite's properties panel. The script, starting with a 'when green flag clicked' event, enters a 'forever' loop. Inside the loop, an 'if touching Paddle?' condition triggers a sequence of actions: starting a 'water\_drop' sound, turning the sprite by a random angle between 160 and 200 degrees, and moving it 15 steps. The properties panel for the 'Ball' sprite shows it is visible and has a size of 120. A 'Paddle' sprite is also visible in the background.

# Computational Concepts

- The activities last week involved using some different **Computational Concepts**, such as:
  - Parallelism
  - Conditionals
  - Operators
  - Data

# Computational Concept: Parallelism (Fish Chomp)

The image displays Scratch code blocks for a 'Fish Chomp' game, illustrating parallelism through multiple instances of fish objects.

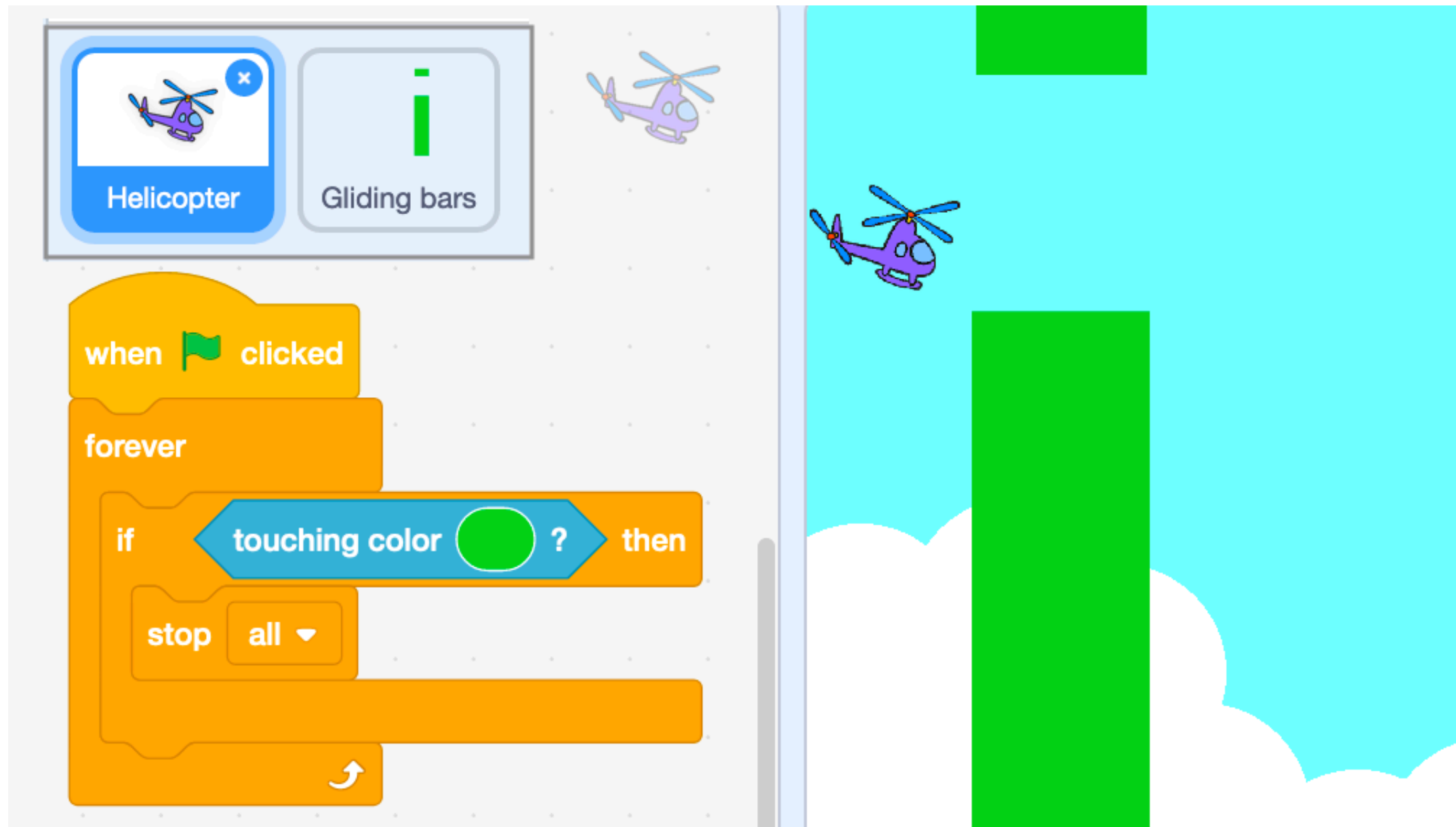
**Left Panel (Hungry Fish Logic):**

- when green flag clicked
- switch costume to fish - open
- forever loop:
  - if distance to mouse-pointer > 10 then:
    - point towards mouse-pointer
    - move 5 steps

**Right Panel (Little Fish Logic):**

- when green flag clicked
- show
- point in direction 90
- forever loop:
  - move 2 steps
  - turn pick random -20 to 20 degrees
  - if on edge, bounce
  - if color purple is touching yellow? then:
    - broadcast got-me
    - hide
    - wait 3 seconds
    - go to x: -200 y: pick random -200 to 200
    - show

# Computational Concept: Conditionals (Scrolling)



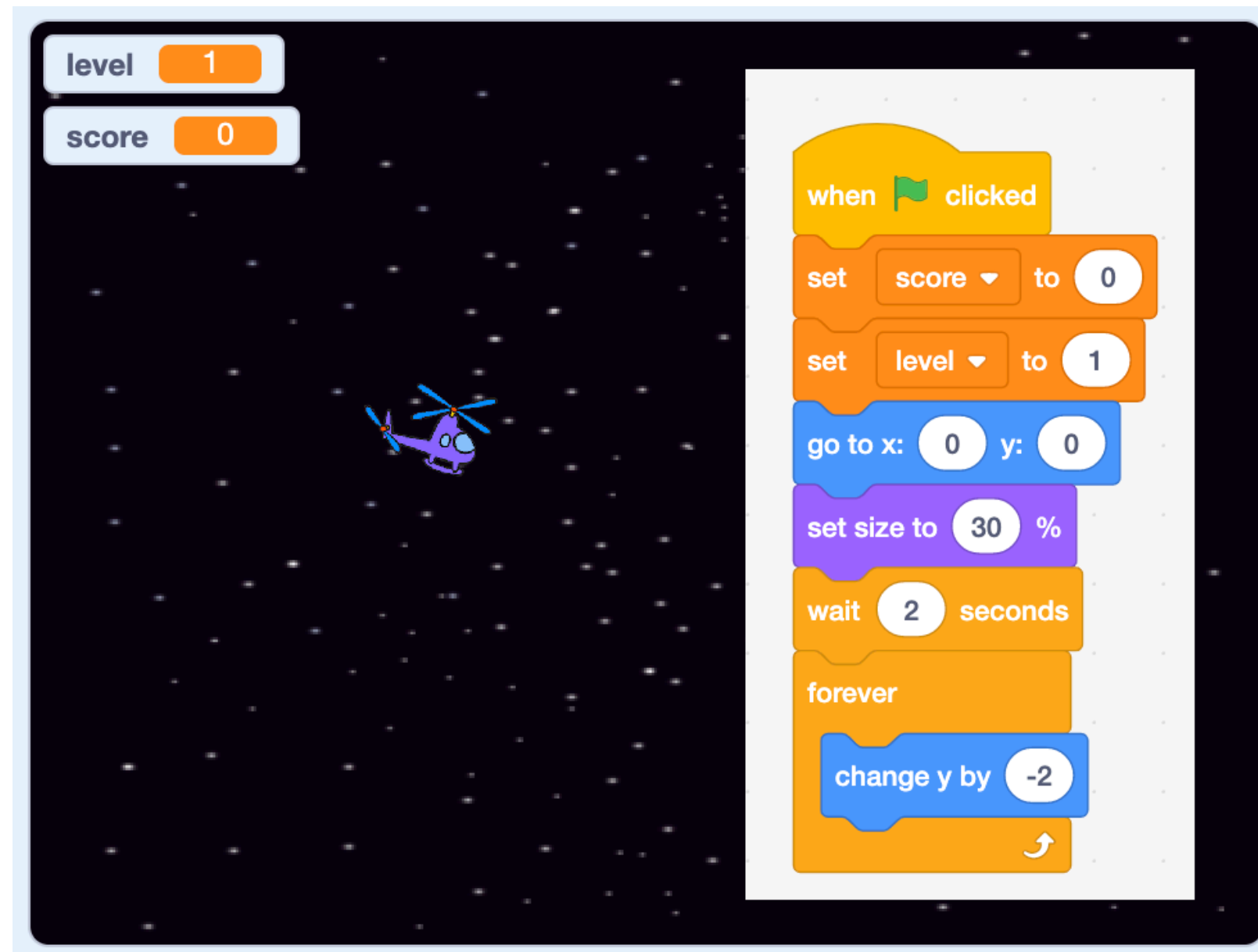
# Computational Concept: Operators (Scrolling)

The image displays the Scratch 'Costumes' tab. On the left, three costumes are listed: 'costume1' (254 x 387), 'costume2' (254 x 396), and 'costume3' (254 x 387). The 'Costumes' tab is active, and the 'Code' tab is also visible. The central script area contains the following code blocks:

```
when I start as a clone  
switch costume to pick random 1 to 3  
go to x: 240 y: 0  
show  
glide 8 secs to x: -240 y: 0  
change score by 1  
delete this clone
```

On the right, two costume thumbnails are shown: 'Helicopter' and 'Gliding bars'.

# Computational Concept: Data



The image shows a Scratch project window. On the left, a stage with a starry background contains a purple helicopter. In the top-left corner, there are two monitors: 'level' with the value '1' and 'score' with the value '0'. On the right, the script area contains the following code blocks:

- when green flag clicked
- set score to 0
- set level to 1
- go to x: 0 y: 0
- set size to 30 %
- wait 2 seconds
- forever loop containing:
  - change y by -2