Session 2: Fish game

To work out the rules for a simple chase game

Let's learn

Computer games are just like the Scratch programs you have worked on. They are created by programmers, who often work in a team together. The programmers have to think really carefully about the **algorithm** they want to use and what they want to happen in their game.

Once the programmers are clear about the steps or rules of their algorithm, they write a **program**, in code, based on the algorithm that the player can then play.

Do you remember what an **algorithm** and a **program** are?

Click on this box to see the definitions.

Let's discuss

Other than a programmer, who else do you think is involved in making a computer game?

Let's do

Make a list of the other jobs involved in your groups.

Click here for possible answers!

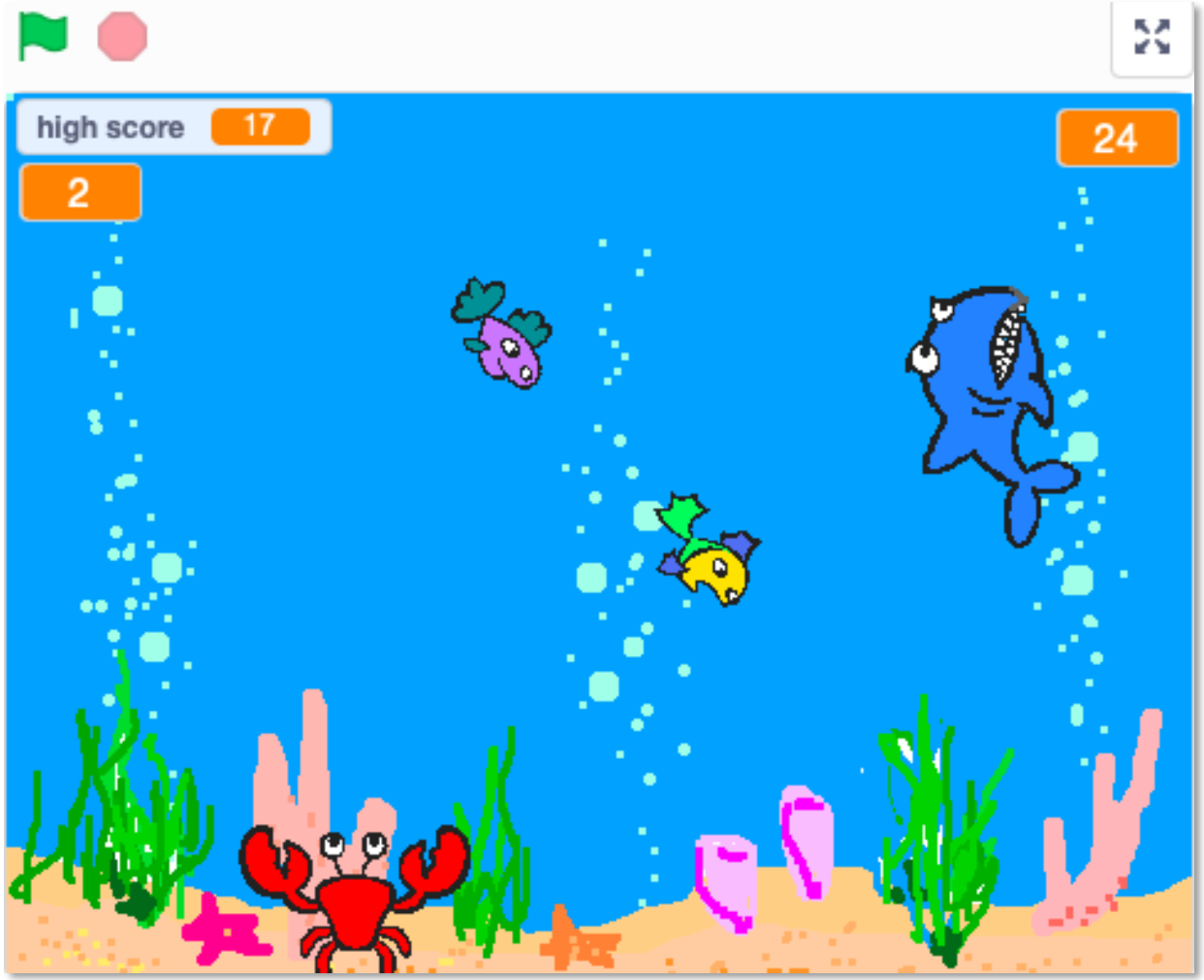
Let's learn

Today we are playing the Fish game. In this game, the shark must catch as many fish as possible in 30 seconds.

- Click the green flag to start the game.
- Use the left and right arrow keys to control the shark.
- Reset the high score by pressing 'R'.

Let's try

1. Can you describe what is happening in the game?
2. Did anything happen differently the second time we played?



Click on the image to open the game!

Let's learn

Play the Fish game a few times in groups. What was the highest score in the class?

Let's discuss

Does it remind you of any games you have played in school or at home?

Let's do

1. Make predictions using **logical reasoning** about how the Fish game works.
2. Test your predictions.
3. Were your predictions right or not?

Do you remember what **logical reasoning** is?
Click on this box to see the definition.

Let's learn

Decomposition is a way of taking a big problem and breaking it down into smaller problems.

Let's do

1. Decompose the game: how many different **sprites** are there?
2. Which of these behave in the same way?

Click here for answers!

Do you remember what a **sprite** is?

Click on this box to see the definition.

Let's do

Use the 'See inside' button to view the **source code**.

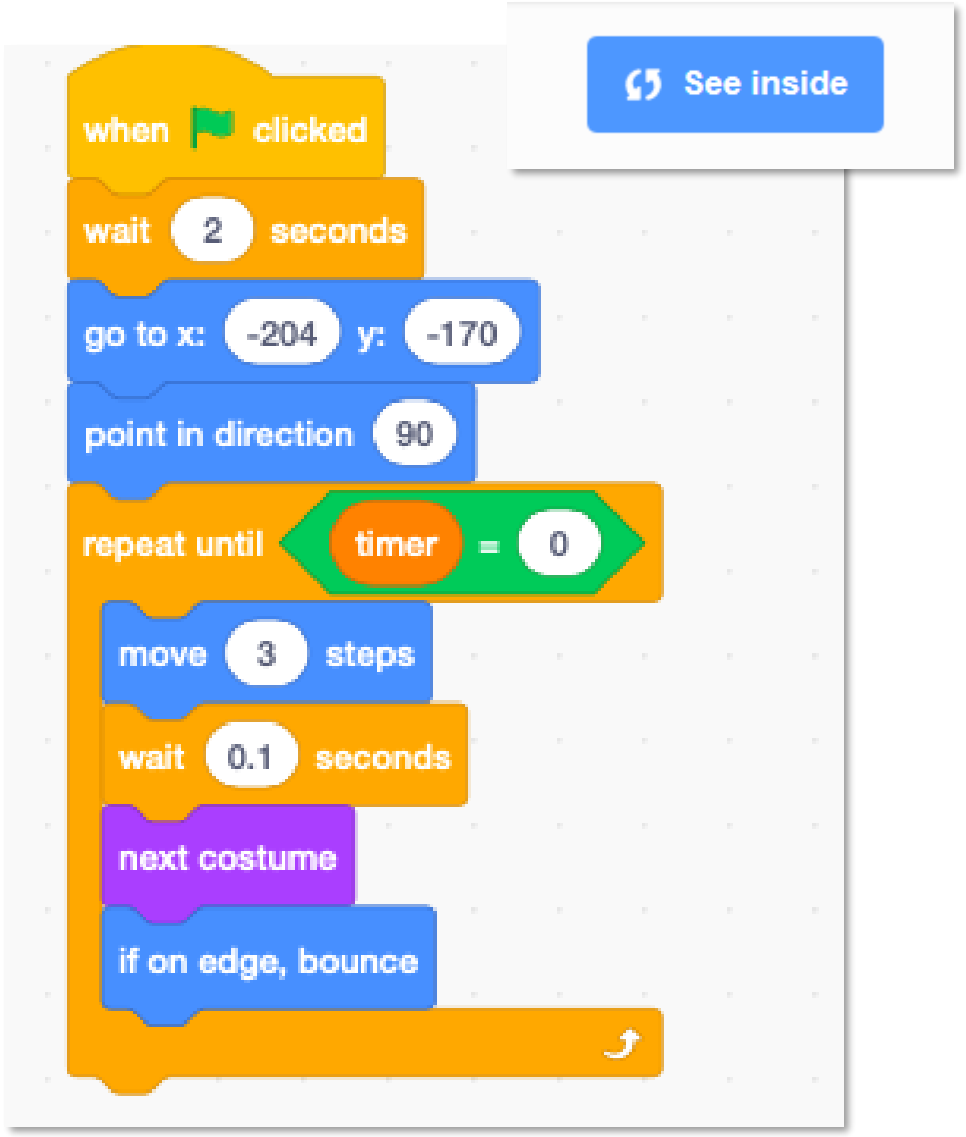
In pairs, work out the algorithm the computer follows:

1. When the arrow keys are pressed
2. To make each of the fish move
3. To decide if a fish gets eaten
4. To open the shark's mouth
5. To keep track of the score and the time
6. To update the high score.

Answers on the next slide!

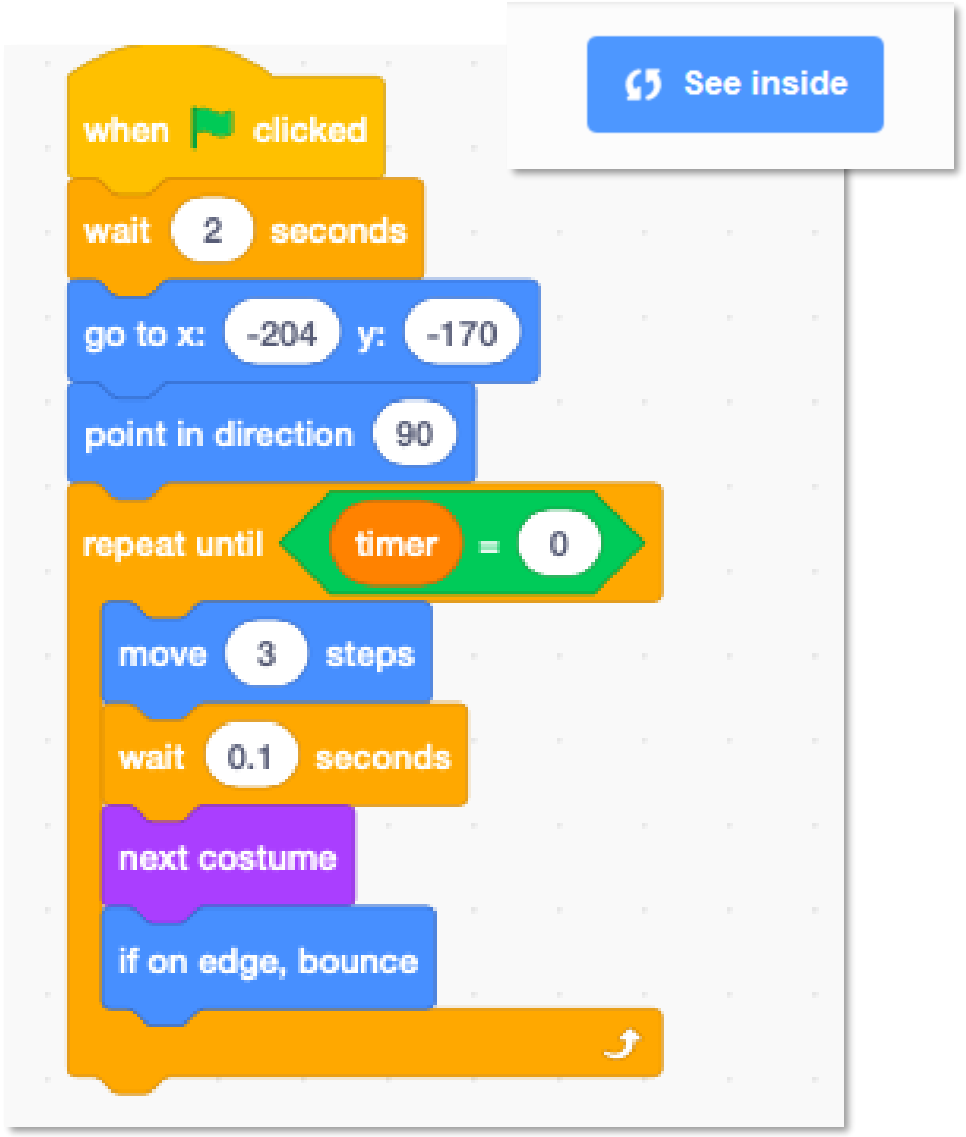
Do you remember what **source code** means?

Click on this box to see the definition.



Let's review

1. The arrow keys make the shark turn that direction.
2. Fish move by moving forward and turning.
3. The computer decides that a fish is eaten if it touches the same colour as the shark. The fish then disappears, 1 is added to the score and then after 1–3 seconds it will reappear.
4. If the shark touches one of the other fish it switches to a costume where its mouth is open.
5. The computer keeps track of the score by increasing it by one every time a fish is eaten.
6. If the score is bigger than the high score then the computer updates the high score.

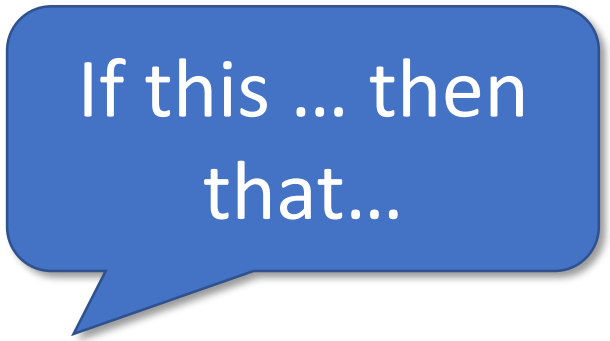


Let's learn

Many modern programs are made up of rules that are followed if certain things happen – we call this event-driven programming.

Let's do

1. Work with a partner to record your observations about the rules that the sprites are following.
2. Can you work out what examples of **input**, **output** and **repetition** there must be in this game? Use your knowledge of ScratchJr space programs from **Unit 2.1: We are astronauts.**



Do you remember what **input, output** and **repetition** mean?

Click on this box to see the definitions.

Let's learn

Let's look at the source code for the game. Can you see any connection between the rules and the programs that the different sprites follow?

Let's do

How does the shark sprite watch for what keys are pressed and how do the fish sprites respond when touching the shark?

Tip: the controlling programs for scores are in the background!