


















Coding with Scratch: Learning Loops: Scratch Revisited

National Curriculum Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.		Lesson Duration This lesson will last approximately 60 minutes.	
Aim To write algorithms using coding blocks in Scratch.			
Success Criteria I can explain what an algorithm is. I can identify different types of coding blocks in Scratch and know where to find them. I can create a sequence of blocks to write an algorithm. I know that it is important to test and debug an algorithm.		Key Vocabulary Algorithm, block code, coding, customise, debug, programming, sequence, test.	
Resources Lesson Pack PC devices, such as laptops, Chromebooks and/or tablets Scratch Online version accessed via _____		Preparation In order to activate prior knowledge and develop a purpose for learning through interests, we recommend you begin the _____ prior to starting this unit. _____ as required Scratch Blocks Quiz Answer Sheets - one per child Writing an Algorithm: Muddled Instructions - one per pair or group (optional) Differentiated Meet and Greet Scratch Task - one per child or per pair It is recommended to set up a 'Teacher's Account' on _____	

Prior Learning: A basic familiarity with using Scratch for simple coding activities would be beneficial. Having previous experience of being able to add, delete and change sprites and backdrops and knowing how to drag coding blocks into the programming area would scaffold children's new learning in this unit.

Learning Sequence

	Remember It: Use the Lesson Presentation to recall prior learning. Share the Scratch interface and remind children of the different areas they will be using. This is a good opportunity to remind children about what coding is and revisit the vocabulary related to coding.	
	Knowledge Organiser: Introduce the topic and Knowledge Organiser . Explain that the Knowledge Organiser has all the key information and vocabulary that the children will need to learn and understand by the end of this unit. Look through this together discussing what they already know and underline words that they don't yet know. We recommend completing one per pair at LKS2.	
	Writing an Algorithm: Teach children about writing an algorithm, using the five steps shown in the Lesson Presentation . Why do you think that there is a double headed arrow between Step 4 and Step 5? Ensure that children understand that testing and debugging is a cyclical procedure and that when writing an algorithm they may have to move between these two steps a number of times. Children are presented with muddled instructions for how a robot might meet and greet a new friend. The same muddled instructions can be found in the Writing an Algorithm: Muddled Instructions resource found within the Lesson Pack . This provides the option for children to physically sort the instructions themselves in a pair or small group. Children can act out the role of robot and friend to test the algorithm. An error has been deliberately included within the instructions so that children have the opportunity to test and debug the code.	
	Coding Blocks in Scratch: Use the quiz activity in the Lesson Presentation to recall the block categories and coding blocks in Scratch. Children can record their answers on the Scratch Blocks Quiz Answer Sheets . What are the advantages of using visual blocks instead of writing lines of code? Children should understand that coding blocks makes coding more visual. It is easier for younger coders as they do not have to learn a programming language. It also reduces the possibility for errors in code which are more likely to occur when writing lines of code.	

 	<p>Algorithms in Scratch: Use the Lesson Presentation to recap the key points of writing algorithms using the coding blocks in Scratch.</p> <p>Why is it useful to be able to customise coding blocks? Children should recognise that coding blocks can be customised by adding text or numbers. By customising coding blocks, algorithms can be very flexible and solve a range of problems more accurately.</p> <p>Ensure that children can access Scratch on a digital device. You can share the link https://scratch.mit.edu/ or children can type 'scratch' into a search engine.</p> <p>Introduce the Meet and Greet Scratch Task to the children and remind them of the five steps they need to follow when writing an algorithm.</p> <div style="display: flex; justify-content: space-around;"> <div data-bbox="215 436 582 548">  <p>Children write the code to make two sprites walk towards each other and greet one another.</p> </div> <div data-bbox="614 436 981 593">  <p>Children write the code for two sprites to walk towards each other, take it in turns to greet one another and then walk past each other.</p> </div> <div data-bbox="1005 436 1372 649">  <p>Children write the code for two sprites to walk towards each other, take it in turns to greet one another, change appearance, say goodbye and then walk past each other.</p> </div> </div>	
	<p>Share It: Invite children to share their projects with one another. This could be done in partners, in small groups or as a class using an interactive board or large screen. Encourage children to use this as a learning opportunity and a chance to see similarities and differences in the code people have used. Ensure that children understand that problems can be solved in multiple ways and that there is no single correct answer.</p>	

Exploreit

Commandit: Instruct children to work with a partner. They take it in turns to be the robot or the controller. The controller should give instructions to move their 'robot' safely around the classroom or playground, avoiding obstacles. They should try to make their instructions as clear and accurate as possible.

Matchit: Provide children with this _____ to increase their familiarity with the different types of blocks in Scratch and in which category they can be found. They could use the paper-based resource in conjunction with Scratch on their computer or laptop so that they investigate and find the correct answers.

Assessment Notes:

Disclaimers:

External Links:

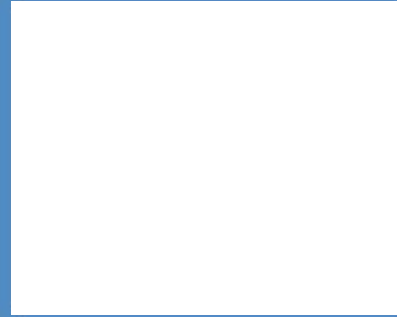
This resource contains links to external websites and/or external apps. Please be aware that the inclusion of any links in this resource should not be taken as an endorsement of any kind by Twinkl of the linked website and/or app, or any association with its operators. You should also be aware that we have no control over the availability of the linked pages and/or apps. If the link is not working, please let us know by contacting TwinklCares and we will try to fix it although we can assume no responsibility if this is the case. We are not responsible for the content of external sites and/or external apps.

Scratch Safety:

Showing or creating the flashing sprite effect could be problematic for children with conditions such as epilepsy. Discretion is advised.

Scissors:

To ensure the safety of the children in your setting, it is your responsibility to assess whether adult supervision or other appropriate safety measures are required when using scissors.



Computing

Coding with Scratch: Learning Loops

Coding
with Scratch

Scratch Revisited



Question Marks

**This is Quizby.
He is a question mark who
loves to ask questions.**



When you see a question mark icon like this in the **Lesson Presentation**, it can be clicked on to reveal one of Quizby's questions.



The questions that appear next to these question marks will help you to think about the key learning throughout the lesson.

Aim

To write algorithms using coding blocks in Scratch.

Success Criteria

I can explain what an algorithm is.

I can identify different types of coding blocks in Scratch and know where to find them.

I can create a sequence of blocks to write an algorithm.

I know that it is important to test and debug an algorithm.

Remember It

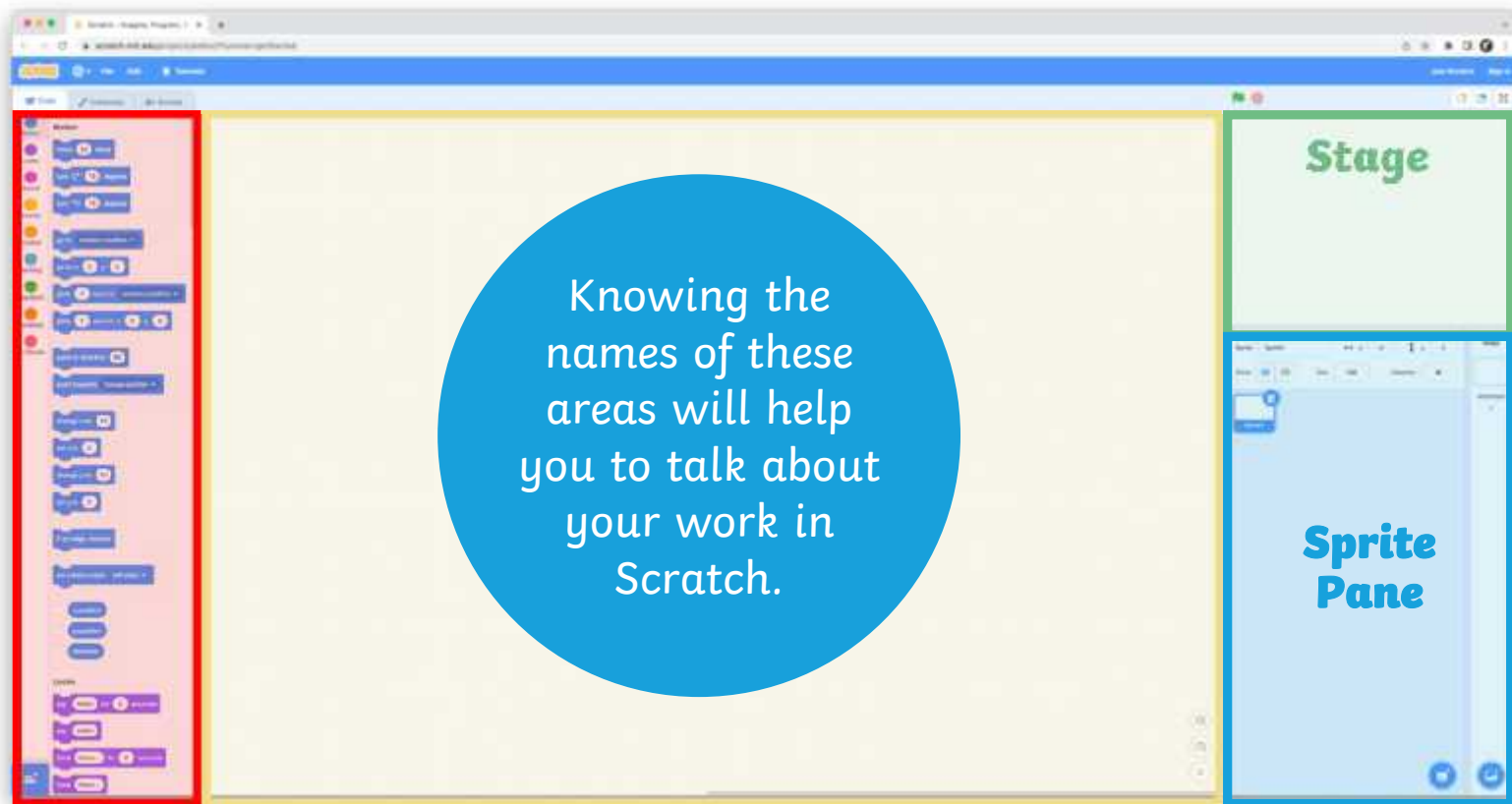
In this series of lessons you will be learning about loops and how to use these within algorithms. You will be using Scratch, an online coding platform that lets you write code using visual blocks.

Have you used Scratch before? Talk to your partner about what you have done on Scratch. What can you remember about it?



Remember It

Have a look at the Scratch interface.
Can you remember what each area is called?



Knowing the names of these areas will help you to talk about your work in Scratch.

Block Palette

Remember It

What do you already know about algorithms and writing code?
Do you know anything about loops?

Have a look at these prompt questions and then talk to your partner.
You have two minutes to share your ideas.

- What is an algorithm?
- How do you write code in Scratch?
- What is a loop?
- What does coding help us to do?
- What is it called when you check code and correct mistakes?



2 minutes

Start



Remember It

What Is Coding?

Coding is writing a set of instructions that will tell a computer what to do. Coding is also known as **computer programming**.

When coding, we need to **sequence** our instructions in a **logical** order that makes a computer program work.

This sequence of instructions is called an **algorithm**.

In Scratch, algorithms are referred to as scripts. You create your algorithms by dragging coding blocks into the **Code Area**.



Knowledge Organiser

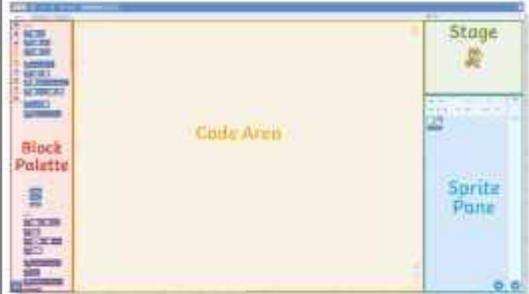
The **Knowledge Organiser** shows you all the important knowledge, vocabulary and skills you will learn during this unit.

Look at the **Knowledge Organiser** for this unit of lessons. Talk to your partner about any interesting things you have noticed. You can use this throughout the unit to support your learning.

Coding with Scratch: Learning Loops

Year 3

Key Vocabulary		What Is Scratch?
algorithm	A sequence of ordered instructions. In Scratch, algorithms are referred to as scripts.	Scratch is a free, online program where you can use a coding language to create digital stories, games and animations using characters known as sprites . Scratch uses a visual block-based coding language. Blocks are joined together to create algorithms .
block	A puzzle-shaped piece of code. They can connect to other blocks to create algorithms .	
code	A set of instructions written in a programming language that a computer can understand.	
condition	A block of code that will only run if a certain event is true or false.	
loop	A way to repeat a set of instructions over and over again.	
sprite	An image that can be created and programmed in Scratch.	
variable	A value that can be recorded in the memory of Scratch. A variable can be edited.	

Scratch Interface	
	

Writing an Algorithm

There are **five** main steps to writing any algorithm.



Let's try using these steps to write our own algorithm.

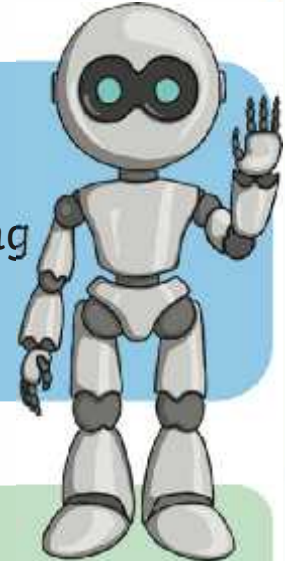
X Why do you think that there is a double-headed arrow between Step 4 and Step 5?



Writing an Algorithm

Step 1 Identify It

This little robot has travelled to Earth and doesn't know what to do when she meets a new friend. She has been given some information about shaking hands but is really unsure what to do. Let's write an algorithm to help her.



Step 2 Break It Down

Let's break the problem down into smaller steps. What things will the robot need to do to complete this task? Share your ideas.

The robot will have to travel towards their friend, meet them, shake hands and finally travel away from their friend.

Were your ideas similar to these?

Writing an Algorithm

Step 3

Write It

Look at the code for making the robot meet and shake hands with a friend. It is all muddled up. Can you put it into the correct order to create an algorithm? When you have had a go, click on the robot to see if you were right.

Join hands with your friend.

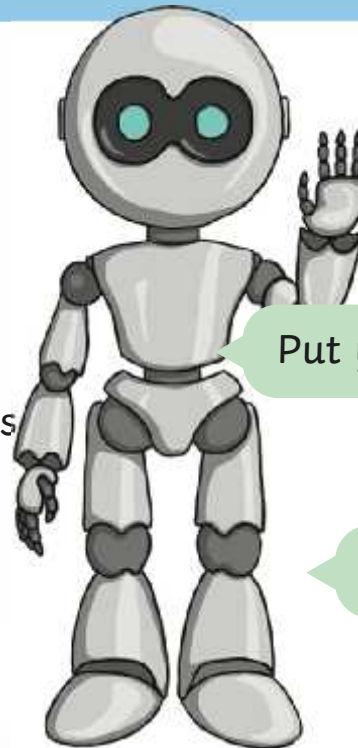
Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times

Put your right hand out in front of you.

Walk towards your friend.



Walk towards your friend.

Stop when you are facing them.

Put your right hand out in front of you.

Join hands with your friend.

Shake hands up and down 3 times.

Walk away from your friend.

Writing an Algorithm

Step 4 Test It

Now work with a partner to test the code. One of you can be the robot and the other one can be the friend. Follow the instructions very carefully. Can you spot an error in the code?

Step 5 Debug It

Did you spot the error? There is no instruction to tell the robot to let go of their friend's hand before they walk away from each other. They could end up pulling each other over! What instruction could you add to debug this algorithm?

Coding Blocks in Scratch

Scratch is a block-based programming language.

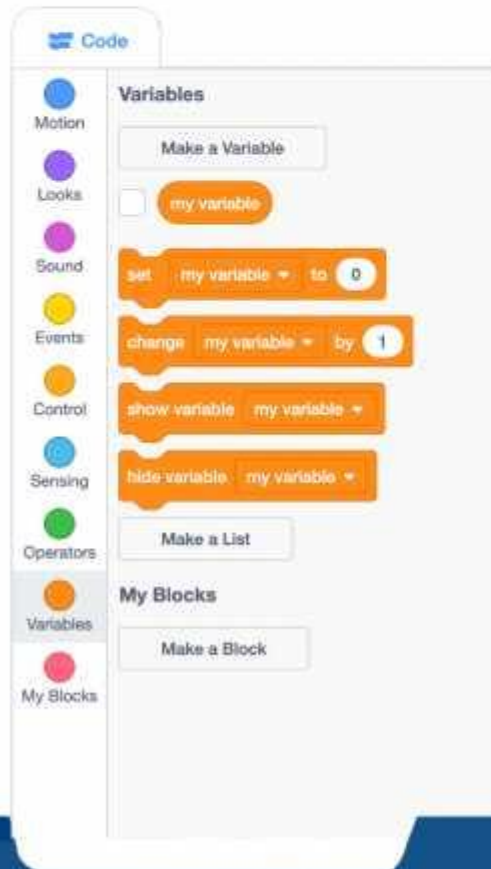
Instead of having to write lines of computer code, blocks can be dragged into a **Code Area** to create algorithms.



The algorithms in Scratch control what happens on the part of the screen called the **Stage**.

Coding Blocks in Scratch

Do you know what some of the blocks in Scratch do?
Do you know where to find them?



The coding blocks in Scratch are found on the left-hand side of the screen under the **Code** tab.

Can you see all the different categories of blocks?

What sort of instructions do you think would be in these different categories?

Coding Blocks in Scratch

Answer the questions in the quiz to see how well you know your way around Scratch.

Question 1

What are the objects and characters in Scratch called?

- A fairies
- B sprites
- C avatars

Next Question

Coding Blocks in Scratch

Question 2

Which of these blocks could you use to start an algorithm?



Next Question

Coding Blocks in Scratch

Question 3

What category would you find this block in?

A purple Scratch 'think' block with the text 'think Hmmmm...' followed by a white circle containing the number '2' and the word 'seconds'.

- A** Looks
- B** Sound
- C** Events

Next Question

Question 4

What would you expect to happen if you use this block?

A purple Scratch block with the text "switch costume to costume2" and a small downward-pointing arrow on the right side.

- A** The sprite would change direction.
- B** The sprite would make a sound.
- C** The sprite would look different in some way.

Next Question

Coding Blocks in Scratch

Question 5

Which of these blocks would make your sprite turn right 90°?



Next Question

Coding Blocks in Scratch

Question 6

Where would you find the block to make the sprite



A



Events

B



Control

C



Sensing

Next Question

Coding Blocks in Scratch

Check your answers to see how well you did.

Question 1

What are the objects and characters in Scratch called?

- A fairies
- B sprites
- C avatars

Next Question

Coding Blocks in Scratch

Question 2

Which of these blocks could you use to start an algorithm?



Next Question

Coding Blocks in Scratch

Question 3

What category would you find this block in?



A



Looks

B



Sound

C




Events

Next Question

Coding Blocks in Scratch

Question 4

What would you expect to happen if you use this block?

A purple Scratch block with the text "switch costume to costume2" and a small downward-pointing arrow on the right side.

- A** The sprite would change direction.
- B** The sprite would make a sound.
- C** The sprite would look different in some way.

Next Question

Coding Blocks in Scratch

Question 5

Which of these blocks would make your sprite turn right 90°?



Next Question

Coding Blocks in Scratch

Question 6

Where would you find the block to make the sprite



A



Events

B



Control

C



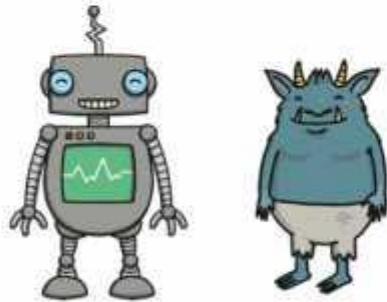
Sensing

Algorithms in Scratch

Step 1 Identify It

Before you can write an algorithm in Scratch you will need to decide what problem you want to solve or what task you want to complete.

Scratch uses objects and characters called sprites. The algorithms you write will tell the sprites what to do.



If you have more than one sprite then you will need to write an algorithm for each sprite.

Step 2 Break It Down

Break down the problem or task into smaller steps. Think about what you want the sprite to do first, then second, etc.

Algorithms in Scratch

Step 3

Write It

Writing algorithms in Scratch is very easy.

All you need to do is drag blocks from the Block Categories into the **Code Area**.

The blocks of code are shaped a bit like jigsaw pieces. When you place the blocks near each other they will click together to start making an algorithm.

If you add a block and then change your mind just drag it back to the **Block Palette** on the left and it will disappear from the **Code Area**.

Top Tip: Remember that if you have more than one sprite on the **Stage**, you will need to click on the correct sprite to add the code.



Algorithms in Scratch

Step 3

Write It

You might have noticed that some of the blocks have a white area within them. If you click in this area you can customise the number or text that is in there.

You can customise the block here, inputting text or numbers.



Why is it useful to be able to customise coding blocks?



Algorithms in Scratch

Step 3

Write It

There are some really important blocks in the **Events** category. These blocks look a little bit different from the others. They have a rounded top instead of a jigsaw shape and are called **Hat blocks**.



These blocks sit at the top of your algorithm. They are used to begin running the algorithm. You can explore using these in your code.

Algorithms in Scratch

Step 4
Test It

Testing your algorithm in Scratch is very simple. Follow the instruction on your first **Events** block to run your code.



Click on this button to make the **Stage** full screen.

Step 5
Debug It

If your program does not run as you expected it to, search for the error. When you find the incorrect code you can delete it or edit it.

You will then need to test your code again.

Sometimes you may find you have to move between testing and debugging a number of times before you are happy that your code is just right. Don't worry as this is exactly what computing programmers and coders have to do all the time!

Algorithms in Scratch



Now it is your turn! Use all the things you have learnt about algorithms to write some code in Scratch.

Look carefully at the task **Meet and Greet**. Remember to follow the **five** steps for writing an algorithm.

Step 1: Identify It

Step 2: Break It Down

Step 3: Write It

Step 4: Test It



Step 5: Debug It

Meet and Greet

To write algorithms using coding blocks in Scratch.

Step 1: Identify It

Can you write the code to make two sprites meet, take turns to greet one another and then walk on past?

Step 2: Break It Down

Select a backdrop and two sprites. Make sure the sprites are facing each other.

Make your sprites walk towards each other. How many steps will they need to move?

Make your sprites greet one another in turn. What could they say to each other? How can you make the sprites take it in turn to speak?

Make your sprites walk past each other.

Step 3: Write It

These code blocks may help you to write your code. Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?



Step 5: Debug It

Edit your code and test it again.

★ Top Tip:

To change the **Direction** a sprite faces in:

Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue



Algorithms in Scratch

Why not share your Scratch project with your partner, group or class?

Looking at each other's work is a great way of learning more about coding.

Tell each other what you liked about the project.

Suggest things that could be added to the project. What are the next steps to make it even better?



Aim

To write algorithms using coding blocks in Scratch.

Success Criteria

I can explain what an algorithm is.

I can identify different types of coding blocks in Scratch and know where to find them.

I can create a sequence of blocks to write an algorithm.

I know that it is important to test and debug an algorithm.



Meet and Greet

To write algorithms using coding blocks in Scratch.



Step 1: Identify It

Can you write the code to make two sprites meet and greet one another?

Step 2: Break It Down

Select a backdrop and two sprites.

Make sure the sprites are facing each other.

Make your sprites walk towards each other.

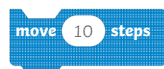
How many steps will they need to move?

Make your sprites greet one another when they meet.

What could they say to each other?

Step 3: Write It

These code blocks may help you to write your code. Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?



Step 5: Debug It

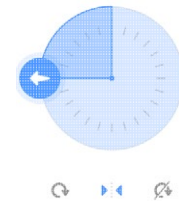
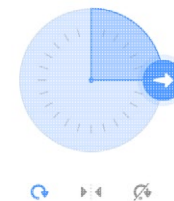
Edit your code and test it again.

★ Top Tip:

To change the direction a sprite faces in:

Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue arrow around the circle until it faces in the opposite direction.



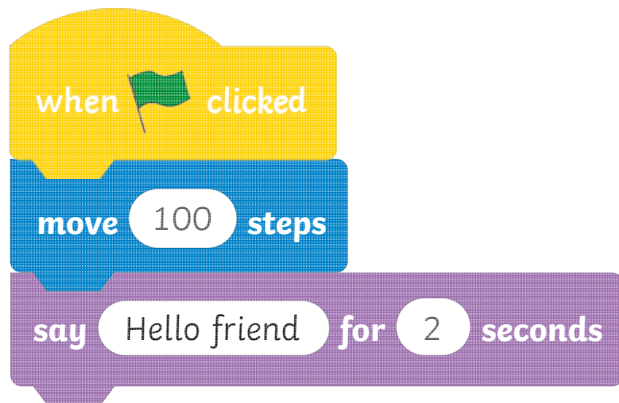
Meet and Greet Answers

To write algorithms using coding blocks in Scratch.

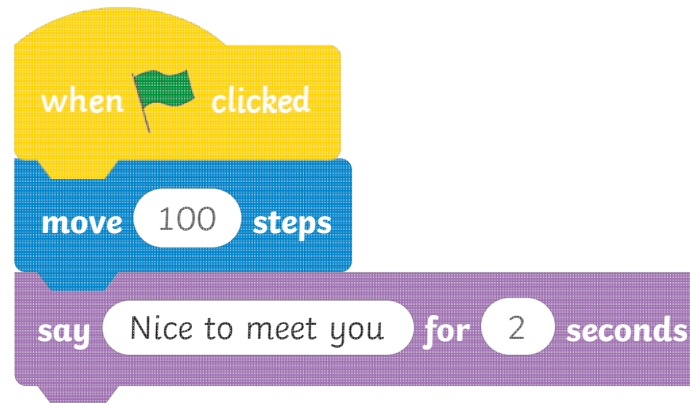


The code below shows one possible answer. Yours may look different.

sprite 1



sprite 2



Meet and Greet

To write algorithms using coding blocks in Scratch.



Step 1: Identify It

Can you write the code to make two sprites meet, take turns to greet one another and then walk on past?

Step 2: Break It Down

Select a backdrop and two sprites.
Make sure the sprites are facing each other.

Make your sprites walk towards each other.
How many steps will they need to move?

Make your sprites greet one another in turn.
What could they say to each other? How can you make the sprites take it in turn to speak?

Make your sprites walk past each other.

Step 3: Write It

These code blocks may help you to write your code.
Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?



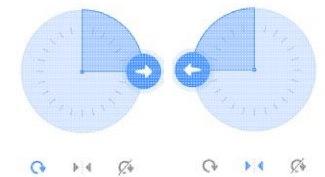
Step 5: Debug It

Edit your code and test it again.

★ Top Tip:

To change the direction a sprite faces in:
Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue arrow around the circle until it faces in the opposite direction.



Meet and Greet Answers

To write algorithms using coding blocks in Scratch.



The code below shows one possible answer. Yours may look different.

sprite 1

```
when green flag clicked
  move 100 steps
  say Hello friend for 2 seconds
  wait 2 seconds
  move 200 steps
```

sprite 2

```
when green flag clicked
  move 100 steps
  wait 2 seconds
  say Nice to meet you for 2 seconds
  move 200 steps
```

Meet and Greet

To write algorithms using coding blocks in Scratch.



Step 1: Identify It

Can you write the code to make two sprites meet, take turns to greet one another and then walk on past?

Step 2: Break It Down

Select a backdrop and two sprites. Make sure the sprites are facing each other.

Make your sprites walk towards each other. How many steps will they need to move?

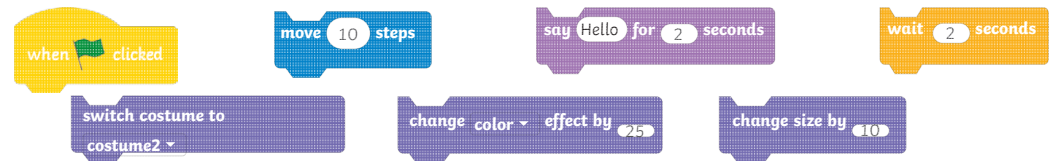
Make your sprites greet one another in turn. What could they say to each other? How can you make the sprites take it in turn to speak?

Make your sprites change their appearance in some way.

Make your sprites say goodbye and walk past each other.

Step 3: Write It

These code blocks may help you to write your code but you will need to customise the text or numbers in the white area. Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?

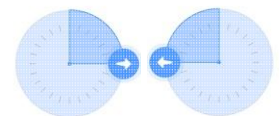


Step 5: Debug It

Edit your code and test it again.

★ **Top Tip:** To change the direction a sprite faces in: Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue arrow around the circle until it faces in the opposite direction.



Meet and Greet Answers

To write algorithms using coding blocks in Scratch.



○

The code below shows one possible answer. Yours may look different.

sprite 1

```
when green flag clicked
  move 100 steps
  wait 2 seconds
  say Hello friend for 2 seconds
  switch costume to costume2
  wait 2 seconds
  Say Goodbye! for 2 seconds
  switch costume to costume1
  move 200 steps
```

sprite 2

```
when green flag clicked
  move 100 steps
  wait 2 seconds
  say Nice to meet you for 2 seconds
  switch costume to dog2-c
  wait 2 seconds
  say See you again for 2 seconds
  switch costume to dog2-a
  move 200 steps
```

Meet and Greet

To write algorithms using coding blocks in Scratch.



Step 1: Identify It

Can you write the code to make two sprites meet and greet one another?

Step 2: Break It Down

Select a backdrop and two sprites.

Make sure the sprites are facing each other.

Make your sprites walk towards each other.

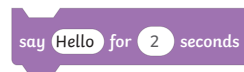
How many steps will they need to move?

Make your sprites greet one another when they meet.

What could they say to each other?

Step 3: Write It

These code blocks may help you to write your code. Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?



Step 5: Debug It

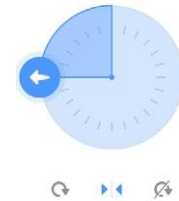
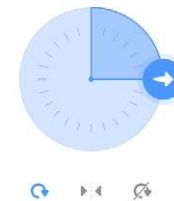
Edit your code and test it again.

★ Top Tip:

To change the direction a sprite faces in:

Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue arrow around the circle until it faces in the opposite direction.



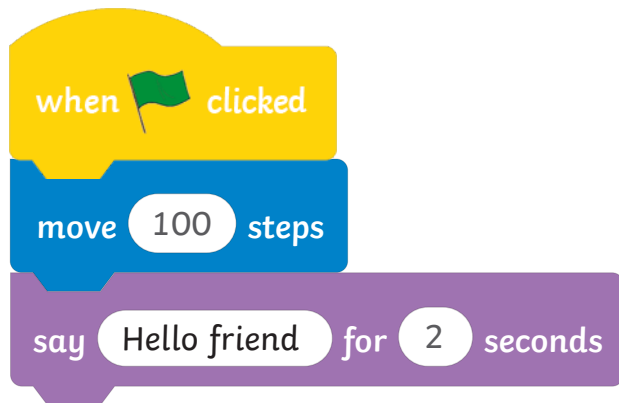
Meet and Greet Answers

To write algorithms using coding blocks in Scratch.

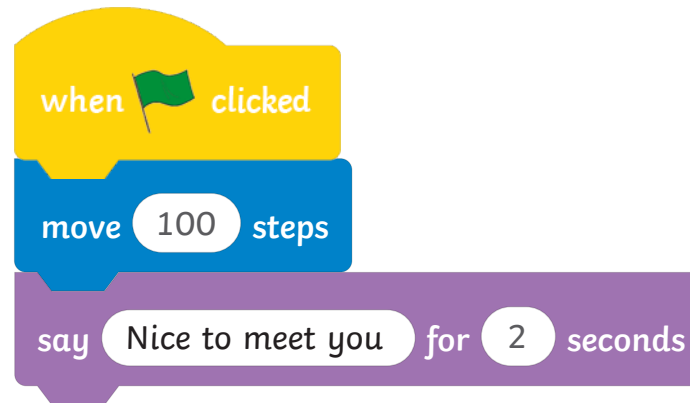


The code below shows one possible answer. Yours may look different.

sprite 1



sprite 2



Meet and Greet

To write algorithms using coding blocks in Scratch.



Step 1: Identify It

Can you write the code to make two sprites meet, take turns to greet one another and then walk on past?

Step 2: Break It Down

Select a backdrop and two sprites.
Make sure the sprites are facing each other.

Make your sprites walk towards each other.
How many steps will they need to move?

Make your sprites greet one another in turn.
What could they say to each other? How can you make the sprites take it in turn to speak?

Make your sprites walk past each other.

Step 3: Write It

These code blocks may help you to write your code.
Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?



Step 5: Debug It

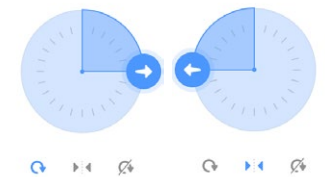
Edit your code and test it again.

★ Top Tip:

To change the direction a sprite faces in:

Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue arrow around the circle until it faces in the opposite direction.



Meet and Greet Answers

To write algorithms using coding blocks in Scratch.



The code below shows one possible answer. Yours may look different.

sprite 1

```
when green flag clicked
  move 100 steps
  say Hello friend for 2 seconds
  wait 2 seconds
  move 200 steps
```

The code for sprite 1 consists of five blocks stacked vertically. The first block is a yellow 'when green flag clicked' block. The second block is a blue 'move 100 steps' block. The third block is a purple 'say Hello friend for 2 seconds' block. The fourth block is an orange 'wait 2 seconds' block. The fifth block is a blue 'move 200 steps' block.

sprite 2

```
when green flag clicked
  move 100 steps
  wait 2 seconds
  say Nice to meet you for 2 seconds
  move 200 steps
```

The code for sprite 2 consists of five blocks stacked vertically. The first block is a yellow 'when green flag clicked' block. The second block is a blue 'move 100 steps' block. The third block is an orange 'wait 2 seconds' block. The fourth block is a purple 'say Nice to meet you for 2 seconds' block. The fifth block is a blue 'move 200 steps' block.

Meet and Greet

To write algorithms using coding blocks in Scratch.



Step 1: Identify It

Can you write the code to make two sprites meet, take turns to greet one another and then walk on past?

Step 2: Break It Down

Select a backdrop and two sprites. Make sure the sprites are facing each other.

Make your sprites walk towards each other. How many steps will they need to move?

Make your sprites greet one another in turn. What could they say to each other? How can you make the sprites take it in turn to speak?

Make your sprites change their appearance in some way.

Make your sprites say goodbye and walk past each other.

Step 3: Write It

These code blocks may help you to write your code but you will need to customise the text or numbers in the white area. Remember that each sprite will need its own code.



Step 4: Test It

Run your code. Does it work as you thought it would?



Step 5: Debug It

Edit your code and test it again.

★ **Top Tip:** To change the direction a sprite faces in: Click on the **Direction** box in the **Sprite Pane**.

Click on the two small arrows below the circle to choose **Left/Right**. Drag the large blue arrow around the circle until it faces in the opposite direction.



Meet and Greet Answers

To write algorithms using coding blocks in Scratch.



○

The code below shows one possible answer. Yours may look different.

sprite 1

```
when green flag clicked
  move 100 steps
  wait 2 seconds
  say Hello friend for 2 seconds
  switch costume to costume2
  wait 2 seconds
  say Goodbye! for 2 seconds
  switch costume to costume1
  move 200 steps
```

sprite 2

```
when green flag clicked
  move 100 steps
  wait 2 seconds
  say Nice to meet you for 2 seconds
  switch costume to dog2-c
  wait 2 seconds
  say See you again for 2 seconds
  switch costume to dog2-a
  move 200 steps
```

Scratch Blocks Quiz Answer Sheets

To write algorithms using coding blocks in Scratch.



Scratch Blocks Quiz

Tick the correct answer.

Q1.

A B C

Q2.

A B C

Q3.

A B C

Q4.

A B C

Q5.

A B C

Q6.

A B C

Scratch Blocks Quiz

Tick the correct answer.

Q1.

A B C

Q2.

A B C

Q3.

A B C

Q4.

A B C

Q5.

A B C

Q6.

A B C

Scratch Blocks Quiz

Tick the correct answer.

Q1.

A B C

Q2.

A B C

Q3.

A B C

Q4.

A B C

Q5.

A B C

Q6.

A B C

Scratch Blocks Quiz Answer Sheets

To write algorithms using coding blocks in Scratch.



Scratch Blocks Quiz

Tick the correct answer.

Q1.

A B C

Q2.

A B C

Q3.

A B C

Q4.

A B C

Q5.

A B C

Q6.

A B C

Scratch Blocks Quiz

Tick the correct answer.

Q1.

A B C

Q2.

A B C

Q3.

A B C

Q4.

A B C

Q5.

A B C

Q6.

A B C

Scratch Blocks Quiz

Tick the correct answer.

Q1.

A B C

Q2.

A B C

Q3.

A B C

Q4.

A B C

Q5.

A B C

Q6.

A B C

Writing an Algorithm: Muddled Instructions

To write algorithms using block code in Scratch.



Cut up the muddled instructions and arrange them into the correct order.

Join hands with your friend.

Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times.

Put your right hand out in front of you.

Walk towards your friend.

Join hands with your friend.

Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times.

Put your right hand out in front of you.

Walk towards your friend.

Join hands with your friend.

Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times.

Put your right hand out in front of you.

Walk towards your friend.

Writing an Algorithm: Muddled Instructions

To write algorithms using block code in Scratch.



Cut up the muddled instructions and arrange them into the correct order.

Join hands with your friend.

Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times.

Put your right hand out in front of you.

Walk towards your friend.

Join hands with your friend.

Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times.

Put your right hand out in front of you.

Walk towards your friend.

Join hands with your friend.

Stop when you are facing them.

Walk away from your friend.

Shake hands up and down 3 times.

Put your right hand out in front of you.

Walk towards your friend.

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		

Coding with Scratch: Learning Loops | Scratch Revisited

To write algorithms using coding blocks in Scratch.		
I can explain what an algorithm is.		
I can identify different types of coding blocks in Scratch and know where to find them.		
I can create a sequence of blocks to write an algorithm.		
I know that it is important to test and debug an algorithm.		