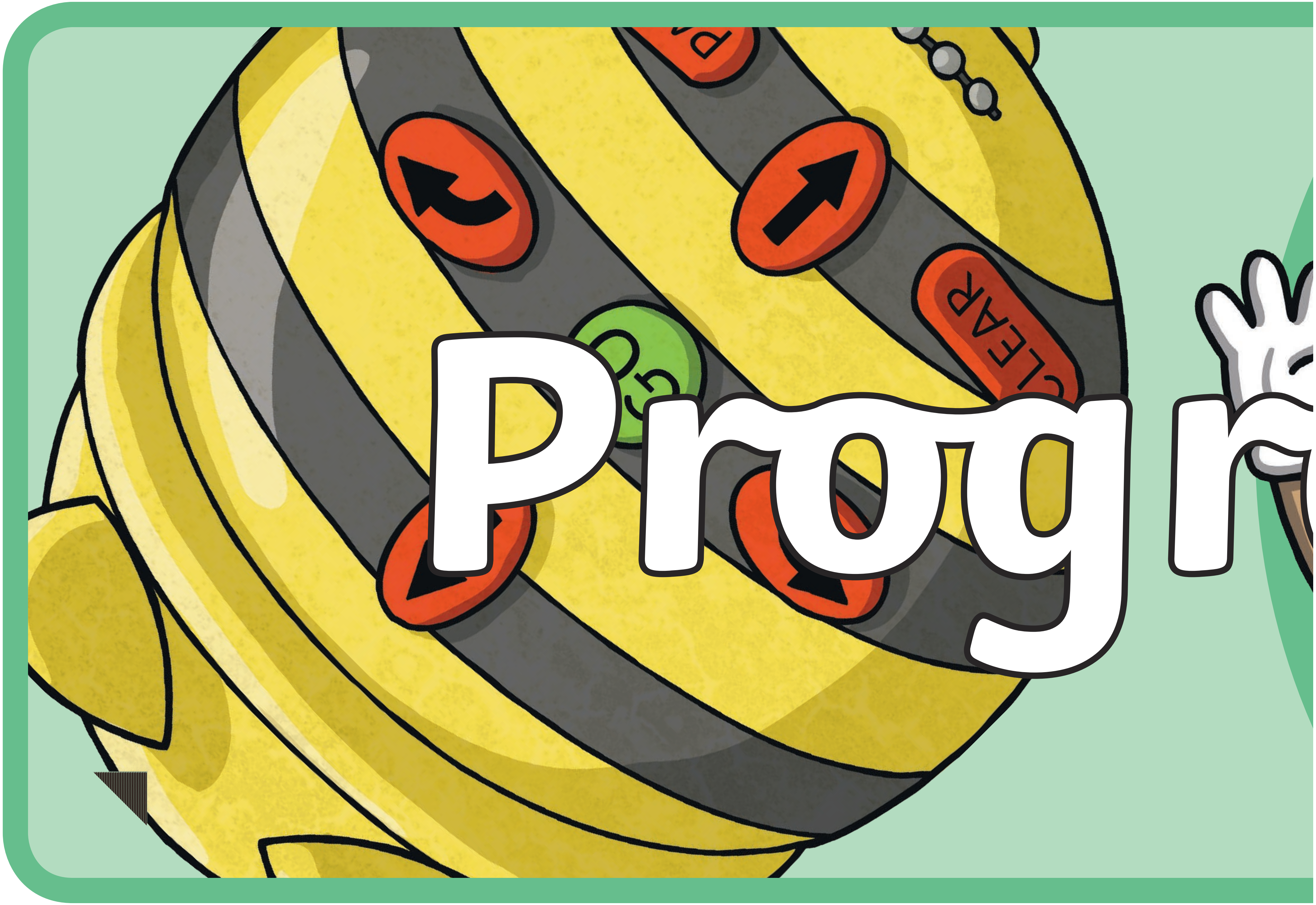
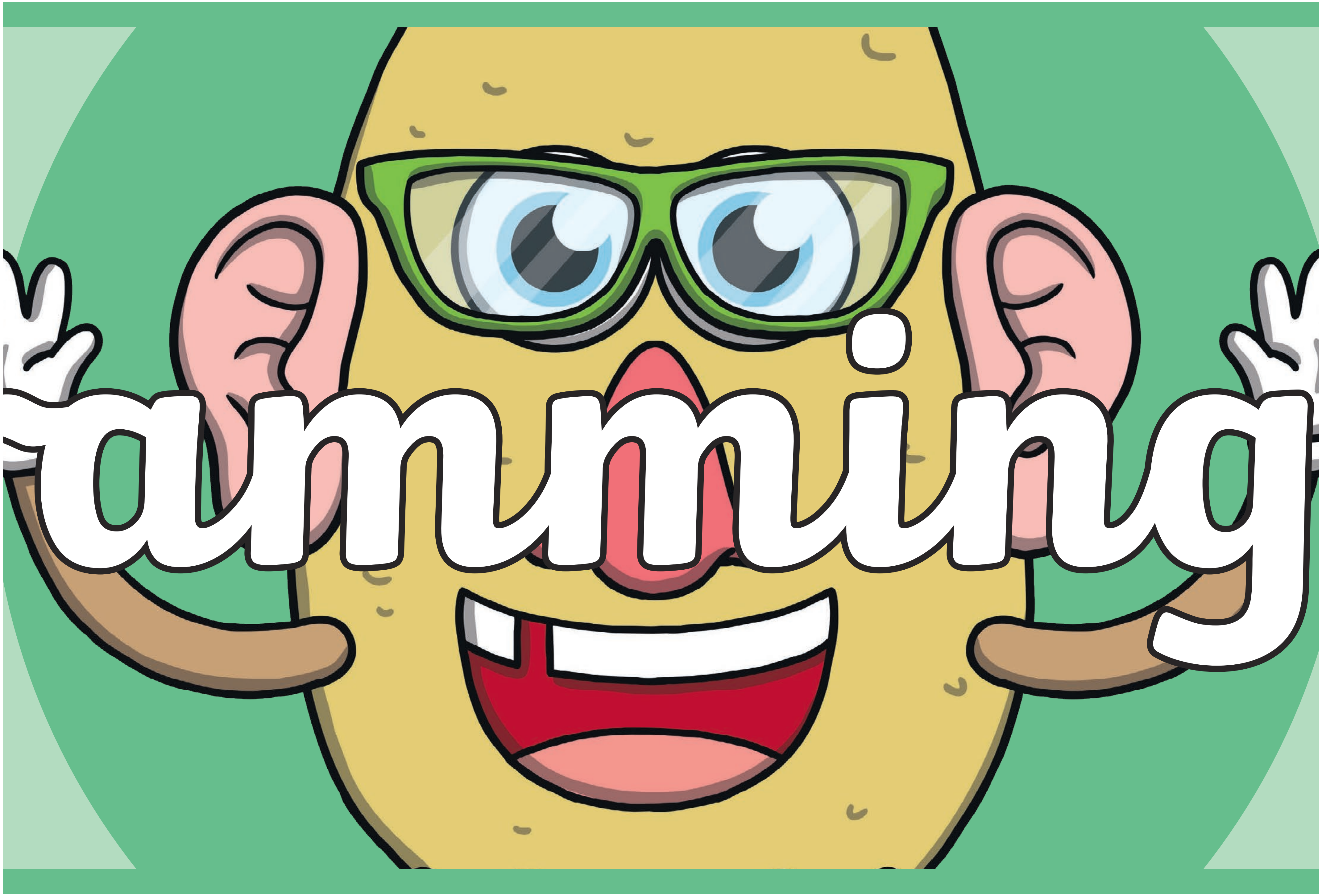
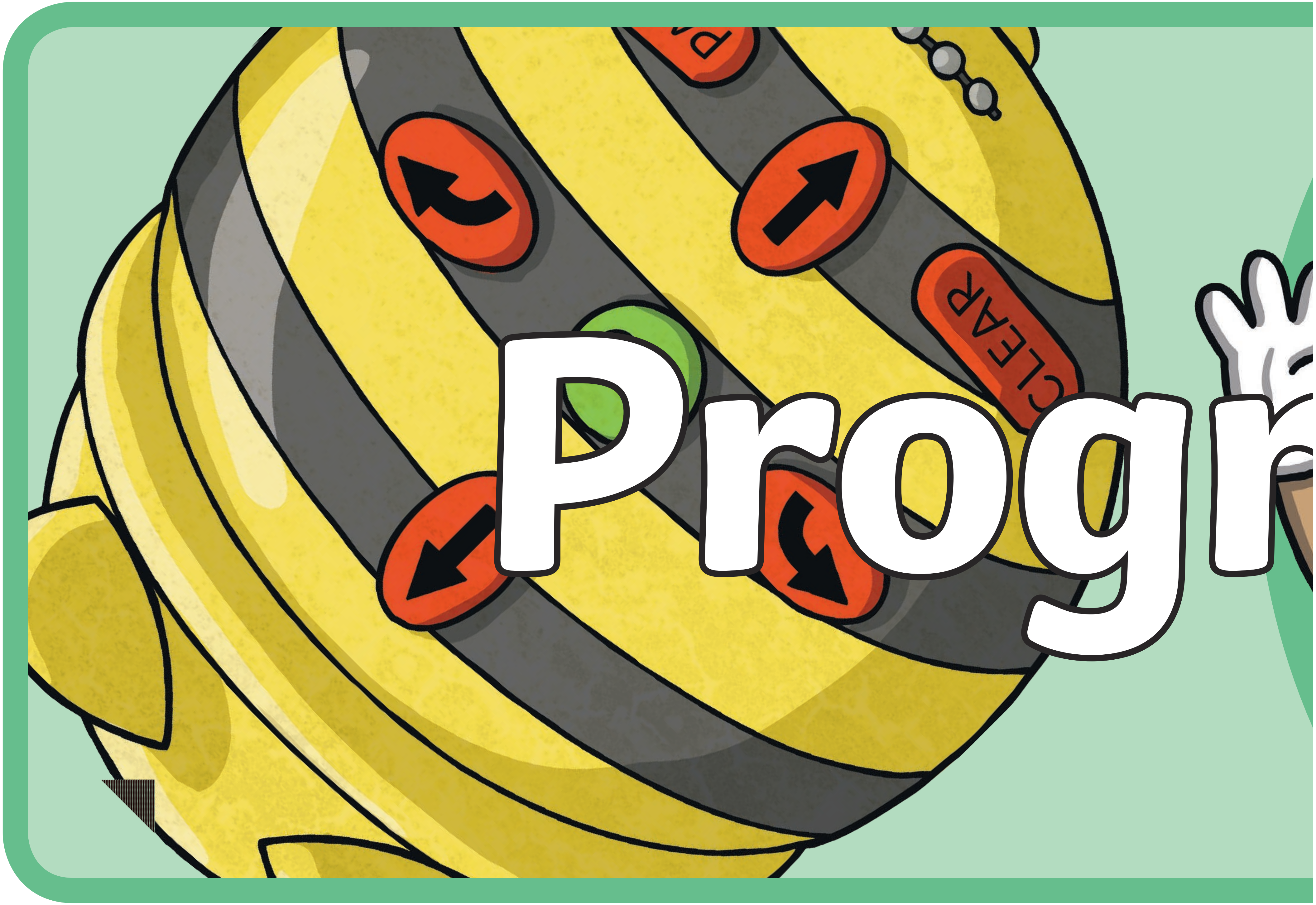
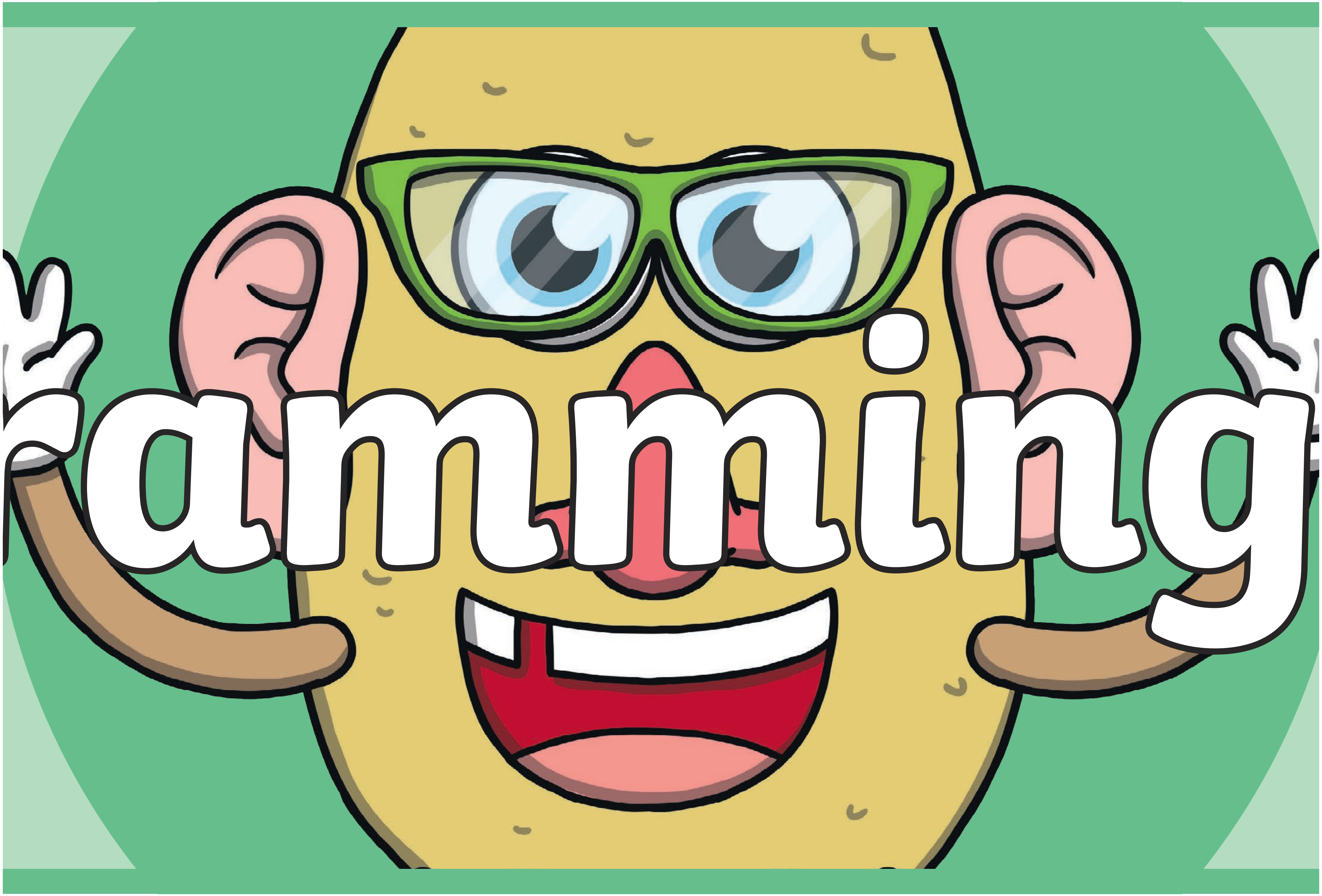

Progr


amming

Toys

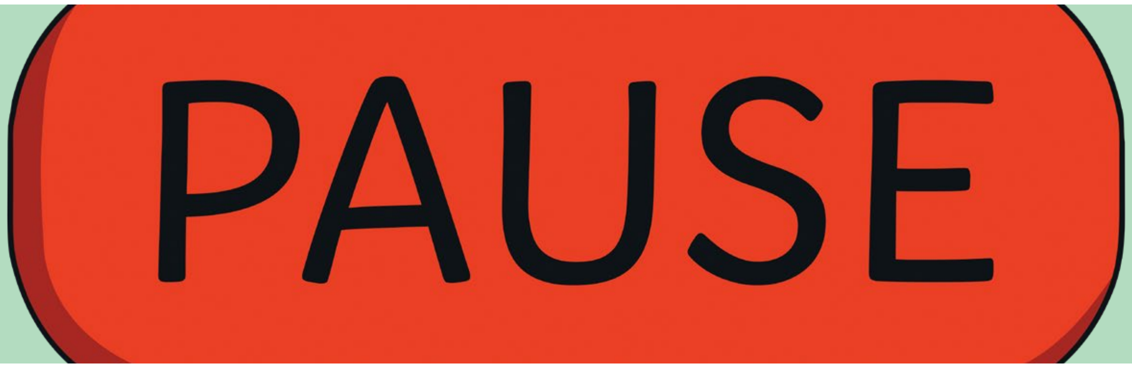
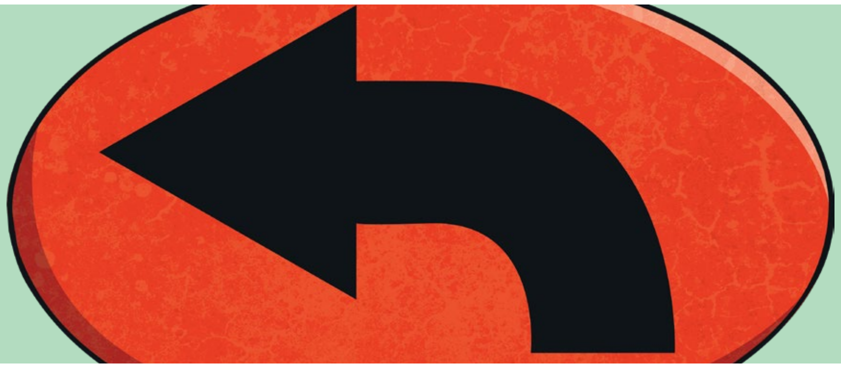
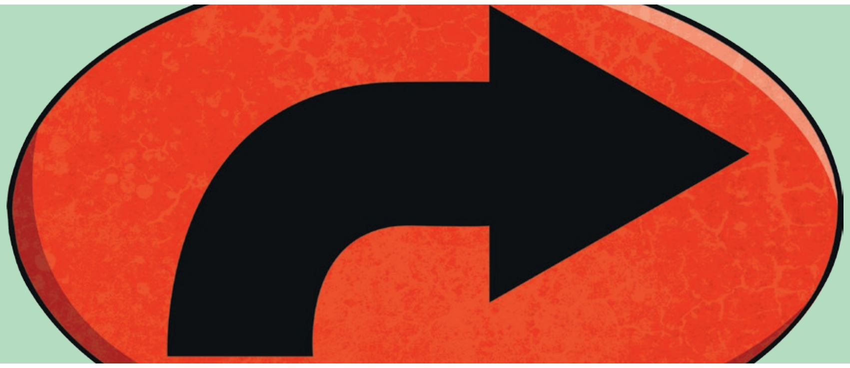
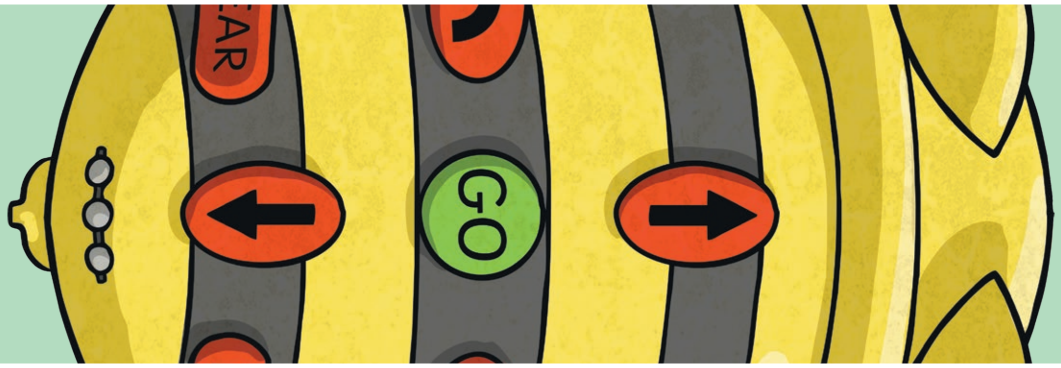


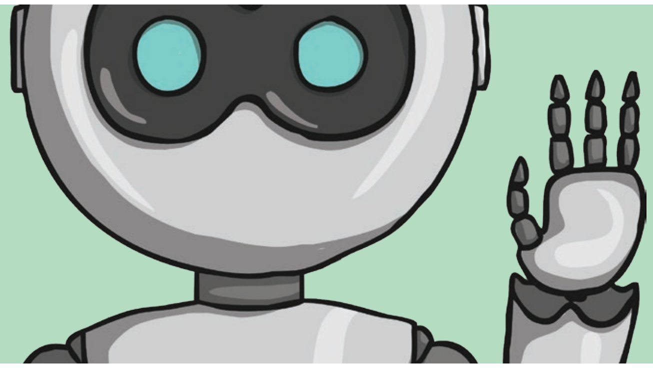
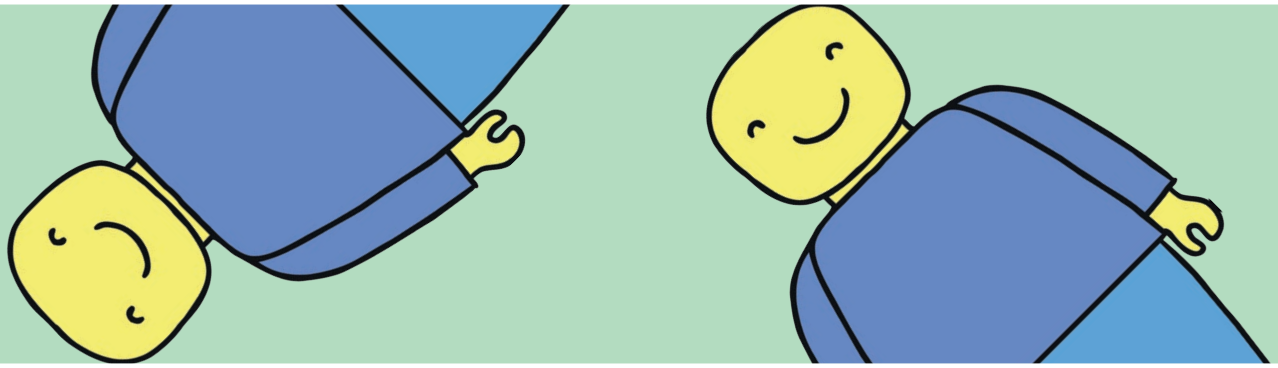
Progr


amming

Toys


GO

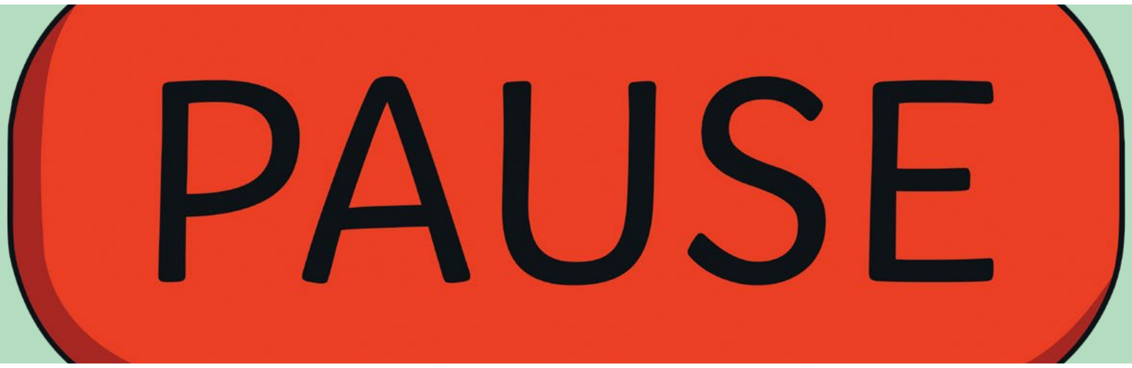
CLEAR

PAUSE



GO

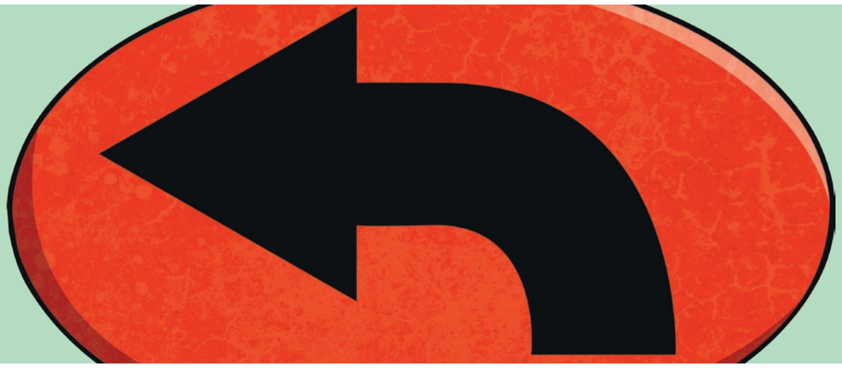



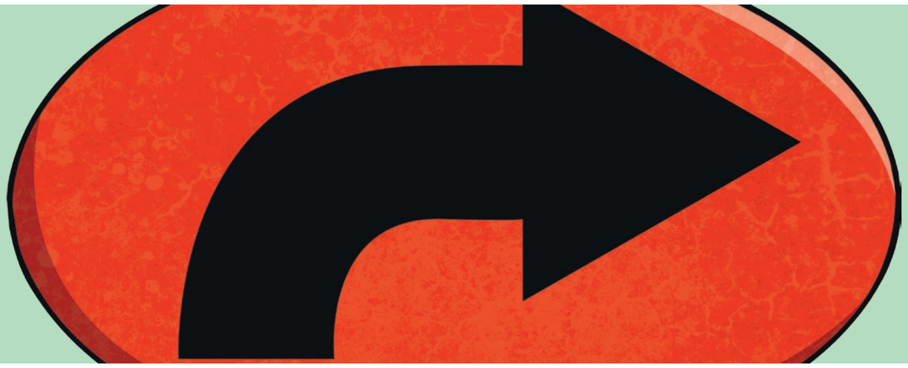


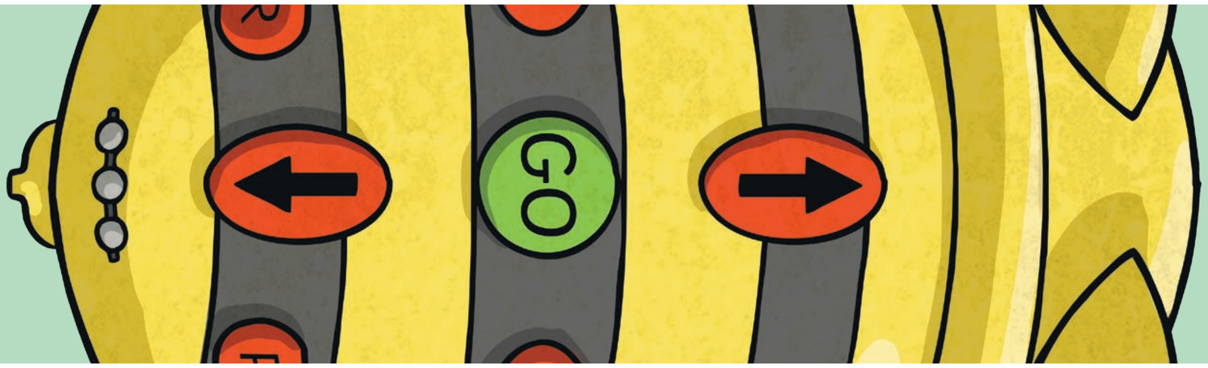
GO

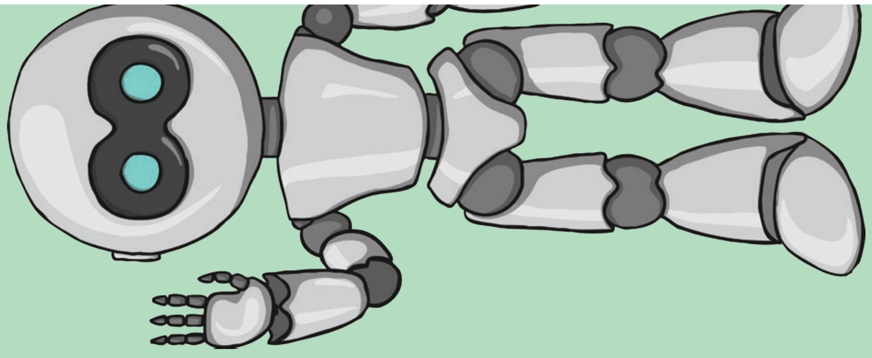

CLEAR

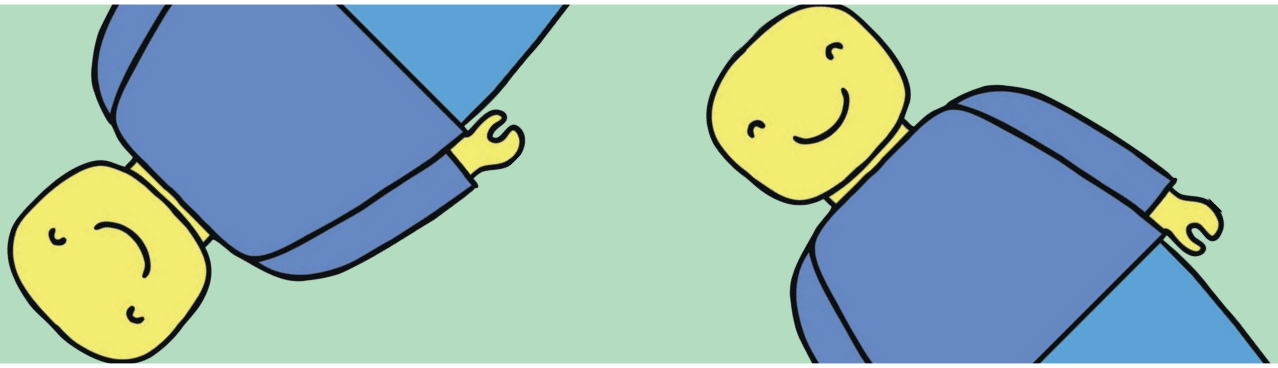

PAUSE

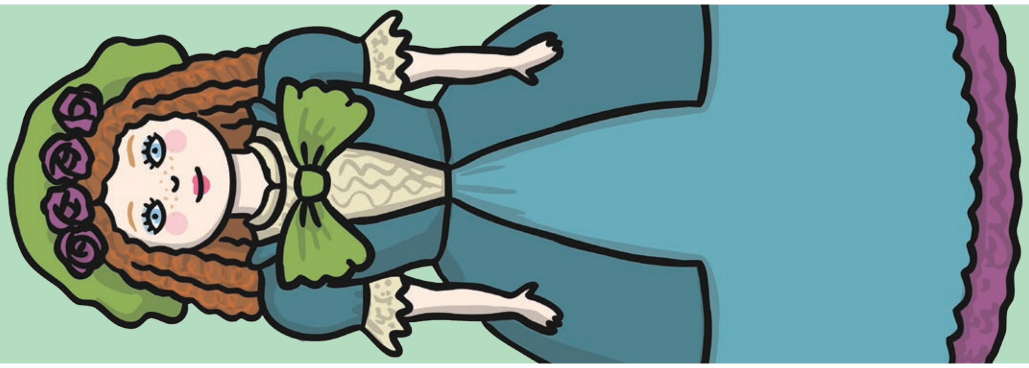





GO

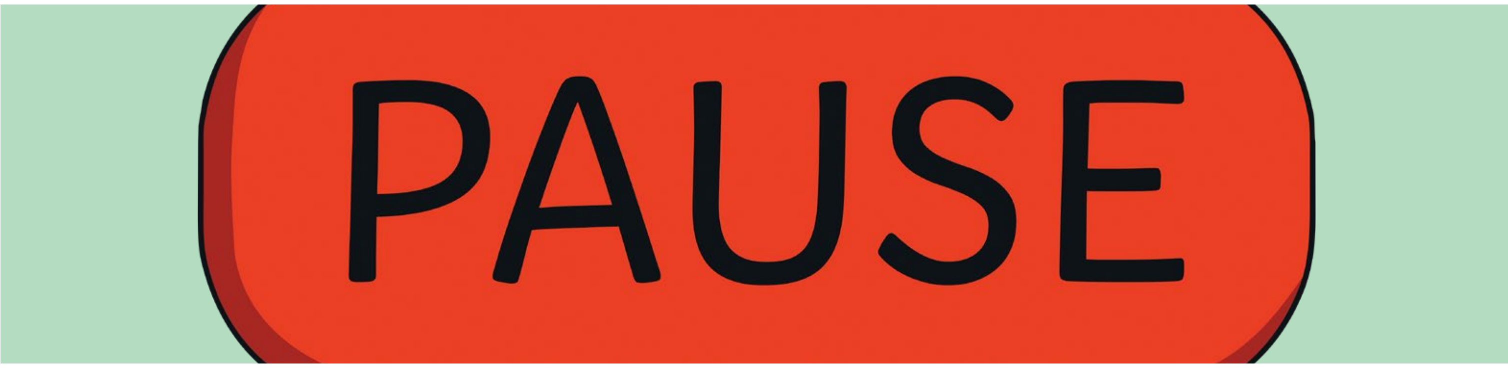


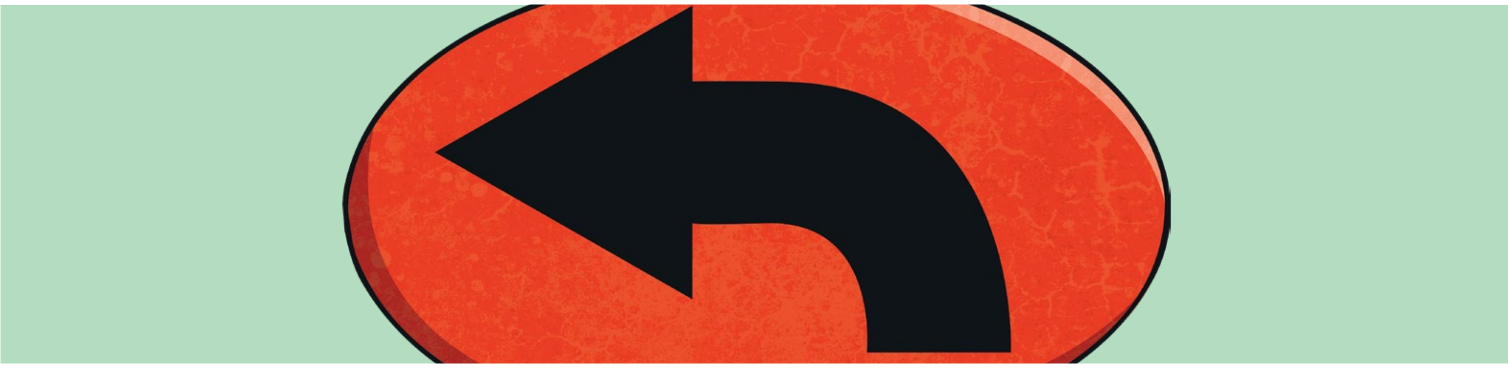


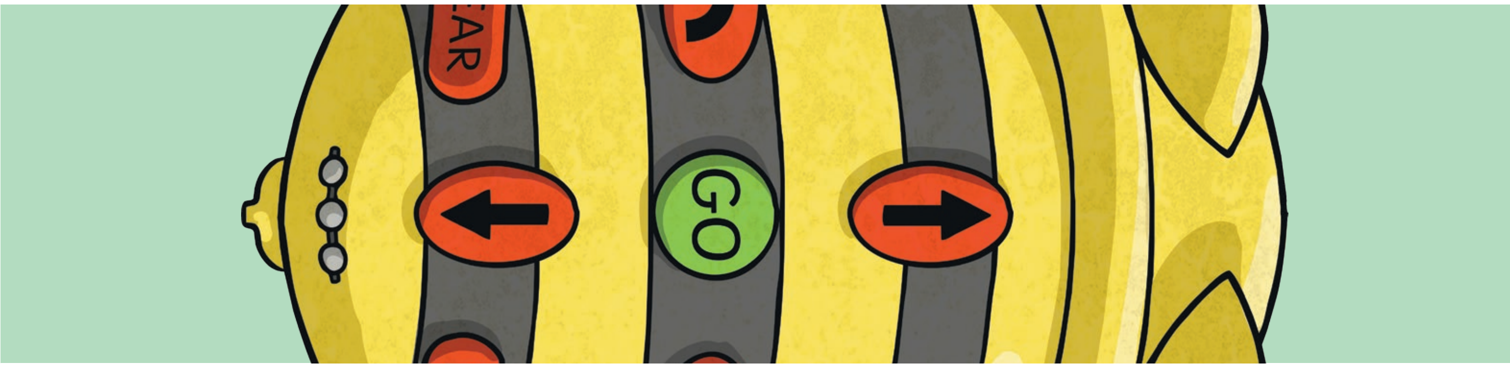


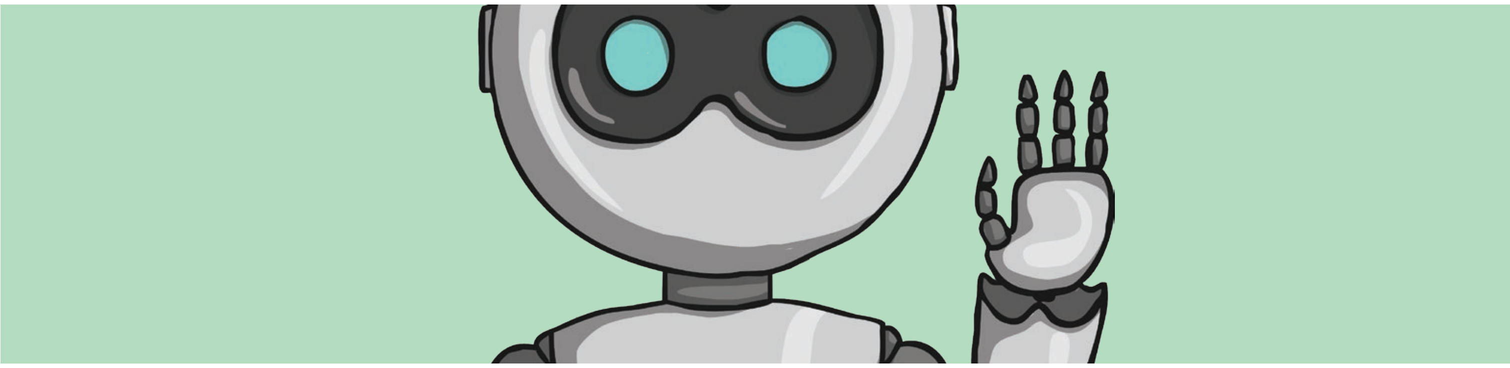

GO

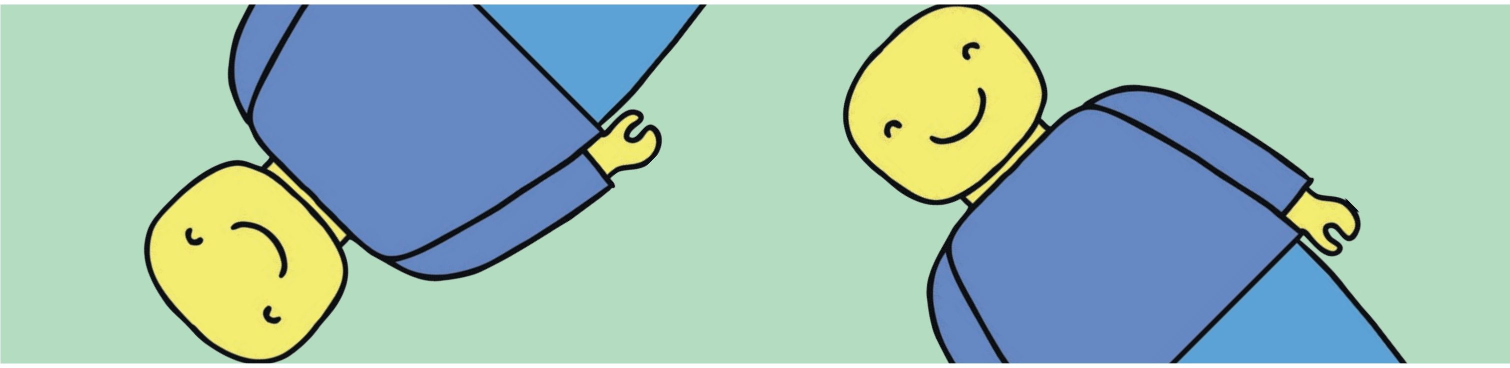

CLEAR


PAUSE




GO

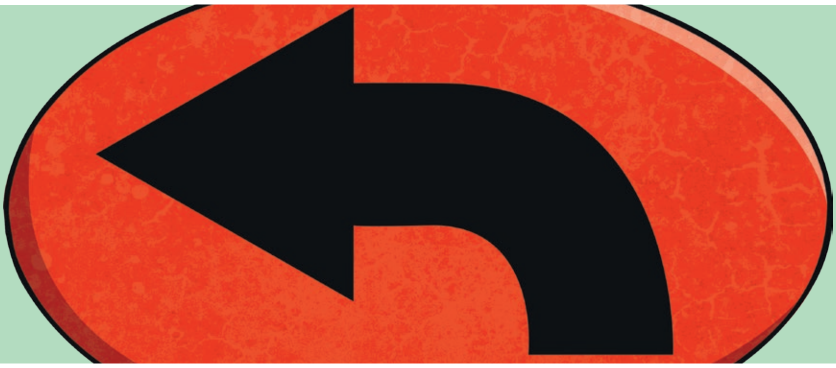


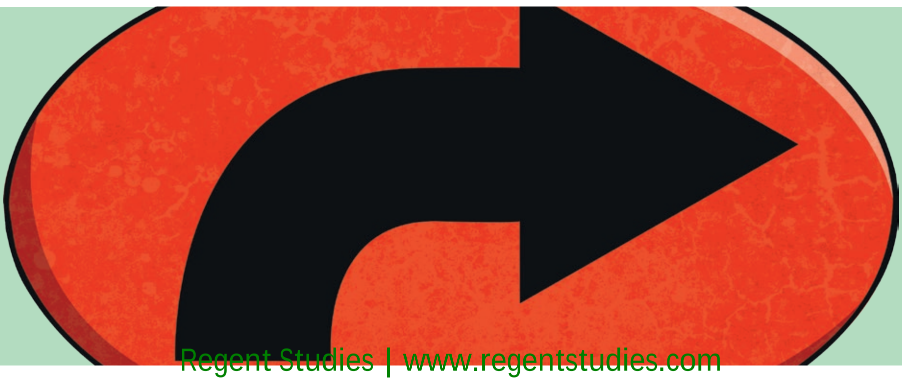


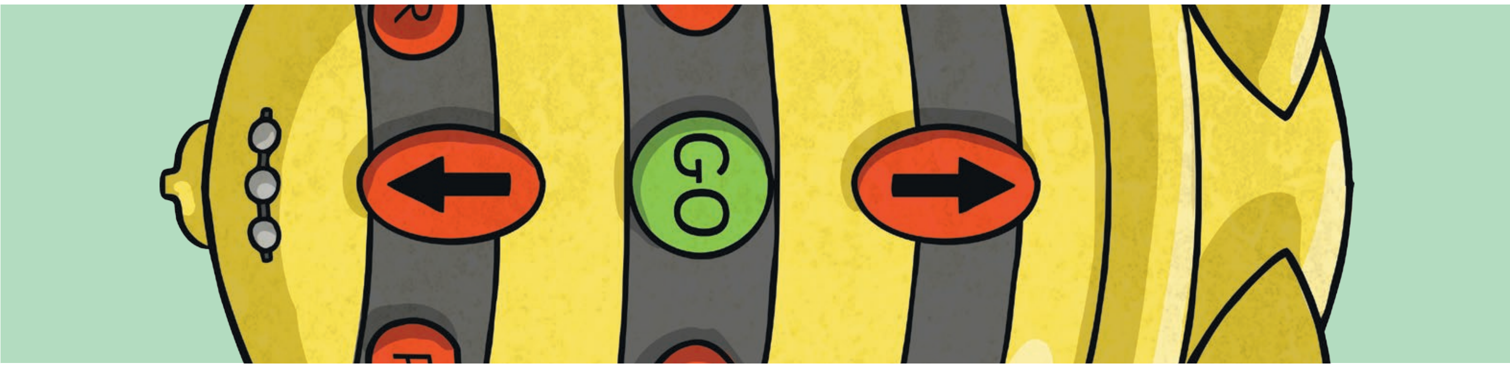
GO

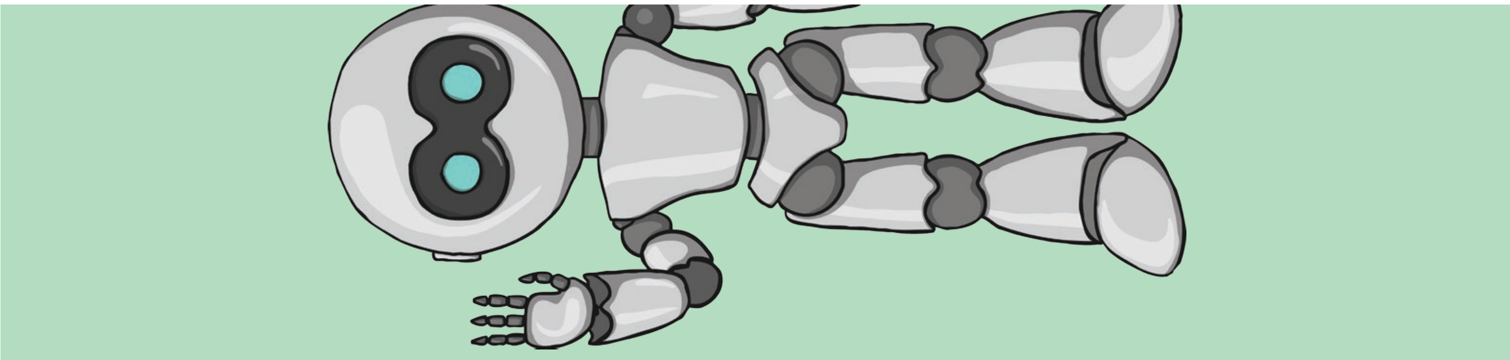
CLEAR

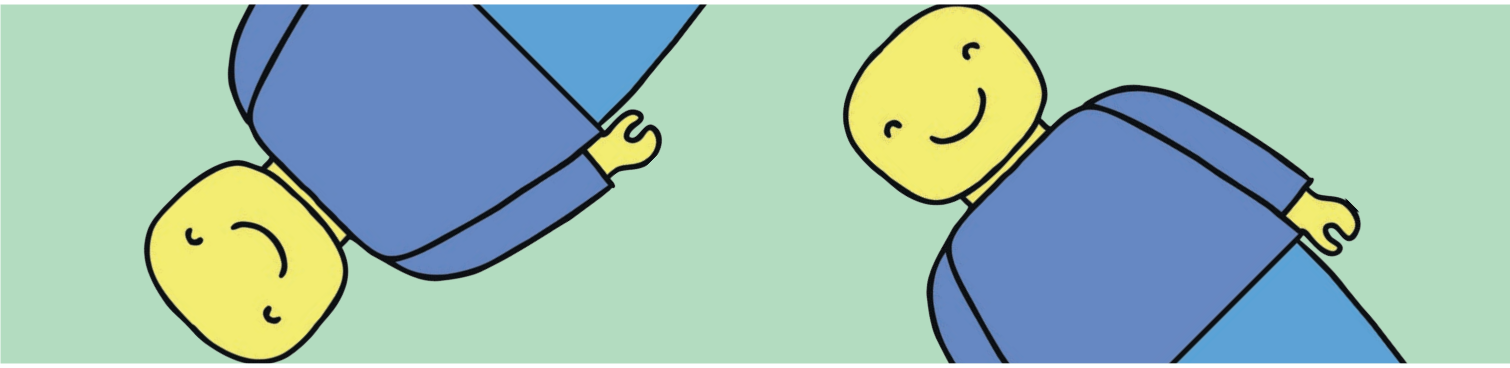
PAUSE

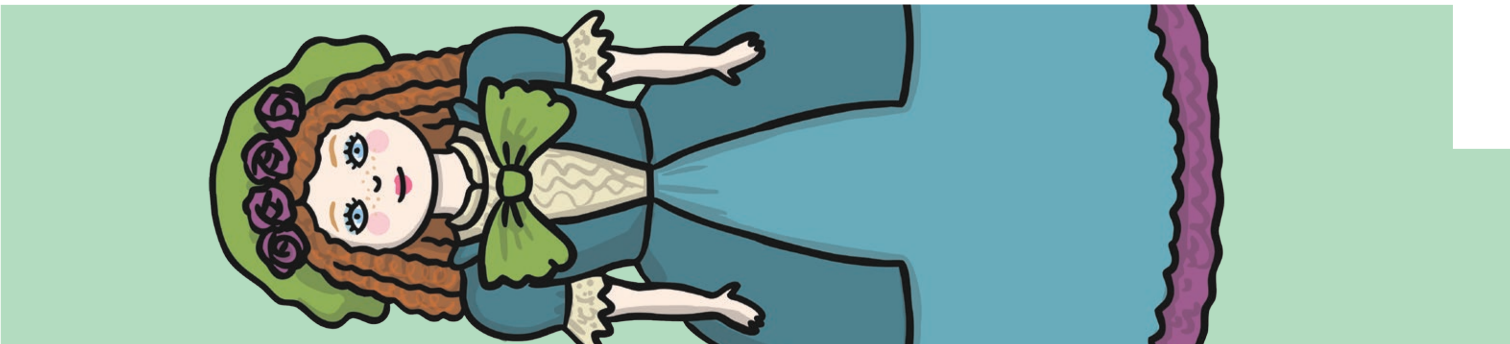




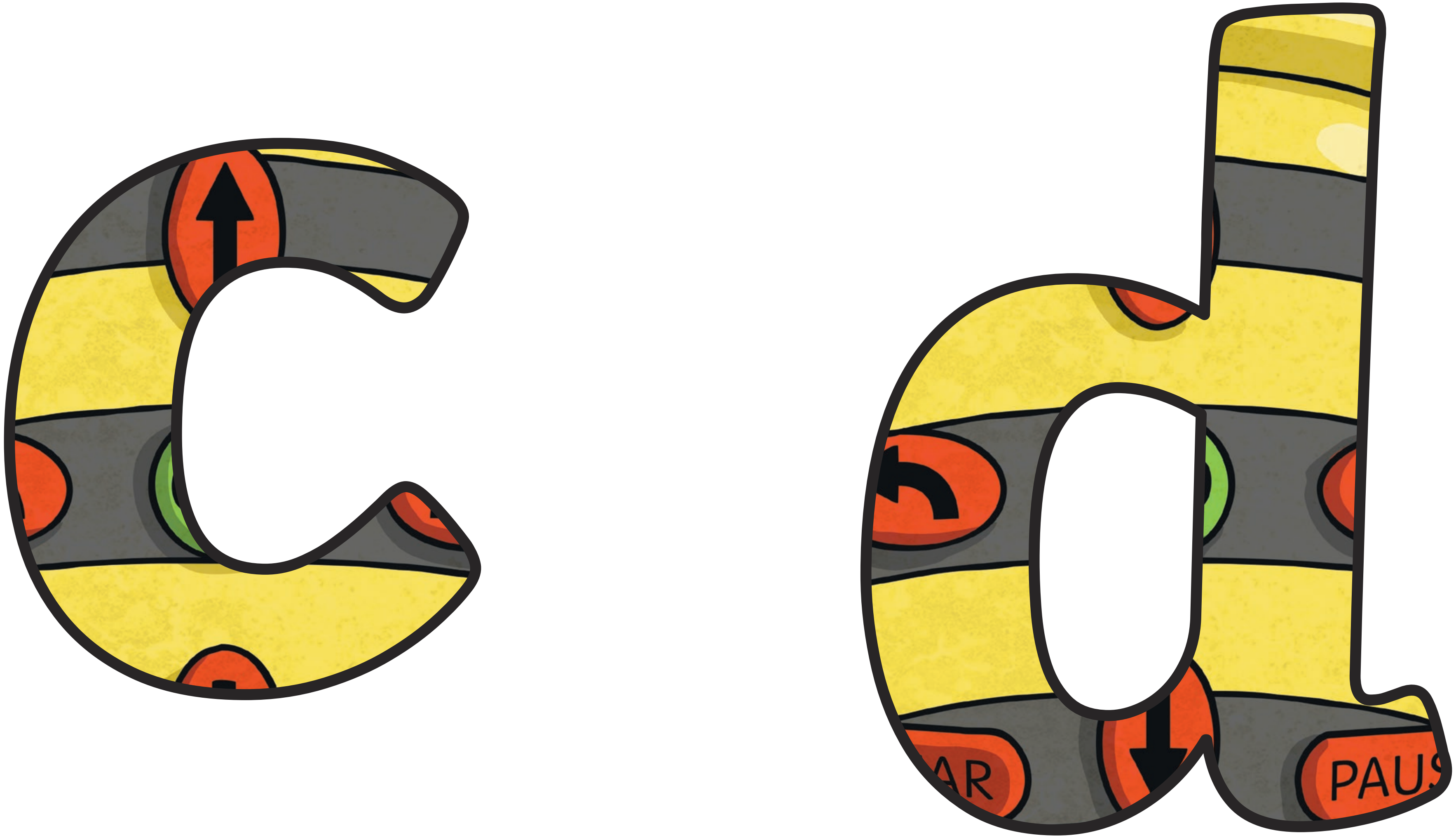

GO

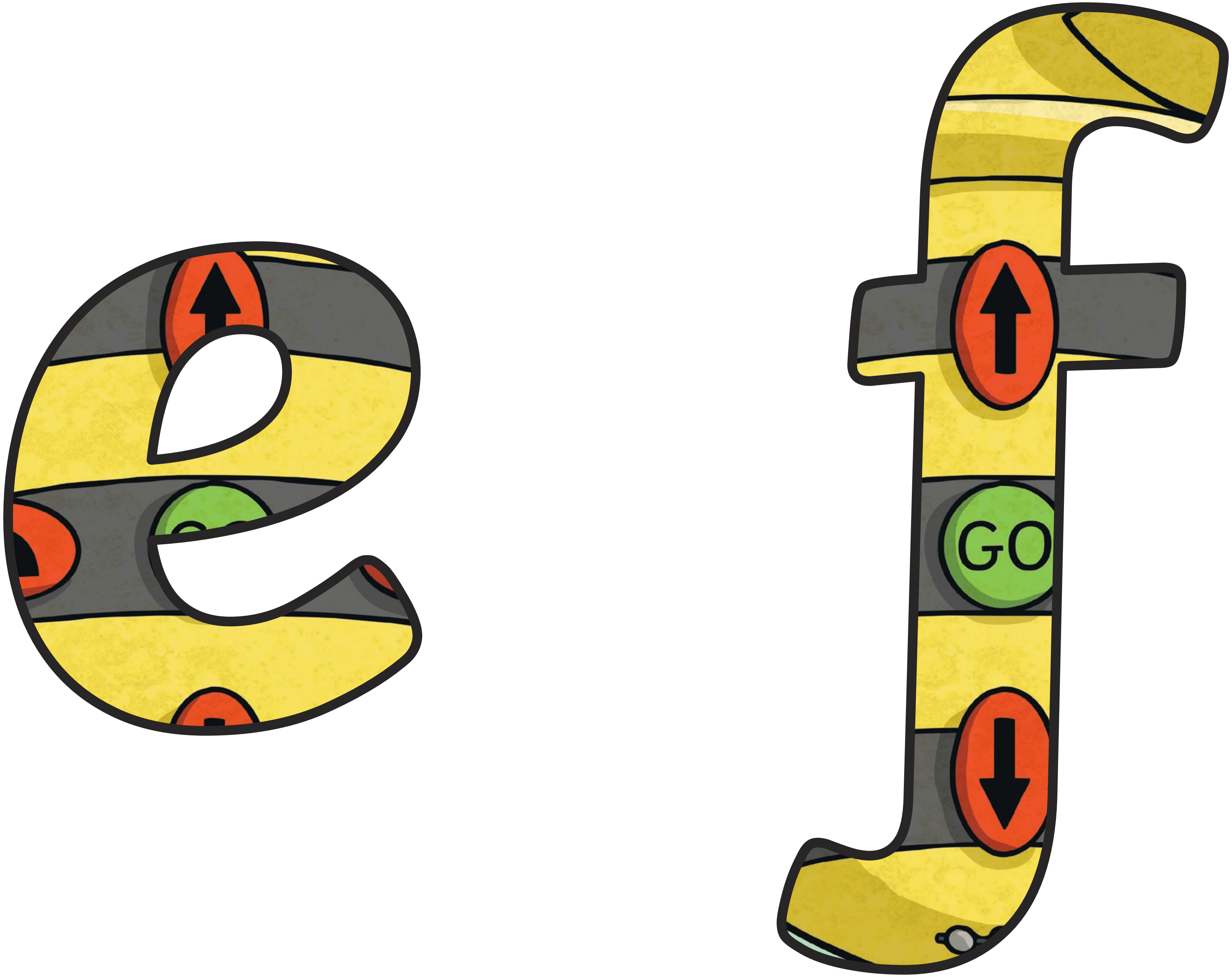






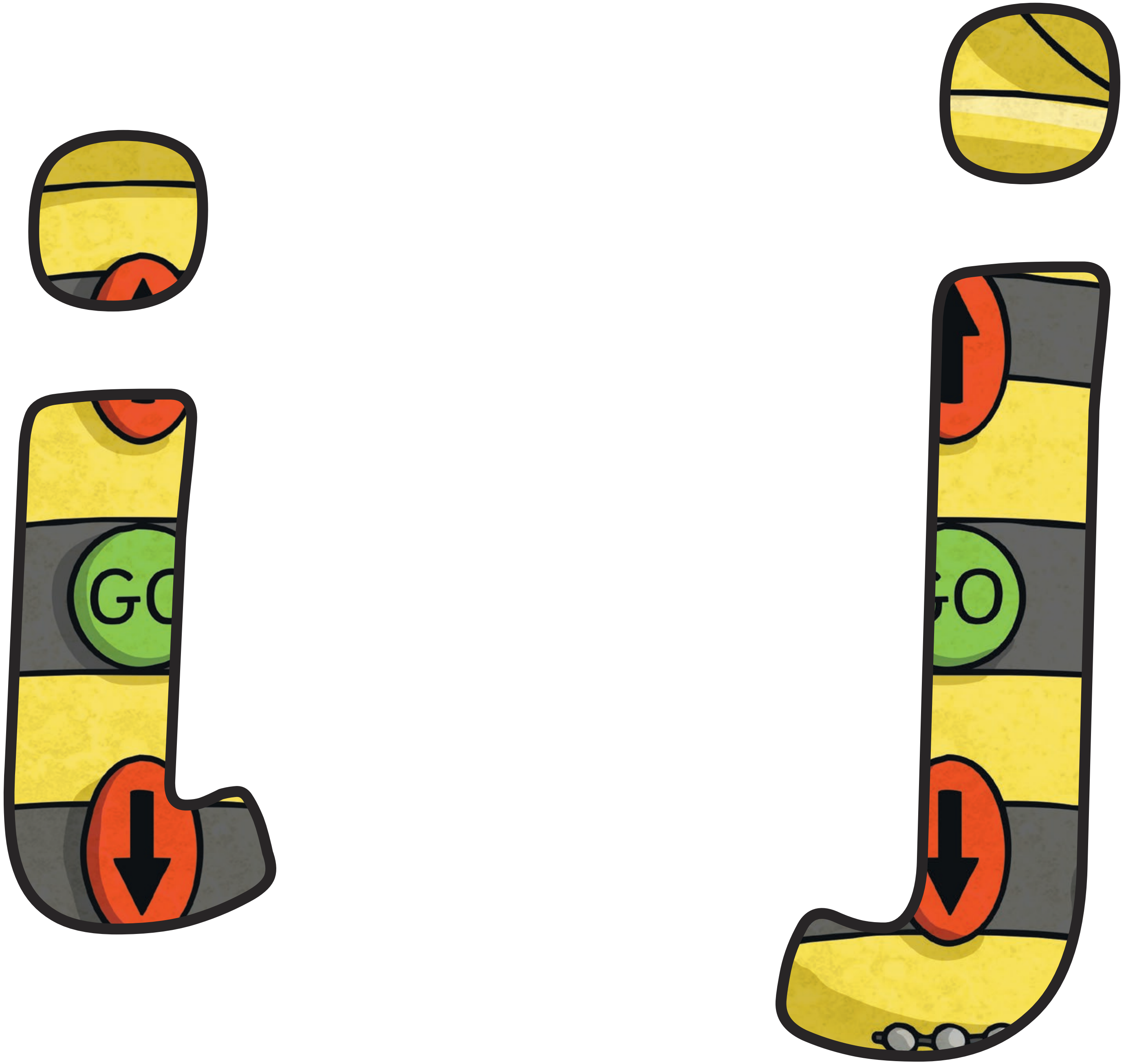
a b

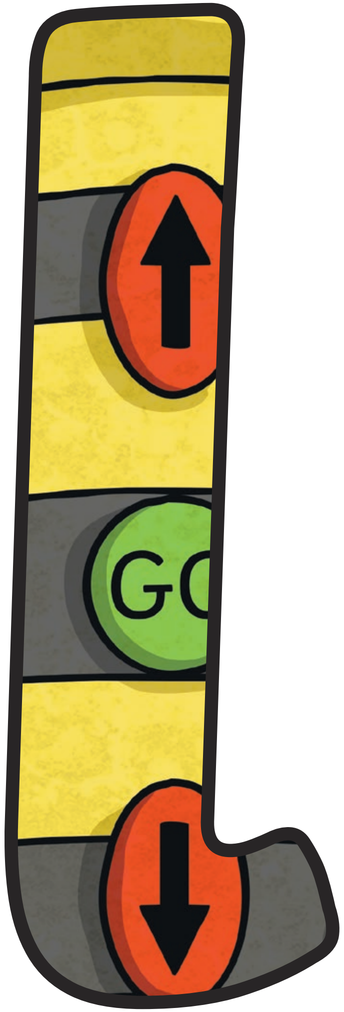

c
d


e
f
GO


g

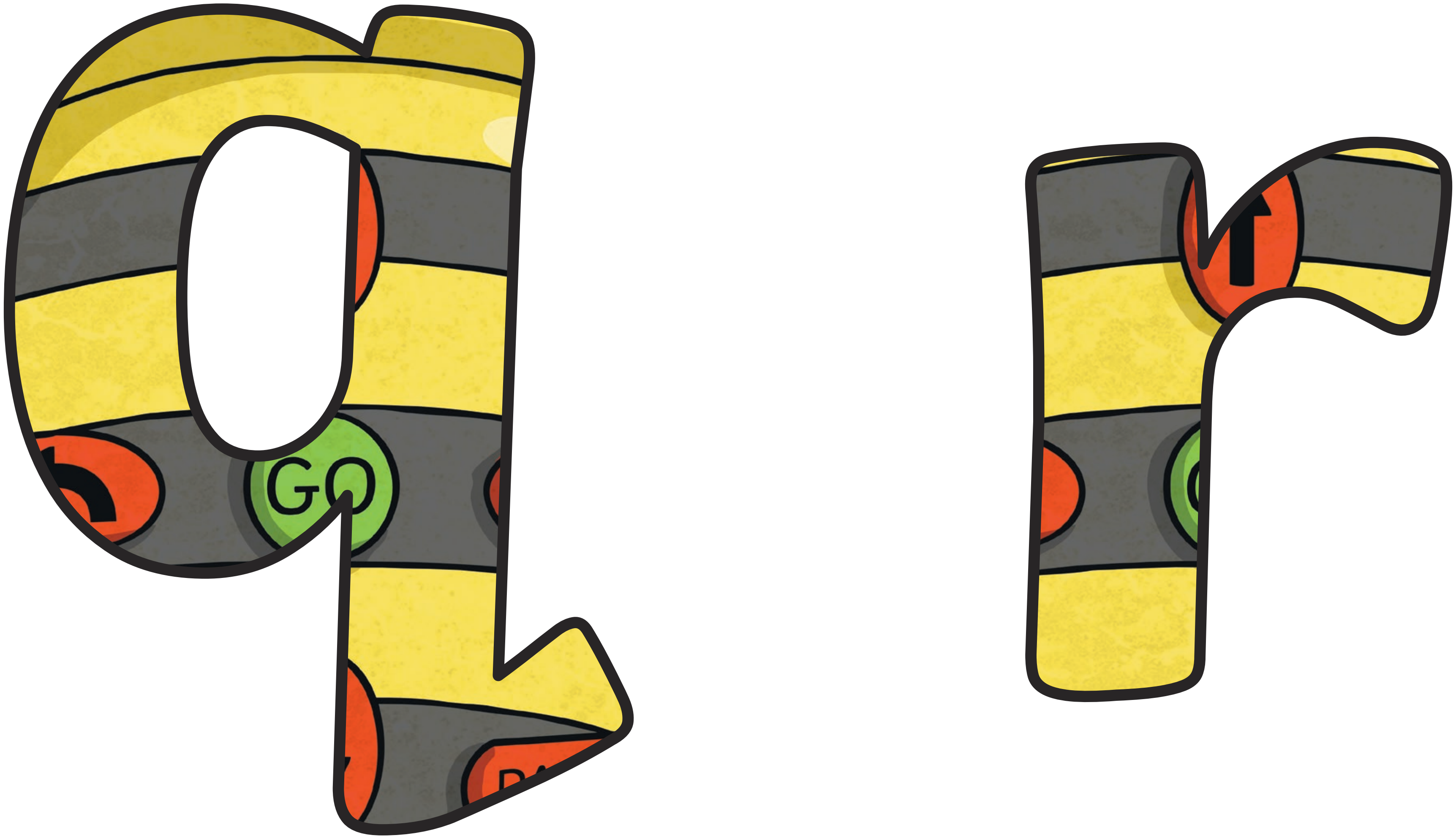

h



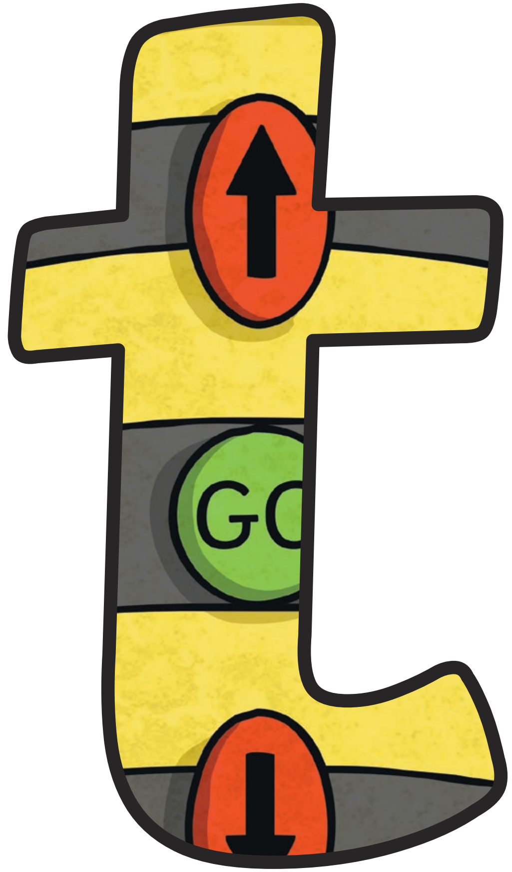



m n

o p


q
r
GO

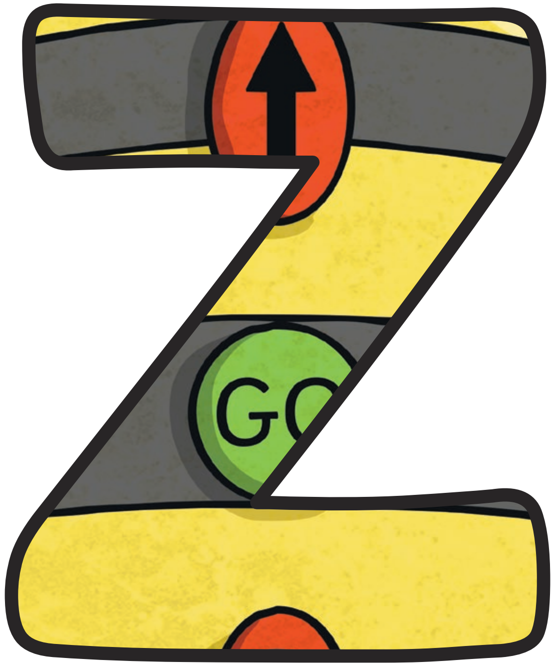

s


t

u v


GO
GO


GO

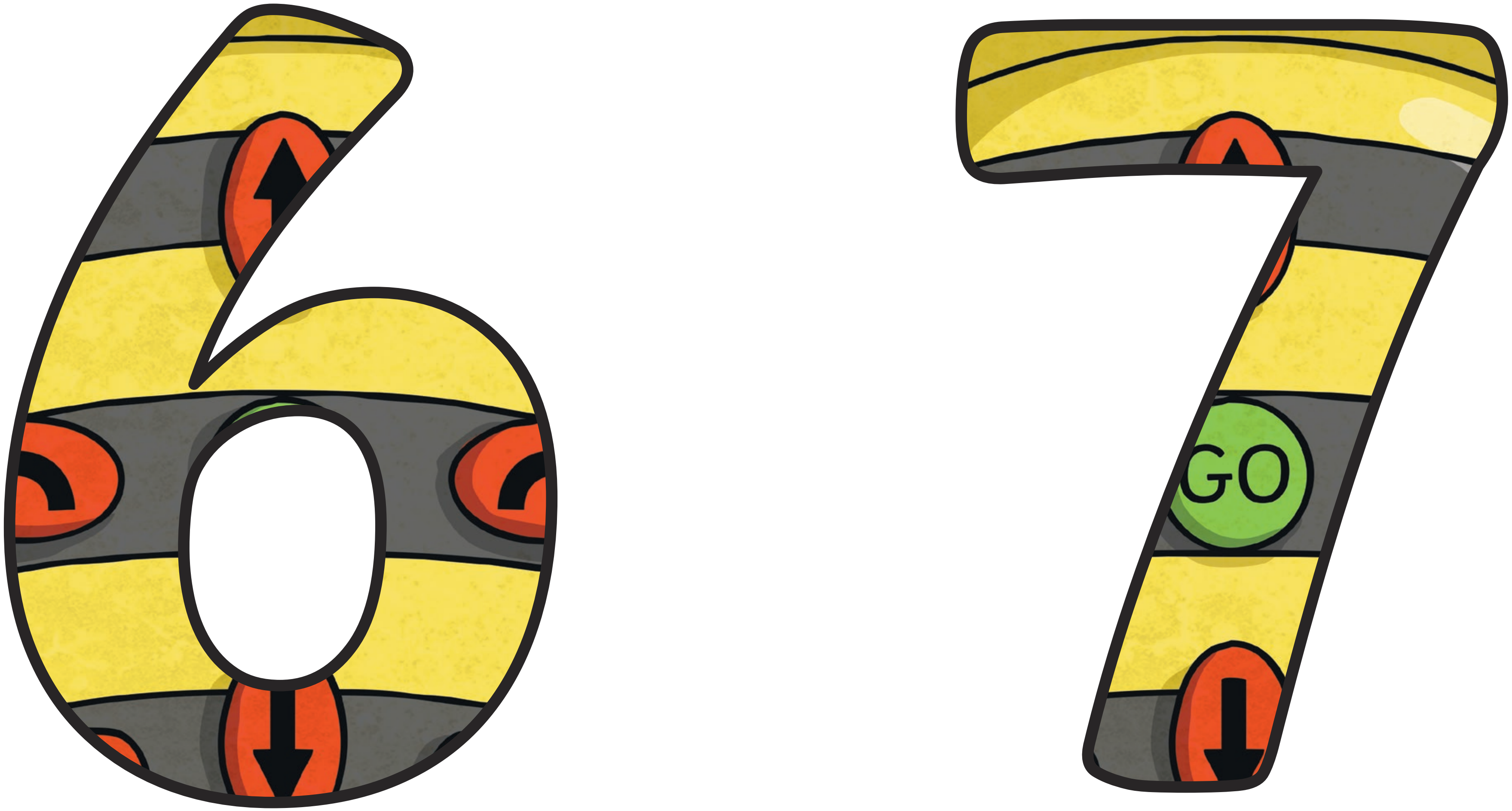


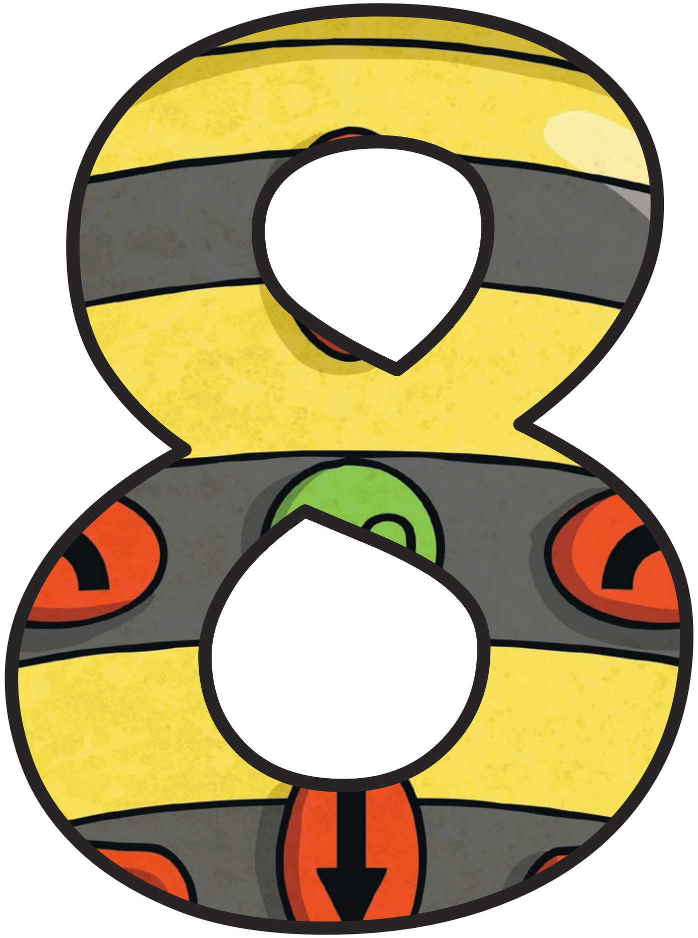
0 1


2
GO

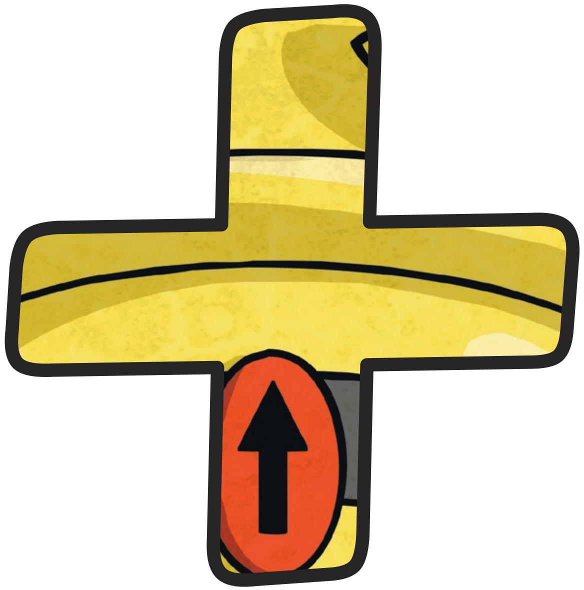

3

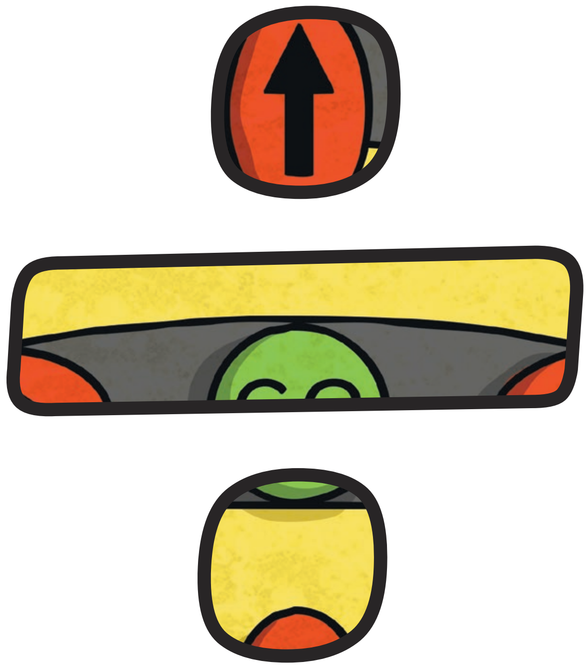
4 5

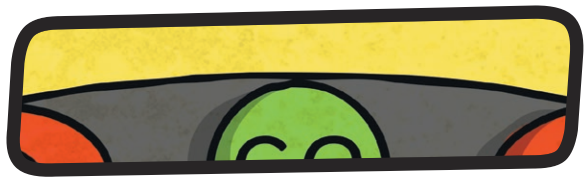

6
7
GO

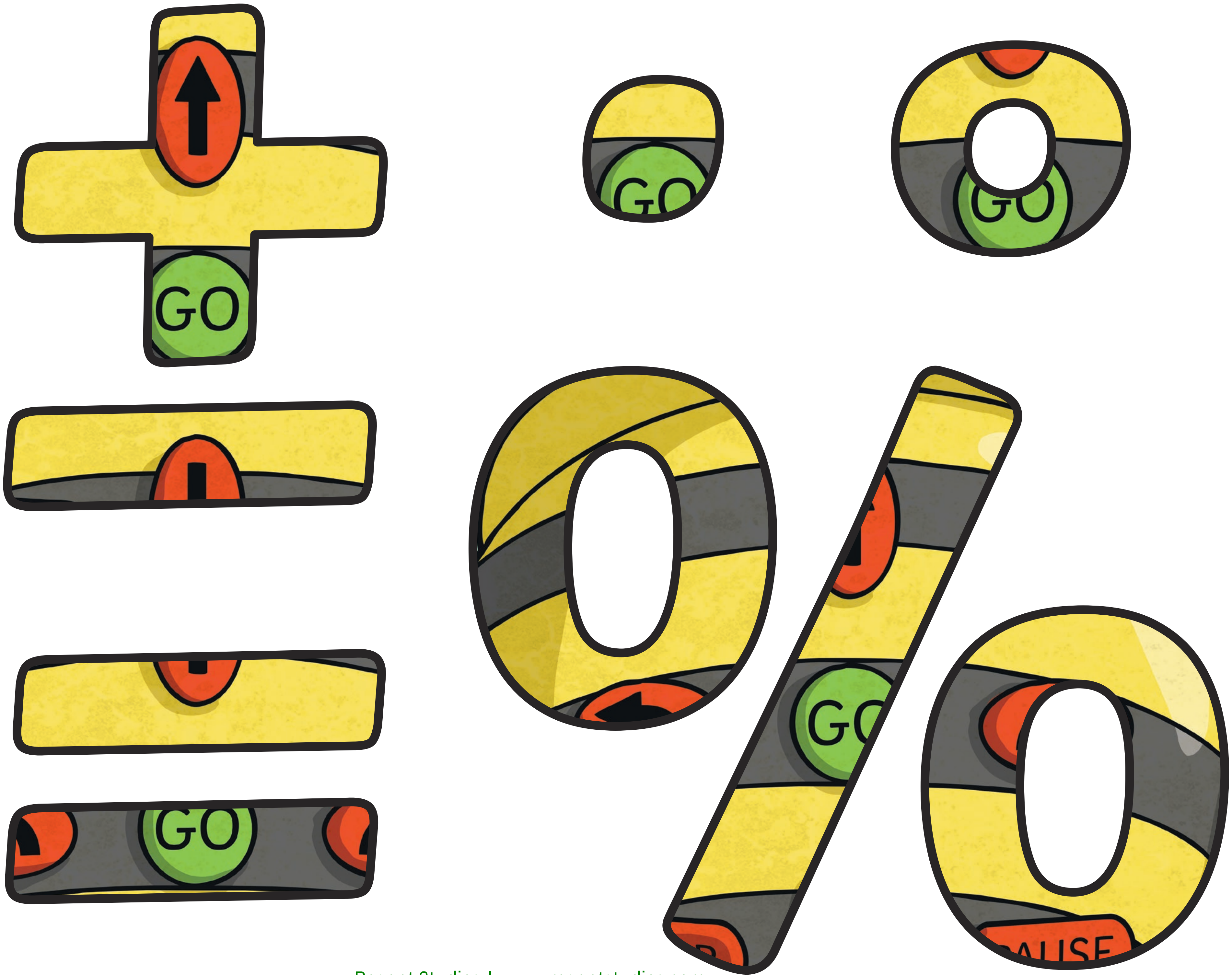

GO

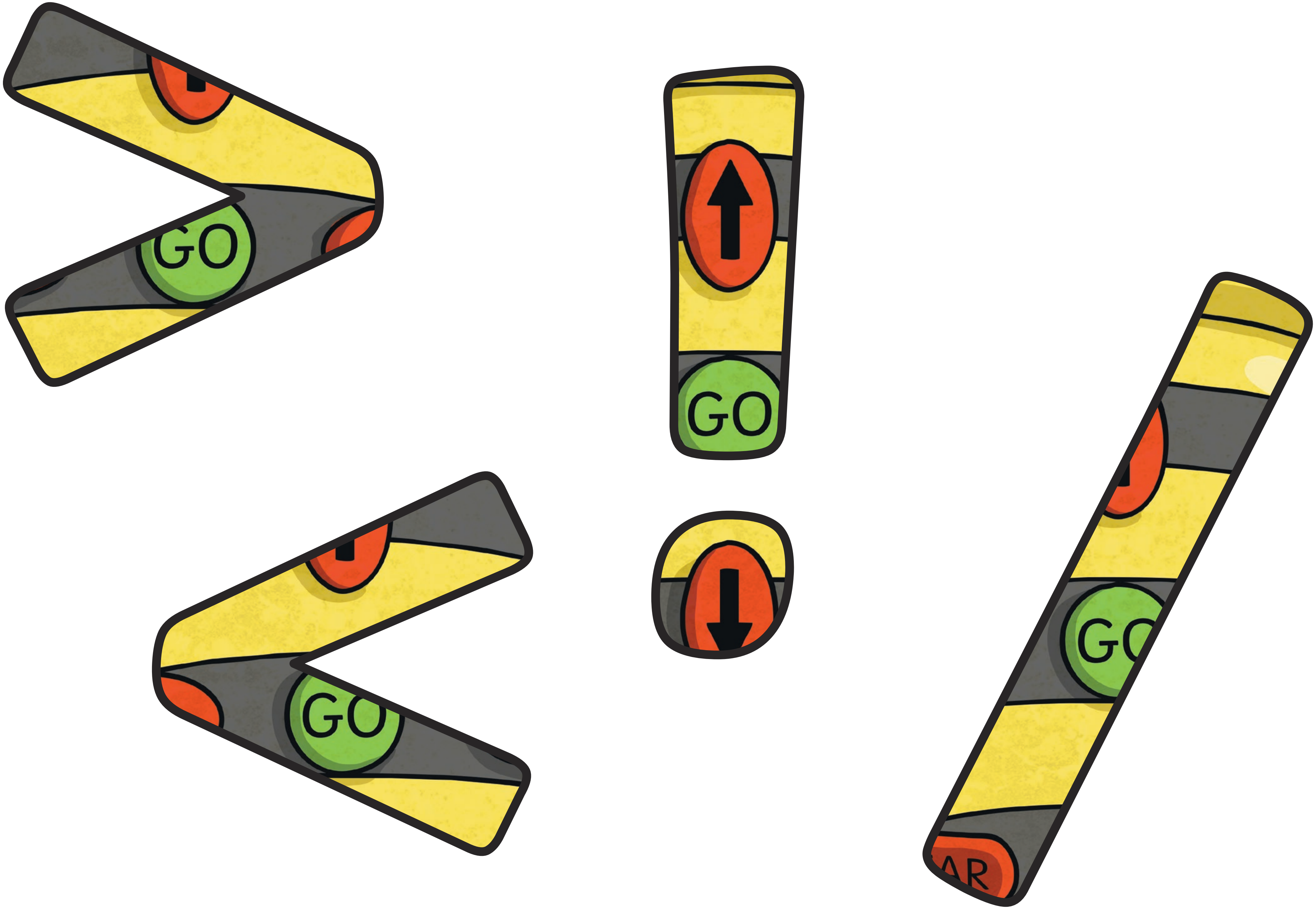
GO

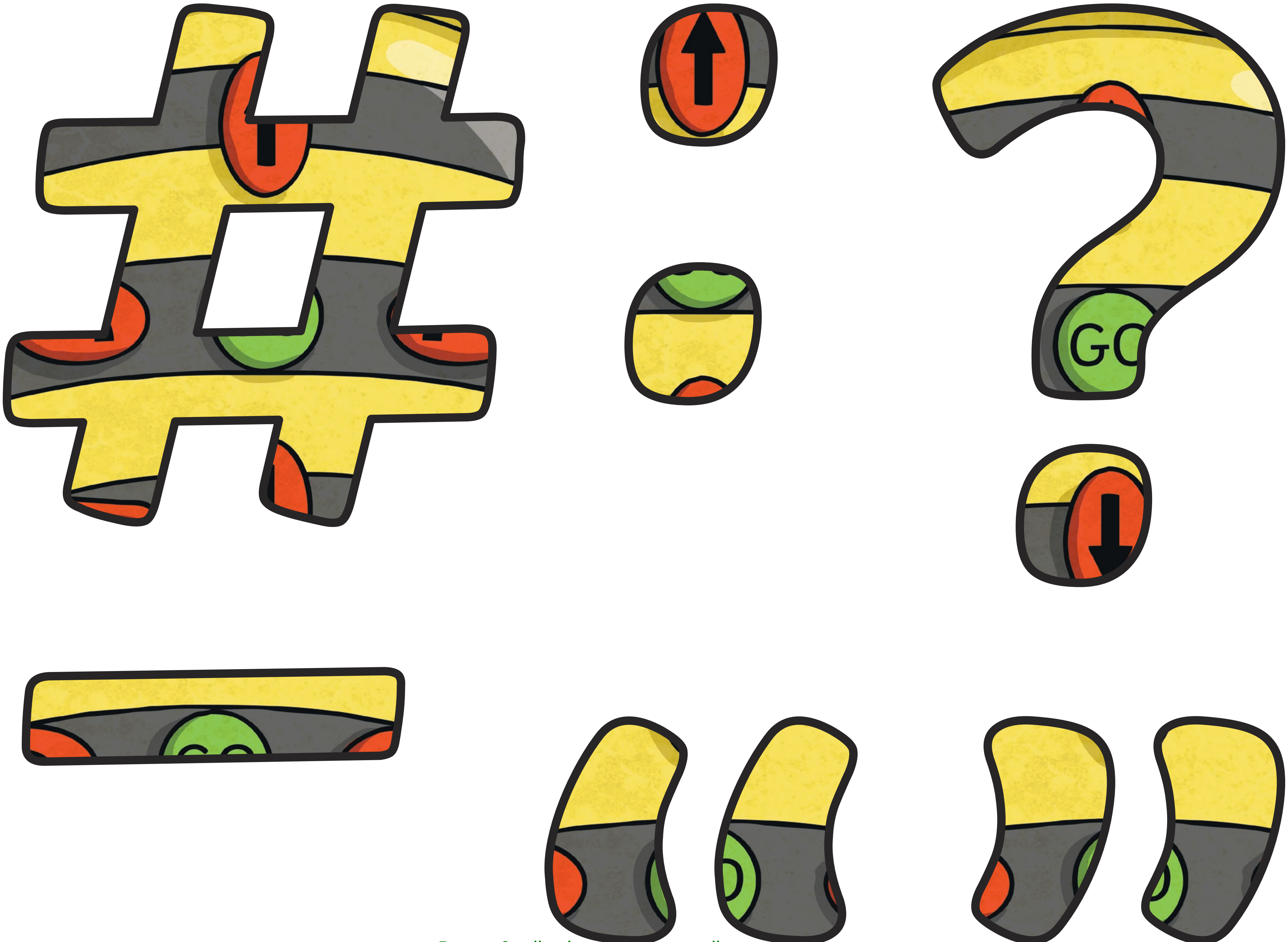


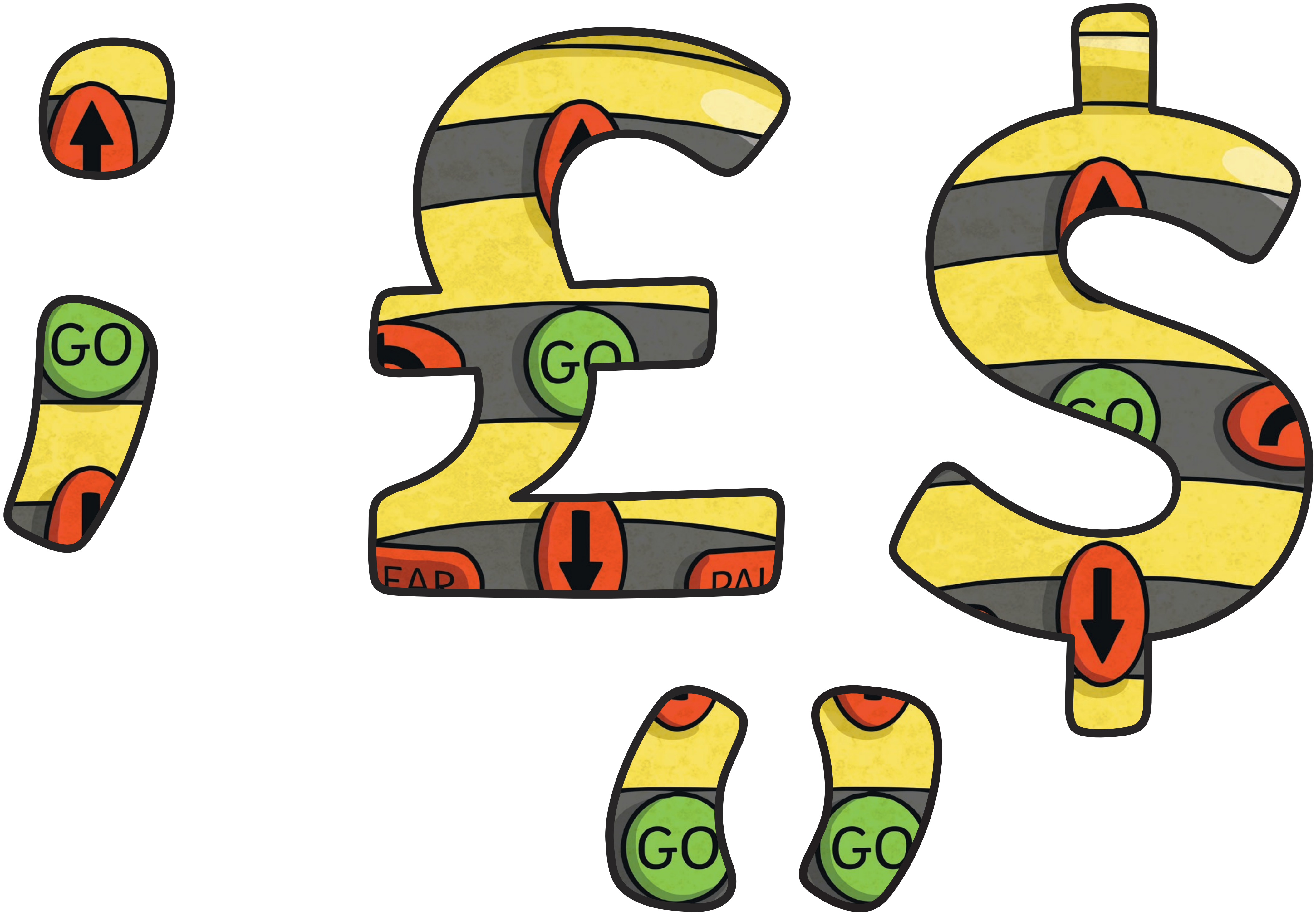


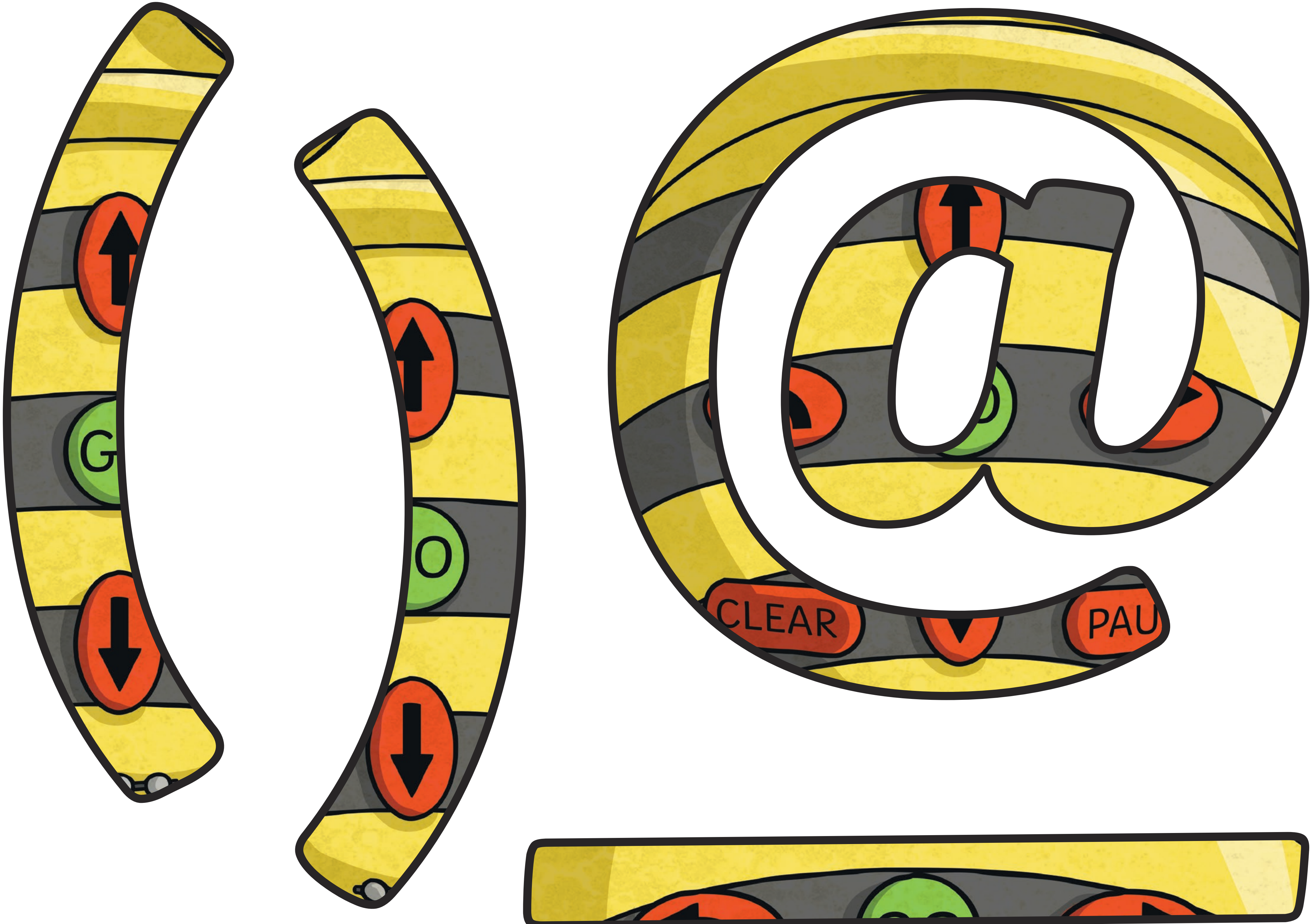


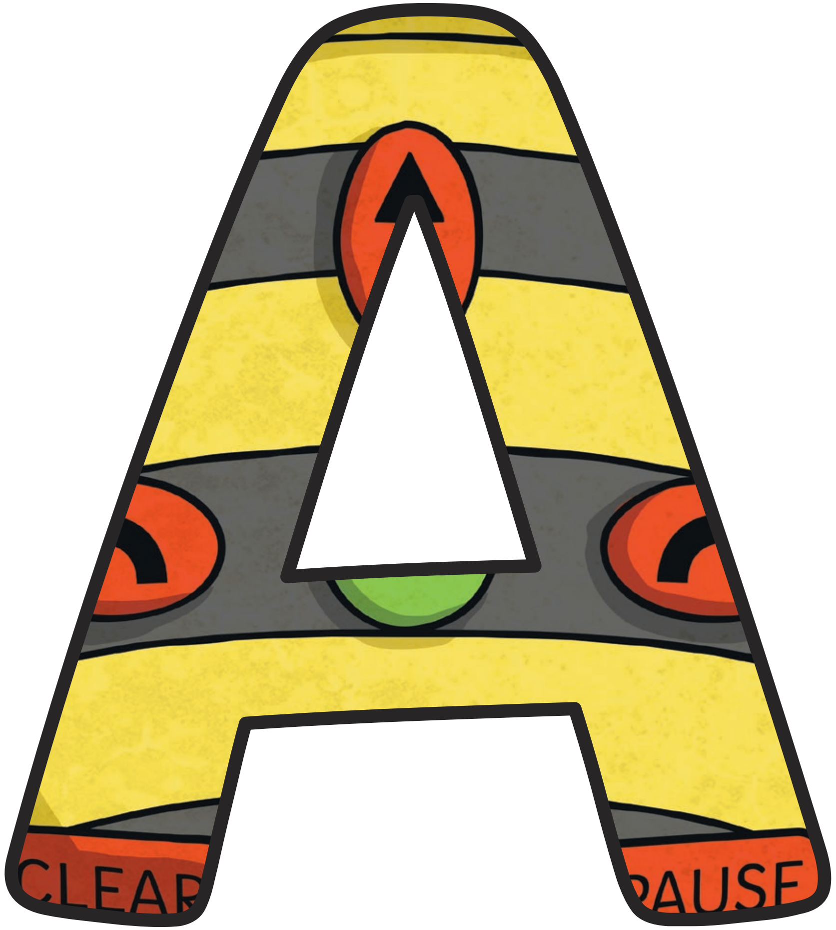

GO
GO
GO
GO
GO
GO
PAUSE


GO
GO
GO
GO

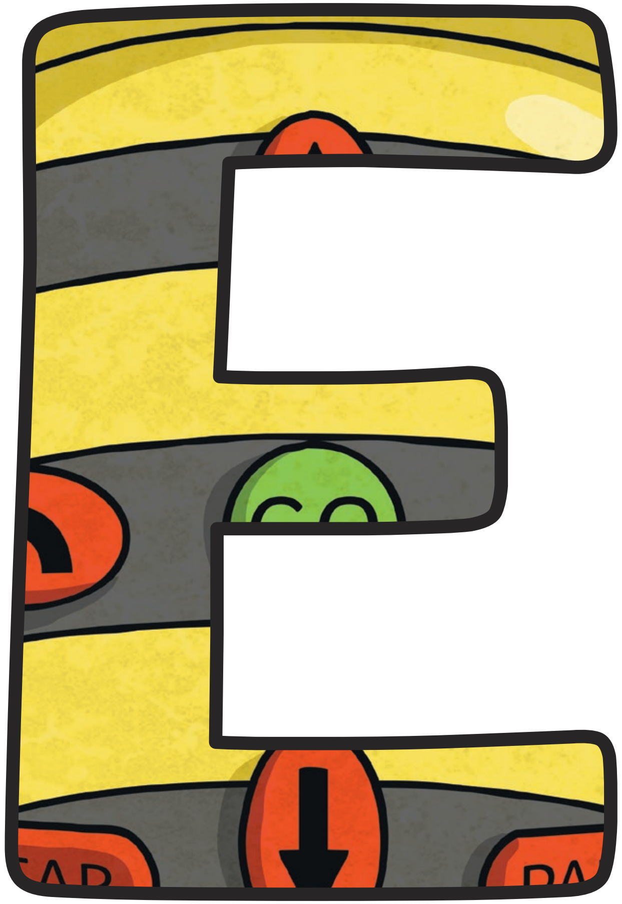


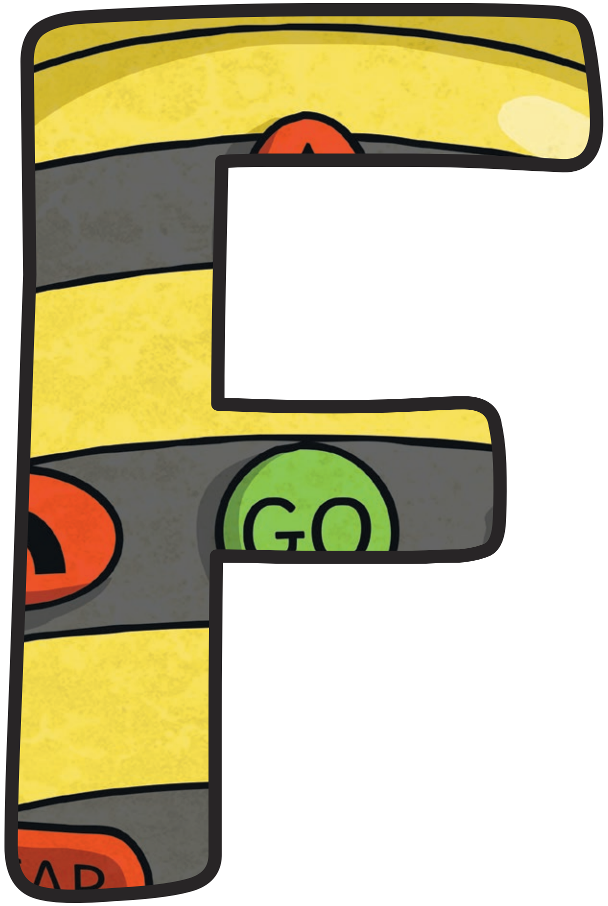

GO
GO
GO
GO


CLEAR

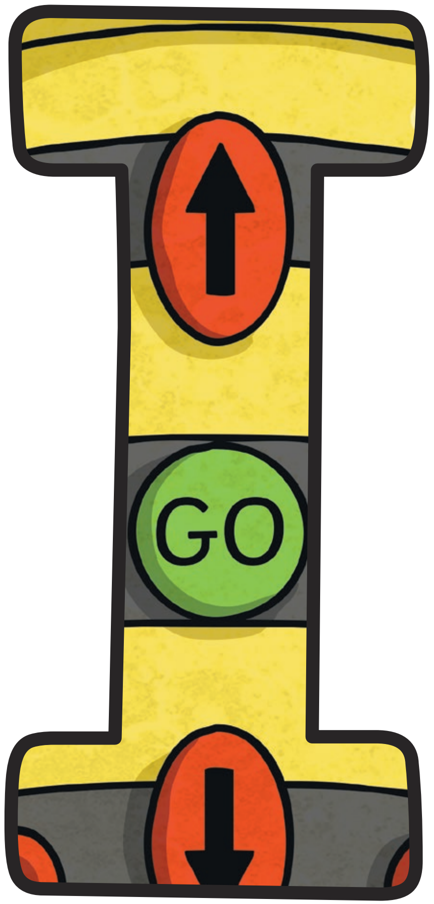


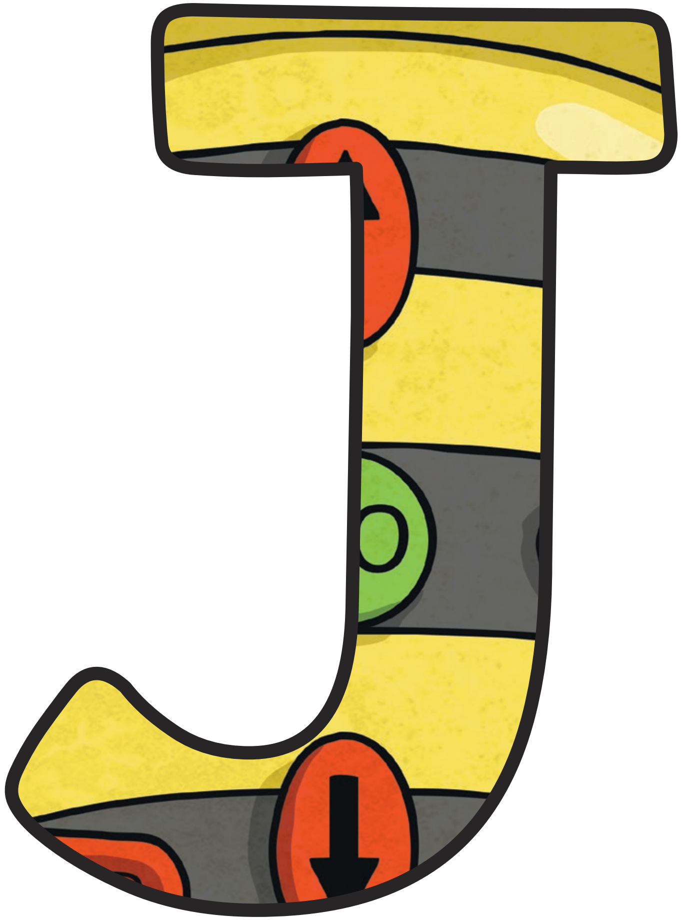
C D

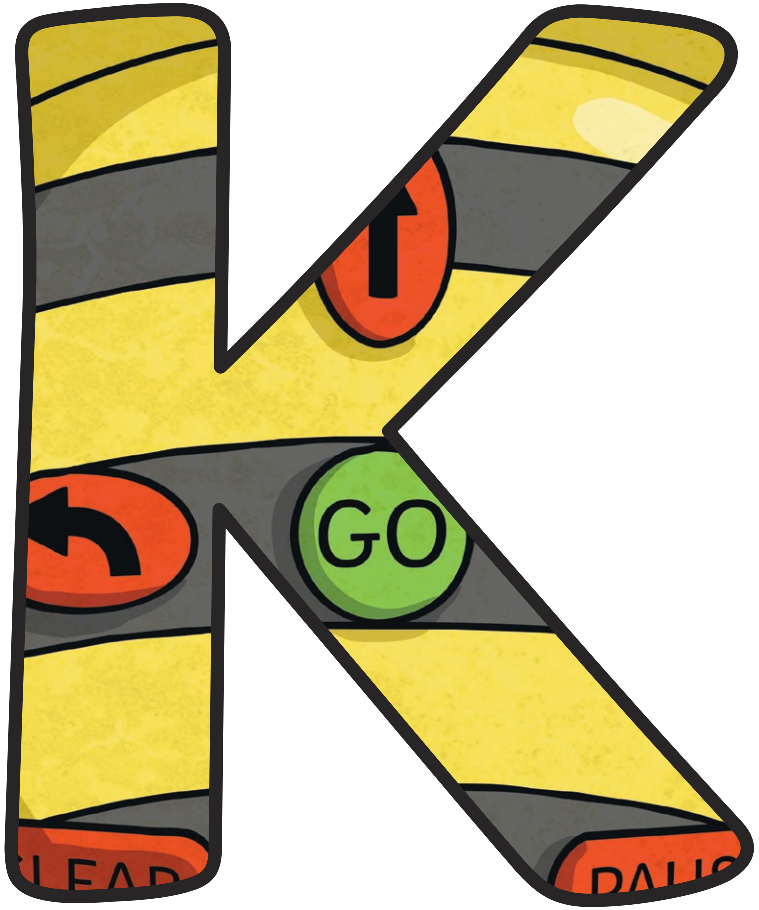

E

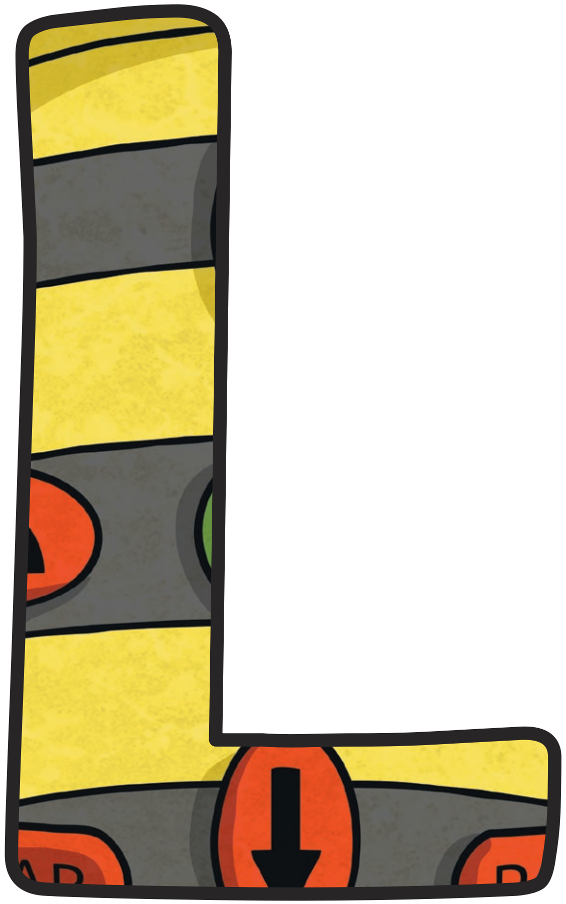

F
GO

G H


GO

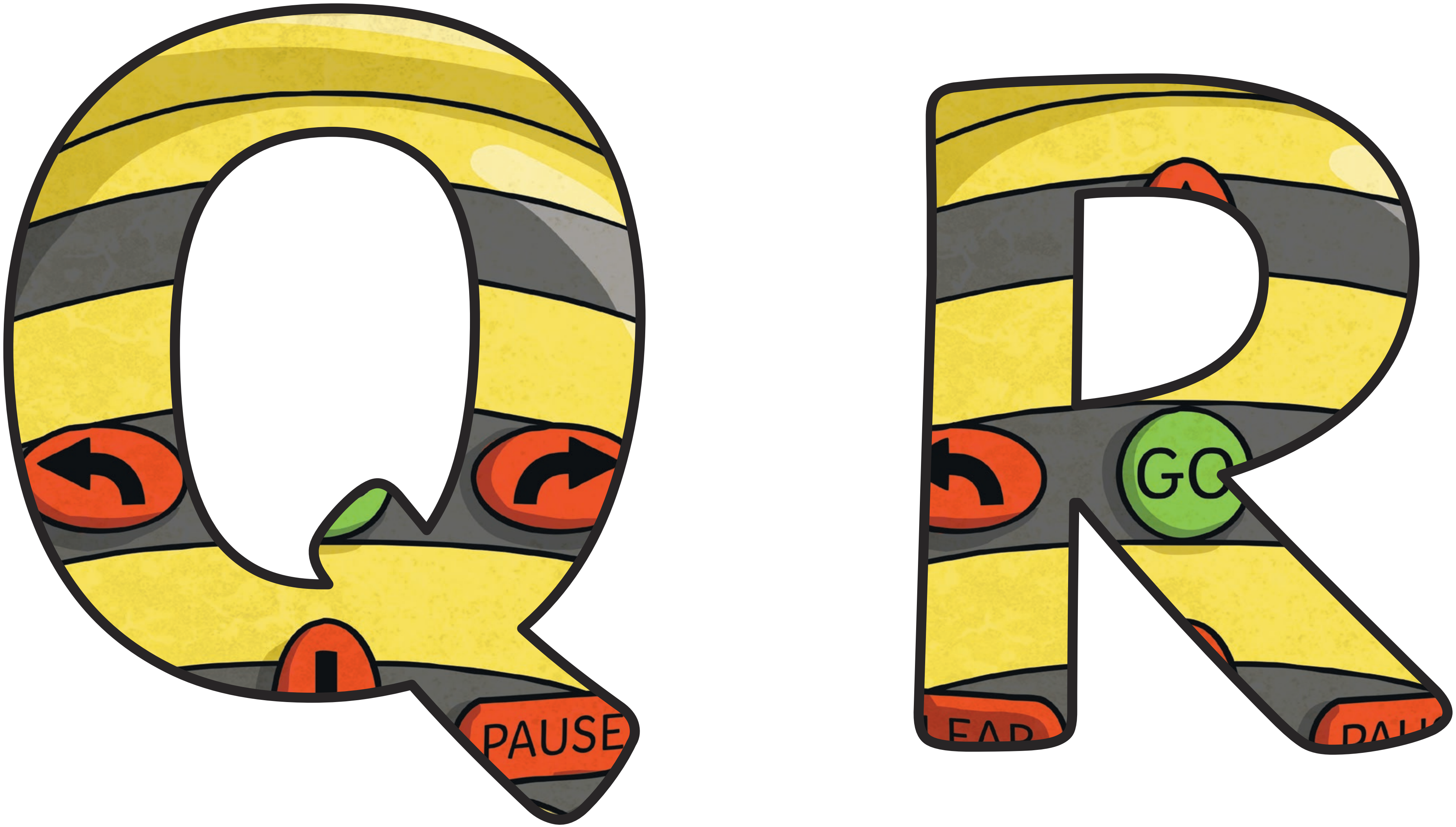


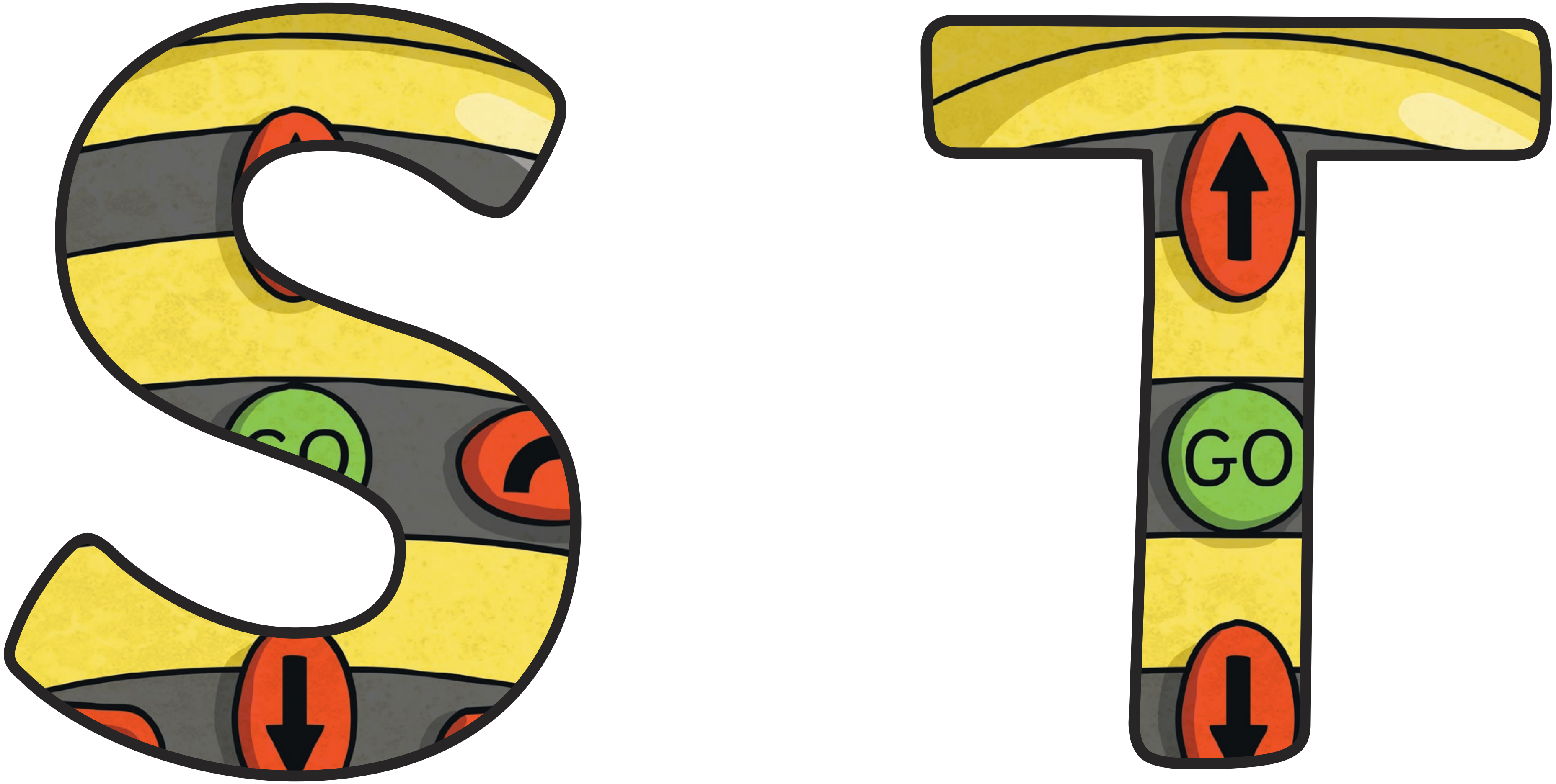

K
GO


L

M N

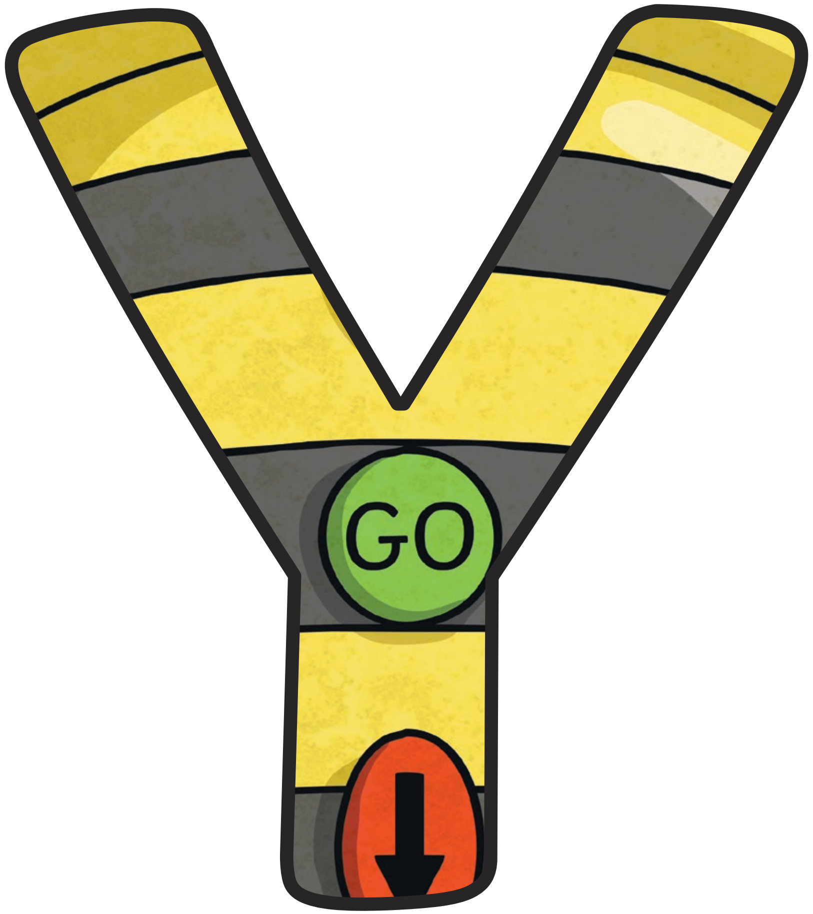

O P
GO


Q
PAUSE
R
GO

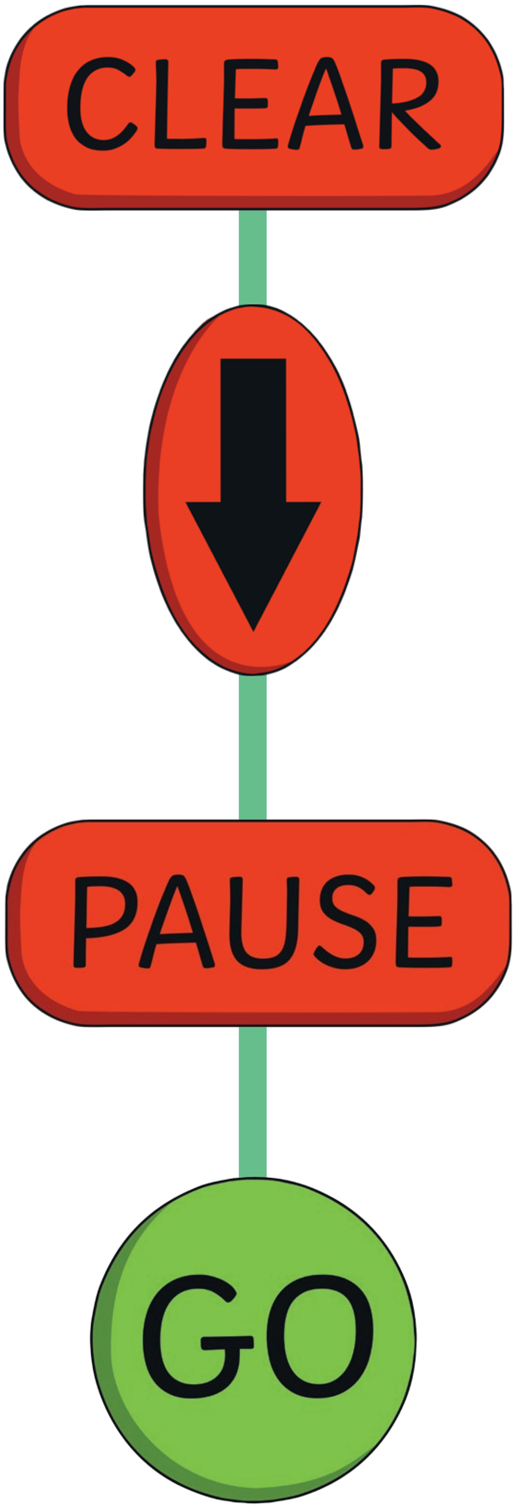

S
T
GO
GO


u v


GO


GO

algorithm



code

0110110101010101

0110101101010101

0101011011100101

0110110101010101

0011010111010101

0110101101010101

0110101101010101

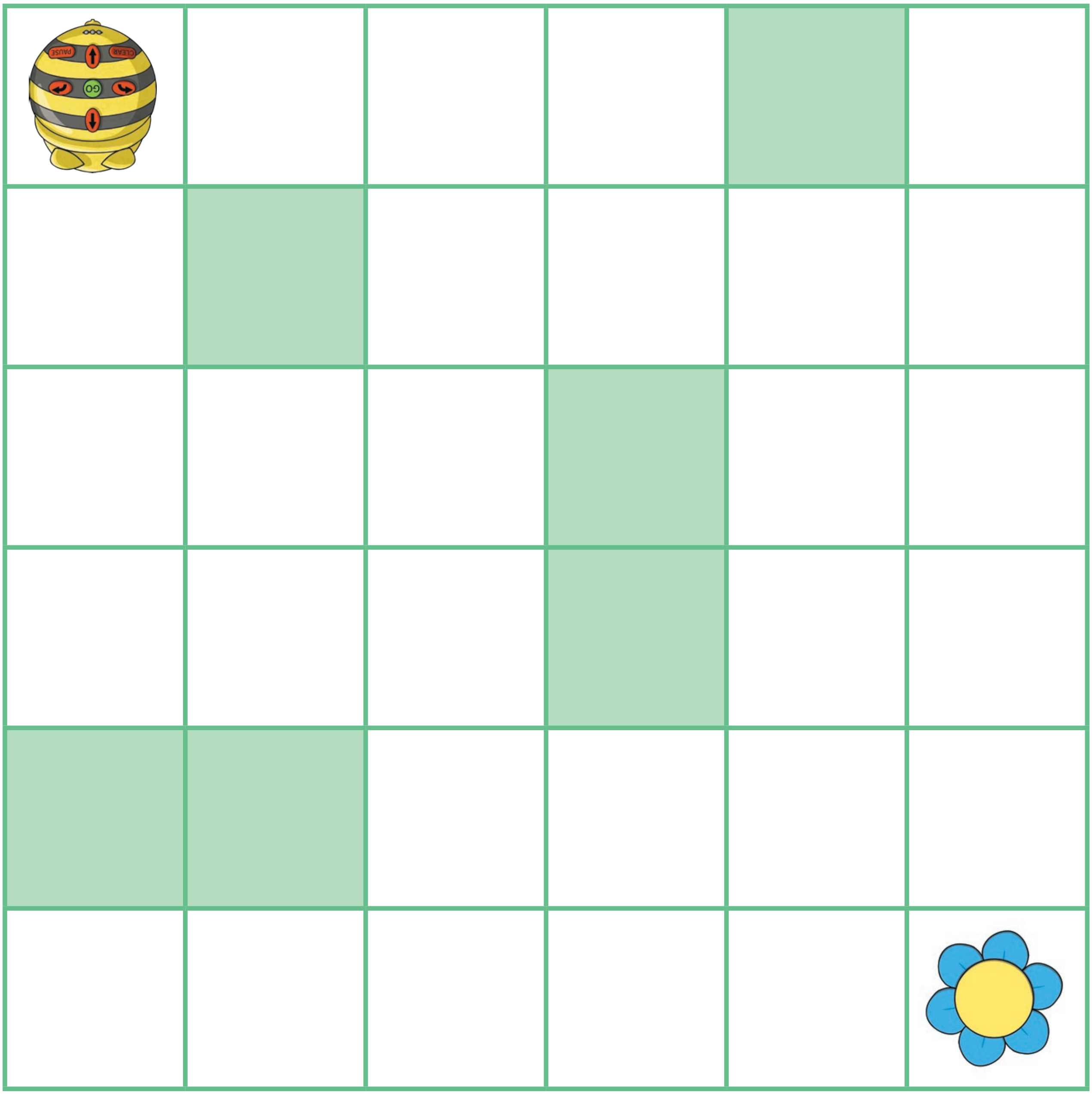
0011010111010101

0110101101010101

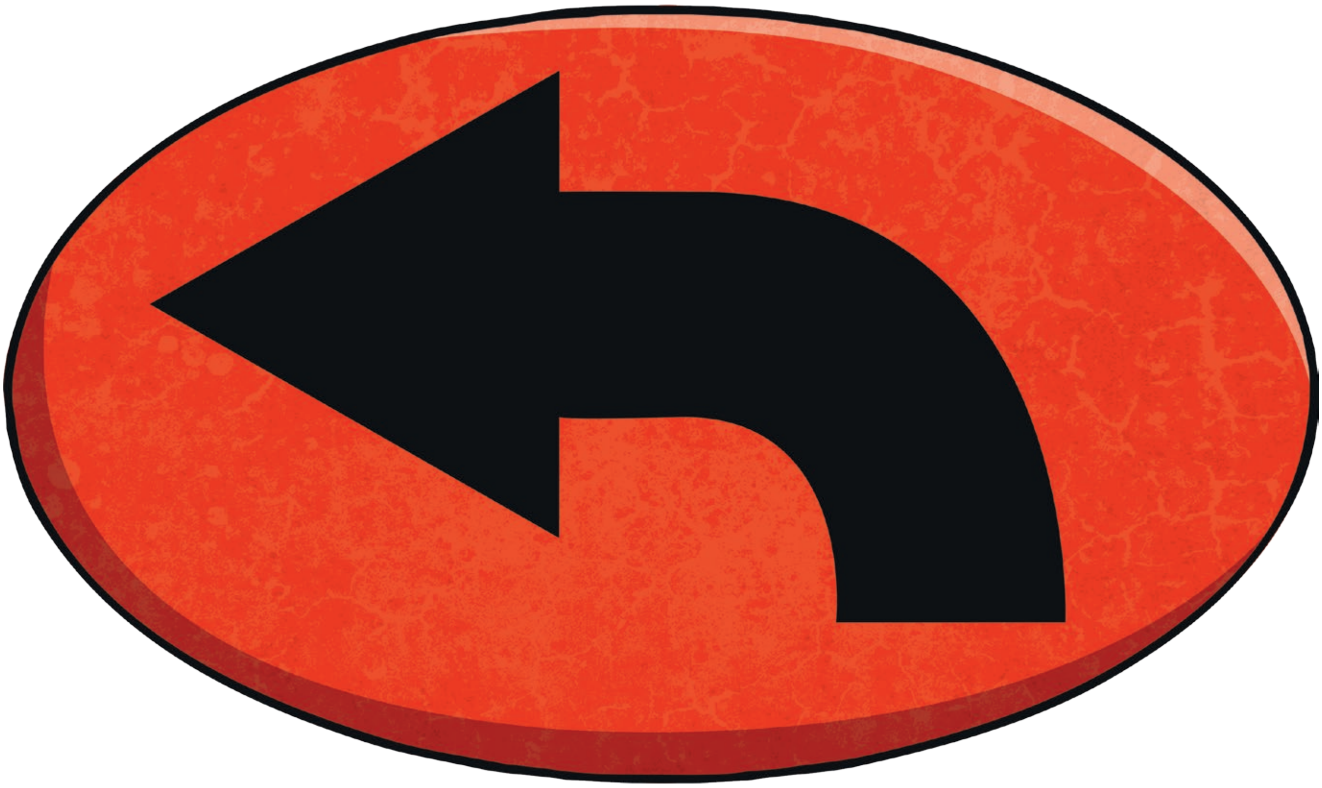
0110110101010101

Instructions

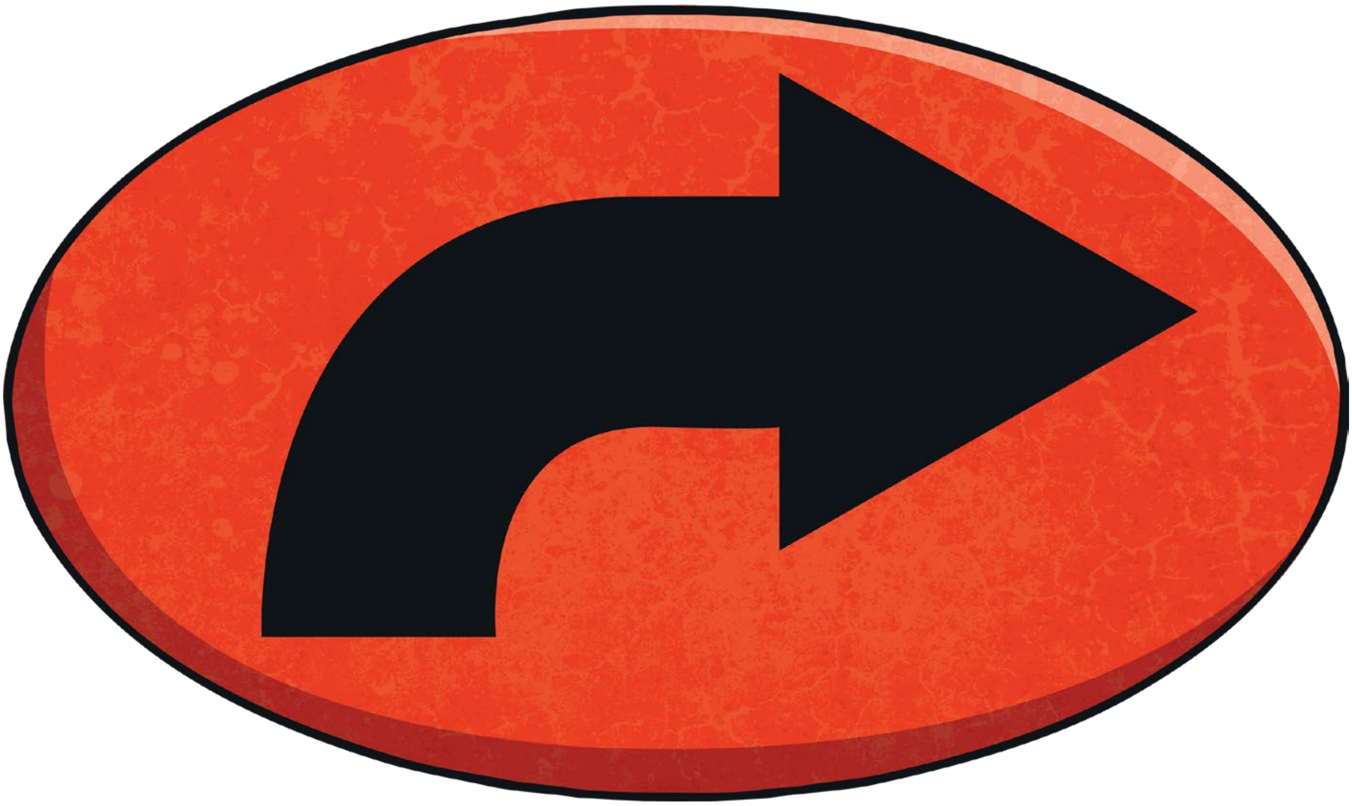
Get the Bee-Bot to the flower.



left



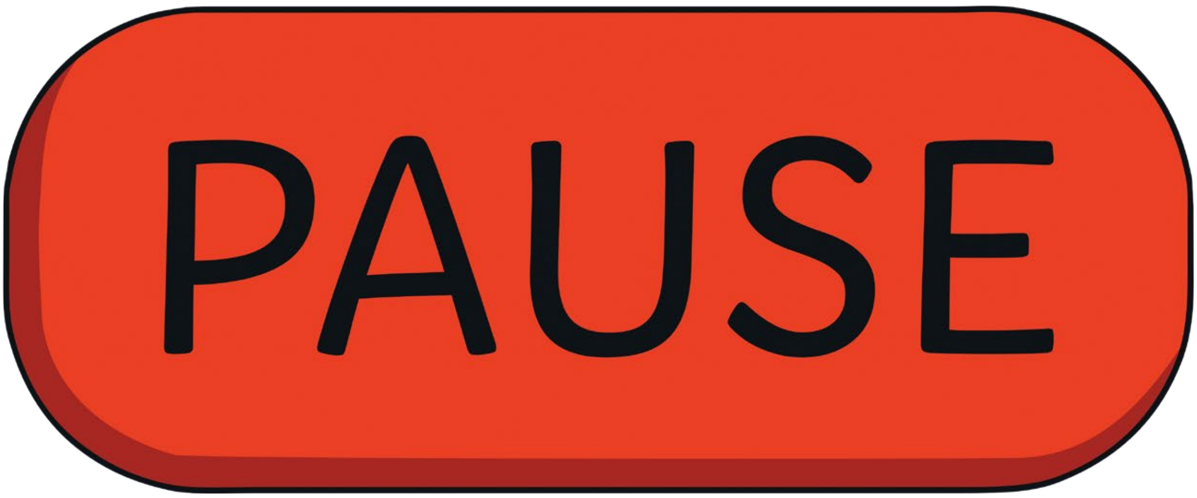
right



forward

backward

pause

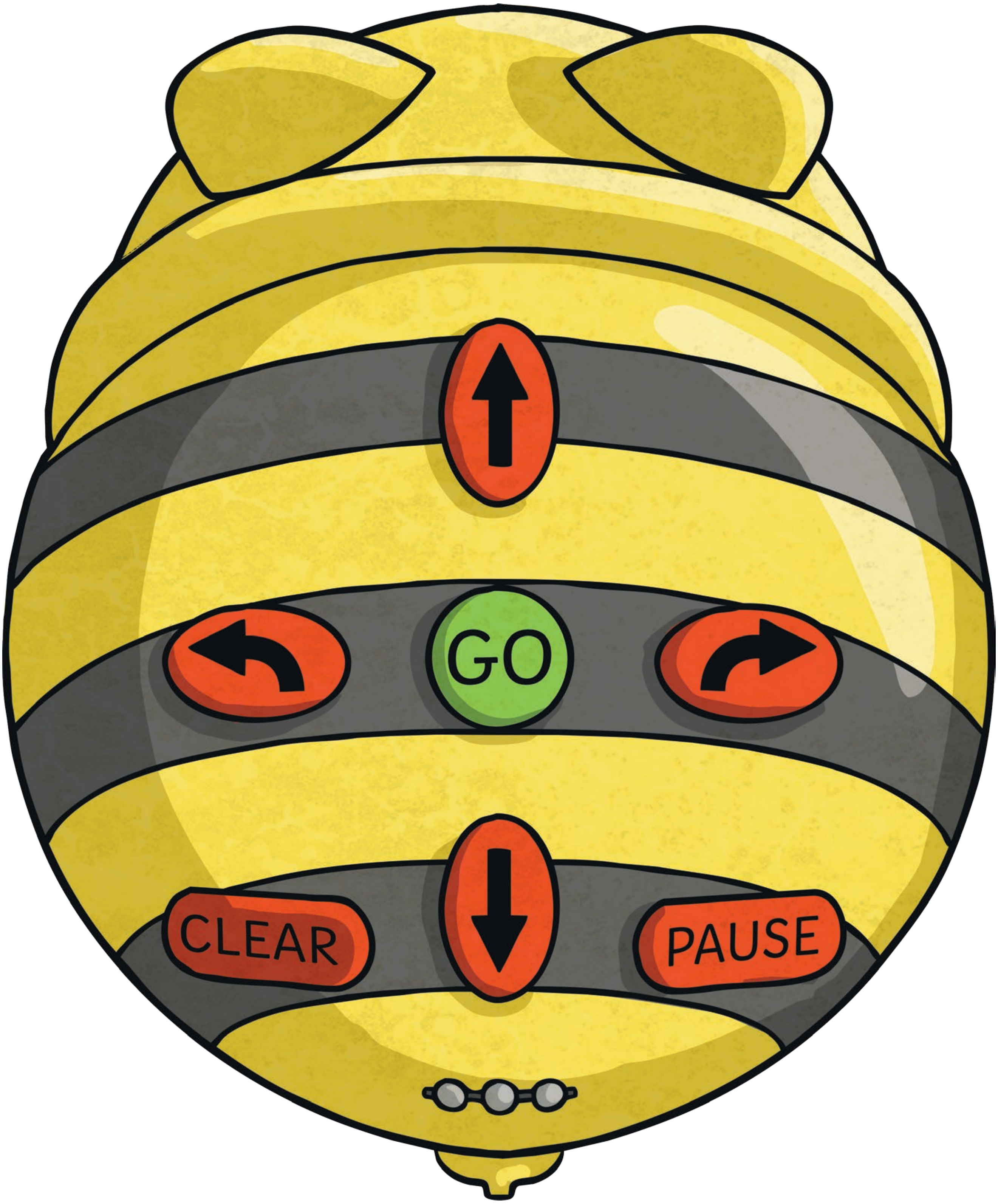


clear

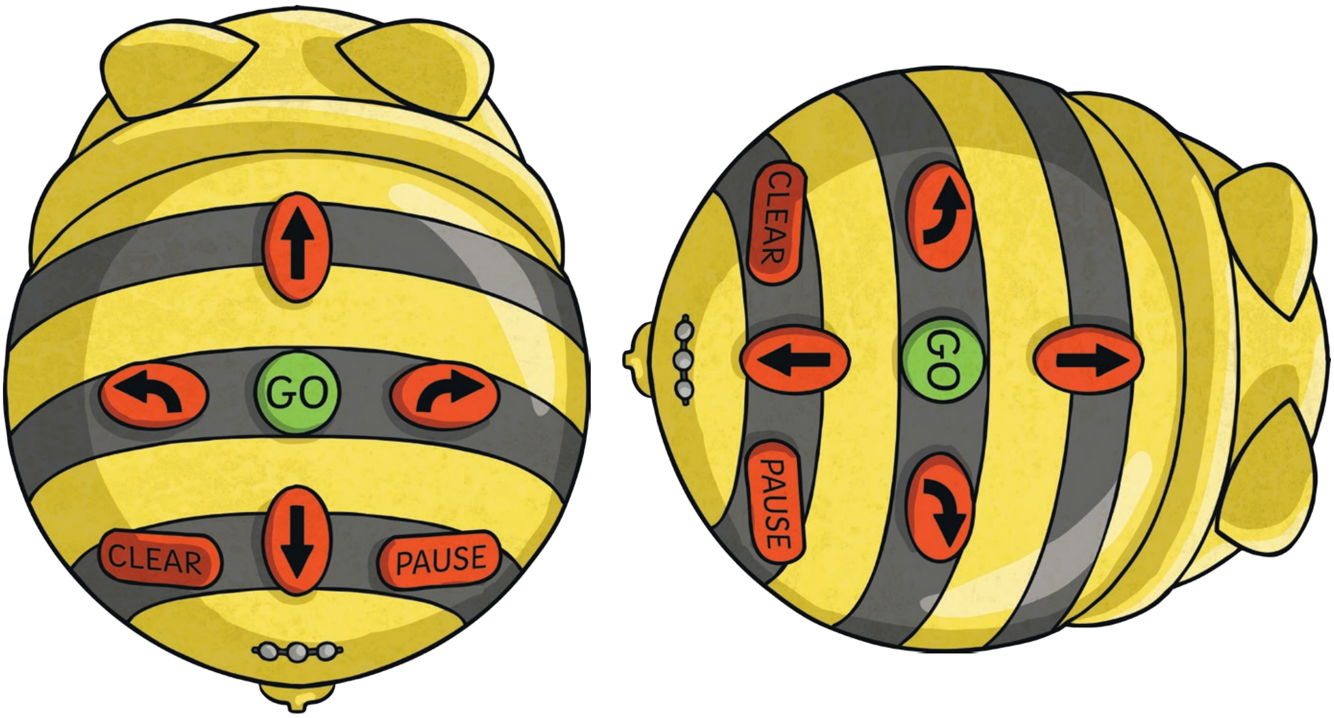
go

program

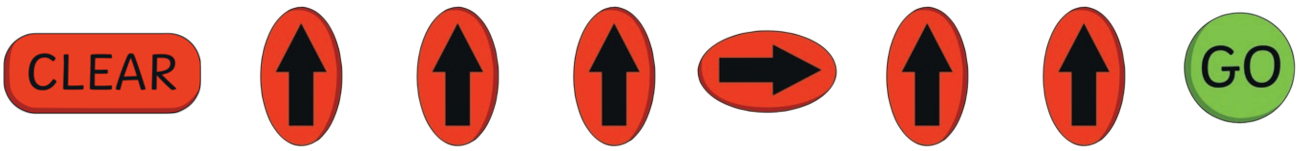
Bee-Bot

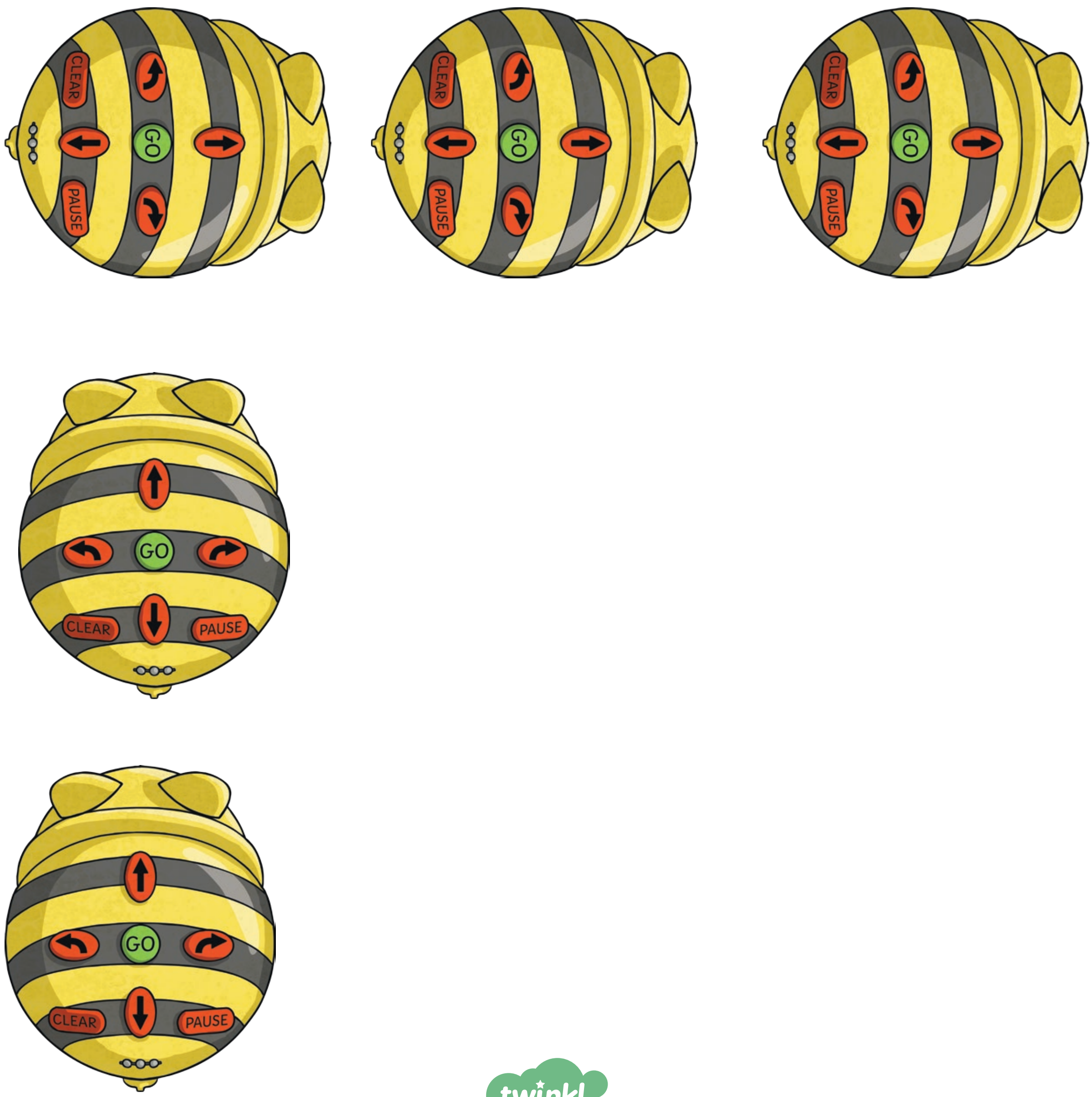


turn

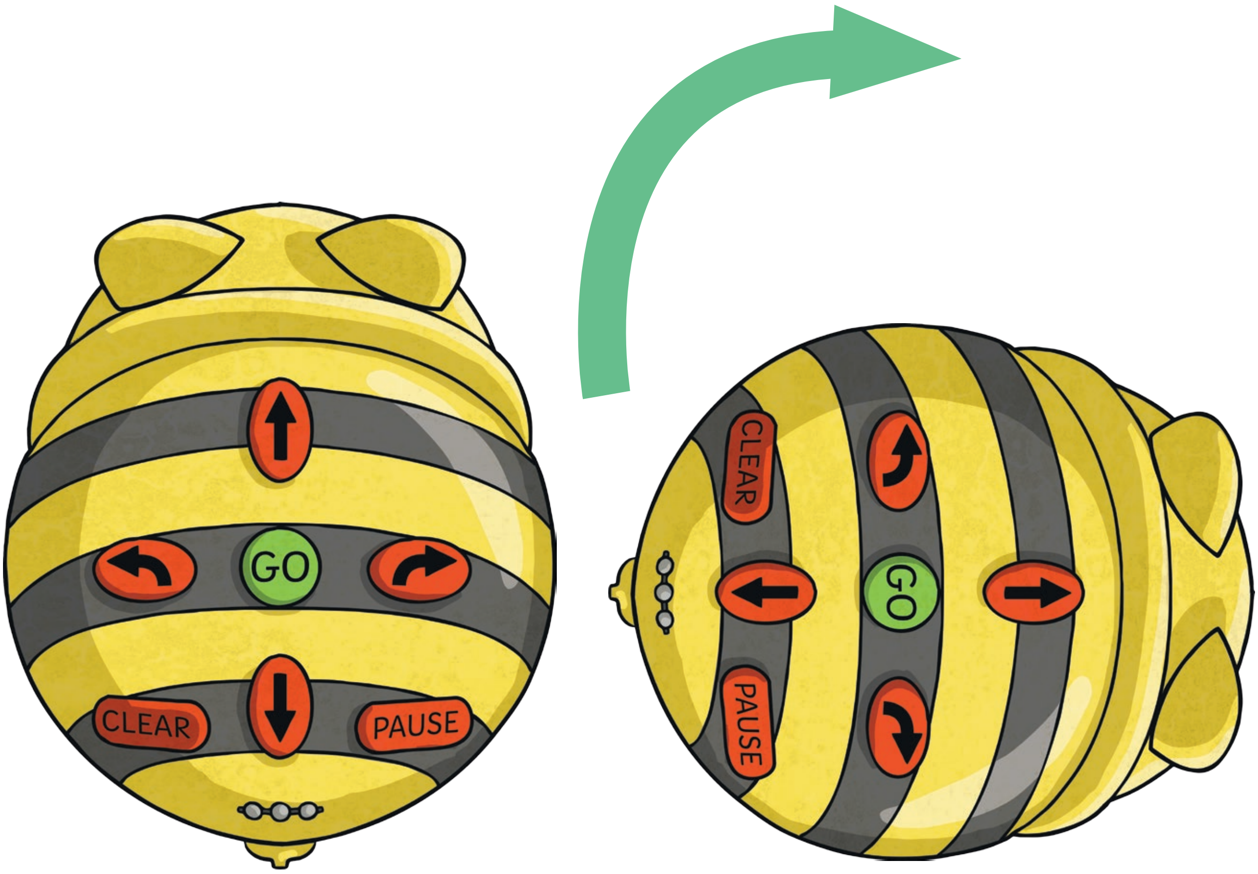


sequence

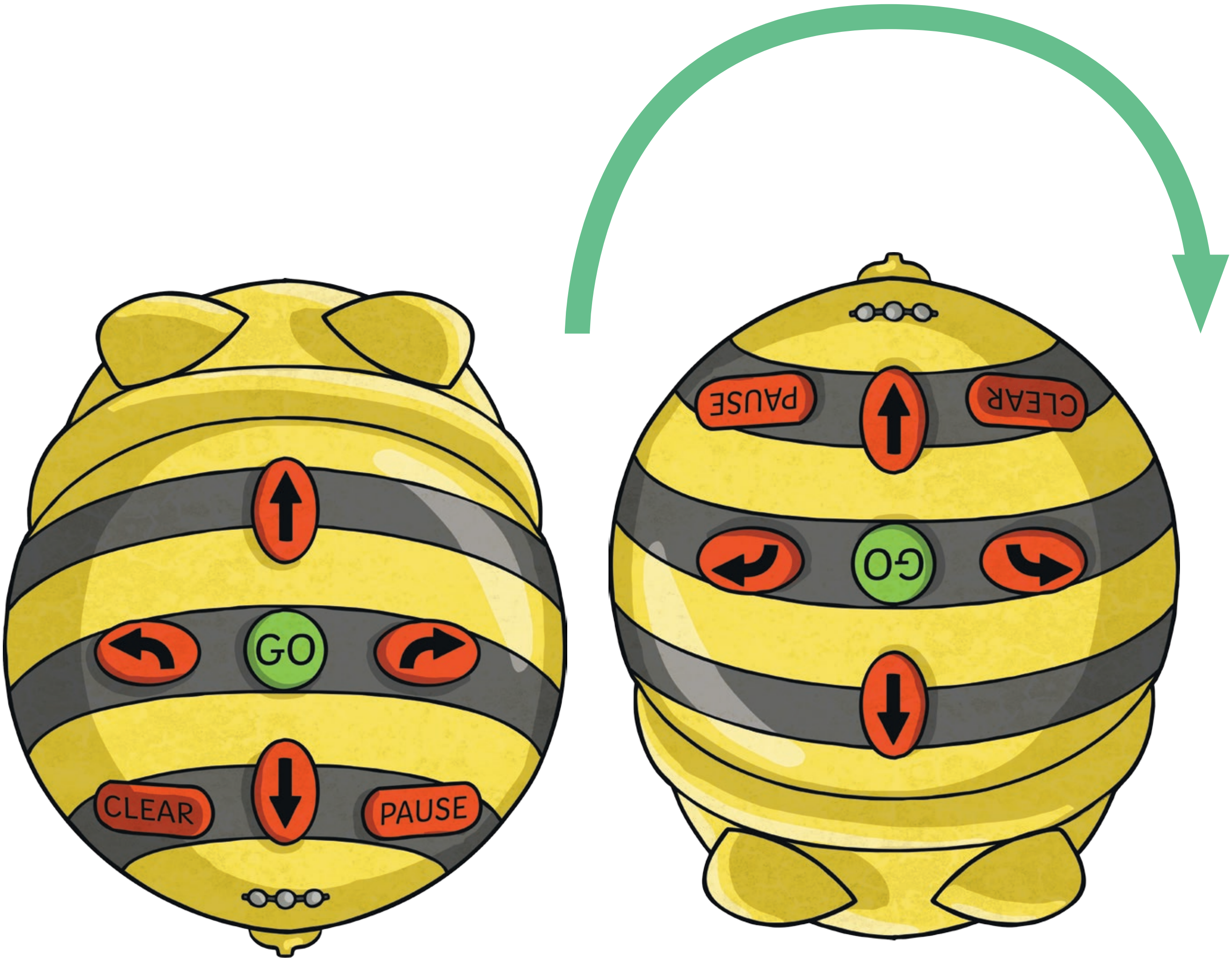




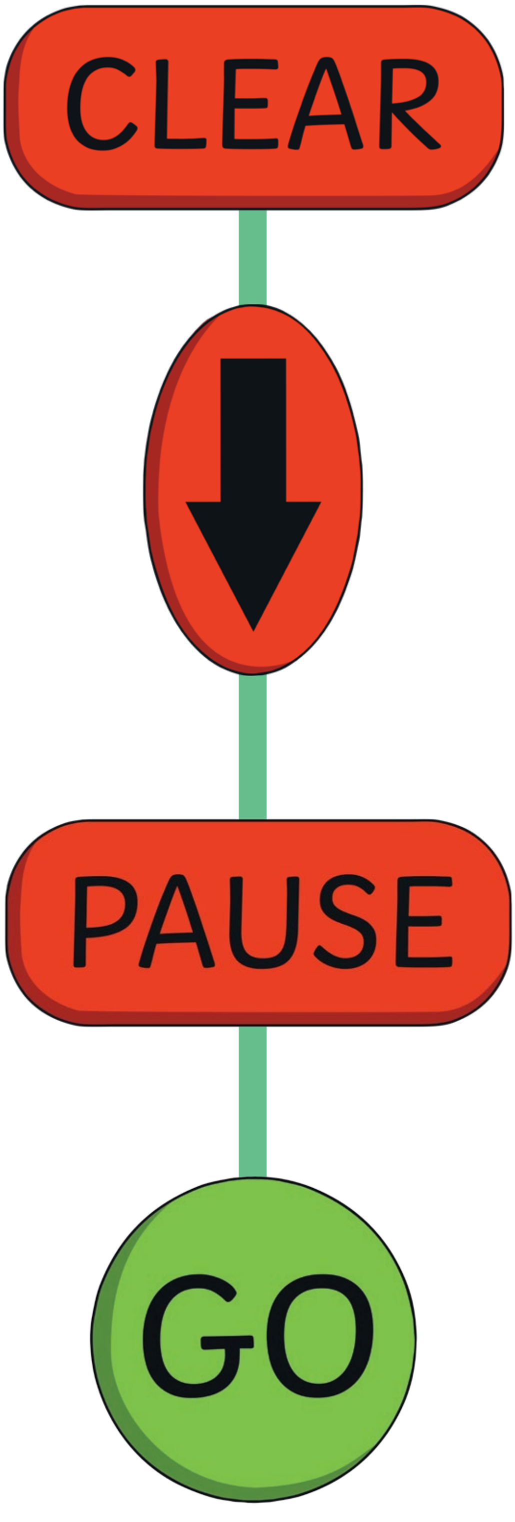
quarter



half



algorithm



code

0110110101010101

0110101101010101

0101011011100101

0110110101010101

0011010111010101

0110101101010101

0110101101010101

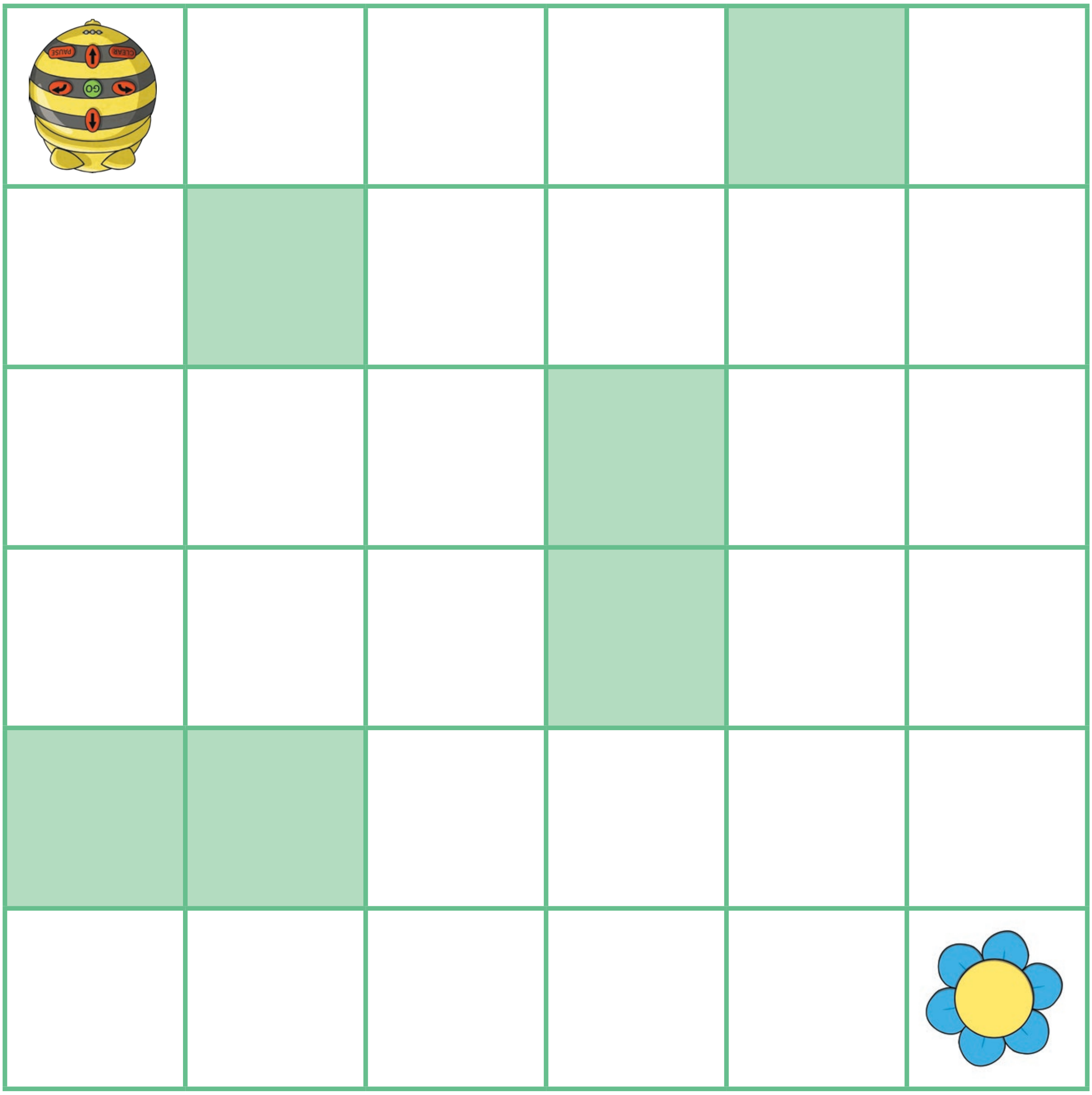
0011010111010101

0110101101010101

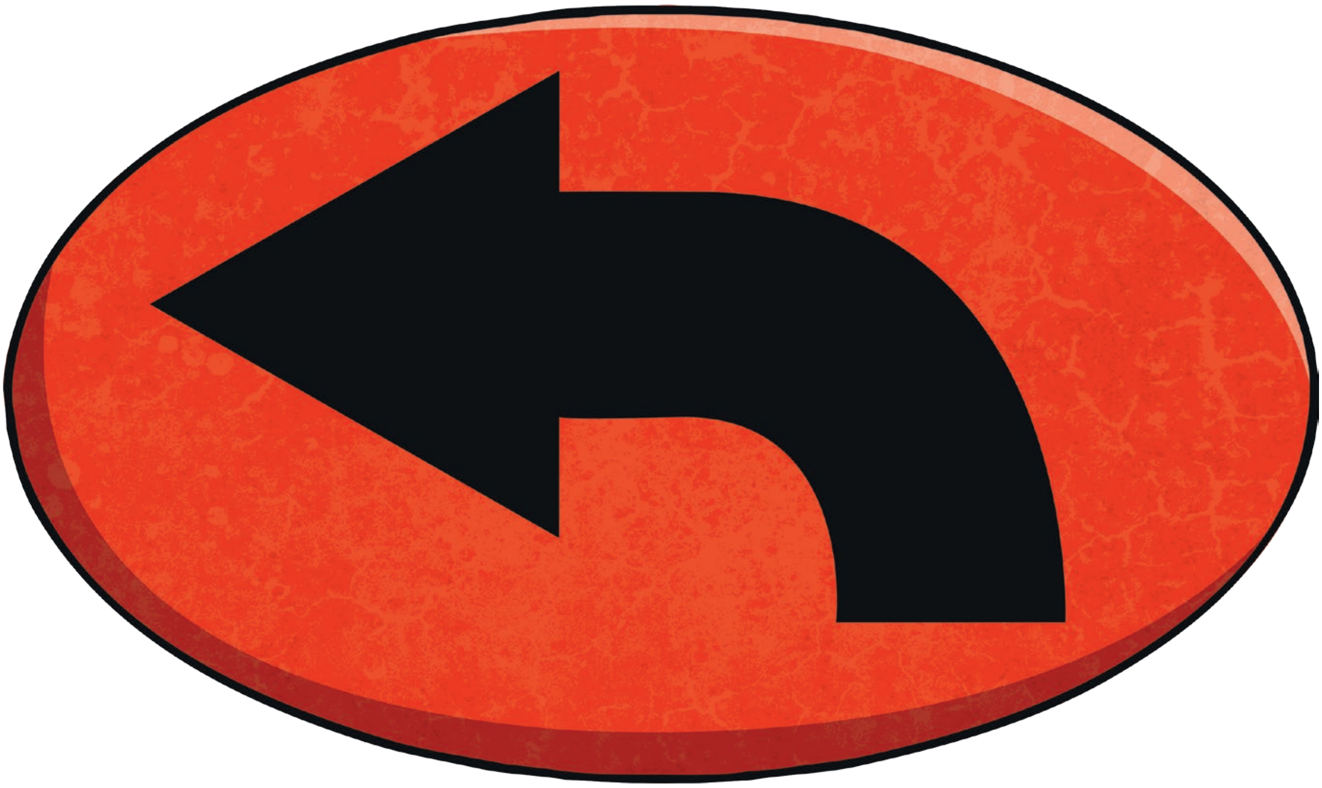
0110110101010101

Instructions

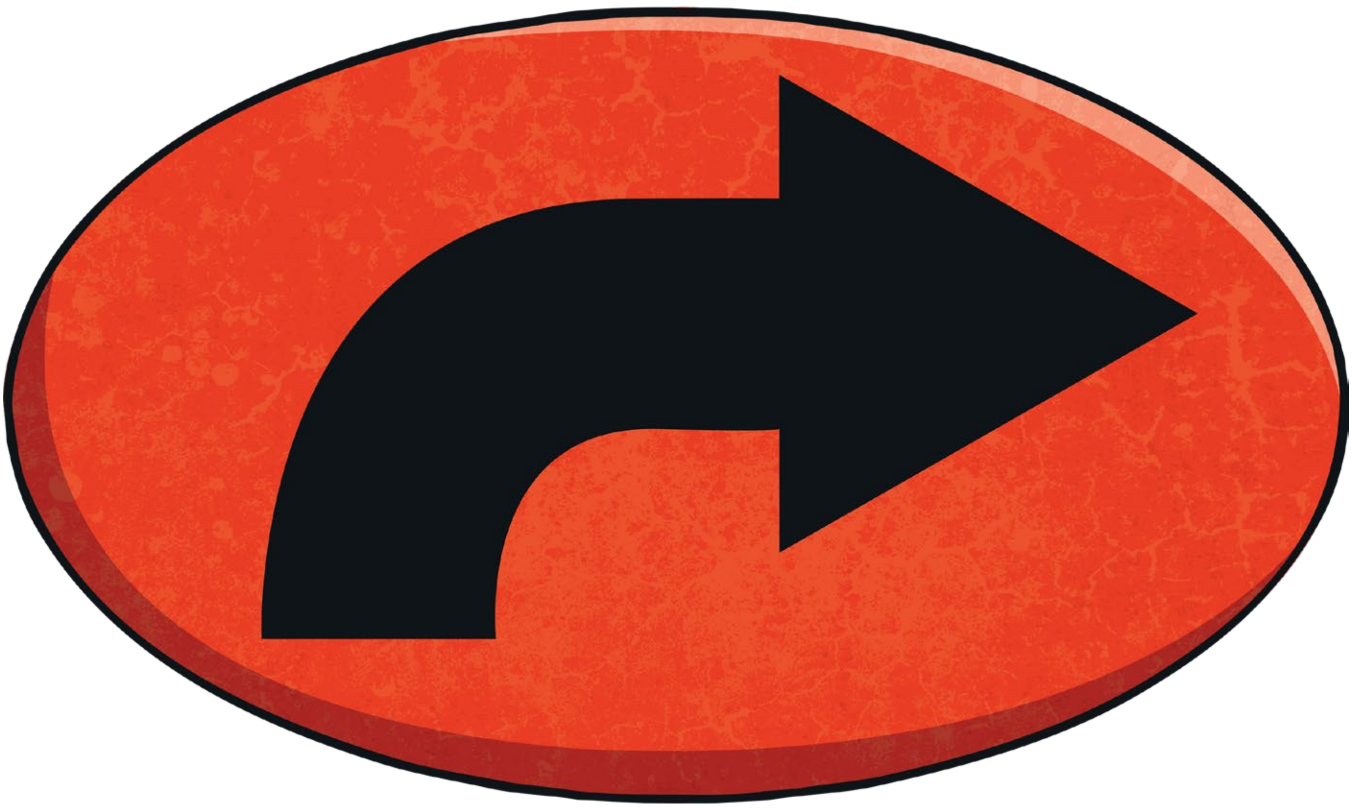
Get the Bee-Bot to the flower.



left



right

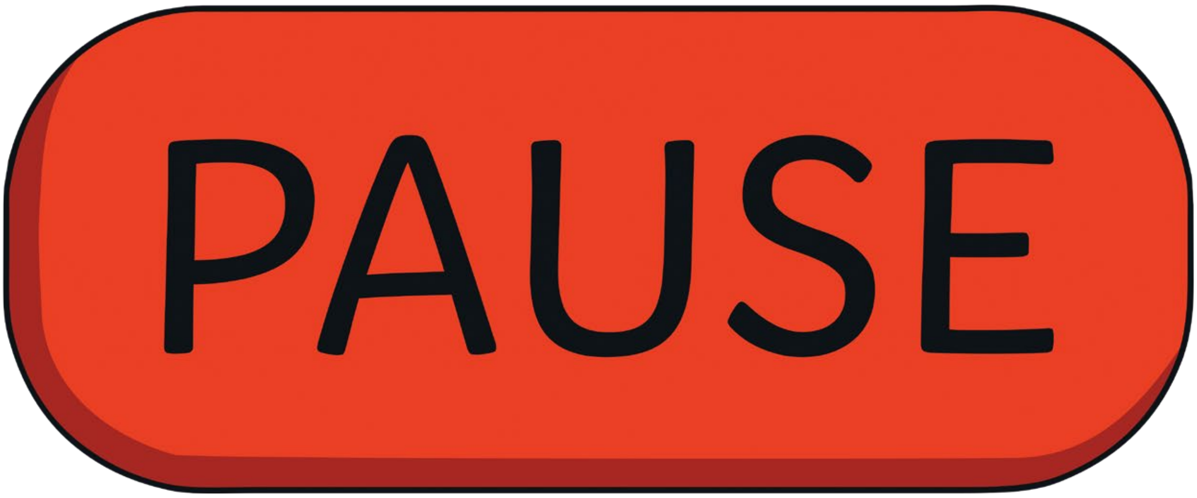


visit twinkl.com

forward

backward

pause

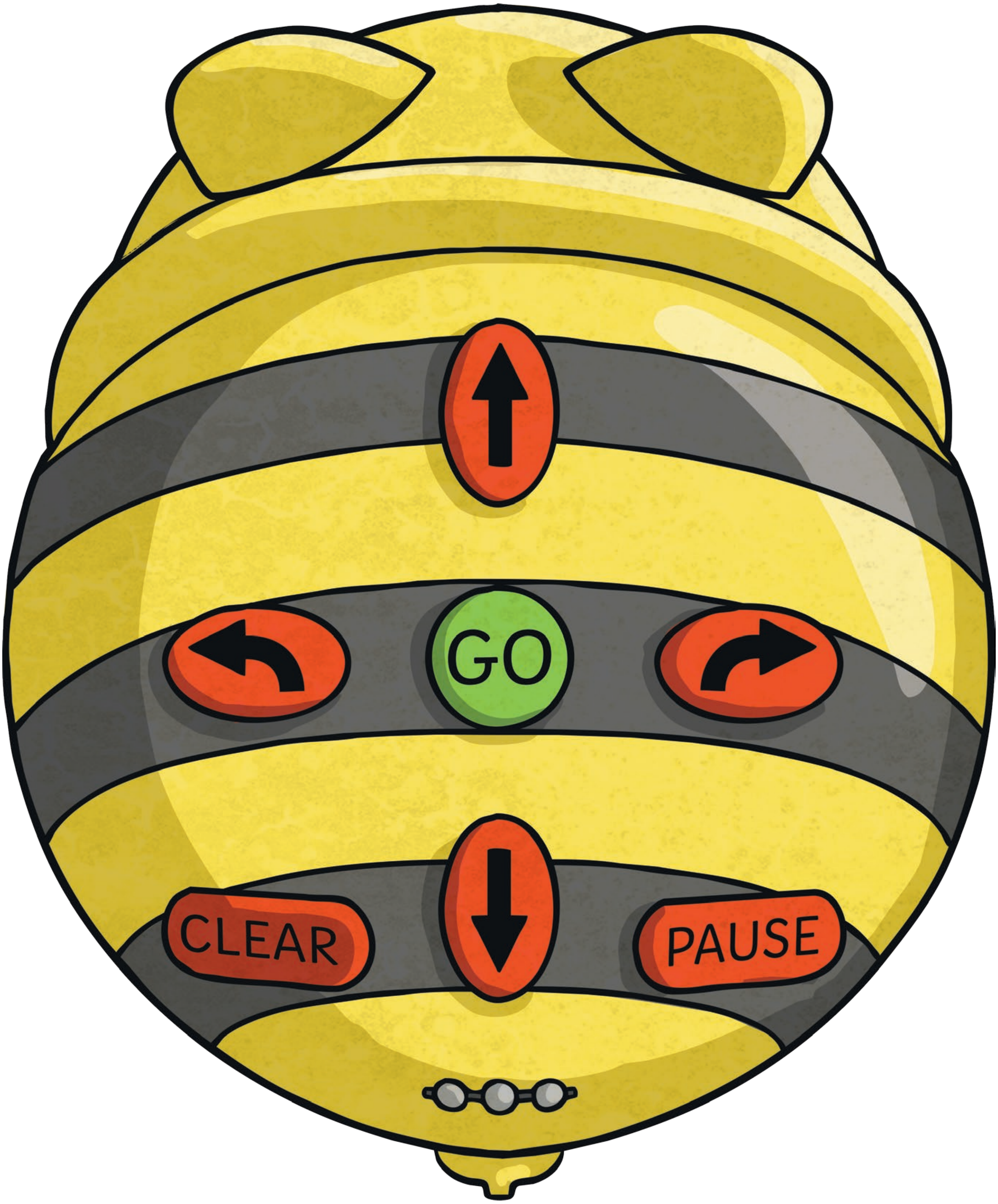


clear

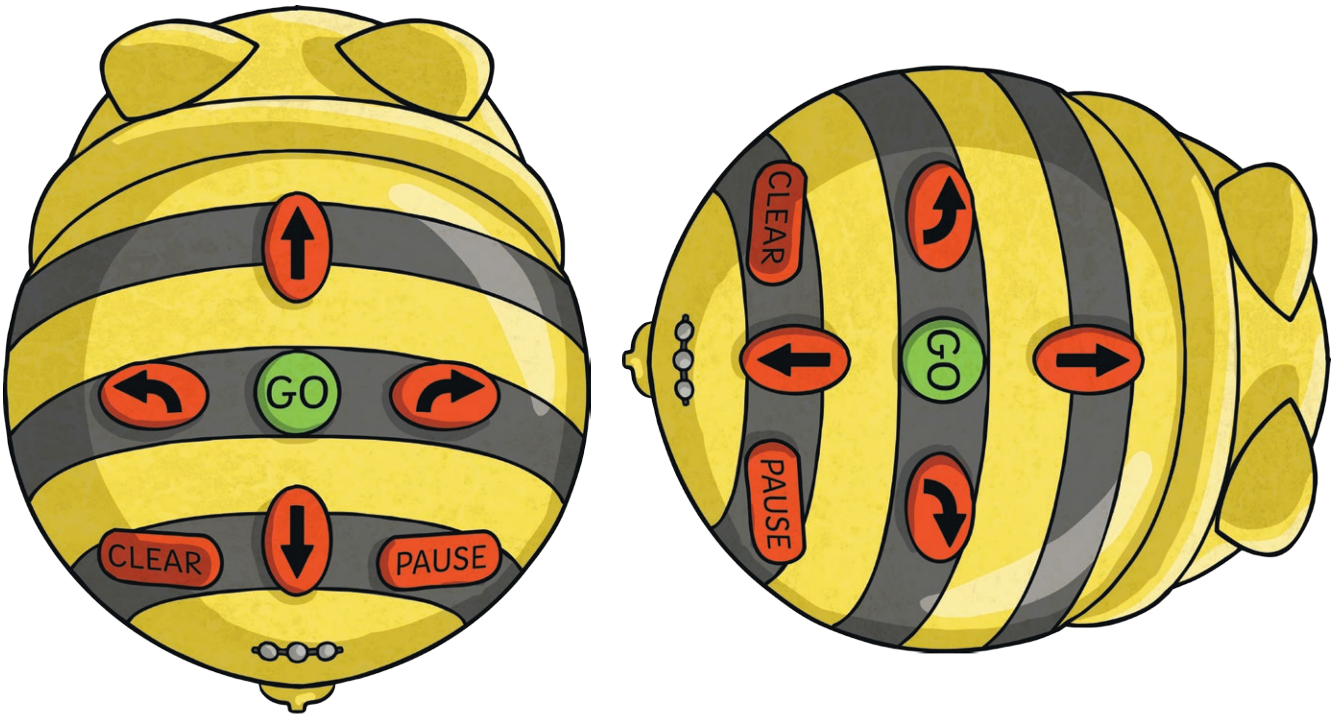
go

program

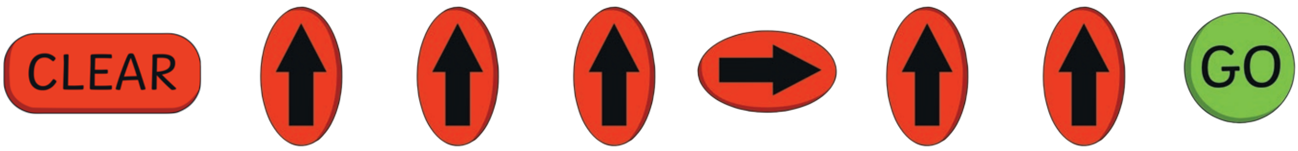
Bee-Bot

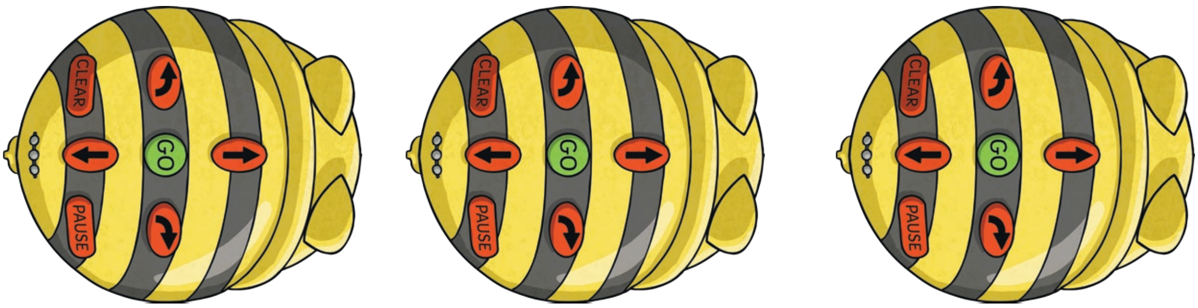


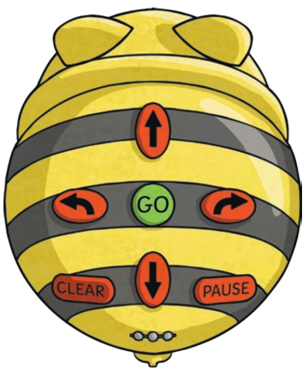
turn

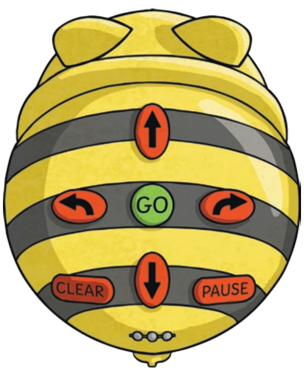


sequence

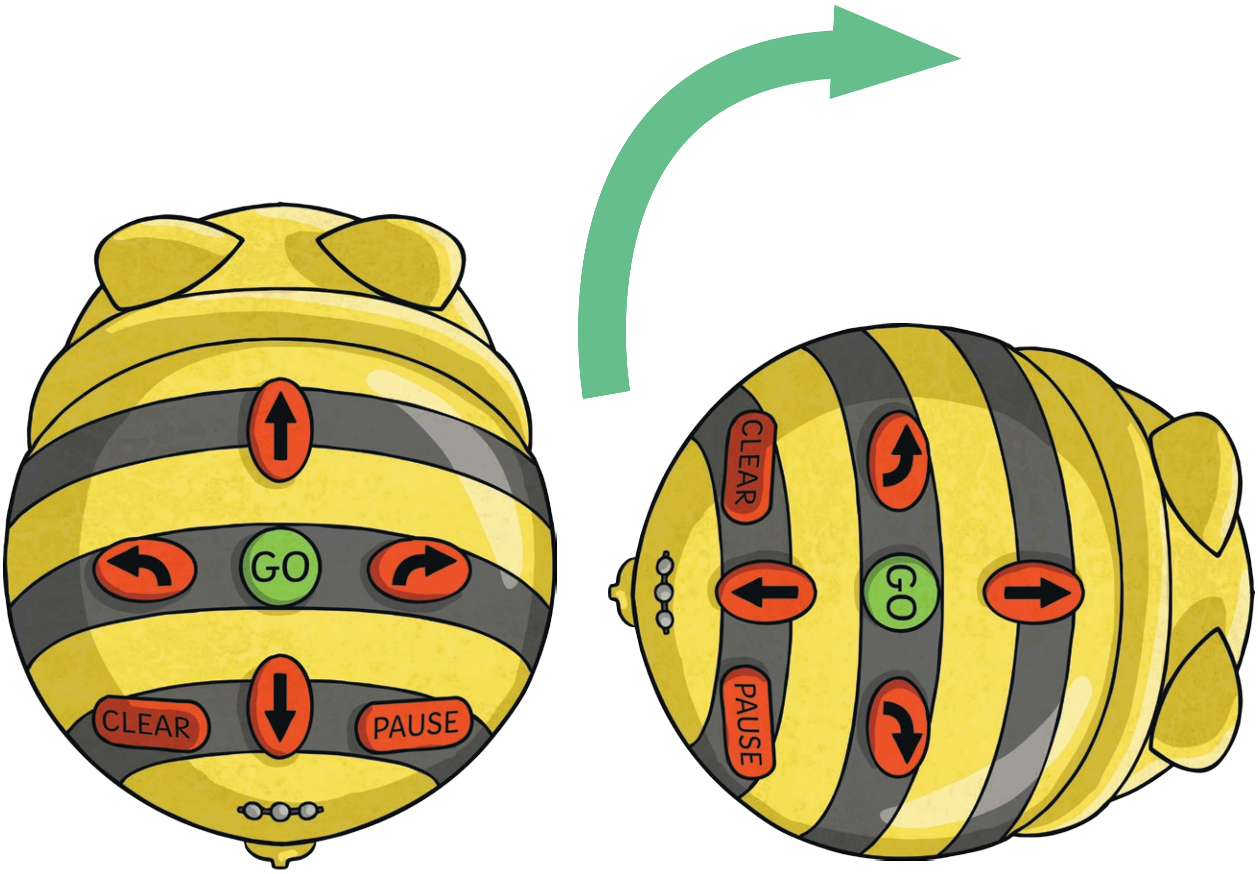

CLEAR
GO


CLEAR
GO
PAUSE
CLEAR
GO
PAUSE
CLEAR
GO
PAUSE

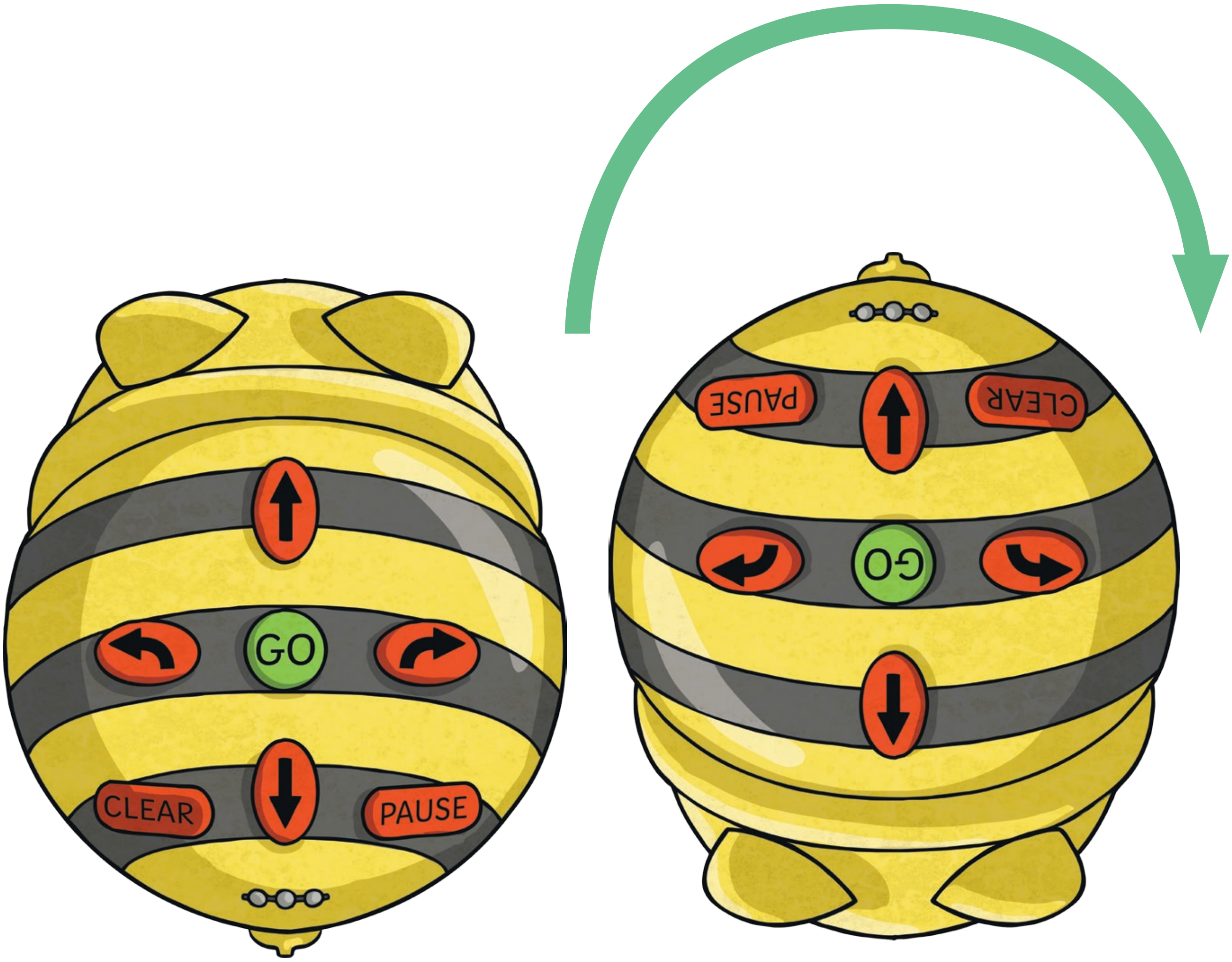

GO
CLEAR
PAUSE


GO
CLEAR
PAUSE

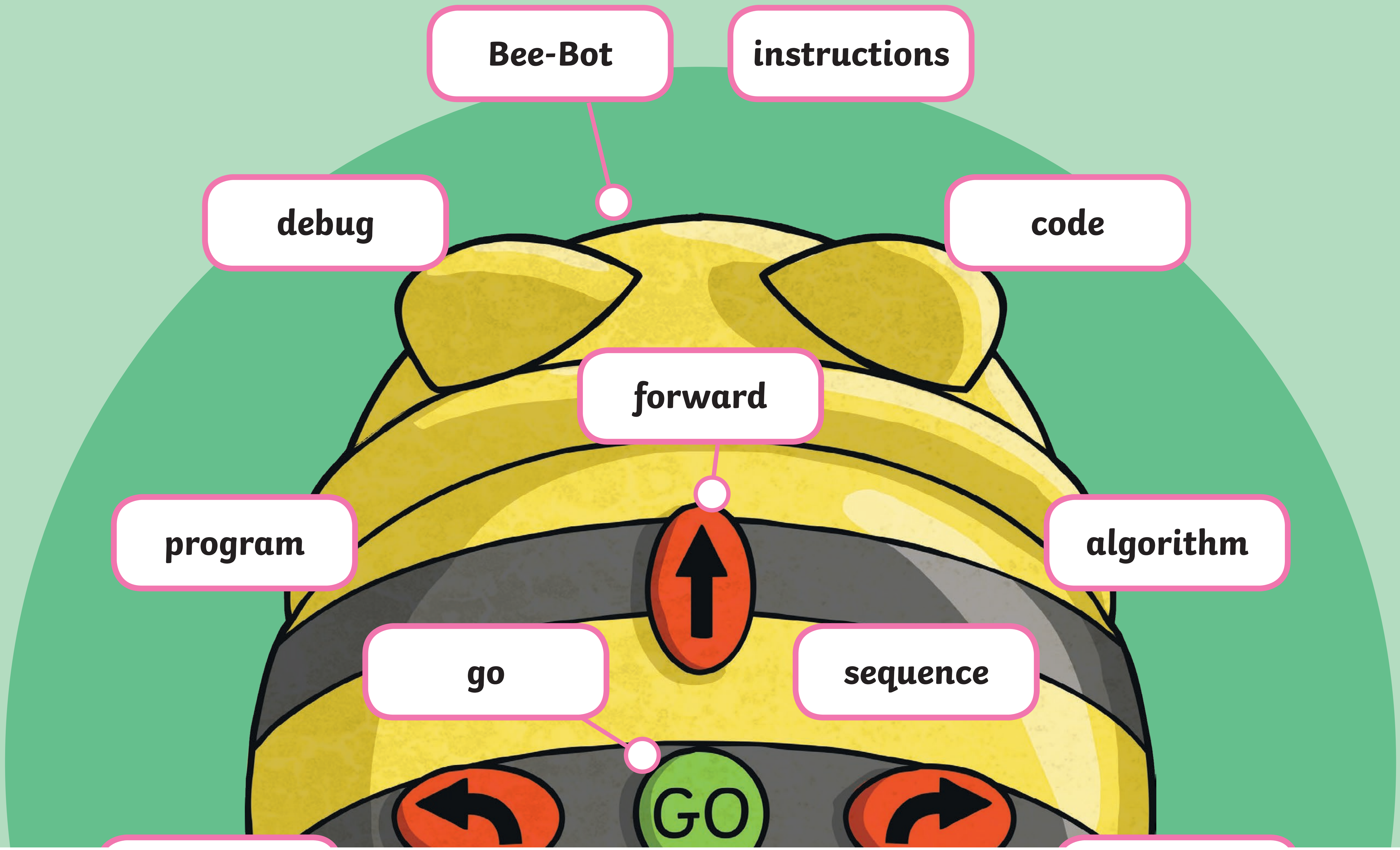
quarter

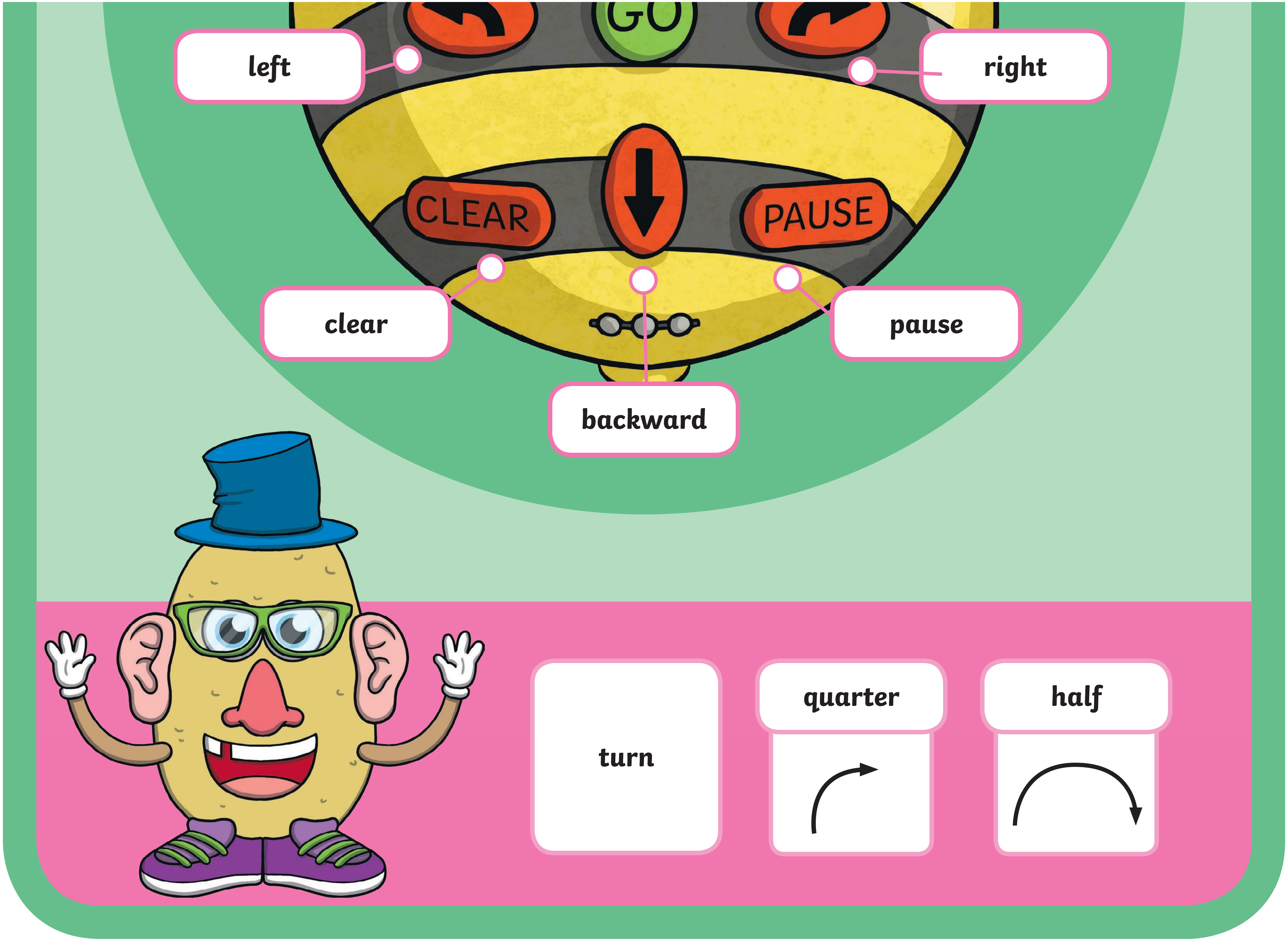


half

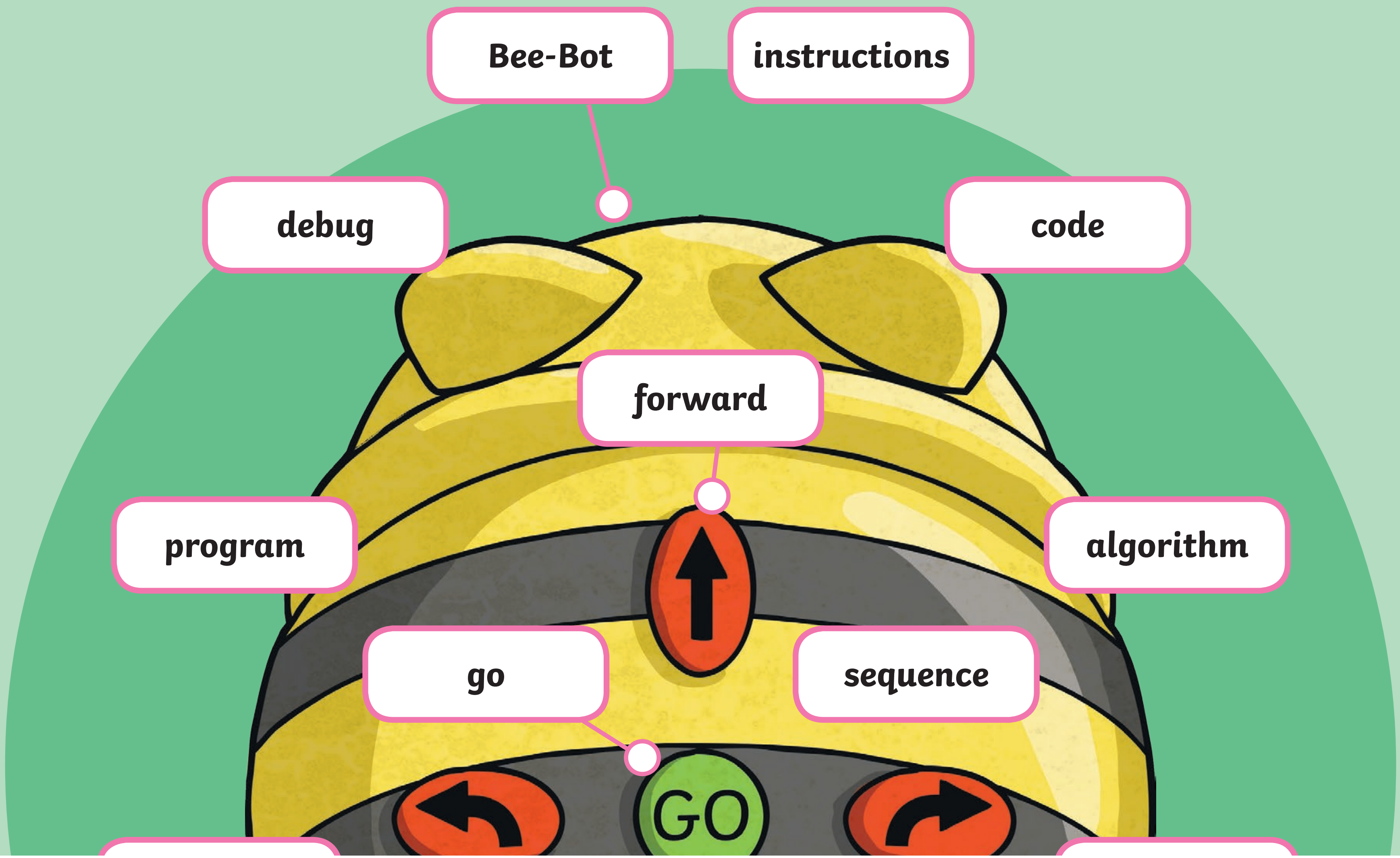


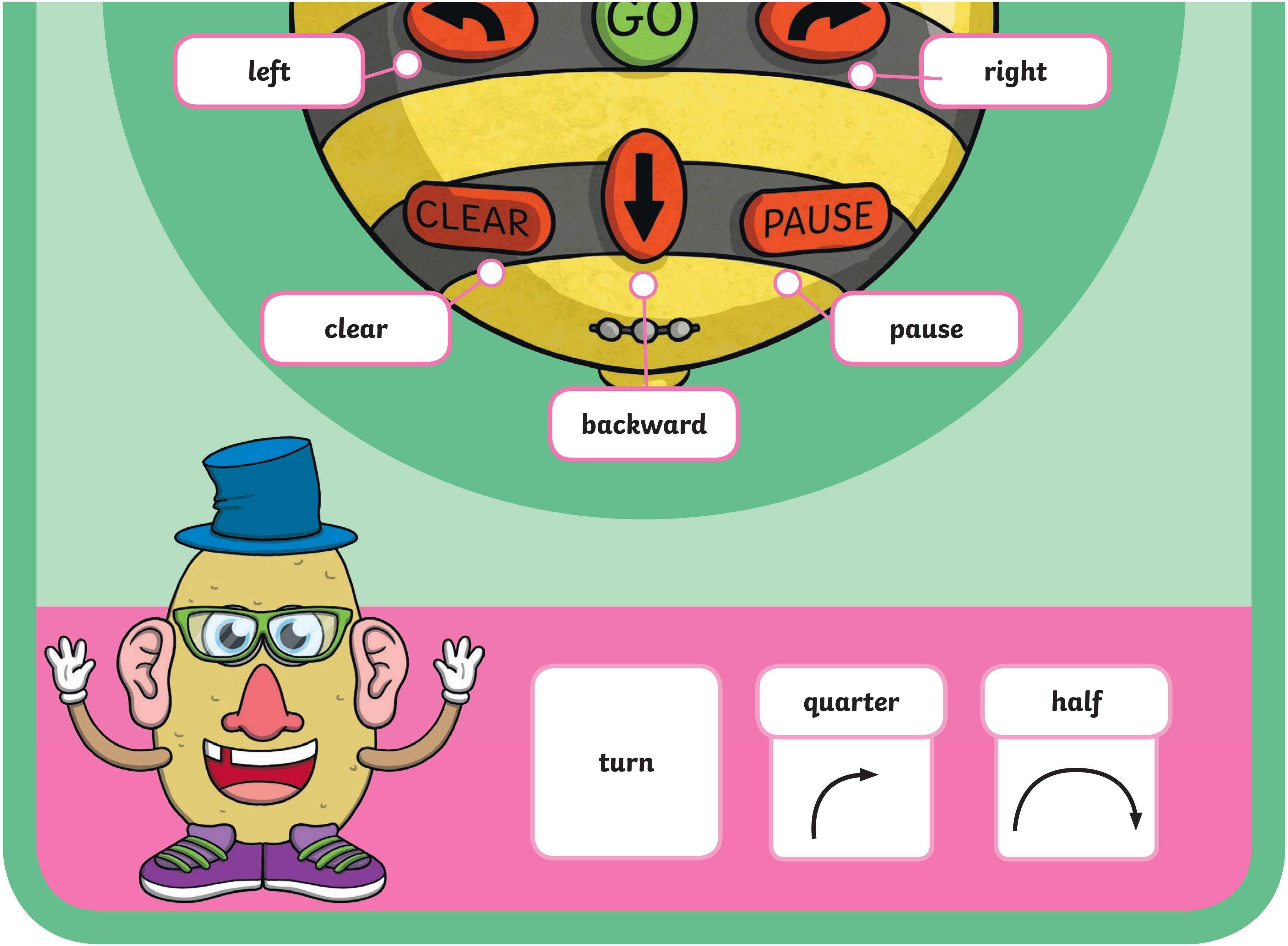
Programming Toys




left
right
CLEAR
PAUSE
GO
clear
pause
backward
turn
quarter
half

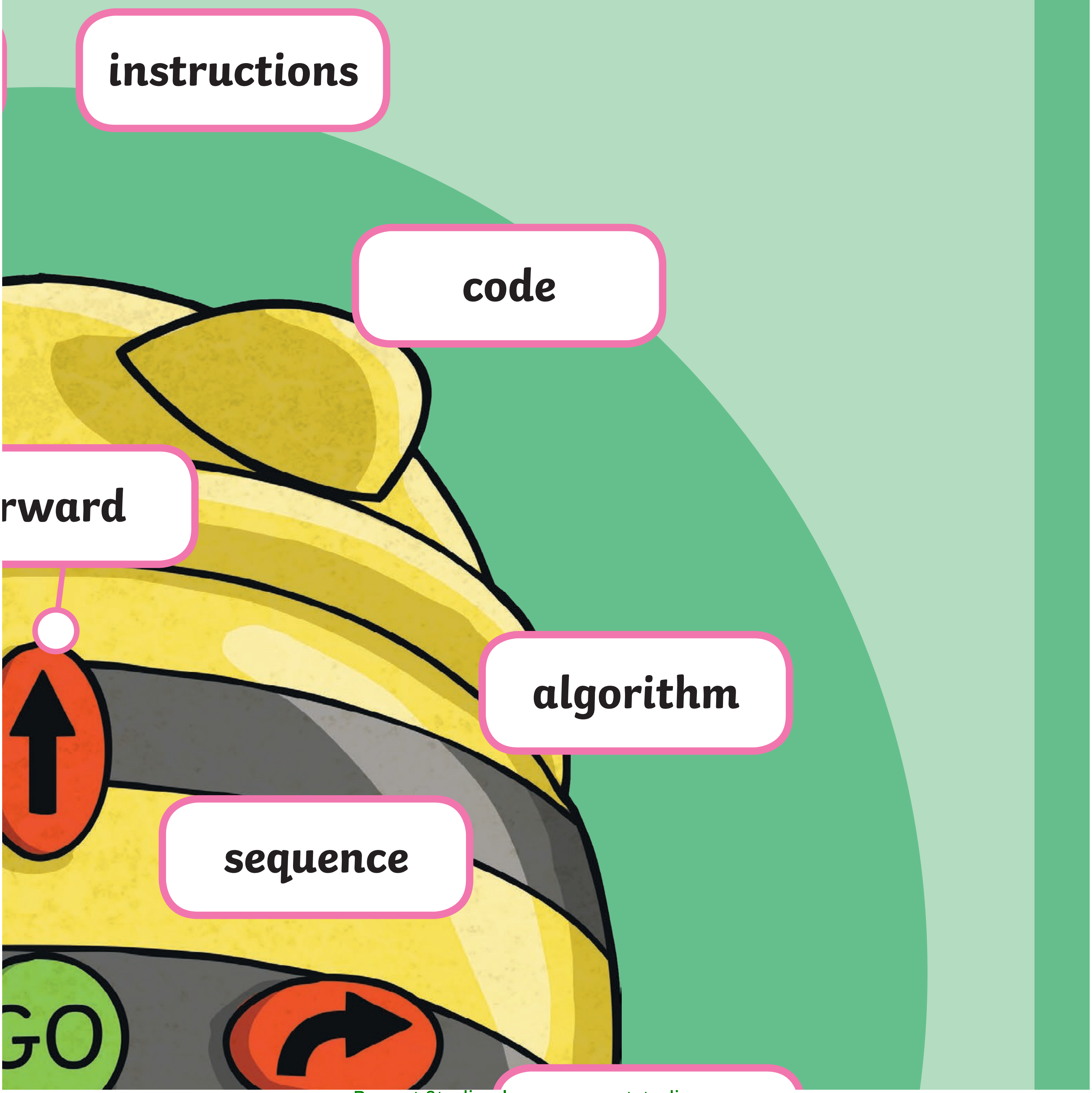
Programming Toys



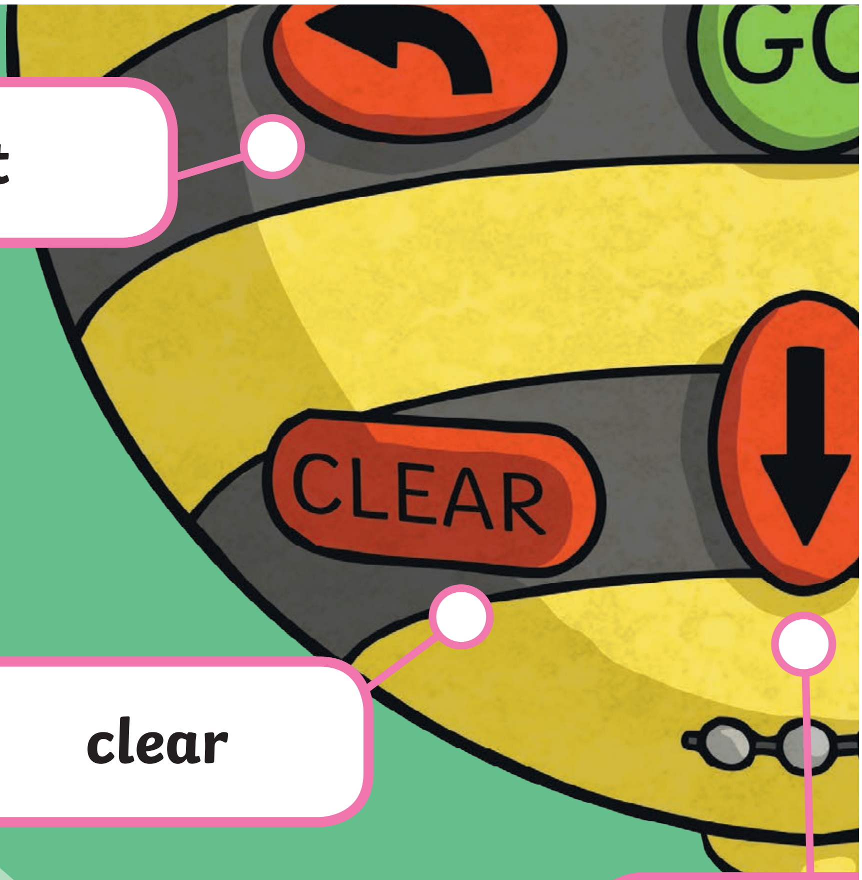

GO
left
right
CLEAR
PAUSE
clear
pause
backward
turn
quarter
half

Programm

ming Toys

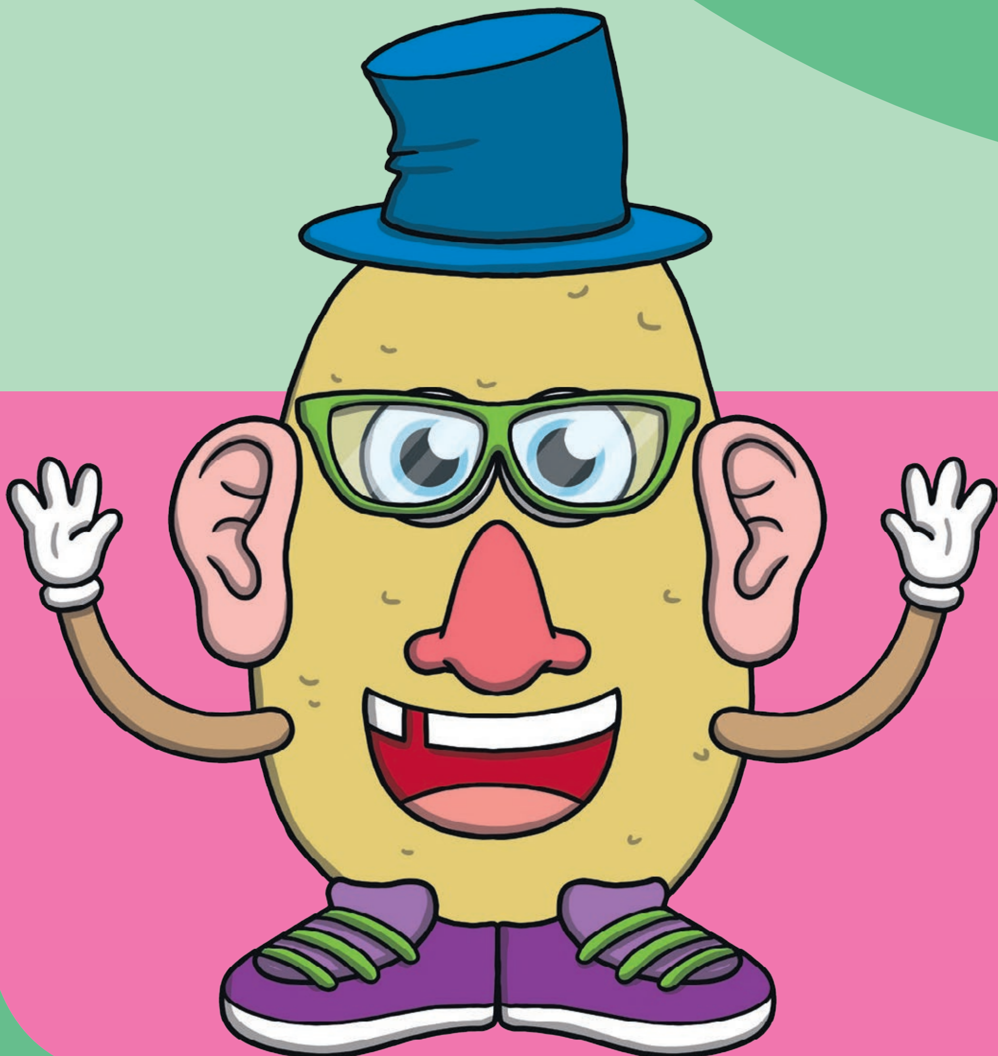


left

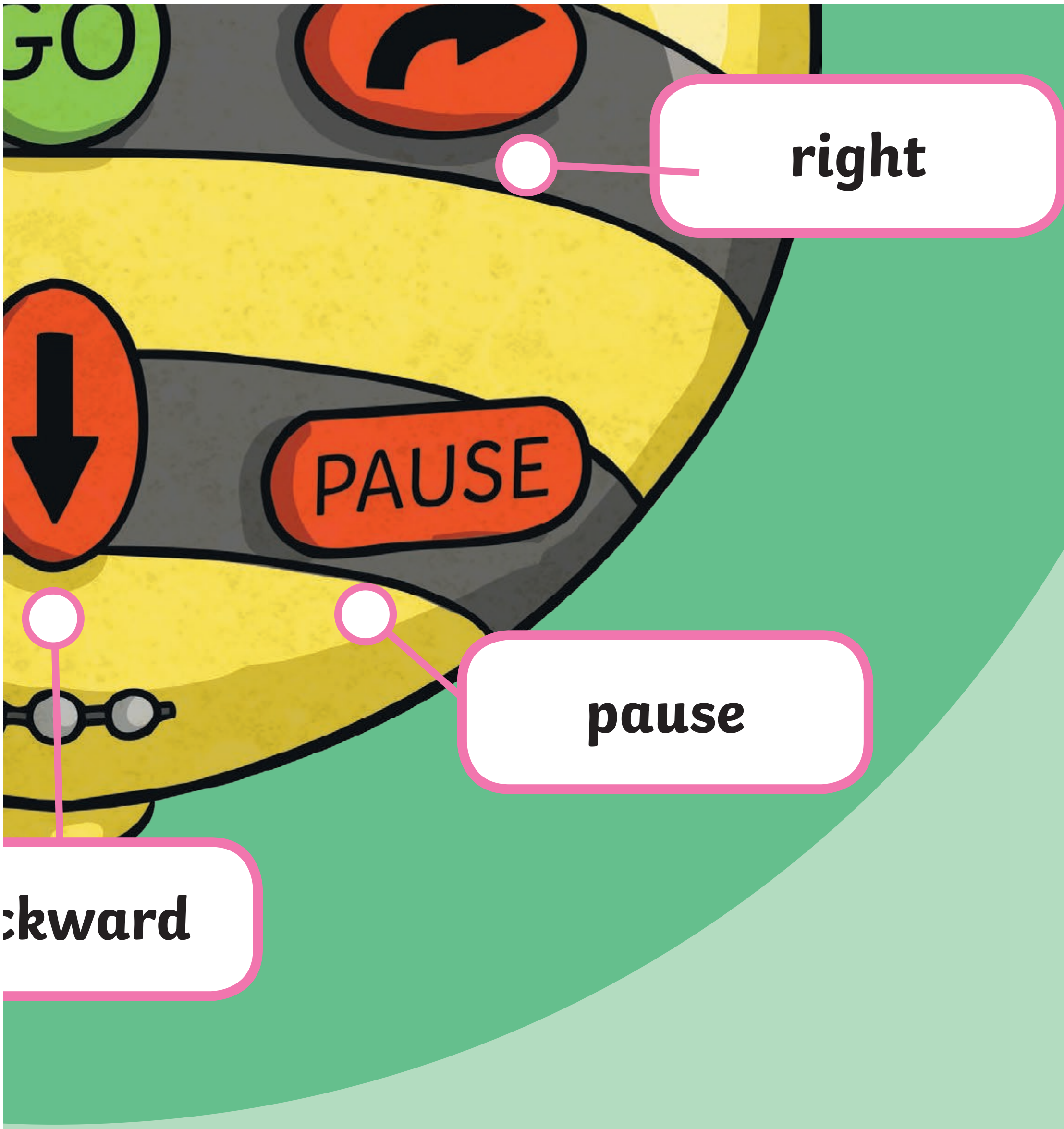


clear

backw

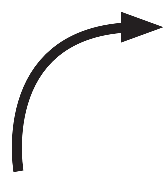


turn

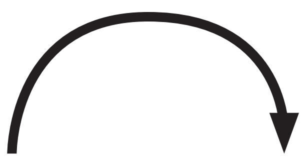


rn

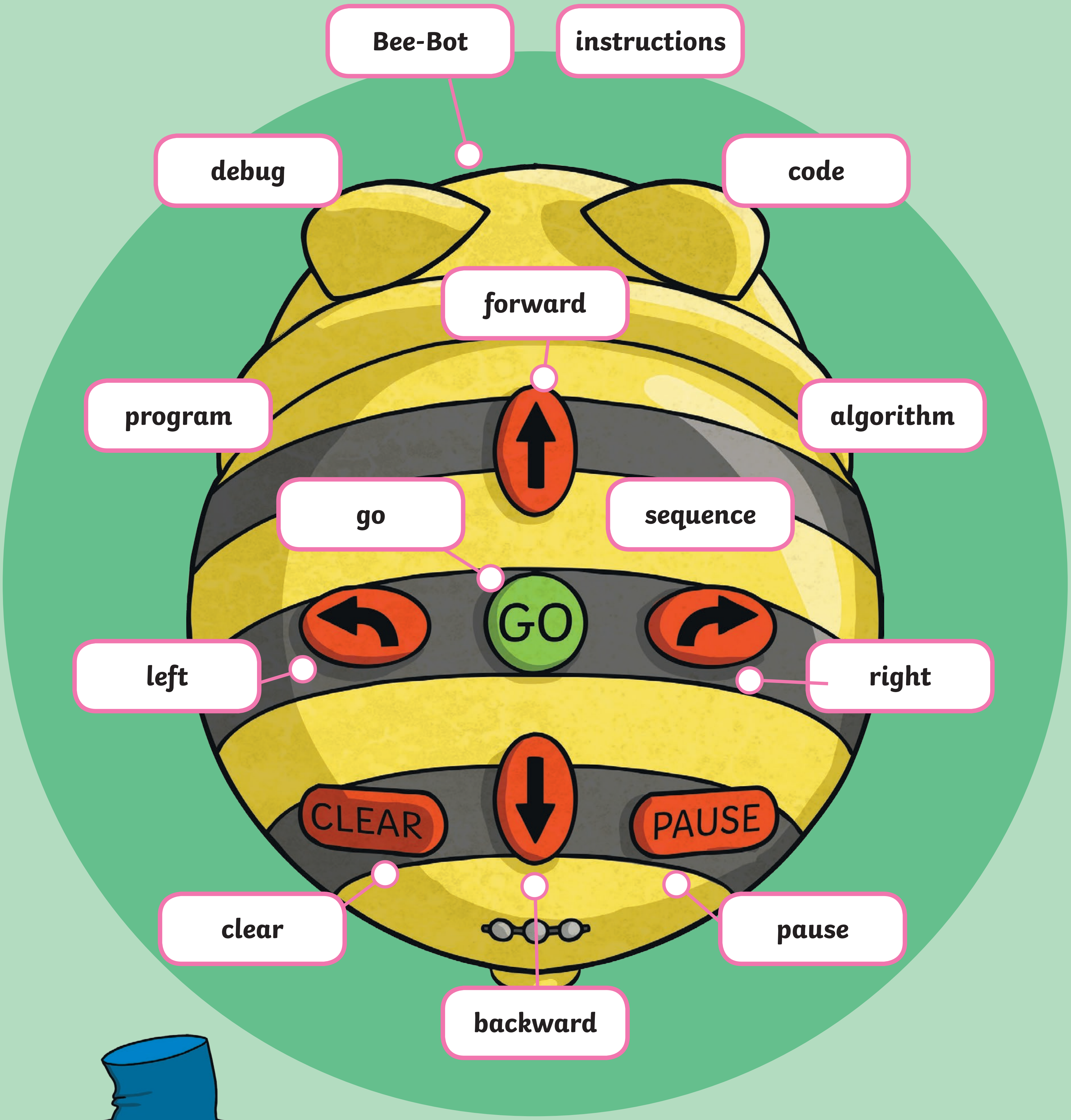
quarter

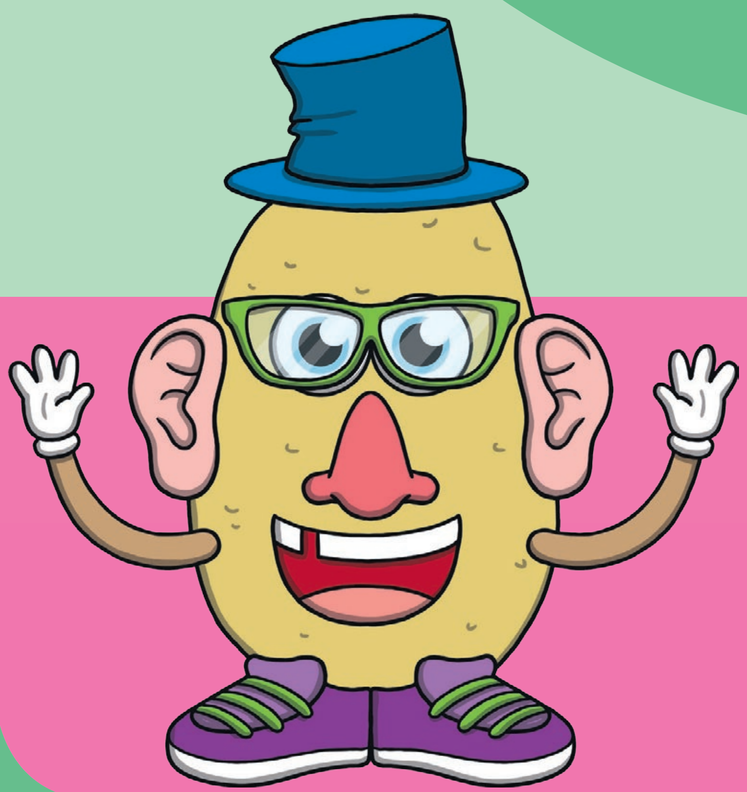


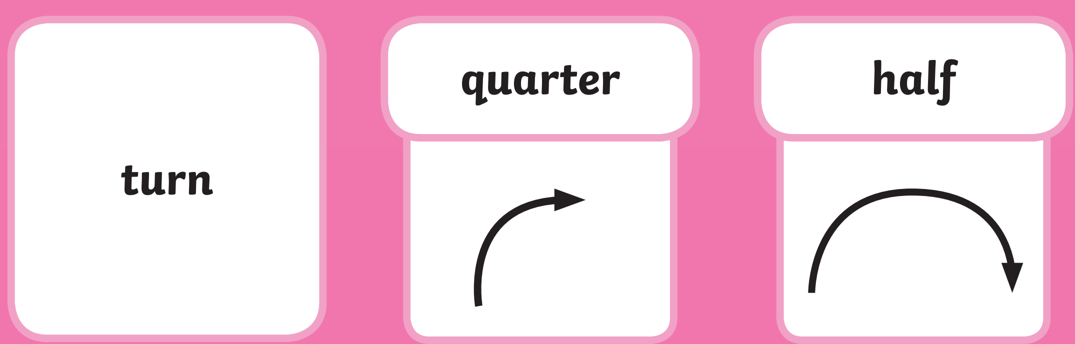
half



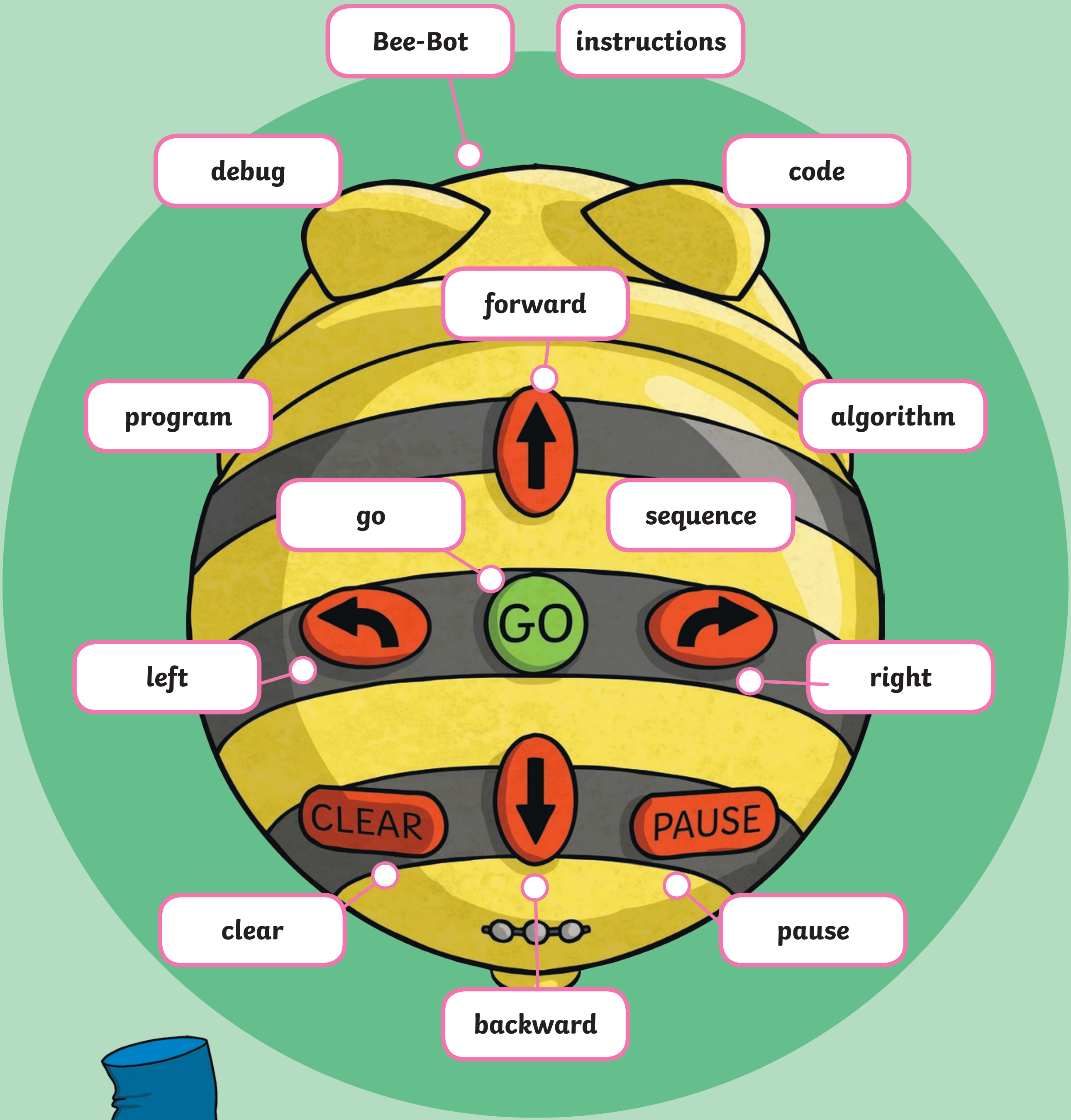
Programming Toys

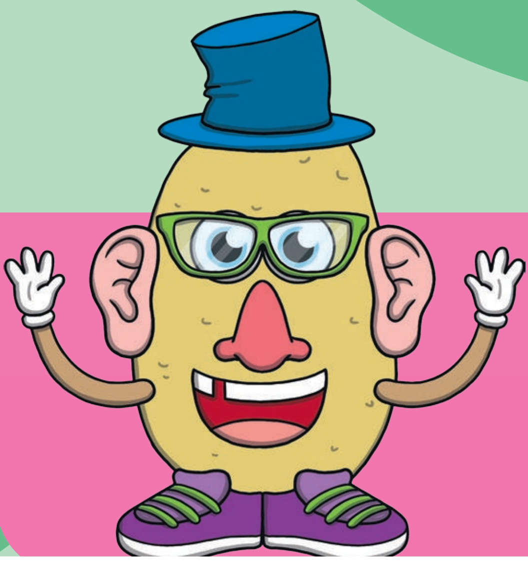


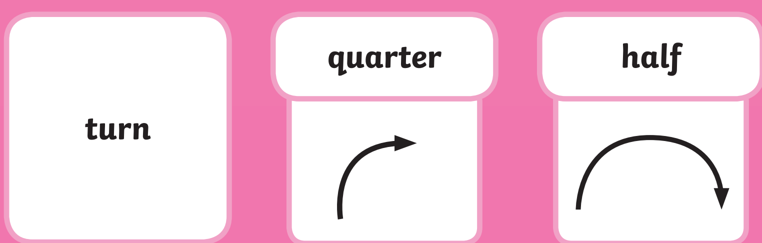




Programming Toys



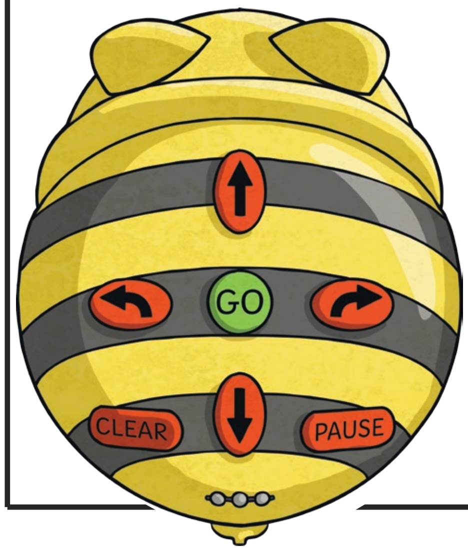


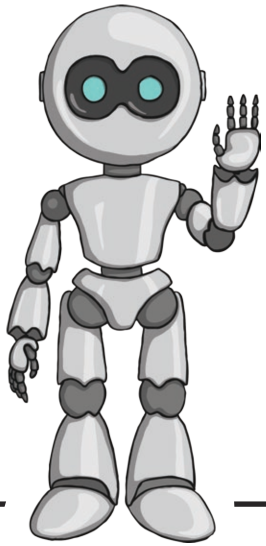


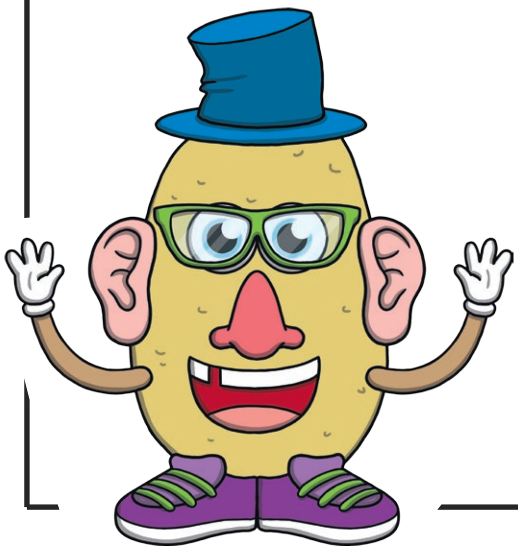

CLEAR

PAUSE


GO

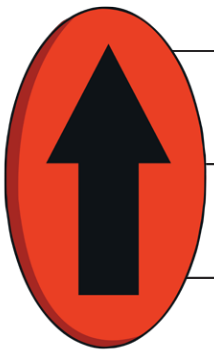

GO
CLEAR
PAUSE



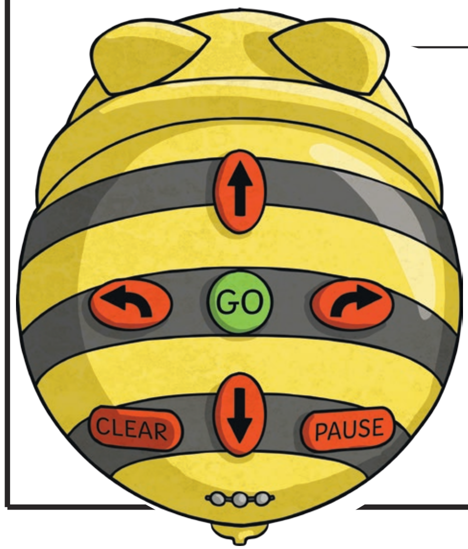


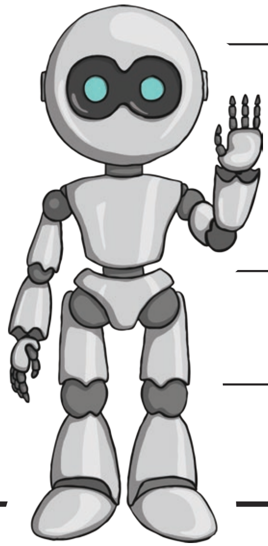

CLEAR

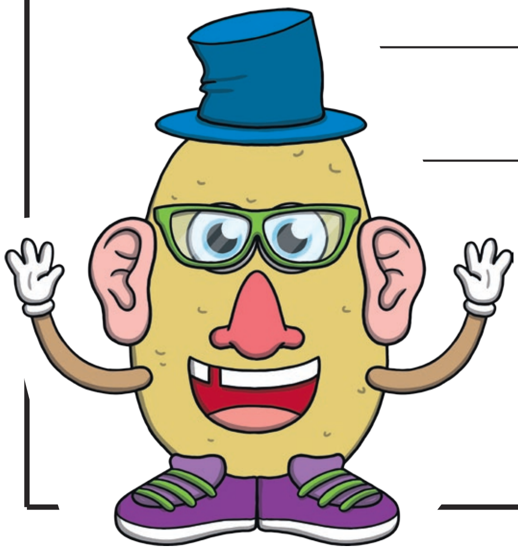
PAUSE



GO

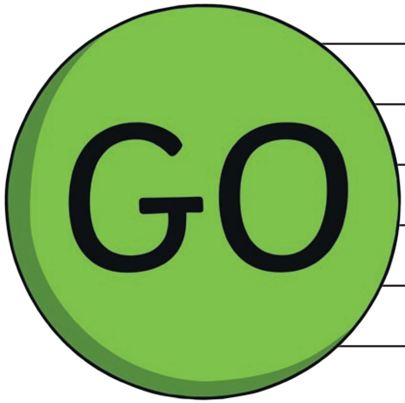

GO
CLEAR
PAUSE

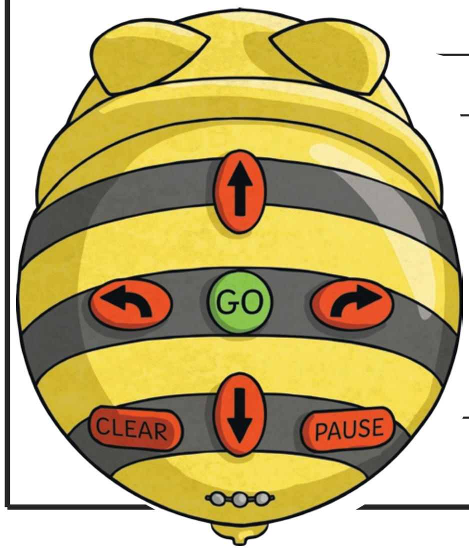


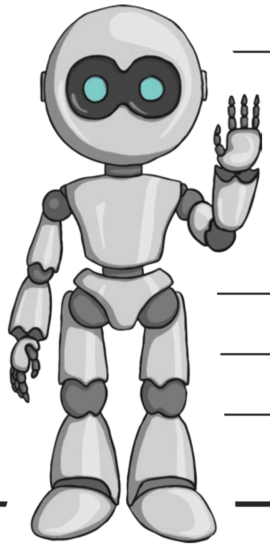


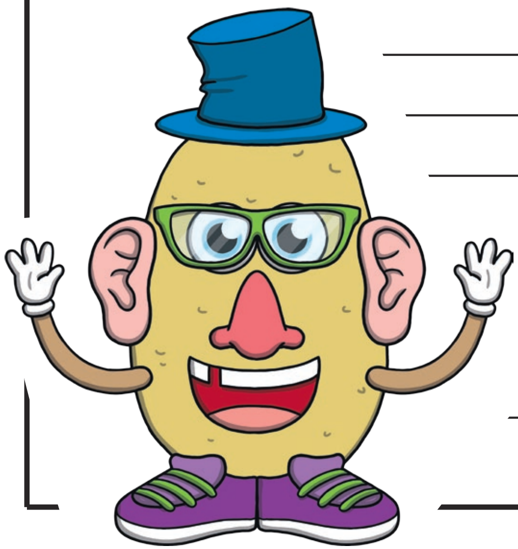
CLEAR

PAUSE


GO

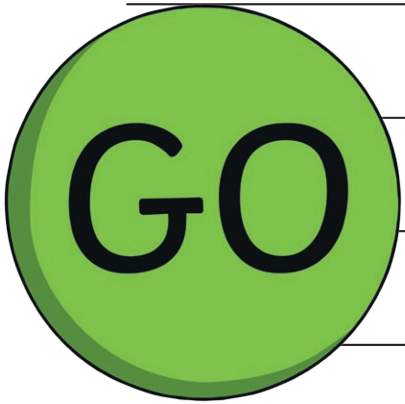

GO
CLEAR
PAUSE

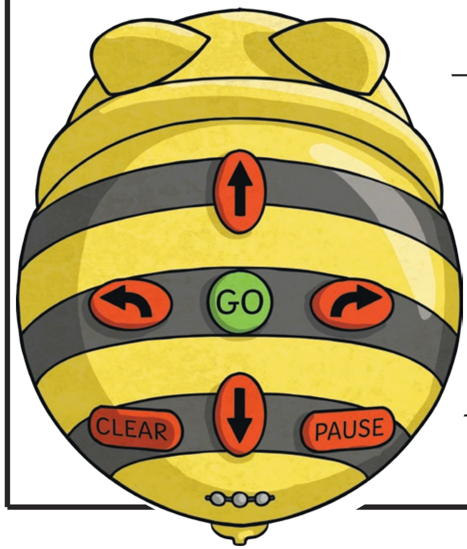


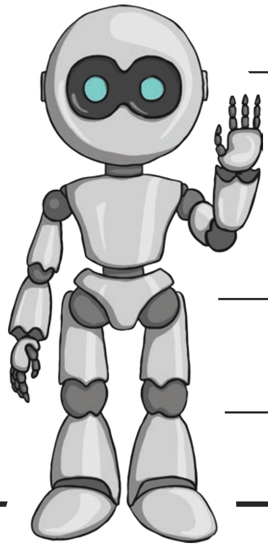


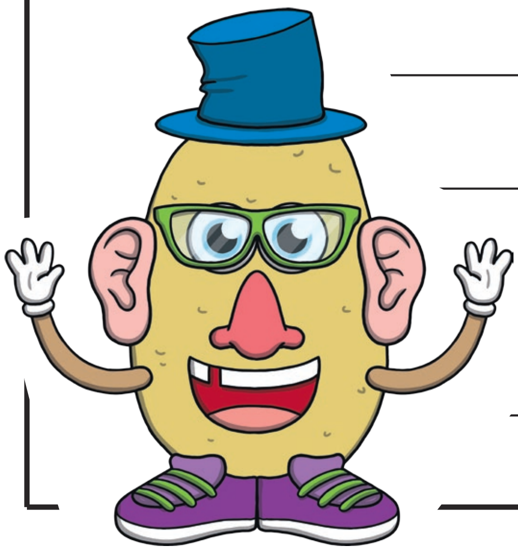
CLEAR

PAUSE


GO

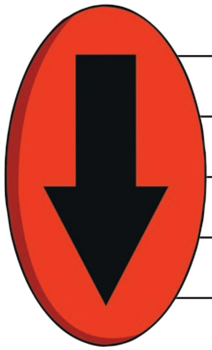

GO
CLEAR
PAUSE

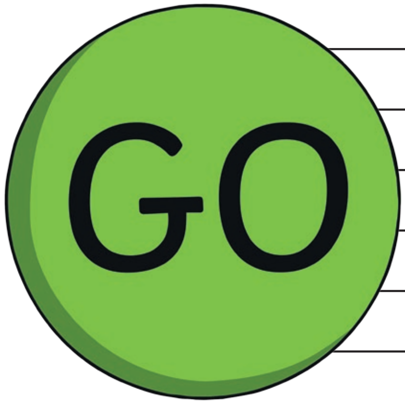


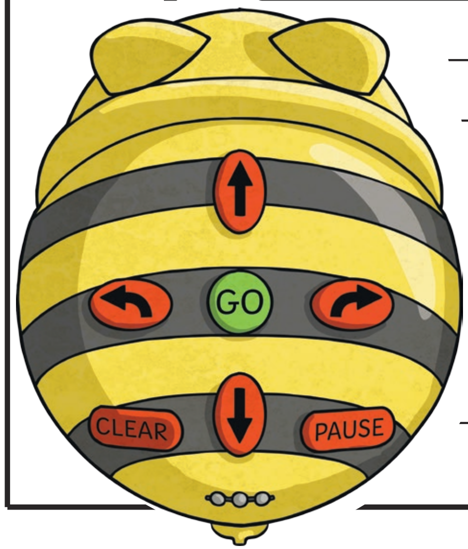


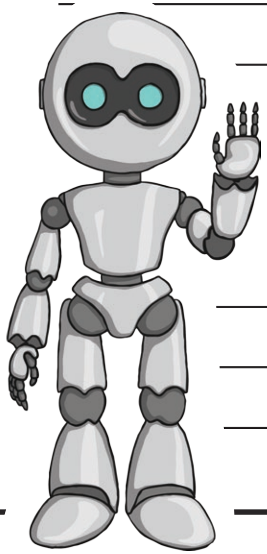
CLEAR

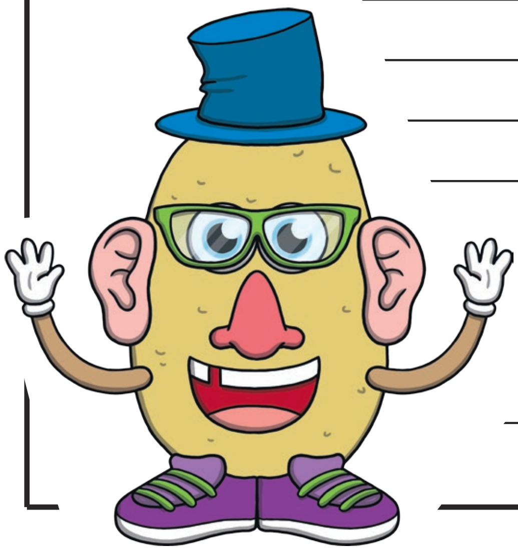
PAUSE




GO


GO
CLEAR
PAUSE

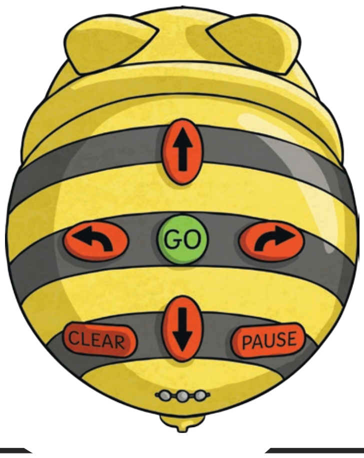


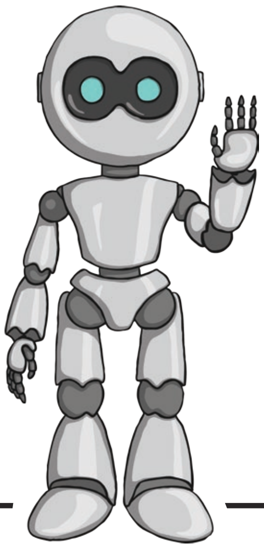


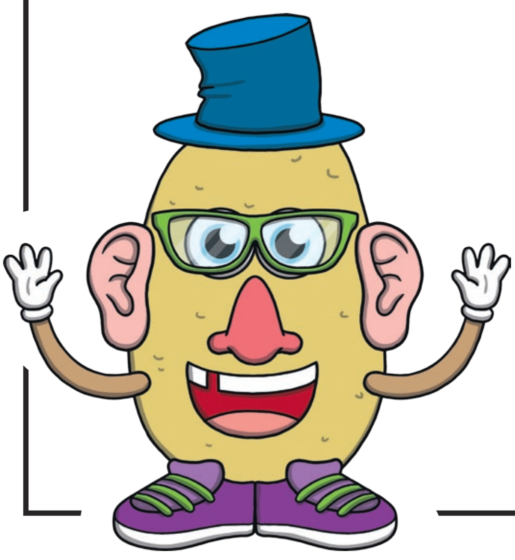

CLEAR

PAUSE


GO

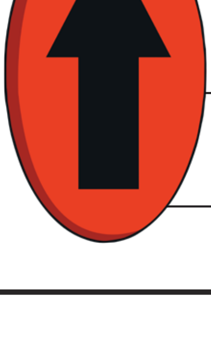

GO
CLEAR
PAUSE



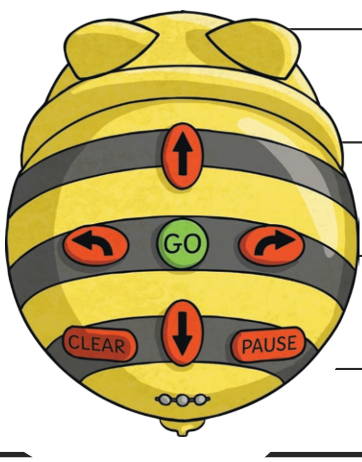


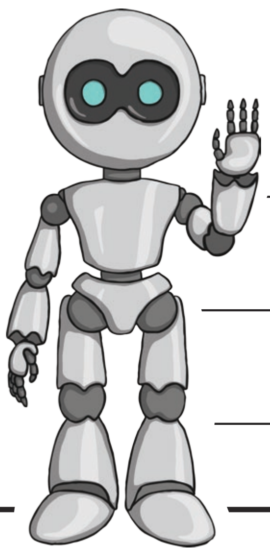
CLEAR

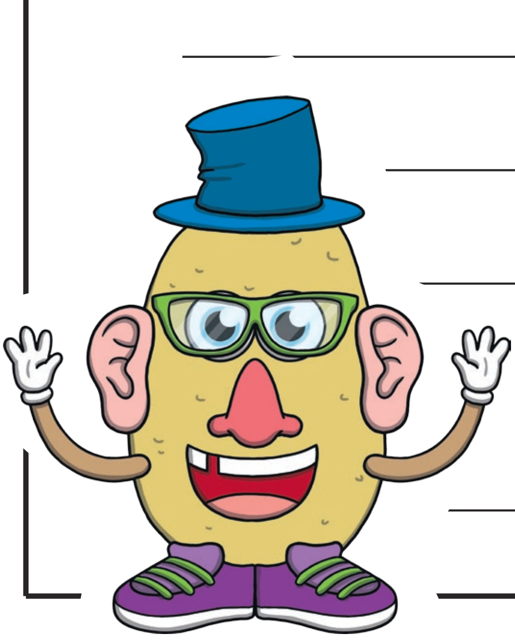

PAUSE



GO


GO
CLEAR
PAUSE

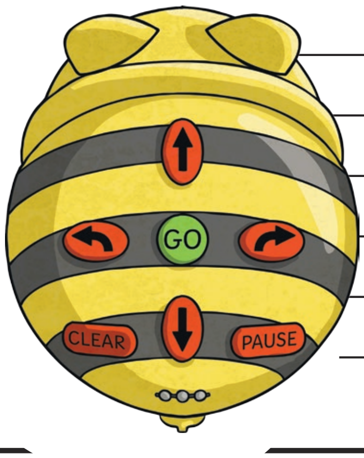


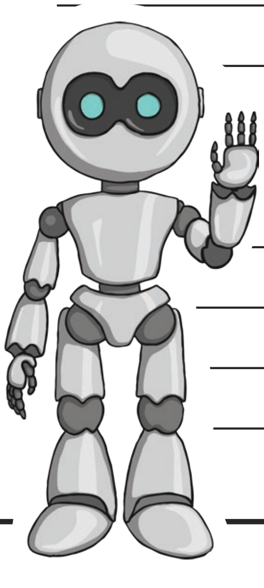


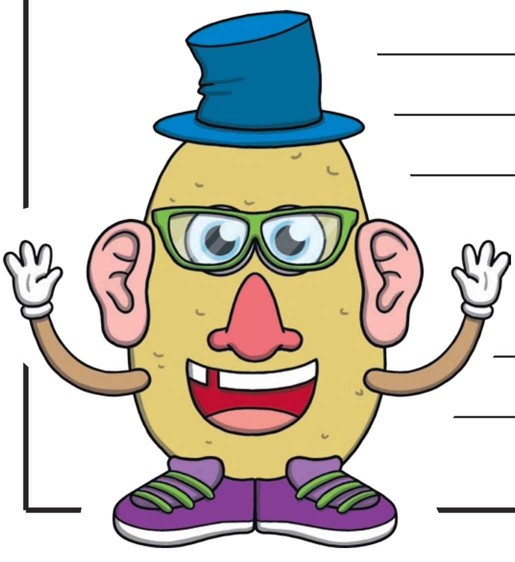
CLEAR

PAUSE

GO


GO
CLEAR
PAUSE

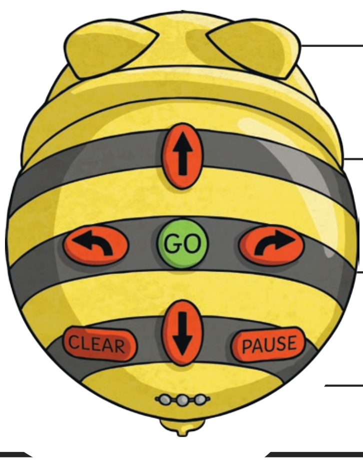


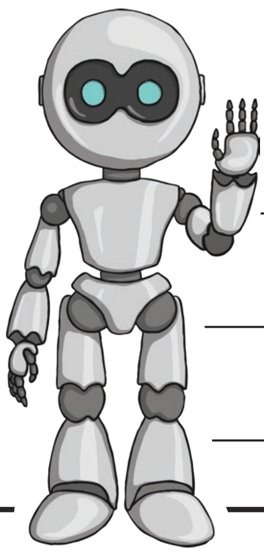


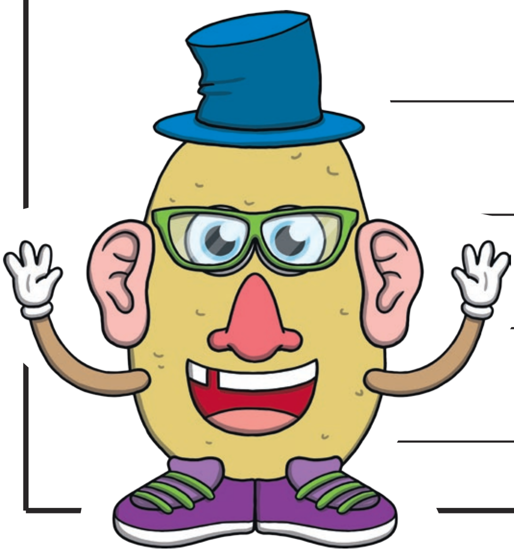

CLEAR

PAUSE

GO

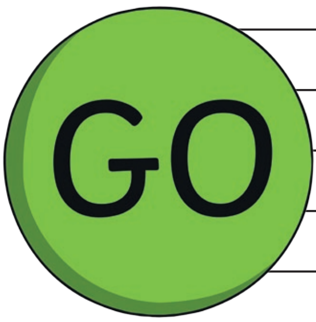

GO
CLEAR
PAUSE

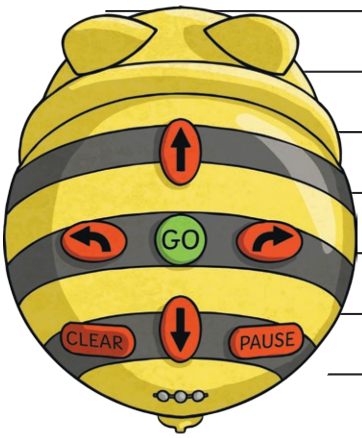


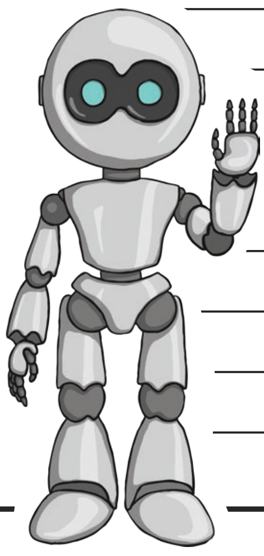


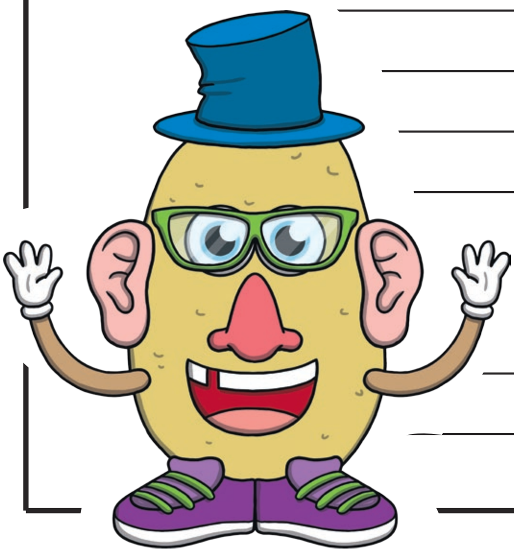

CLEAR

PAUSE

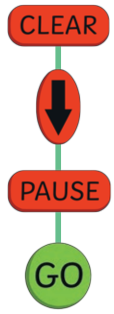

GO


GO
CLEAR
PAUSE





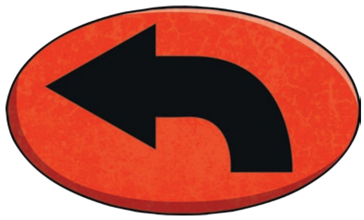
algorithm



code

0110110101010101
0110101101010101
0101011011100101
0110110101010101
0011010111010101
0110101101010101
0110101101010101
0011010111010101
0110101101010101
0110110101010101

left



right

forward

backward

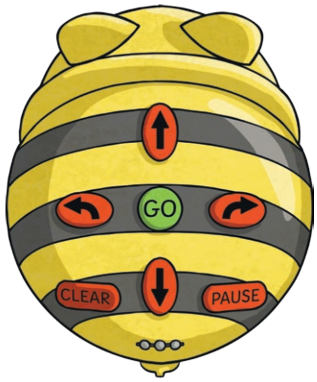
pause

clear

go

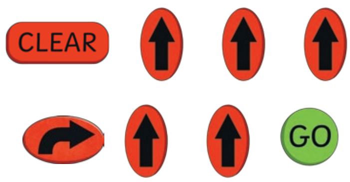
program

Bee-Bot

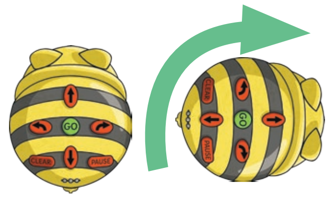


turn

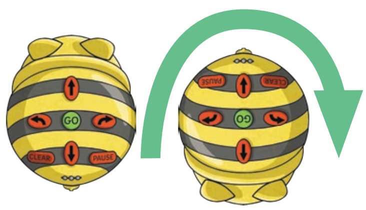
sequence



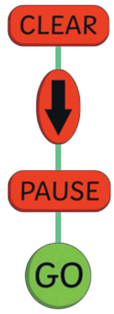
quarter



half



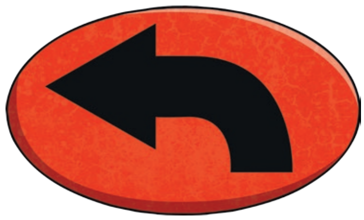
algorithm



code

0110110101010101
0110101101010101
0101011011100101
0110110101010101
0011010111010101
0110101101010101
0110101101010101
0011010111010101
0110101101010101
0110110101010101

left



right

forward

backward

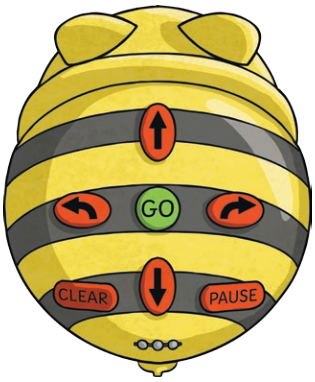
pause

clear

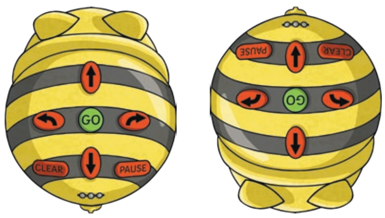
go

program

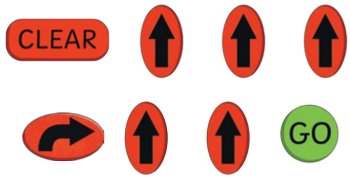
Bee-Bot



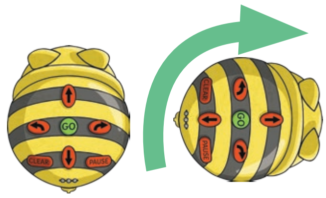
turn



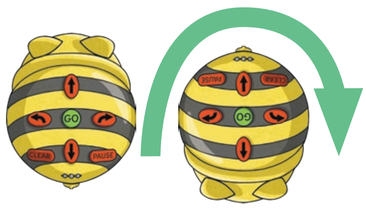
sequence



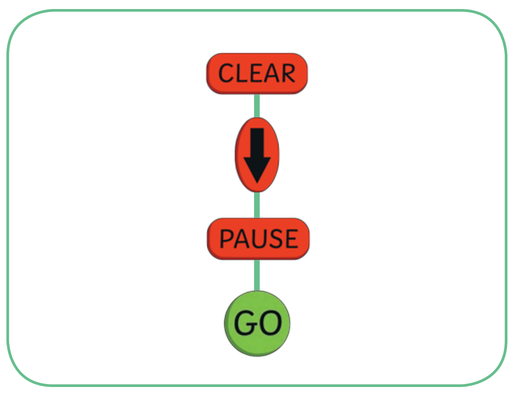
quarter



half

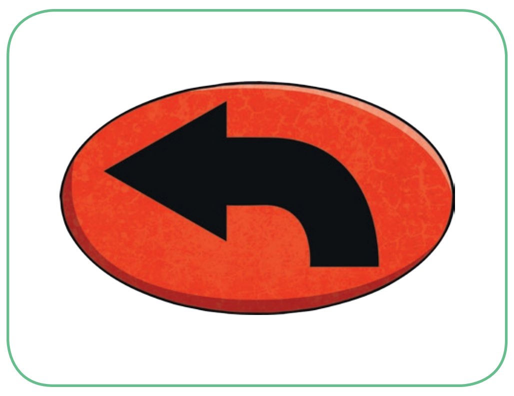


Programming Toys

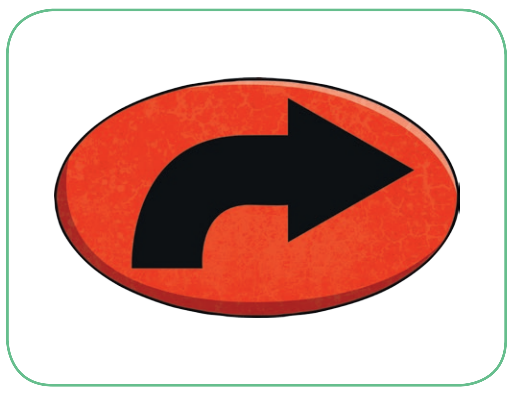


algorithm

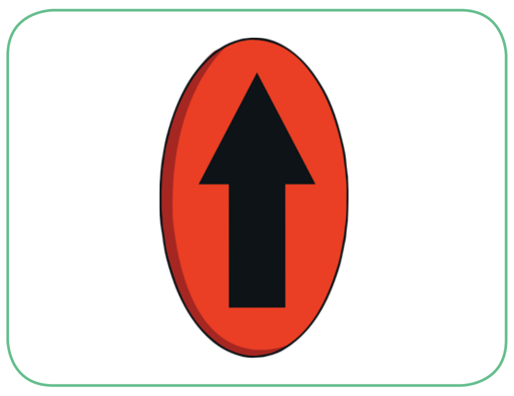
code



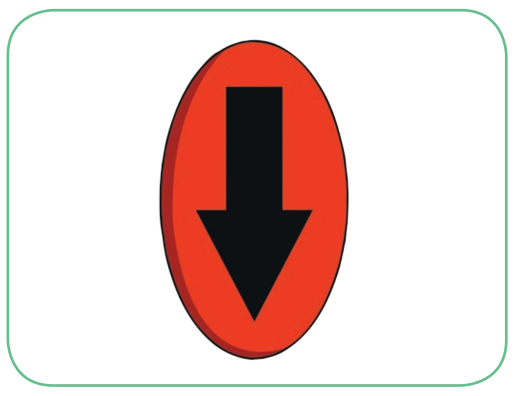
left



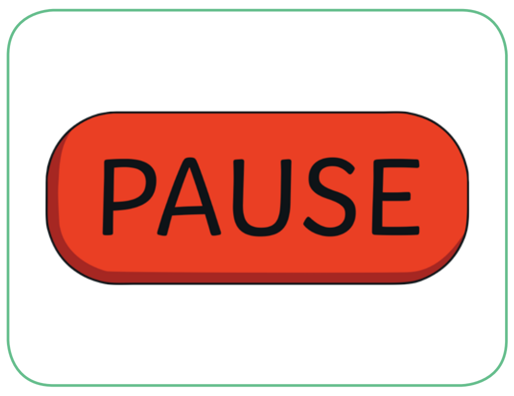
right



forward



backward



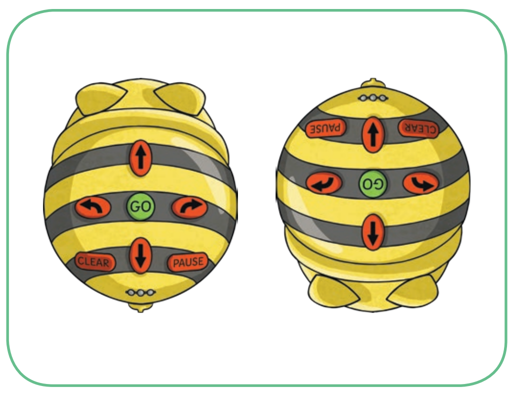
pause

clear

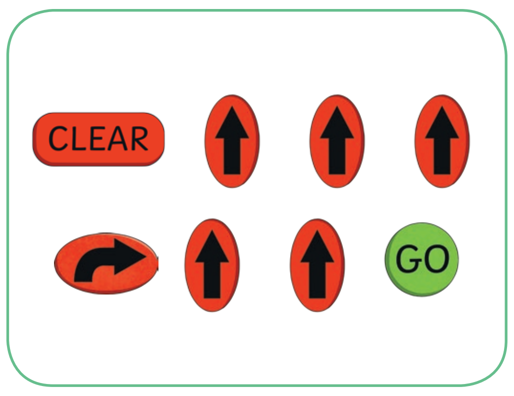
go

program

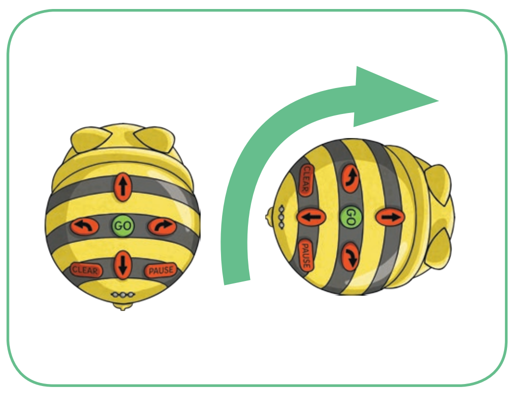
bee-bot



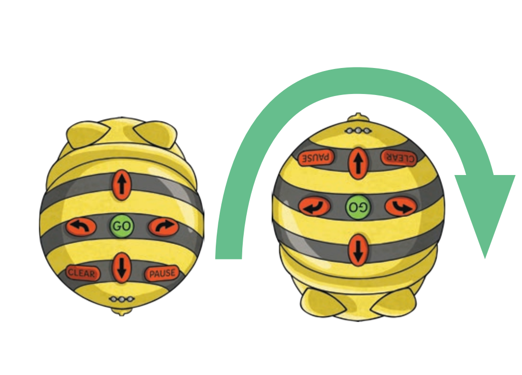
turn



sequence

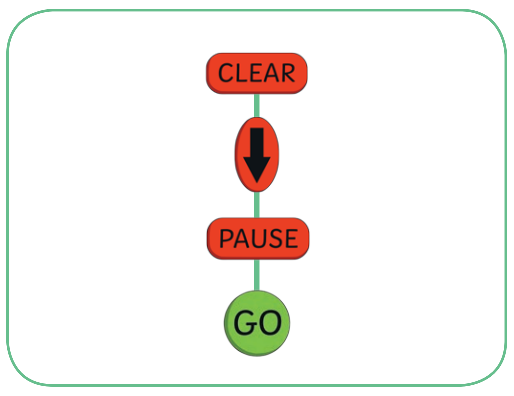


quarter



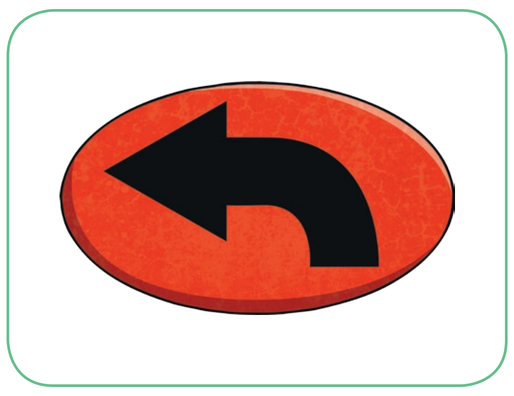
half

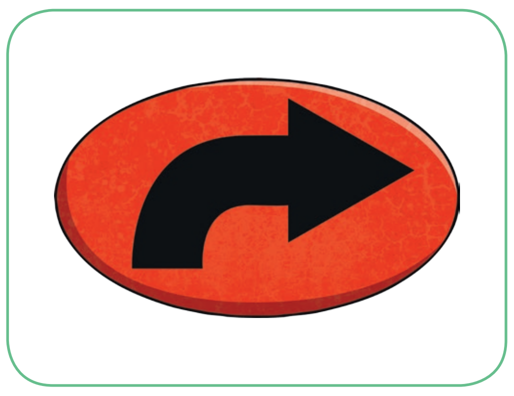
Programming Toys

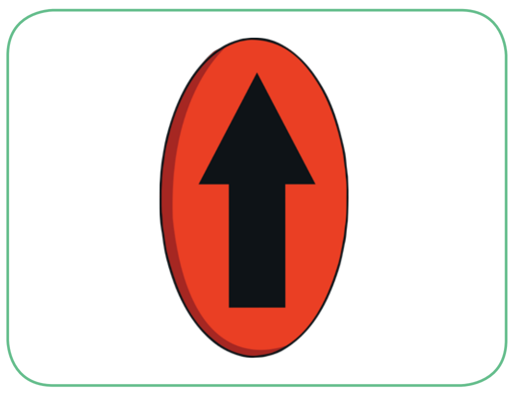


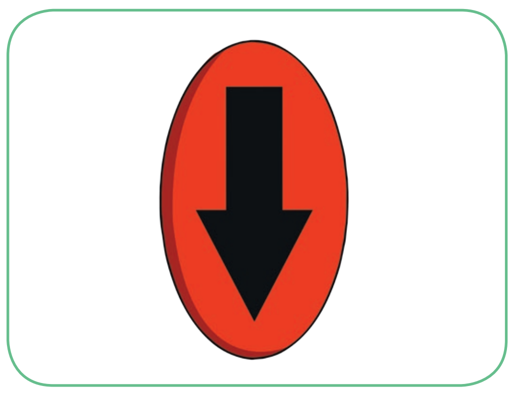
algorithm

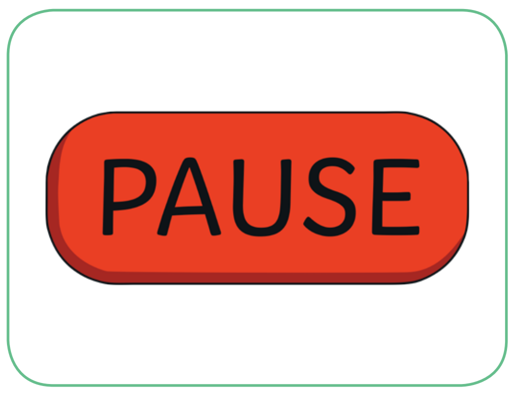
code


left


right


forward


backward



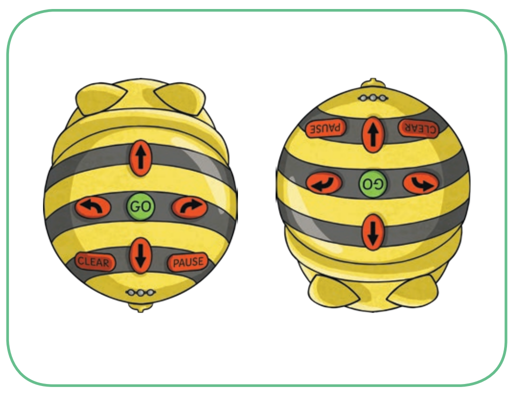
pause

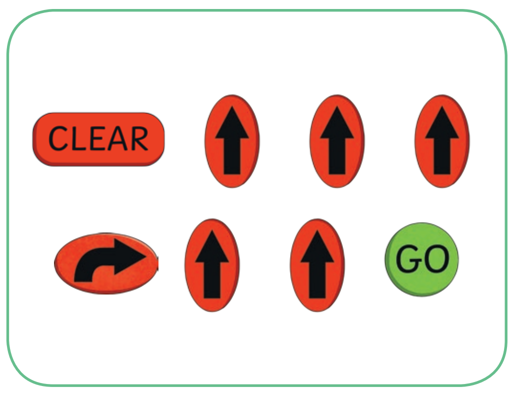
clear

go

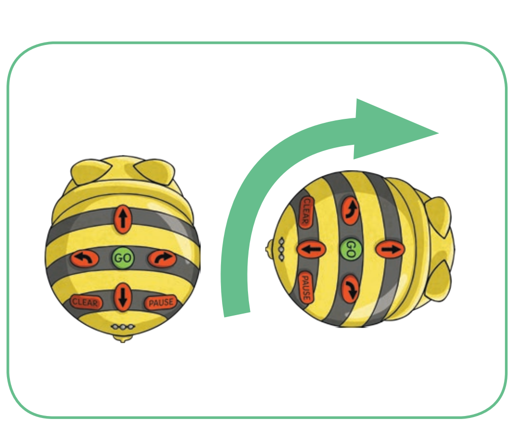

program

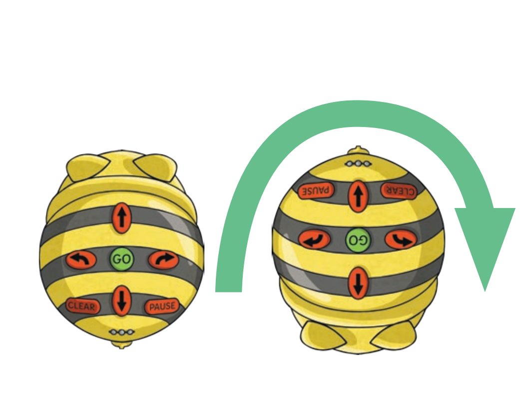

bee-bot


turn



sequence


quarter


half

Programming Toys

f	q	z	e	d	o	c	m	a
o	t	o	y	s	r	m	s	l
g	d	m	r	u	c	t	v	g
l	e	f	t	e	f	d	r	o
y	n	w	c	f	k	w	o	r
p	r	o	g	r	a	m	x	i
y	u	y	a	r	j	e	y	t
h	t	t	o	b	e	e	b	h
n	q	z	t	h	g	i	r	m

left
turn
Bee-Bot
program

code
toys
right
algorithm

Programming Toys **Answers**

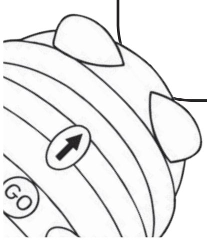
f	q	z	e	d	o	c	m	a
o	t	o	y	s	r	m	s	l
g	d	m	r	u	c	t	v	g
l	e	f	t	e	f	d	r	o
y	n	w	c	f	k	w	o	r
p	r	o	g	r	a	m	x	i
y	u	y	a	r	j	e	y	t
h	t	t	o	b	e	e	b	h
n	q	z	t	h	g	i	r	m

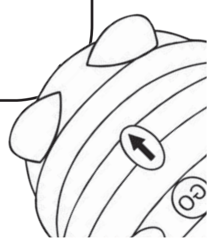
left	code
turn	toys
Bee-Bot	right
program	algorithm

Programming Toys

s	k	d	i	r	e	c	t	i	o	n	x
c	w	q	n	m	f	x	f	l	w	u	f
o	x	q	s	b	i	e	o	w	c	w	c
d	b	g	t	a	s	p	r	e	e	m	a
e	e	m	r	c	b	c	w	s	v	d	l
k	e	a	u	k	h	a	a	e	r	e	g
v	b	r	c	w	a	d	r	q	e	b	o
e	o	g	t	a	l	n	d	u	t	u	r
u	t	o	i	r	f	i	o	e	r	g	i
p	h	r	o	d	q	n	t	n	a	s	t
k	c	p	n	v	n	v	t	c	u	i	h
a	c	b	s	o	t	e	j	e	q	j	m

debug	code	half
sequence	Bee-Bot	quarter
algorithm	forward	direction
instructions	backward	program





Programming Toys **Answers**

s	k	d	i	r	e	c	t	i	o	n	x
c	w	q	n	m	f	x	f	l	w	u	f
o	x	q	s	b	i	e	o	w	c	w	c
d	b	g	t	a	s	p	r	e	e	m	a
e	e	m	r	c	b	c	w	s	v	d	l
k	e	a	u	k	h	a	a	e	r	e	g
v	b	r	c	w	a	d	r	q	e	b	o
e	o	g	t	a	l	n	d	u	t	u	r
u	t	o	i	r	f	i	o	e	r	g	i
p	h	r	o	d	q	n	t	n	a	s	t
k	c	p	n	v	n	v	t	c	u	i	h
a	c	b	s	o	t	e	j	e	q	j	m

debug	code	half
sequence	Bee-Bot	quarter
algorithm	forward	direction
instructions	backward	program

Programming Toys

f	q	z	e	d	o	c	m	a
o	t	o	y	s	r	m	s	l
g	d	m	r	u	c	t	v	g
l	e	f	t	e	f	d	r	o
y	n	w	c	f	k	w	o	r
p	r	o	g	r	a	m	x	i
y	u	y	a	r	j	e	y	t
h	t	t	o	b	e	e	b	h
n	q	z	t	h	g	i	r	m

left
turn
Bee-Bot
program

code
toys
right
algorithm

Programming Toys **Answers**

f	q	z	~~e~~	~~d~~	~~o~~	~~c~~	m	~~a~~
o	~~t~~	~~o~~	~~y~~	~~s~~	r	m	s	~~l~~
g	d	m	r	u	c	t	v	~~g~~
~~l~~	~~e~~	~~f~~	~~t~~	e	f	d	r	~~o~~
y	~~n~~	w	c	f	k	w	o	~~r~~
~~p~~	~~r~~	~~o~~	~~g~~	~~r~~	~~a~~	~~m~~	x	~~i~~
y	~~u~~	y	a	r	j	e	y	~~t~~
h	~~t~~	~~t~~	~~o~~	~~b~~	~~e~~	~~e~~	~~b~~	~~h~~
n	q	z	~~t~~	~~h~~	~~g~~	~~i~~	~~r~~	~~m~~

left	code
turn	toys
Bee-Bot	right
program	algorithm

Programming Toys

s	k	d	i	r	e	c	t	i	o	n	x
c	w	q	n	m	f	x	f	l	w	u	f
o	x	q	s	b	i	e	o	w	c	w	c
d	b	g	t	a	s	p	r	e	e	m	a
e	e	m	r	c	b	c	w	s	v	d	l
k	e	a	u	k	h	a	a	e	r	e	g
v	b	r	c	w	a	d	r	q	e	b	o
e	o	g	t	a	l	n	d	u	t	u	r
u	t	o	i	r	f	i	o	e	r	g	i
p	h	r	o	d	q	n	t	n	a	s	t
k	c	p	n	v	n	v	t	c	u	i	h
a	c	b	s	o	t	e	j	e	q	j	m

debug	code	half
sequence	Bee-Bot	quarter
algorithm	forward	direction
instructions	backward	program

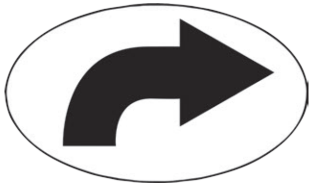
Programming Toys **Answers**

s	k	d	i	r	e	c	t	i	o	n	x
c	w	q	n	m	f	x	f	l	w	u	f
o	x	q	s	b	i	e	o	w	c	w	c
d	b	g	t	a	s	p	r	e	e	m	a
e	e	m	r	c	b	c	w	s	v	d	l
k	e	a	u	k	h	a	a	e	r	e	g
v	b	r	c	w	a	d	r	q	e	b	o
e	o	g	t	a	l	n	d	u	t	u	r
u	t	o	i	r	f	i	o	e	r	g	i
p	h	r	o	d	q	n	t	n	a	s	t
k	c	p	n	v	n	v	t	c	u	i	h
a	c	b	s	o	t	e	j	e	q	j	m

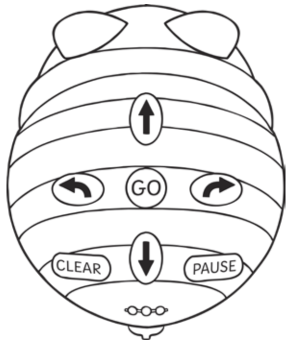
debug	code	half
sequence	Bee-Bot	quarter
algorithm	forward	direction
instructions	backward	program

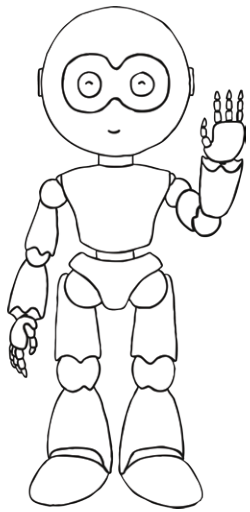
CLEAR

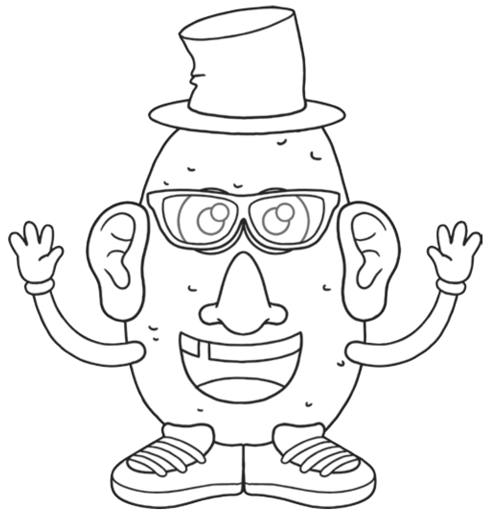
PAUSE



GO

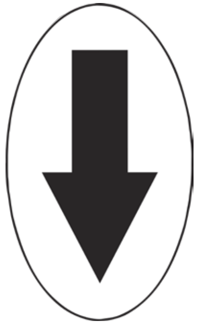

GO
CLEAR
PAUSE

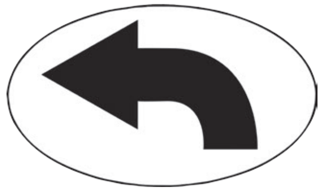


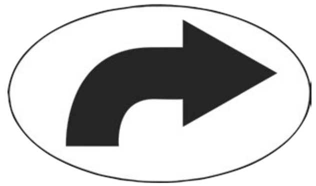


CLEAR

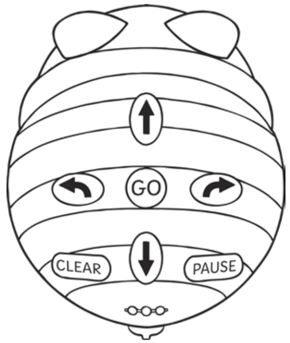
PAUSE

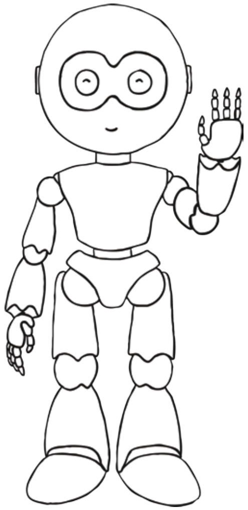


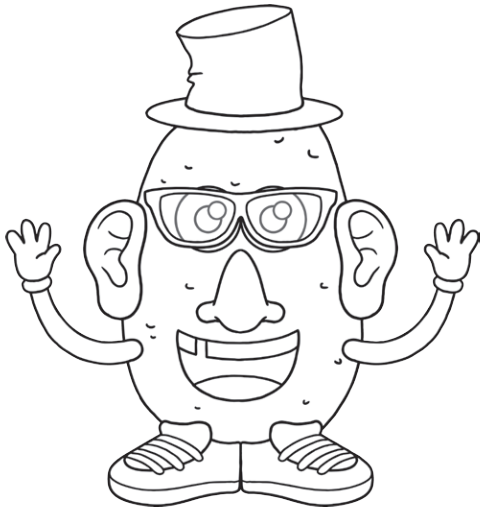




GO

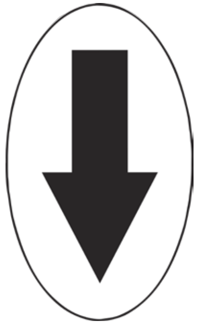

GO
CLEAR
PAUSE

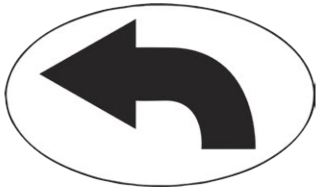


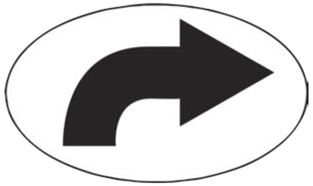


CLEAR

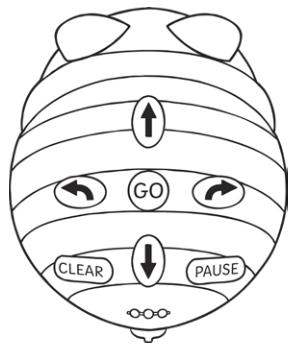
PAUSE

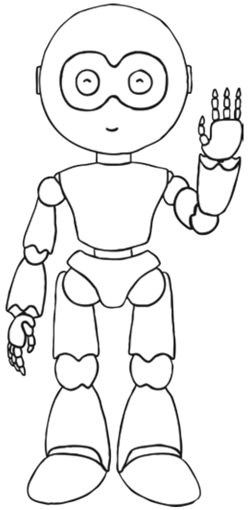


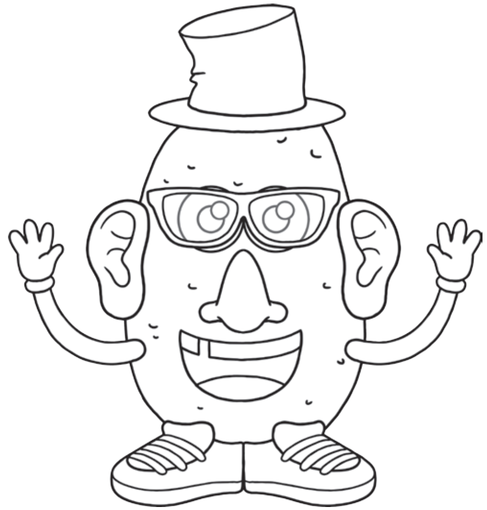




GO

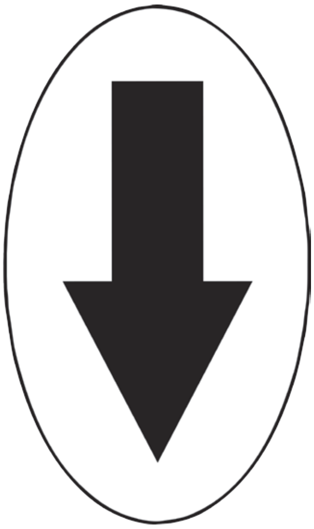

GO
CLEAR
PAUSE

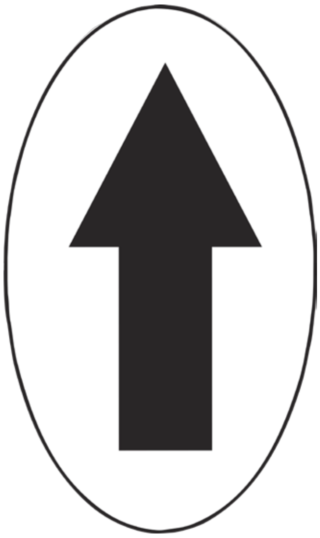


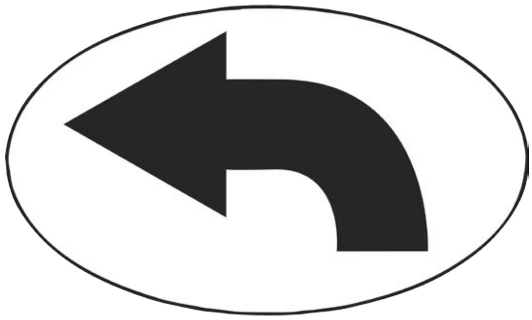


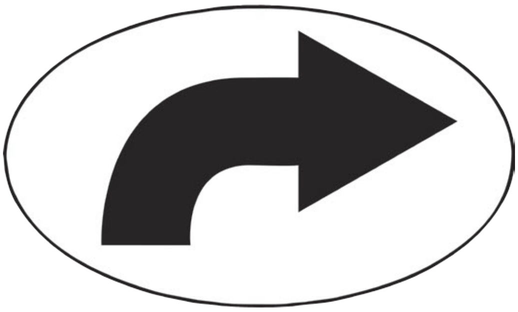
CLEAR

PAUSE

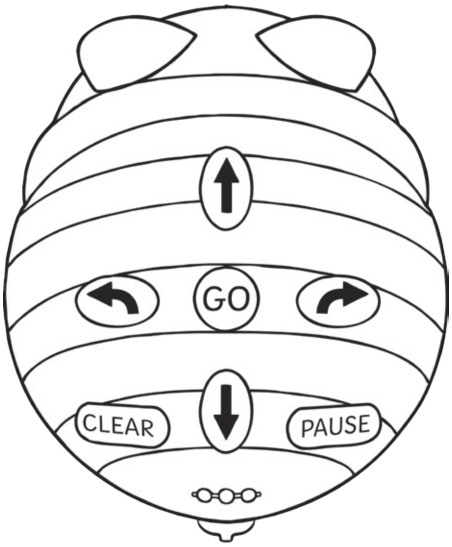


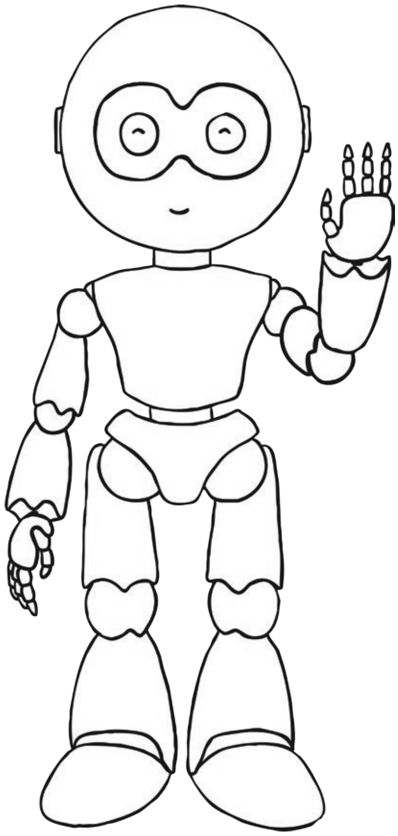


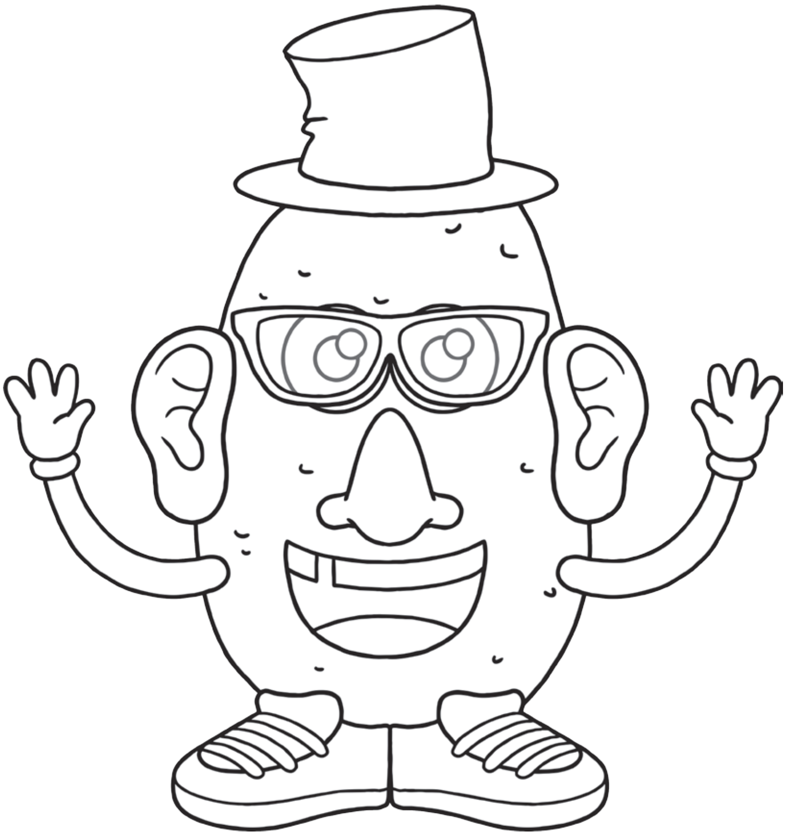




GO

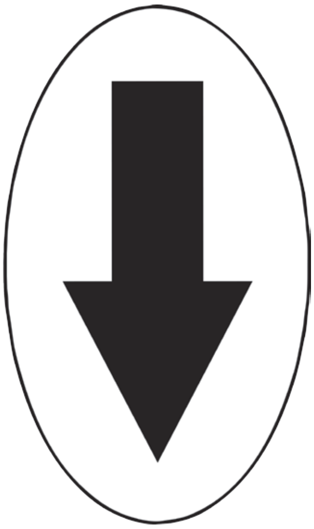

GO
CLEAR
PAUSE

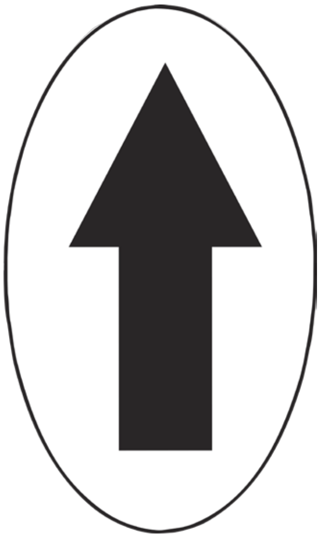


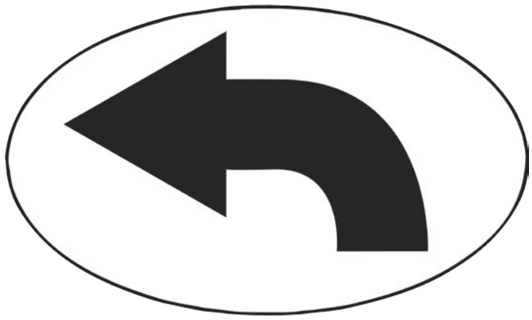


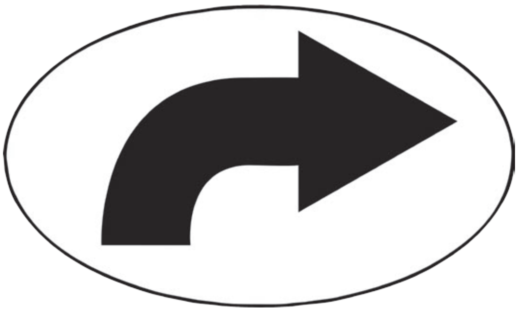
CLEAR

PAUSE

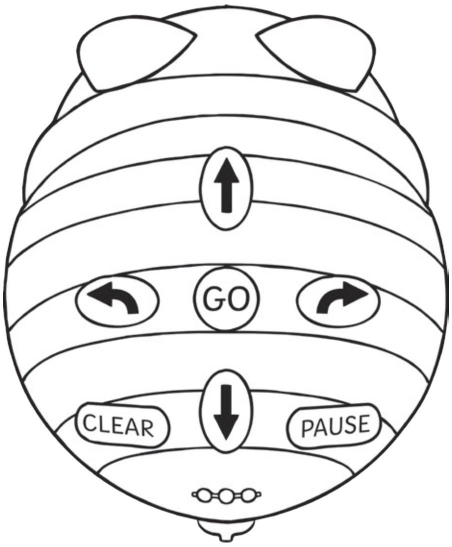


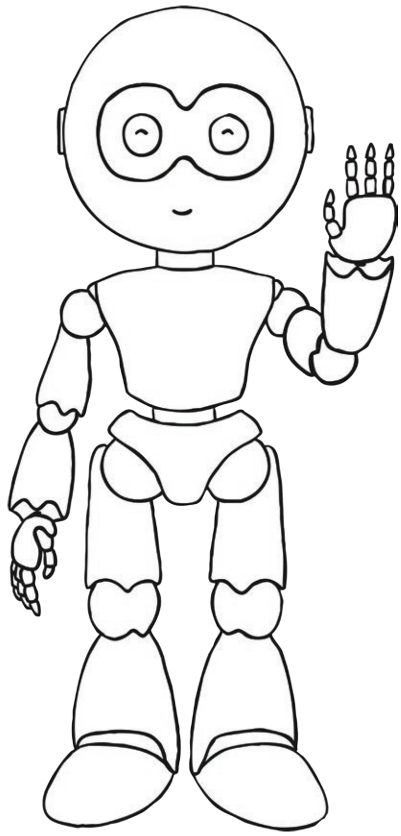


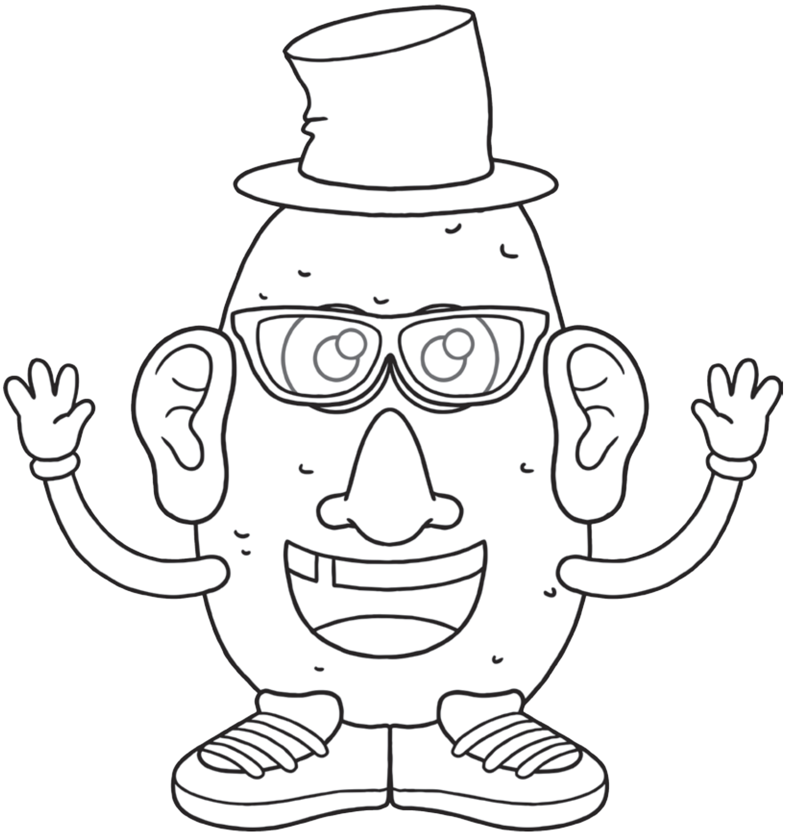




GO

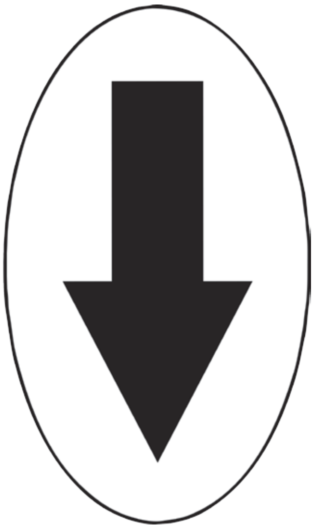

GO
CLEAR
PAUSE

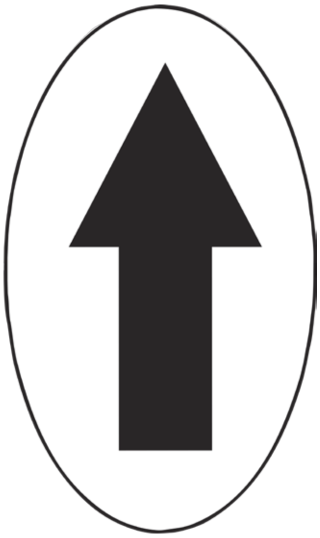


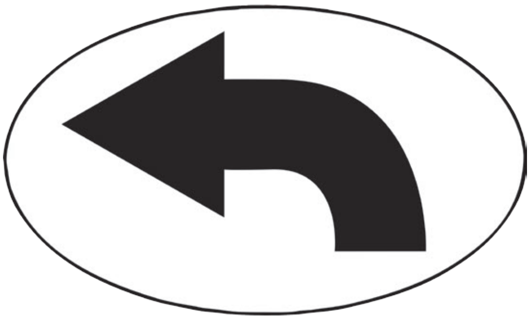


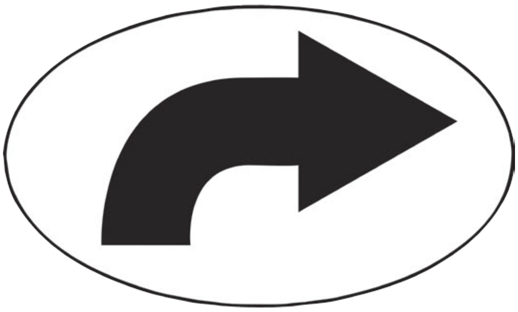
CLEAR

PAUSE

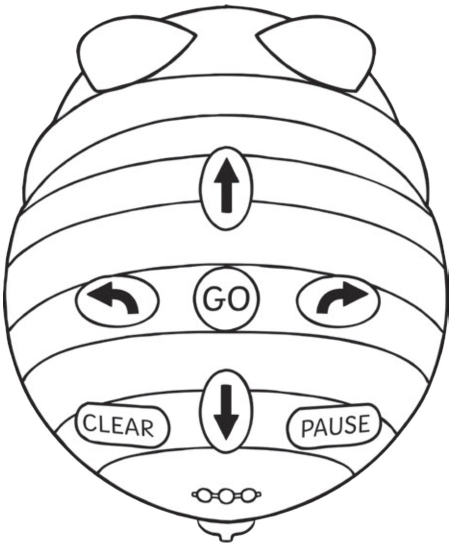


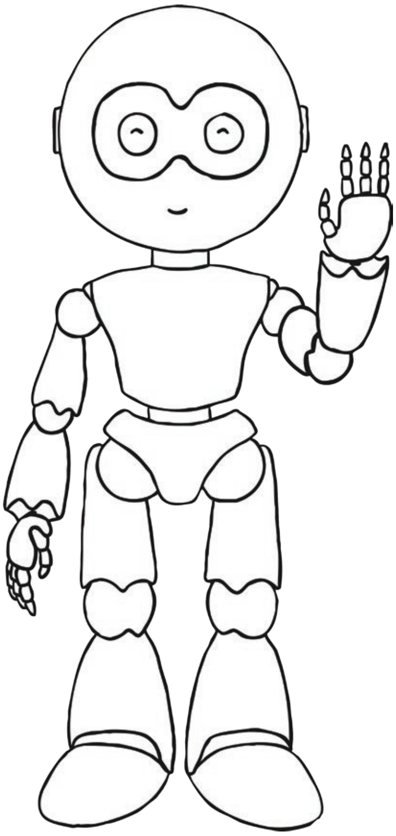


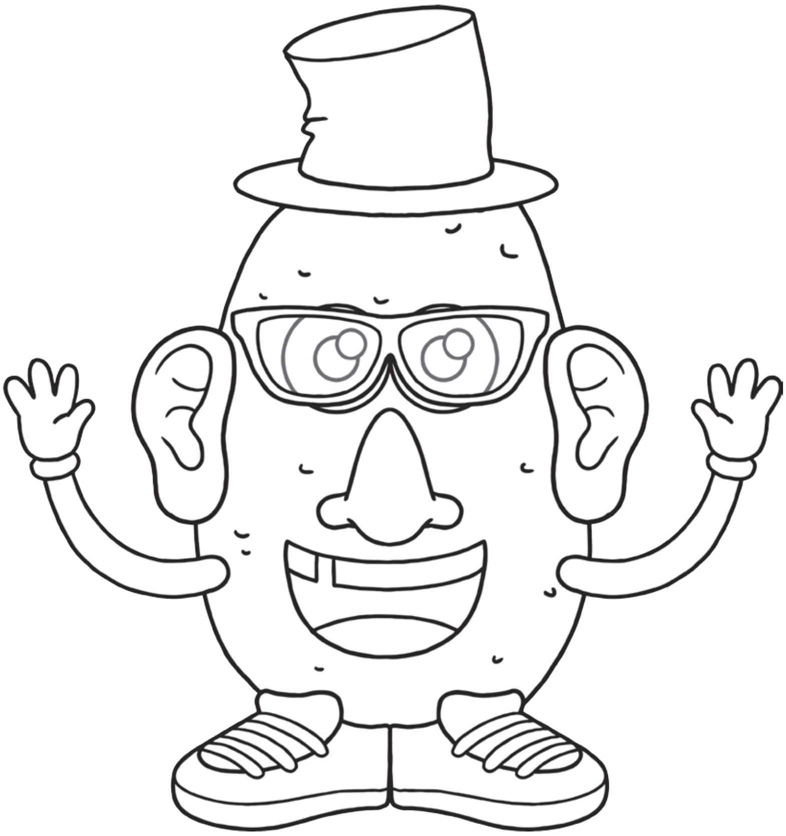




GO


GO
CLEAR
PAUSE

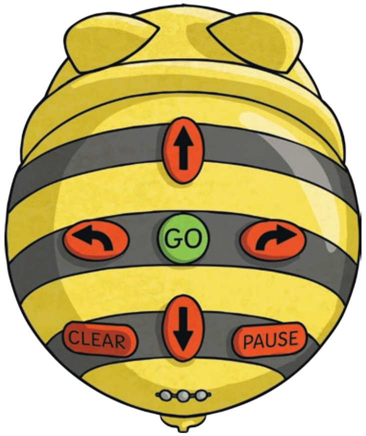


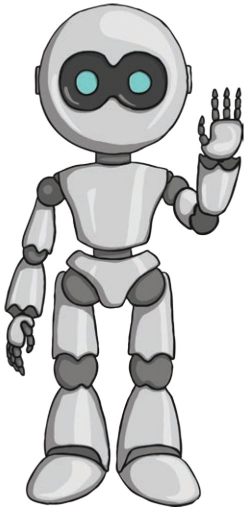


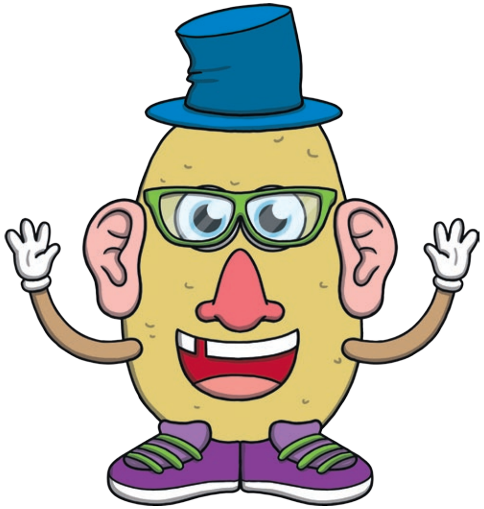

CLEAR


PAUSE


GO

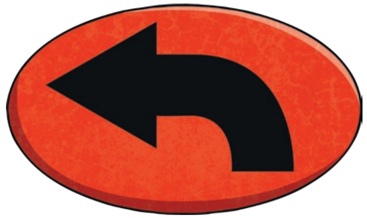

GO
CLEAR
PAUSE

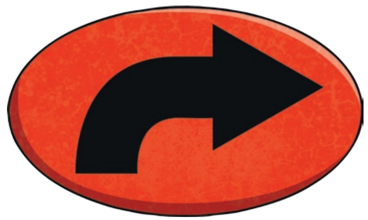




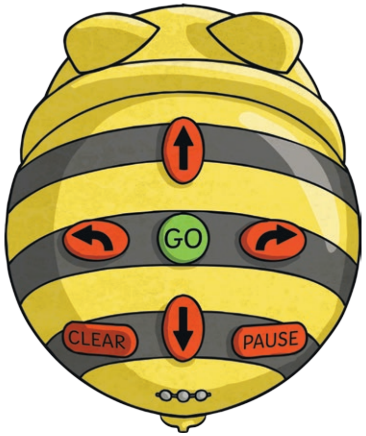

CLEAR

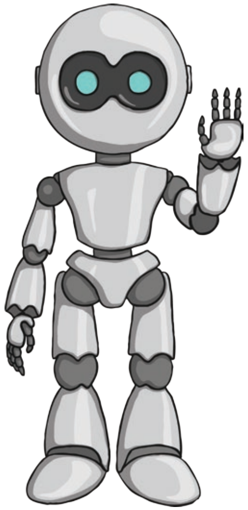

PAUSE

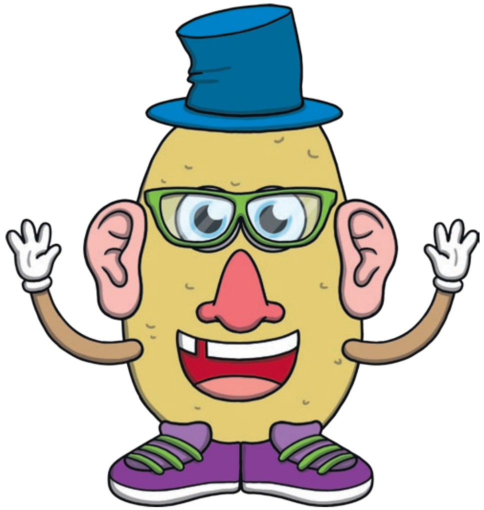





GO


GO
CLEAR
PAUSE

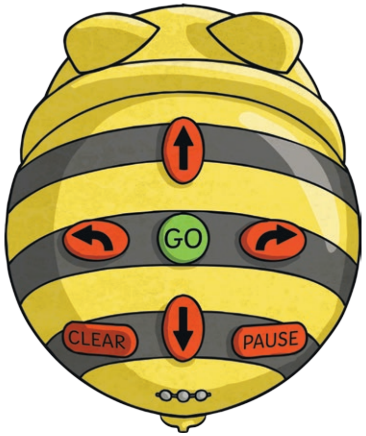


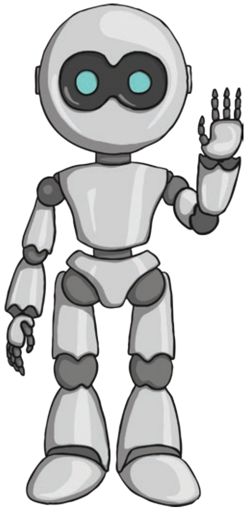


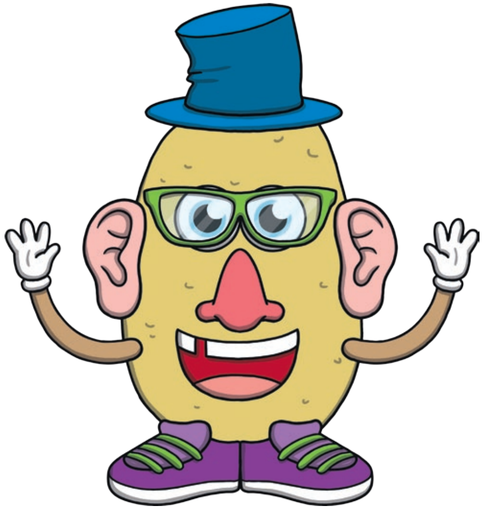

CLEAR


PAUSE

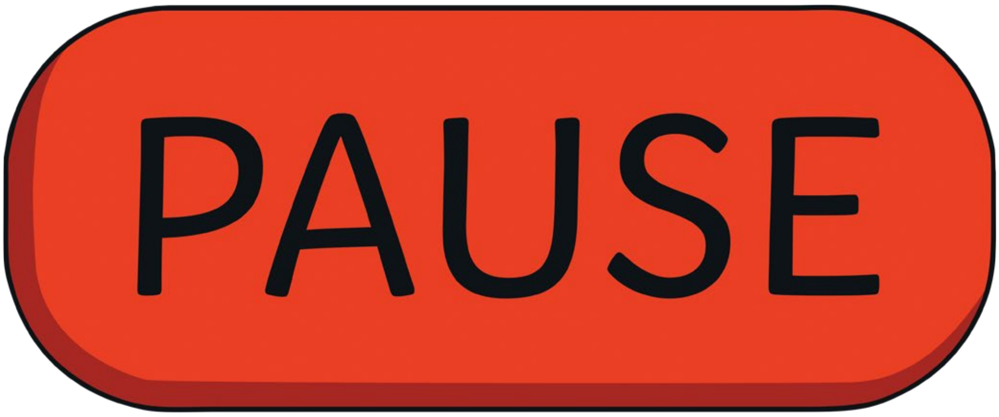

GO

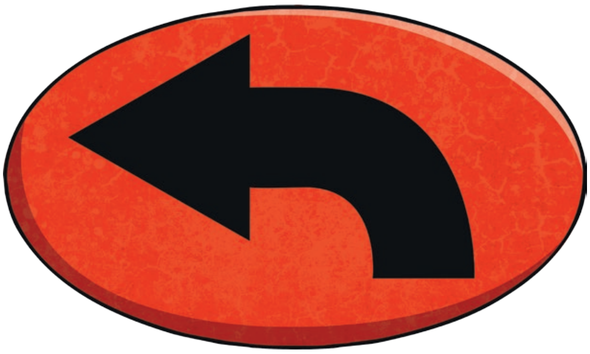

GO
CLEAR
PAUSE

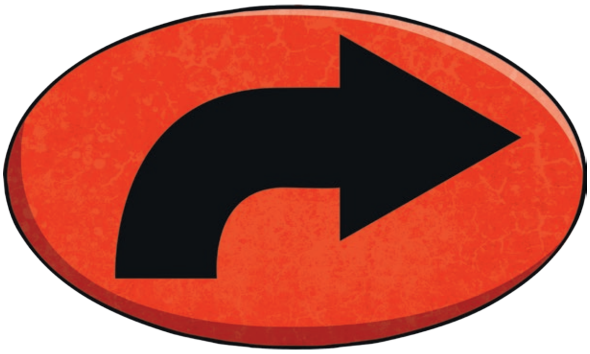




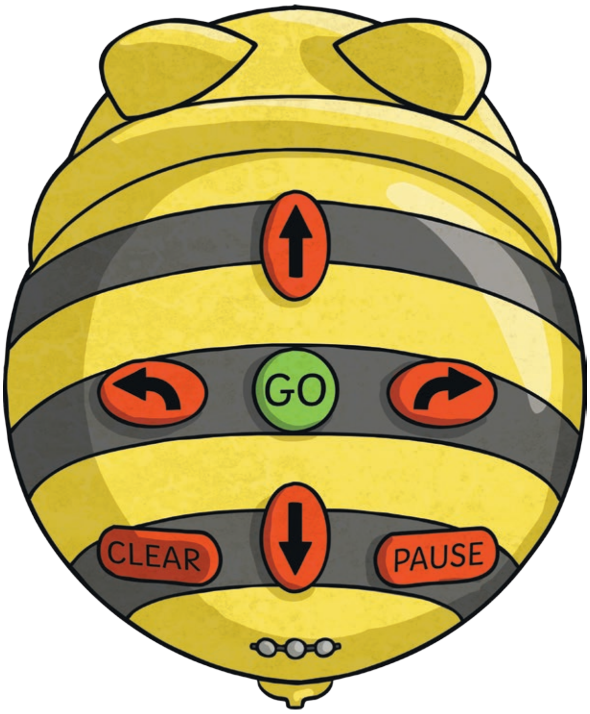

CLEAR

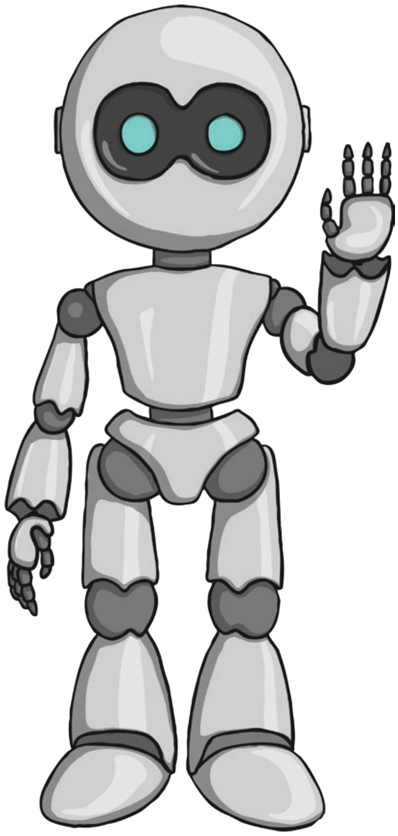

PAUSE

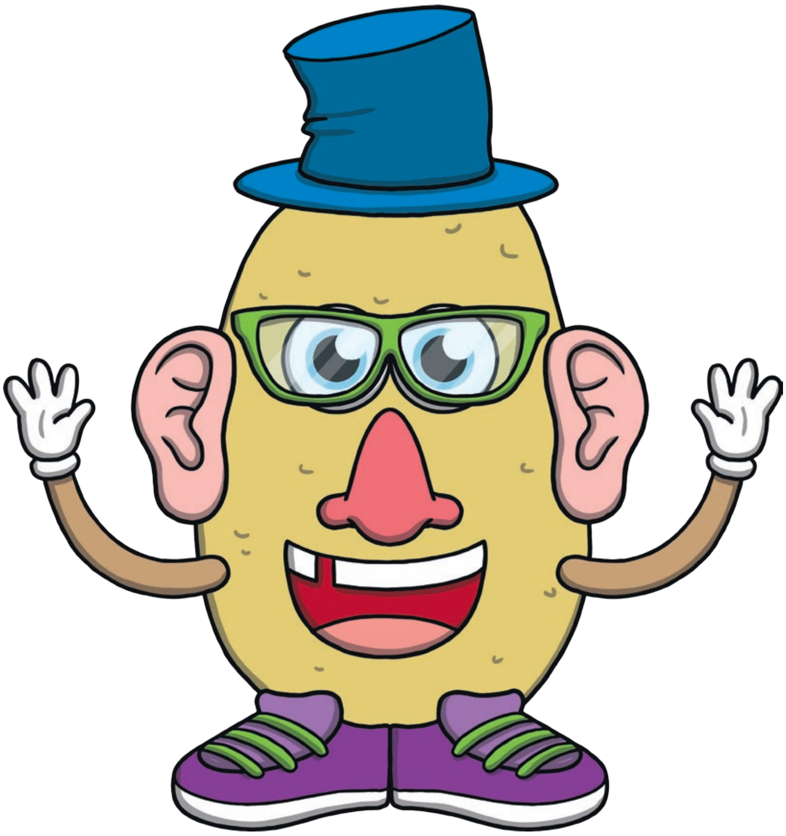




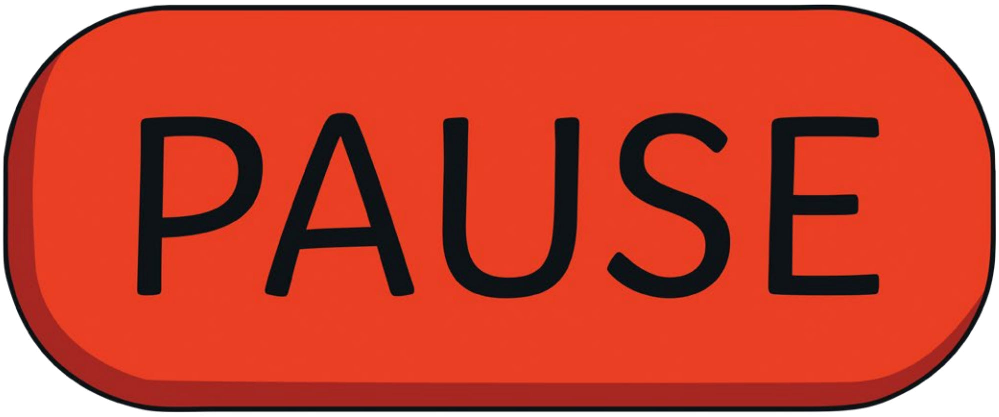

GO

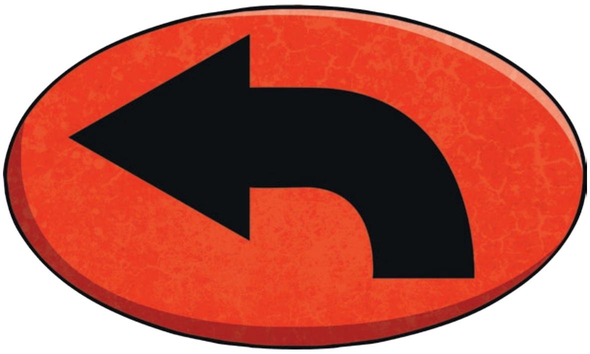

GO
CLEAR
PAUSE

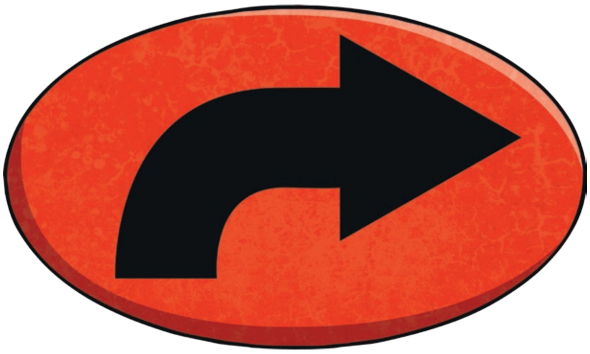




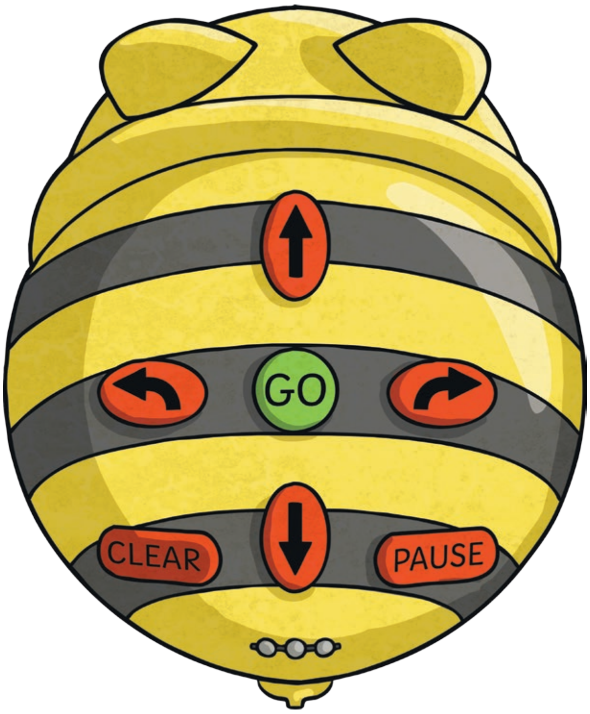

CLEAR

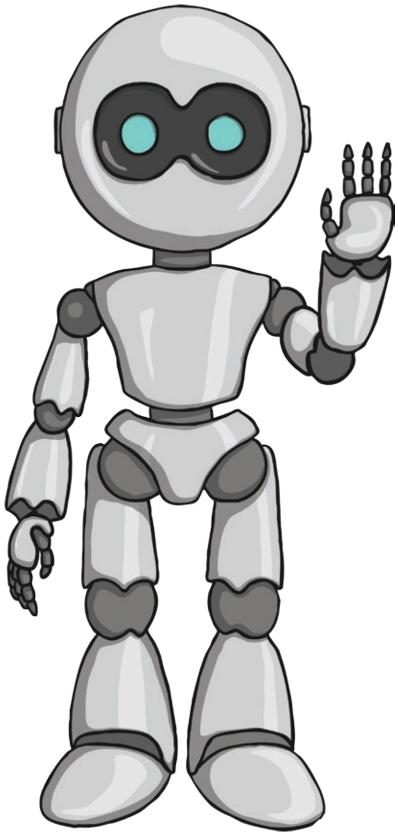

PAUSE

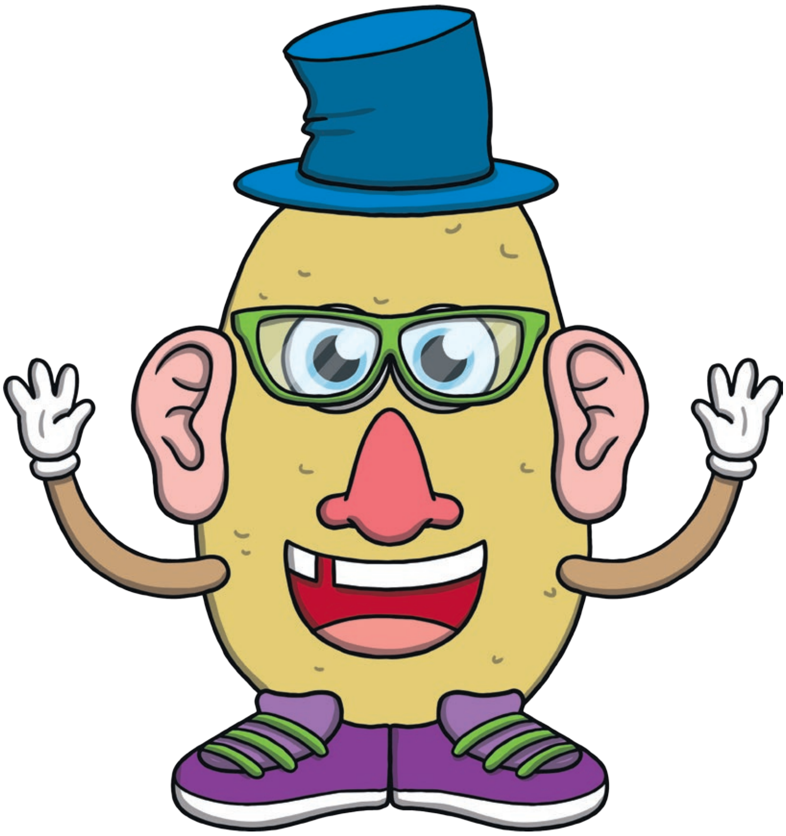




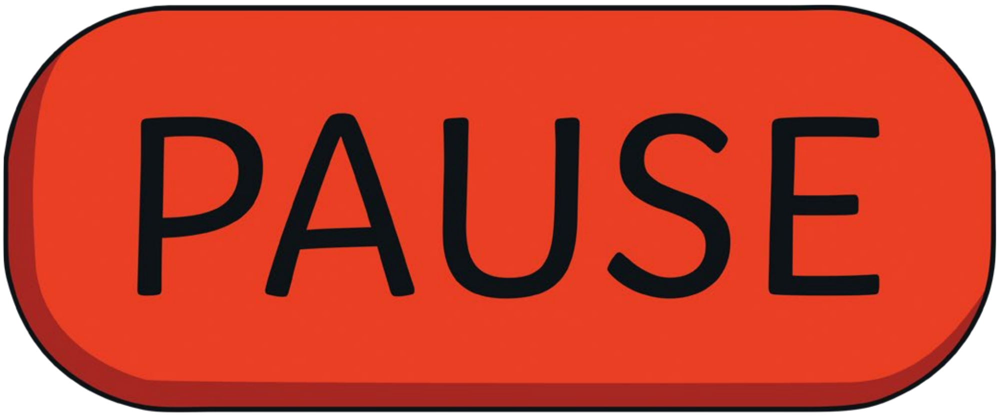

GO

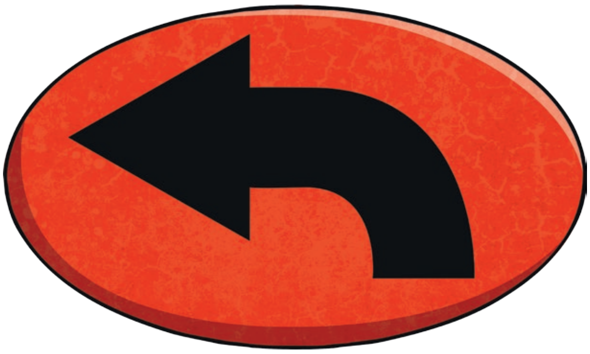

GO
CLEAR
PAUSE

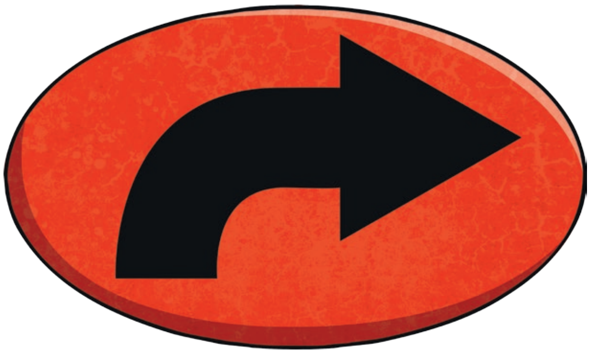




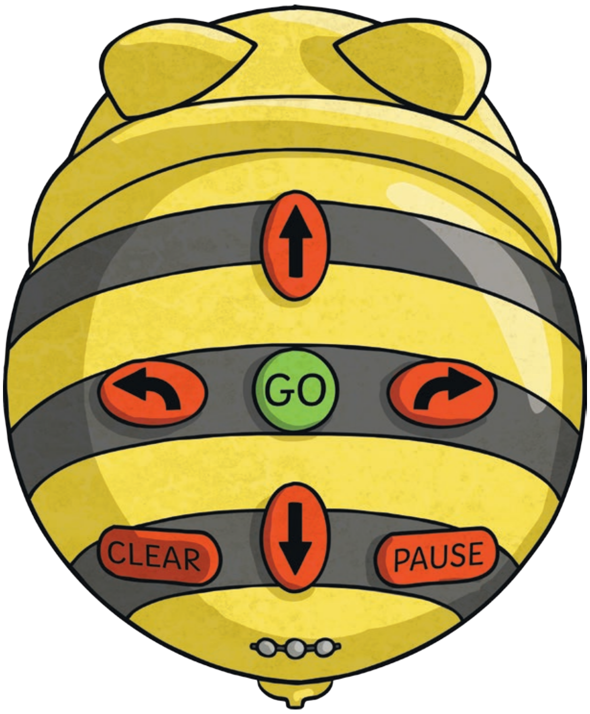

CLEAR

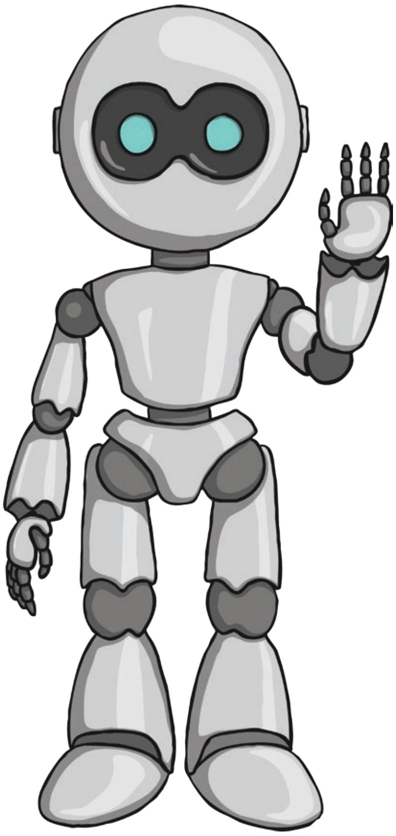

PAUSE

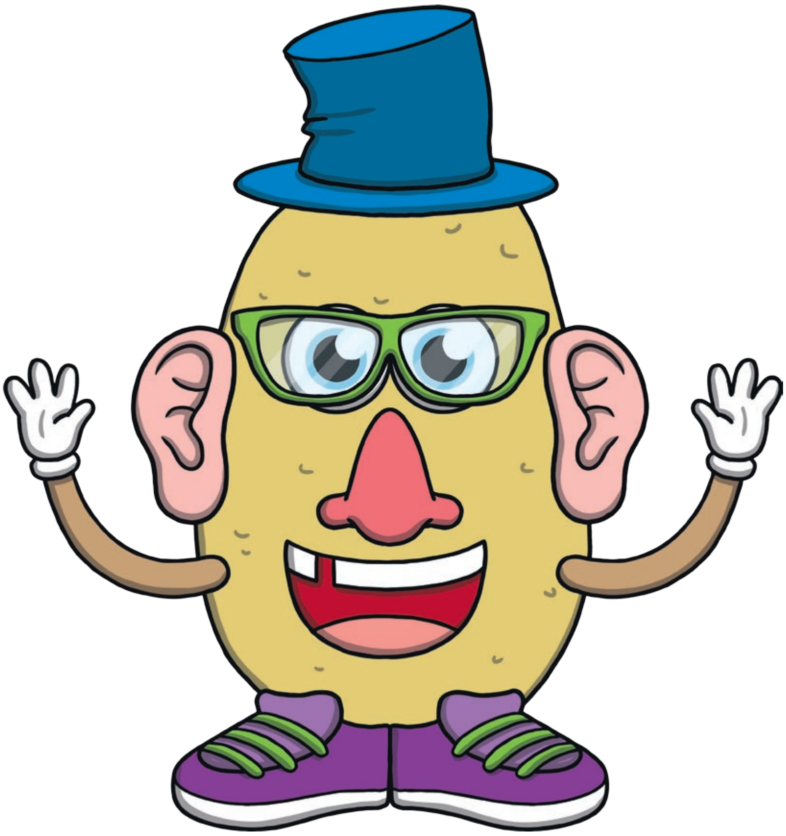





GO


GO
CLEAR
PAUSE





End of Unit Assessment | Computing | Year 1 | Programming Toys

All	Most	Some
Create step-by-step instructions using pictures, write and follow detailed step-by-step instructions, direct a Bee-Bot (or similar programmable toy) to a toy, program a Bee-Bot (or similar programmable toy) one instruction at a time, using the arrow buttons.	Say what an algorithm is, say why it is important to be precise when writing an algorithm, check their work for mistakes (debug); program a Bee-Bot (or similar programmable toy) using the arrow buttons, start their programming sequence again if they need to, check their work for mistakes to debug a program, plan and check an algorithm.	See how a product changes when they change the instructions, evaluate and improve their sequence (debug).
33%	33%	33%
Name	Name	Name
Name	Name	Name
Name	Name	Name
Name	Name	Name

End of Unit Assessment | Computing | Year 1 | Programming Toys

Insert a character against the criteria the child has met. If they have not met the criteria leave it blank.

		Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	% of class
	% met by child	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	
	Has the child met the all and most statements?	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	0%
All	Create step-by-step instructions using pictures.																																			0%
All	Write and follow detailed step-by-step instructions.																																			0%
All	Direct a Bee-Bot (or similar programmable toy) to a toy.																																			0%
All	Program a Bee-Bot (or similar programmable toy) one instruction at a time, using the arrow buttons.																																			0%
Most	Say what an algorithm is.																																			0%
Most	Say why it is important to be precise when writing an algorithm.																																			0%
Most	check their work for mistakes (debug).																																			0%
Most	Program a Bee-Bot (or similar programmable toy) using the arrow buttons.																																			0%
Most	Start their programming sequence again if they need to.																																			0%
Most	Check their work for mistakes to debug a program.																																			0%
Most	Plan and check an algorithm.																																			0%
Some	See how a product changes when they change the instructions.																																			0%
Some	Evaluate and improve their sequence (debug).																																			0%

End of Unit Assessment | Computing | Year 1 | Programming Toys

Lesson Aim — *Success Criteria*

Insert a character against the criteria the child has met. If they have not met the criteria leave it blank.

	Lesson Aim / *Success Criteria*	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	
	% met by child	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	
1	I can create instructions using pictures.																																			0%
	I know what an algorithm is.																																			0%
	I can create step-by-step instructions using pictures.																																			0%
2	I can say why it is important to be precise when writing an algorithm.																																			0%
	I can write and follow detailed instructions.																																			0%
	I can see how a product changes when I change the instructions.																																			0%
3	I can write instructions to program a person like a computer.																																			0%
	I can write step-by-step instructions.																																			0%
	I can check my work for mistakes (debug).																																			0%
4	I can program a Bee-Bot (or similar programmable toy) to move.																																			0%
	I can direct a Bee-Bot (or similar programmable toy) to a toy.																																			0%
	I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.																																			0%
5	I can debug a Bee-Bot (or similar programmable toy).																																			0%
	I can check my work for mistakes to debug a program.																																			0%
	I can start my programming sequence again if I need to.																																			0%
6	I can program a sequence to make a Bee-Bot (or similar programmable toy) move.																																			0%
	I can plan and check an algorithm.																																			0%
	I can evaluate and improve my sequence (debug).																																			0%

NC Aims Covered in the Programming Toys

Use technology purposefully to create digital content.

Understand how [algorithms] are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions

Understand what algorithms are and that programs execute by following precise and unambiguous instructions.

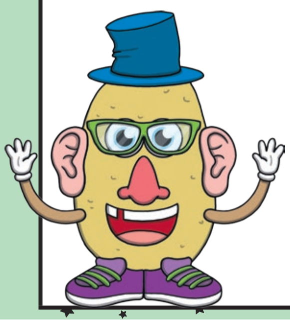
Create and debug simple programs.

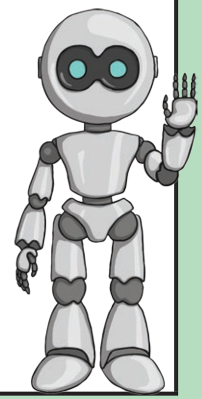
I can... Computing | Year 1 | Programming Toys

Lesson 1	Lesson 2	Lesson 3	Lesson 4	Lesson 5	Lesson 6
I can create instructions using pictures.	**I can say why it is important to be precise when writing an algorithm.**	**I can write instructions to program a person like a computer.**	**I can program a Bee-Bot (or similar programmable toy) to move.**	**I can debug a Bee-Bot (or similar programmable toy).**	**I can program a sequence to make a Bee-Bot (or similar programmable toy) move.**
I know what an algorithm is.	I can write and follow detailed instructions.	I can write step-by-step instructions.	I can direct a Bee-Bot (or similar programmable toy) to a toy.	I can check my work for mistakes to debug a program.	I can plan and check an algorithm.
I can create step-by-step instructions using pictures.	I can see how a product changes when I change the instructions.	I can check my work for mistakes (debug).	I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.	I can start my programming sequence again if I need to.	I can evaluate and improve my sequence (debug).

Computing: Programming Toys

K	W	L
What I know	What I want to know	What I have learnt

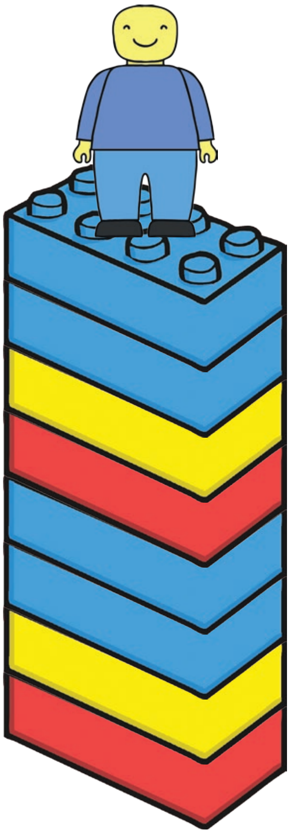




Ordering Instructions

You have been learning about algorithms. They are a set of instructions used to tell a computer program what to do.

Can you order the instructions for building this tower to show which order they should go in? If you have bricks at home, you could try it.



	Stick a yellow brick on top of the red brick.
	Stick a toy man on top.
	Start with a red brick.
	Repeat the pattern again and stick it on top.
	Stick a blue brick on top.
	Stick on another brick of the same colour.

What happens if you change the order? Do you think you would still be able to build the tower? Can you explain why you think this?

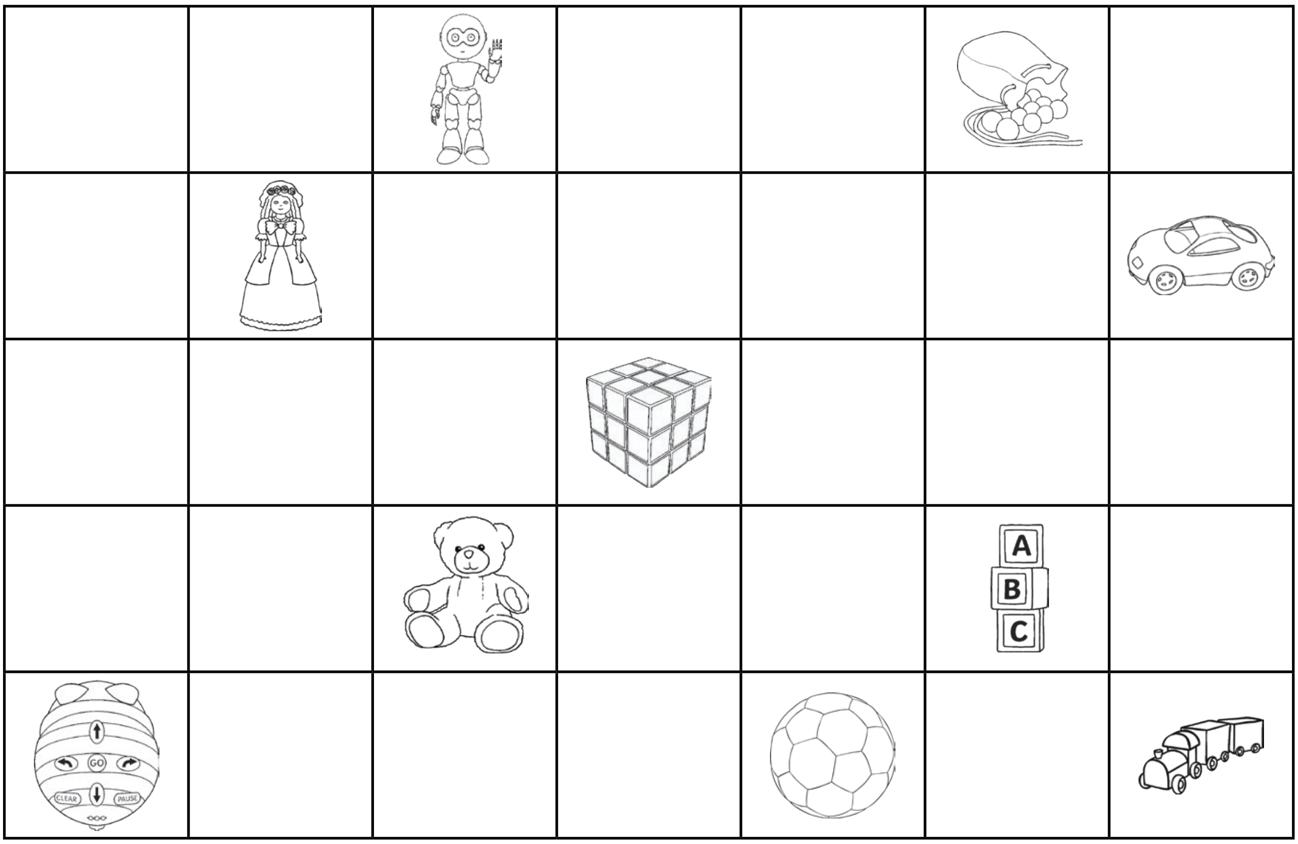
Challenge

Can you think of a way to make these instructions clearer? What else could you include?

Using Symbols in Algorithms

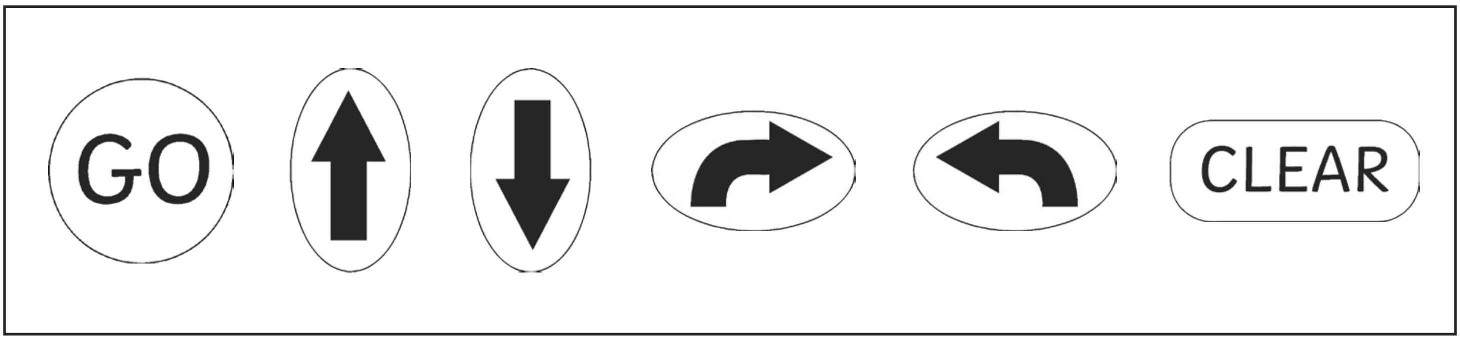
Your programmable toy wants to play with its favourite toy. Can you choose which toy it will play with, then draw arrow symbols to show how it could get there?

You are **not** allowed to go over any other toys!



My programmable toy is going to play with the____.

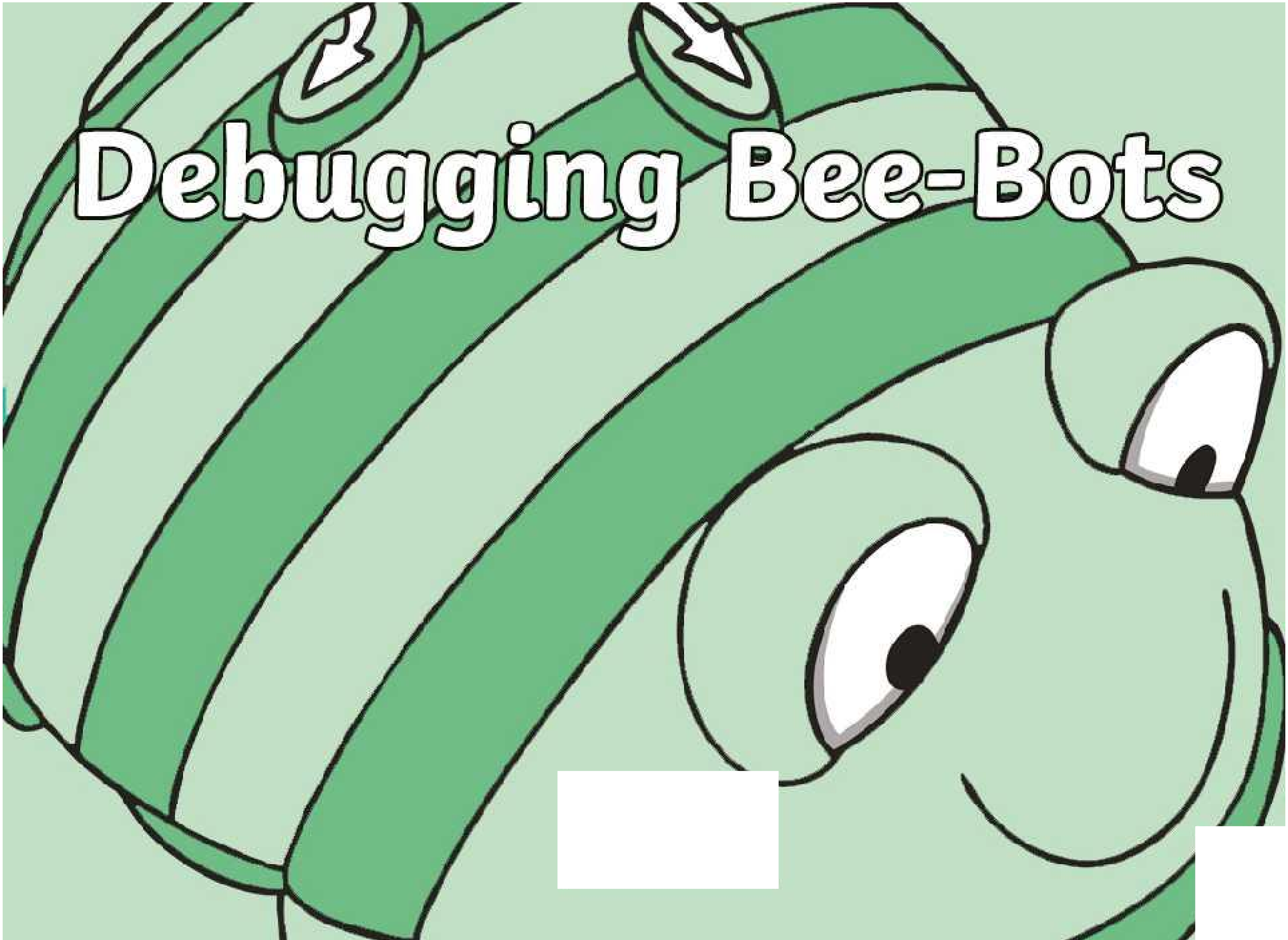
These are the instructions I would use:



Computing

Programming Toys

Debugging Bee-Bots



Aim

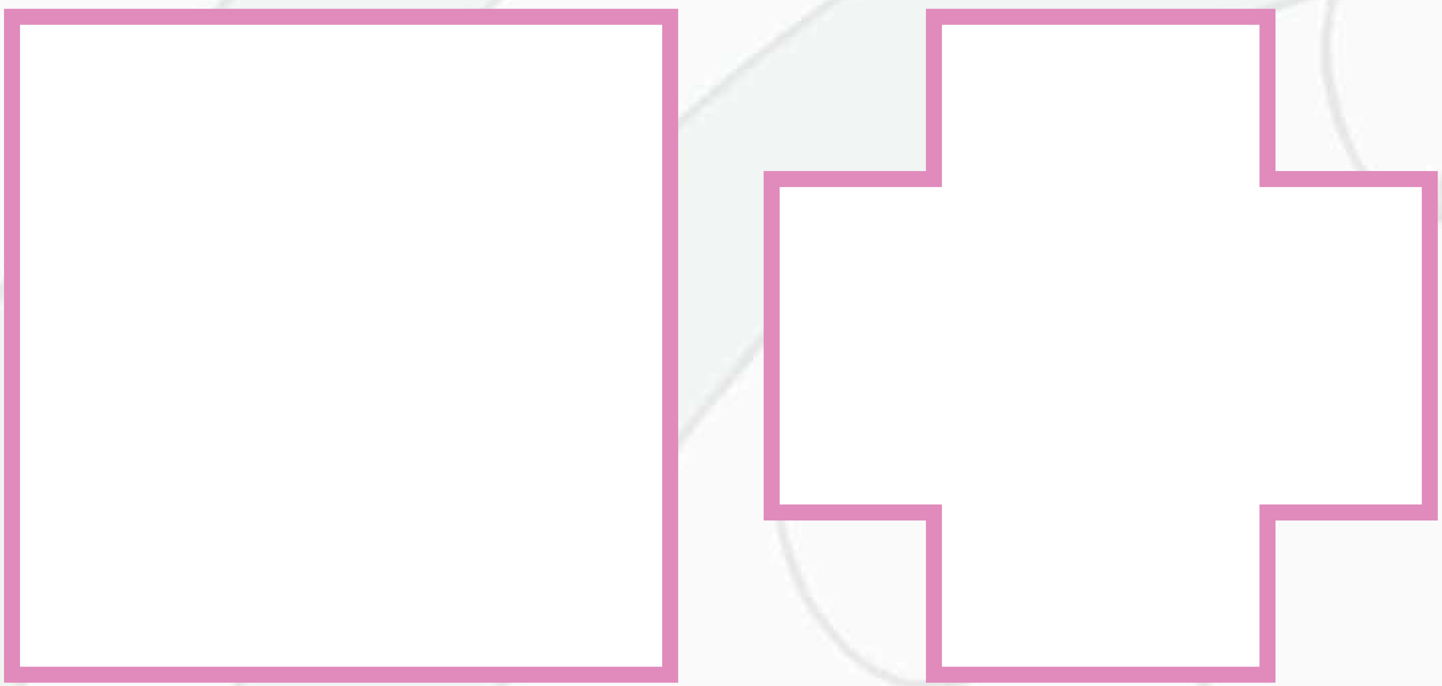
- I can debug a Bee-Bot.

Success Criteria

- I can check my work for mistakes to debug a program.
- I can start my programming sequence again if I need to.

Draw a Shape

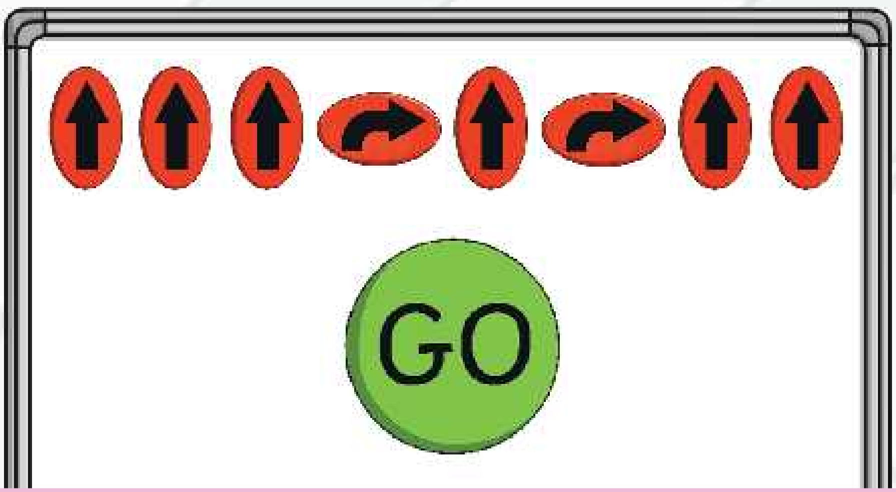
Can you make your Bee-Bot move in a shape? Try one of these:



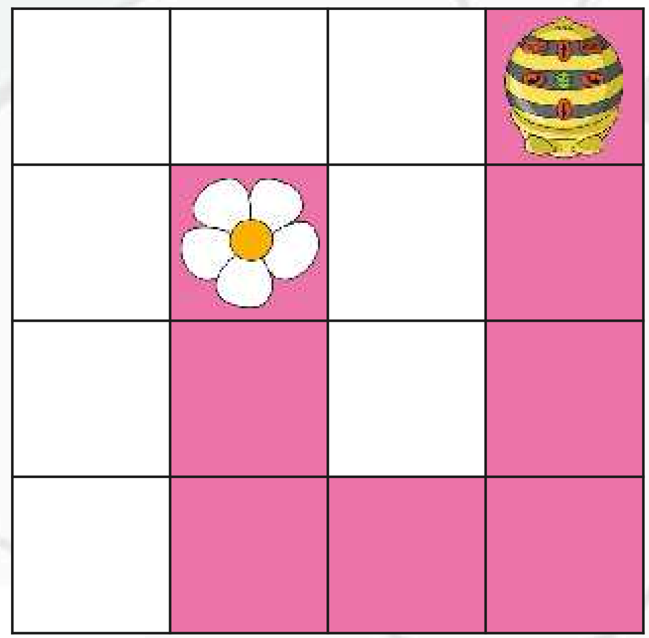
What Went Wrong?

I want to get my Bee-Bot to follow the purple path to the flower.
If I press the buttons shown, where will it end up?

Press GO to find out if you were correct.



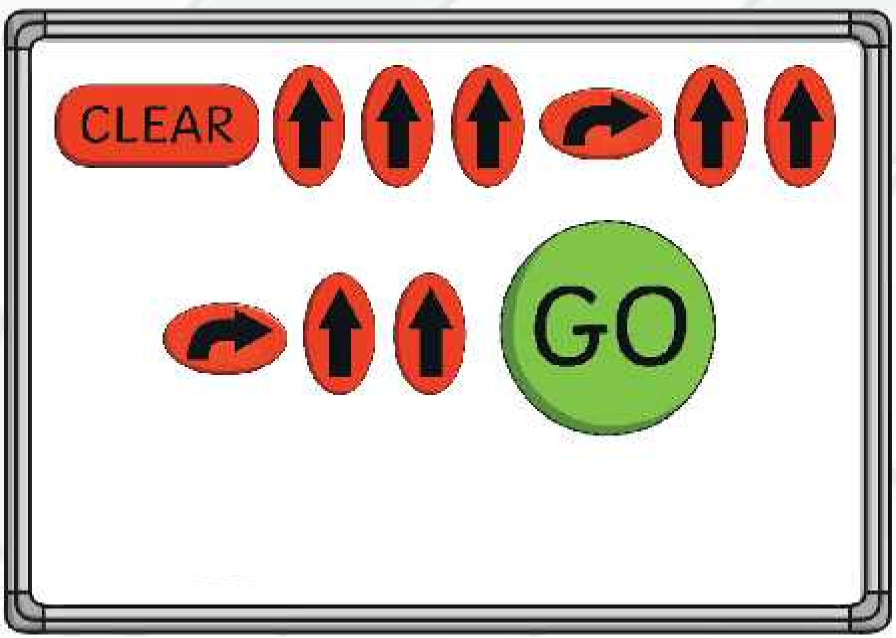
Can you draw the correct buttons to press, to reach the flower?

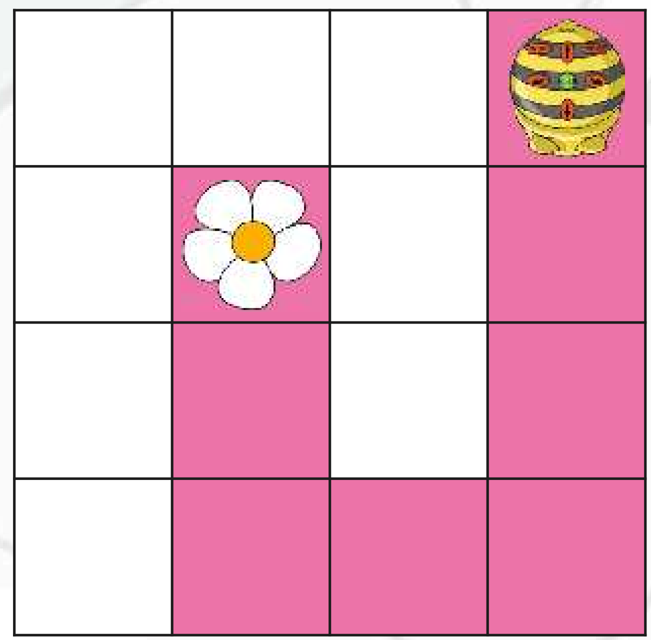


What Went Wrong?

What did you draw?

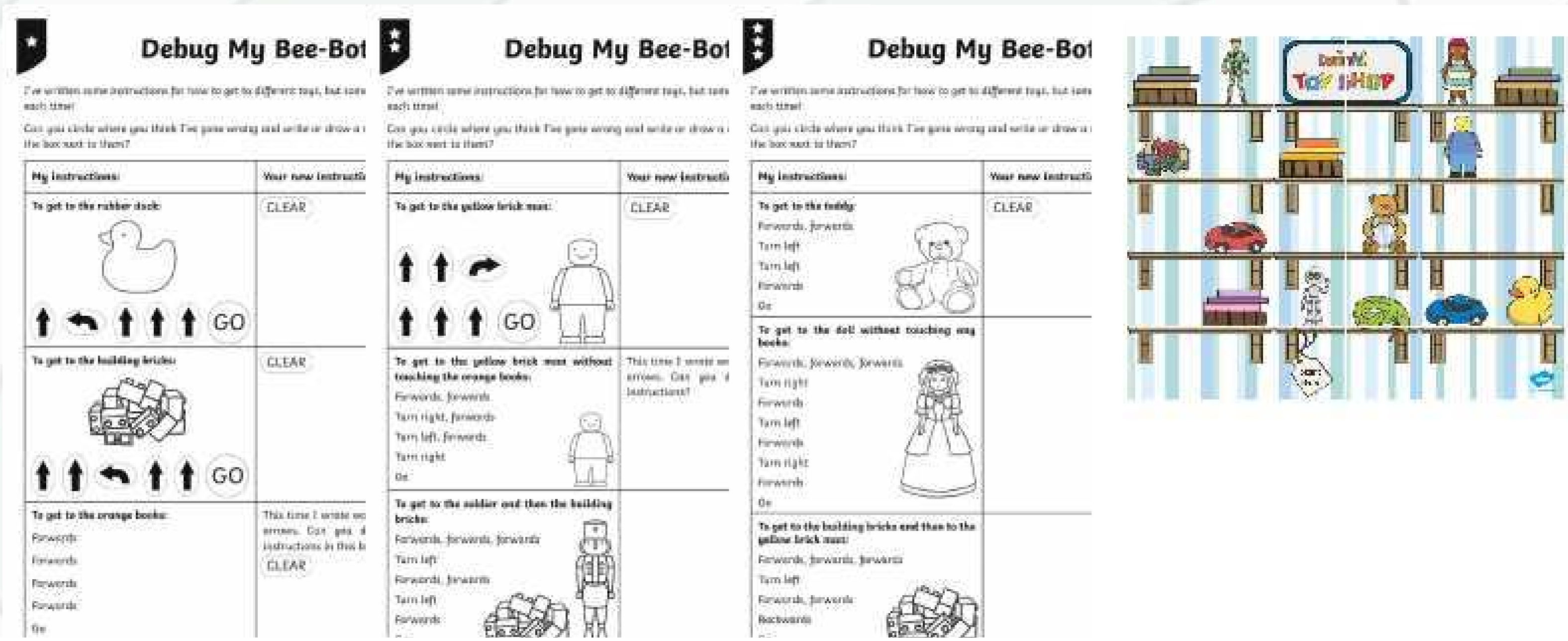
Did you remember to press CLEAR first?
That will clear your Bee-Bot's memory.





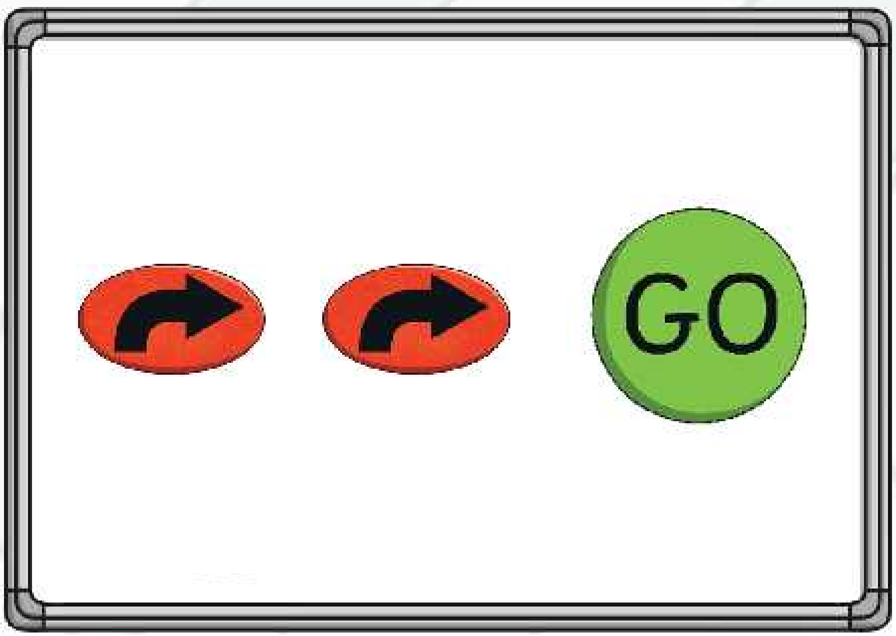
Debugging

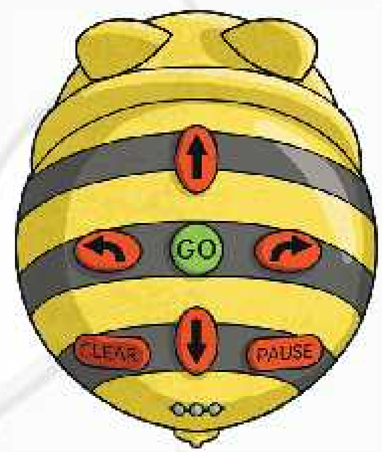
Look at the instructions for each Bee-Bot on your sheet. Can you work out where I have made a mistake and fix it?



Half Turns

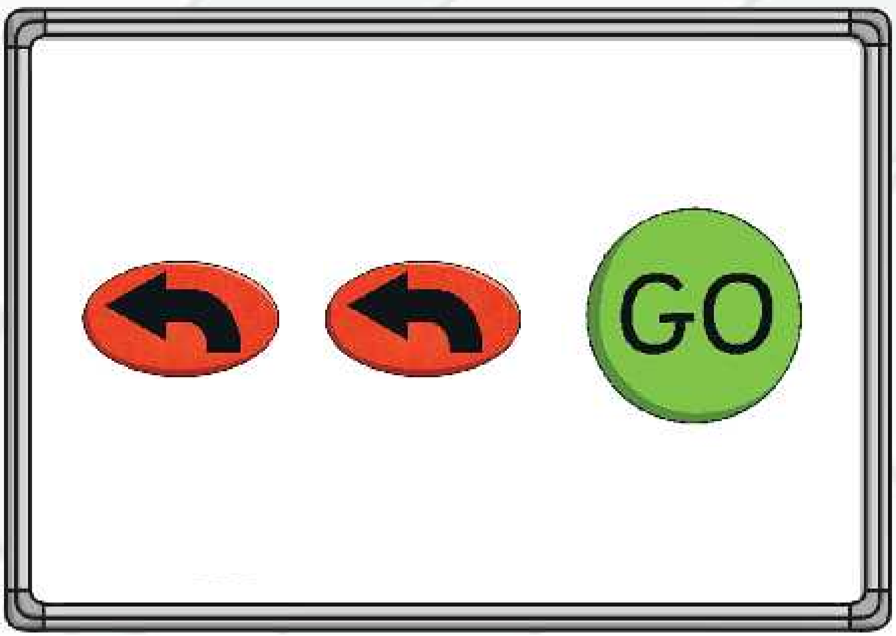
What if I need my Bee-Bot to turn around and go back again?
Some people in here have been practising that today.





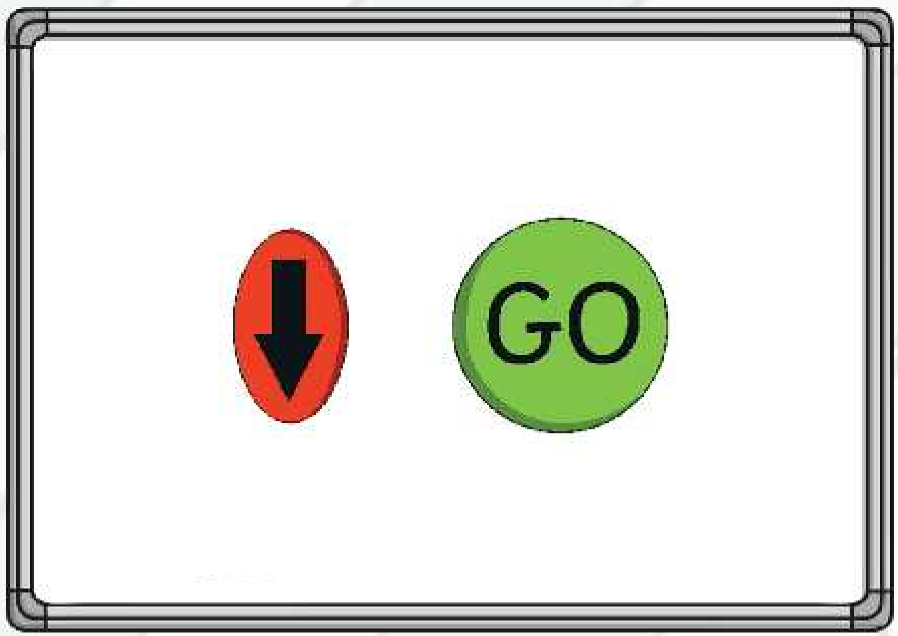
Half Turns

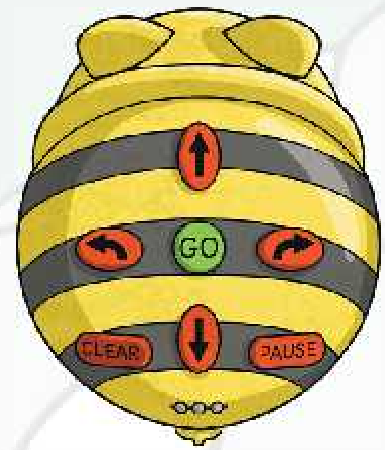
What if I need my Bee-Bot to turn around and go back again?
Some people in here have been practising that today.



Half Turns

What if I need my Bee-Bot to turn around and go back again?



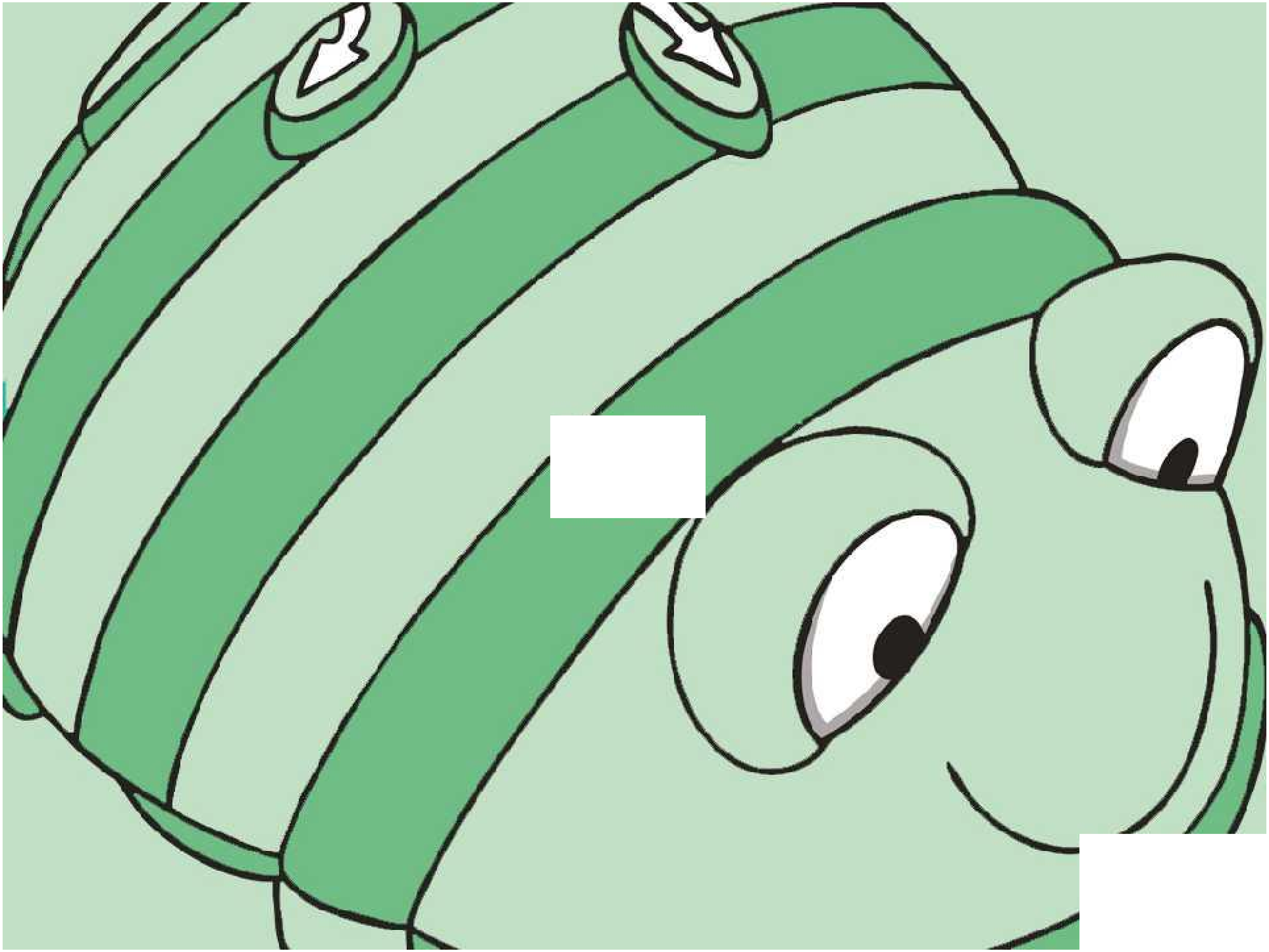


Aim

- I can debug a Bee-Bot.

Success Criteria

- I can check my work for mistakes to debug a program.
- I can start my programming sequence again if I need to.



Programming Toys: Building Bricks

Aim:	Success Criteria:	Resources:
Understand that programs execute by following precise and unambiguous instructions. Create and debug simple programs. Use technology purposefully to create digital content. Children will work within the context of following picture instructions for building shapes. I can create instructions using pictures.	I know what an algorithm is. I can create step-by-step instructions using pictures. **Key/New Words:** Algorithm, photograph, instruction, order.	**Lesson Pack** Building bricks - 5 per pair Tablets with cameras - 1 per pair **Preparation:**

Prior Learning: It will be helpful if children know how to take photos on your chosen device.

Learning Sequence

	What Is an Algorithm? Use the **Lesson Presentation** to define an algorithm. Ask the children to think about an algorithm for getting dressed in the morning, e.g. "What if you put on your coat before your jumper?" Click the link on the **Lesson Presentation** to show the children a video from BBC Bitesize. Highlight that when you write an algorithm, the order of the instructions is very important.	
Whole Class	**What Is an Algorithm?** Using the **Lesson Presentation**, invite children to help you give picture instructions to a robot in order to brush his teeth. Point out that if you try to wet your brush before you turn the tap on, the robot will fail.	
	Build and Snap: Build a simple model using 5 building blocks and show the children how to take pictures of each stage, one block at a time. Show them examples of clear photographs using the **Lesson Presentation**. Can children identify what makes a good picture instruction step?	
	Get Building! The children should build a simple model using 5 blocks (you may choose to add or remove blocks to suit your children), taking a single photograph at each stage. Can children take a clear photograph for someone else to follow? They should then pull apart all their bricks and give their tablet, with the photos open, to another pair. Children must then see if they can follow the pictorial instructions given to them to recreate the model. Are the children able to follow instructions in order?	
Whole Class	**Did You Do a Good Job?** Children show their model to the pair who took the pictures. Evaluate their success using the **Lesson Presentation**. Have the children built the final model correctly?	

Taskit

Whisperit: Chinese whisper building! Can children create a set of picture instructions to follow as a group, where one person completes a single step, then passes it onto the next person? Will the model still look like the picture at the end?

Explainit: Print out some picture instructions from the lesson. Children write an explanation of why photos can be better than a written instruction.

Avery Template: Name Badge Label, 8 per sheet | **Compatible Products**: 15395, 25395, 42395, 45395, 48395, 5395, 8395, 88395, 85395.

Programming Toys | Building Bricks

I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

Programming Toys | Building Bricks

I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

Programming Toys | Building Bricks

I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

Programming Toys | Building Bricks

I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

Programming Toys | Building Bricks

I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

Programming Toys | Building Bricks

I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

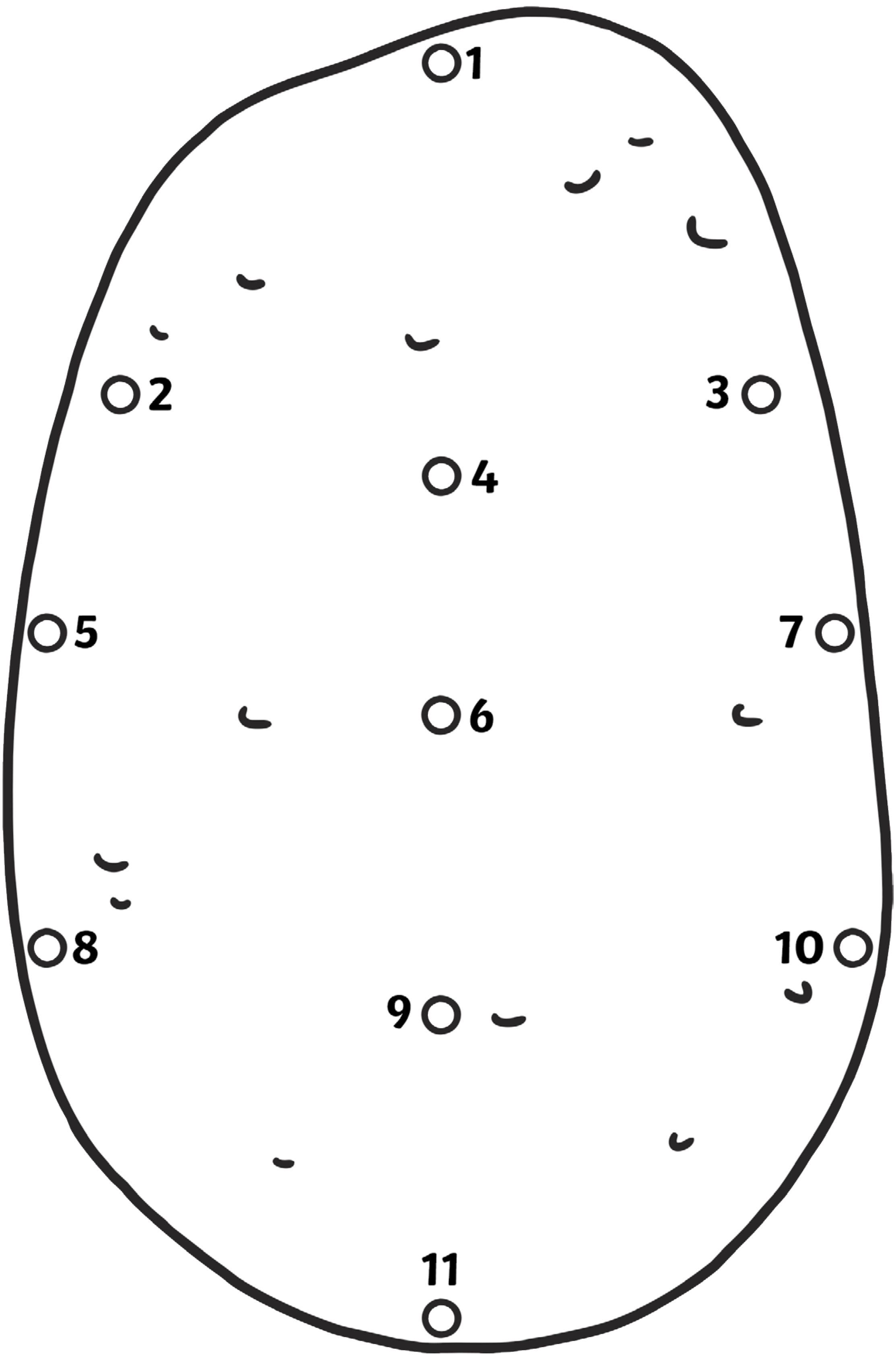
Programming Toys | Building Bricks

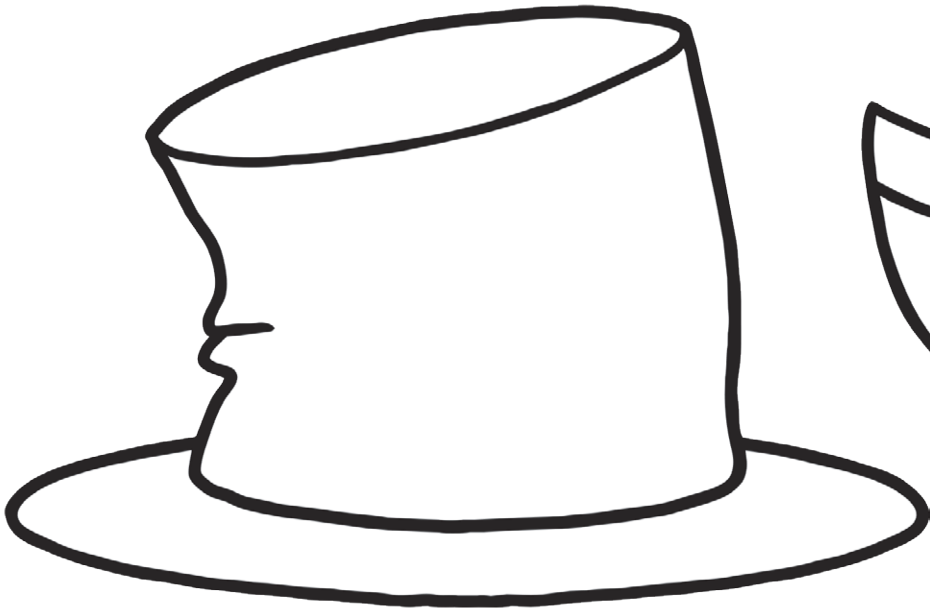
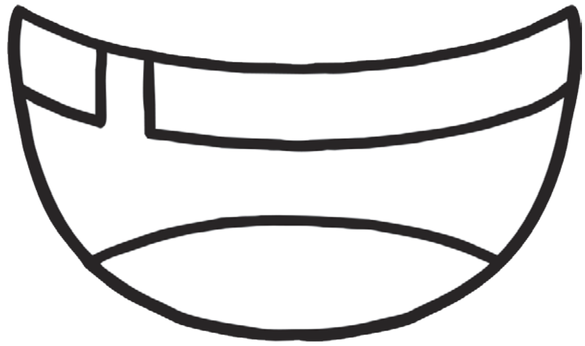
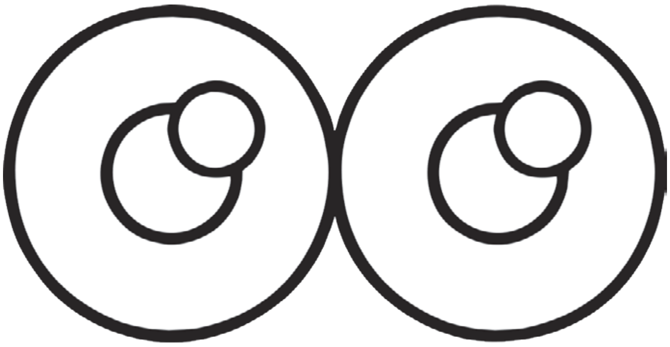
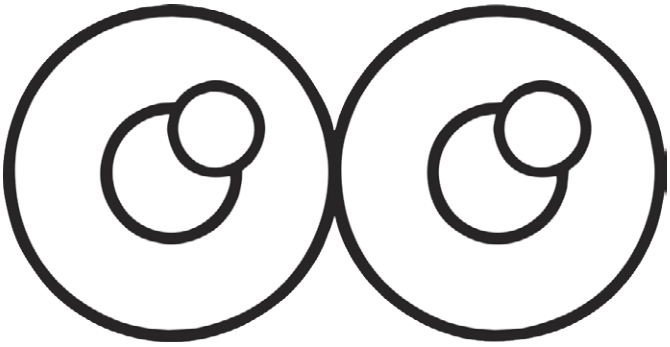
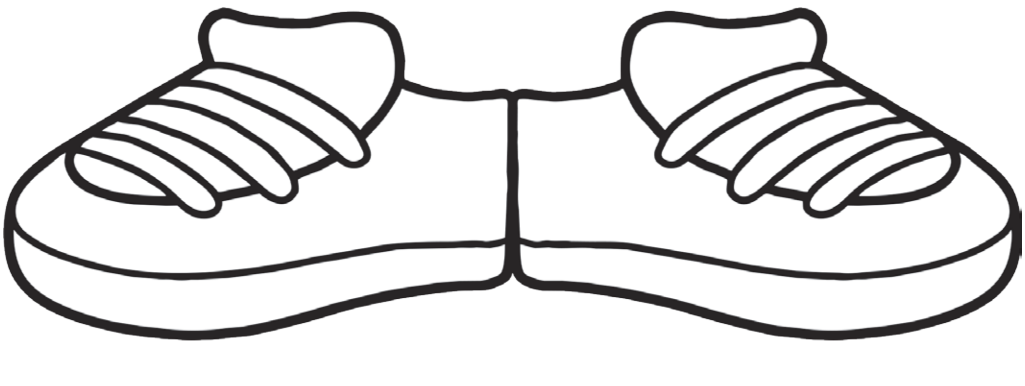
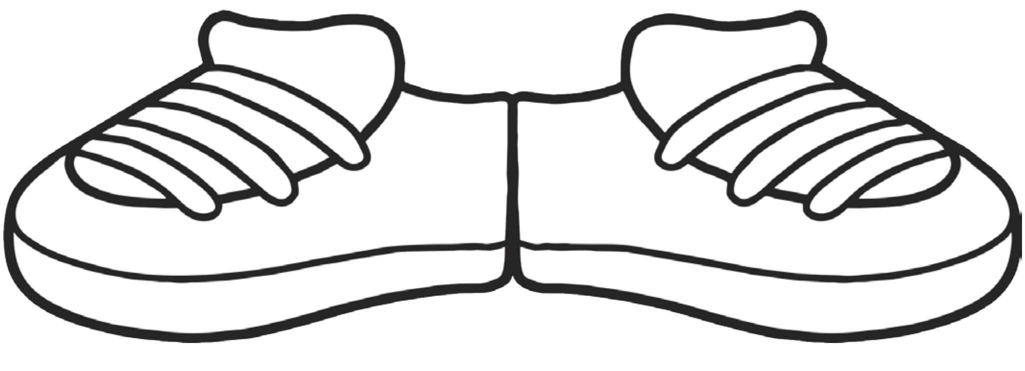
I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

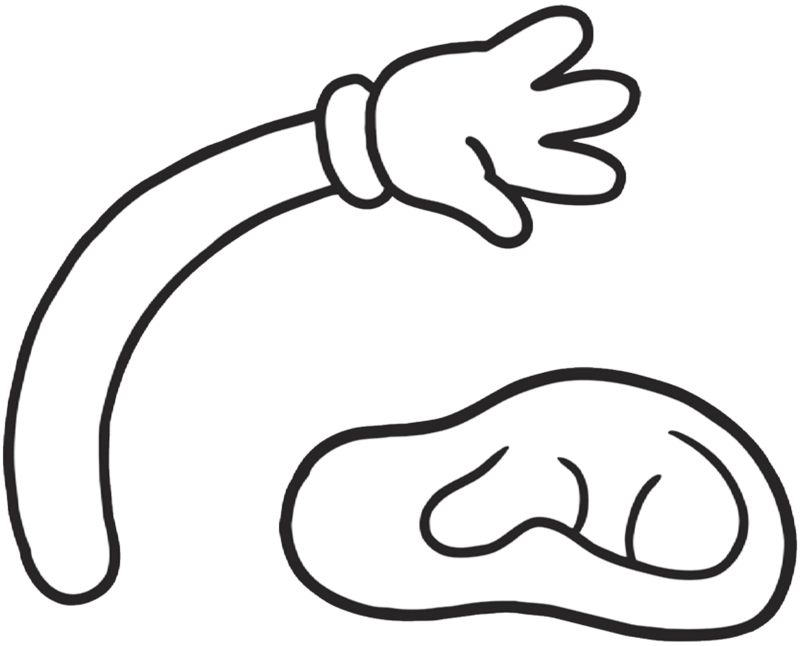
Programming Toys | Building Bricks

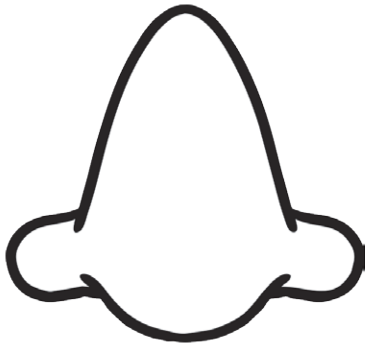
I can create instructions using pictures.		
I know what an algorithm is.		
I can create step-by-step instructions using pictures.		

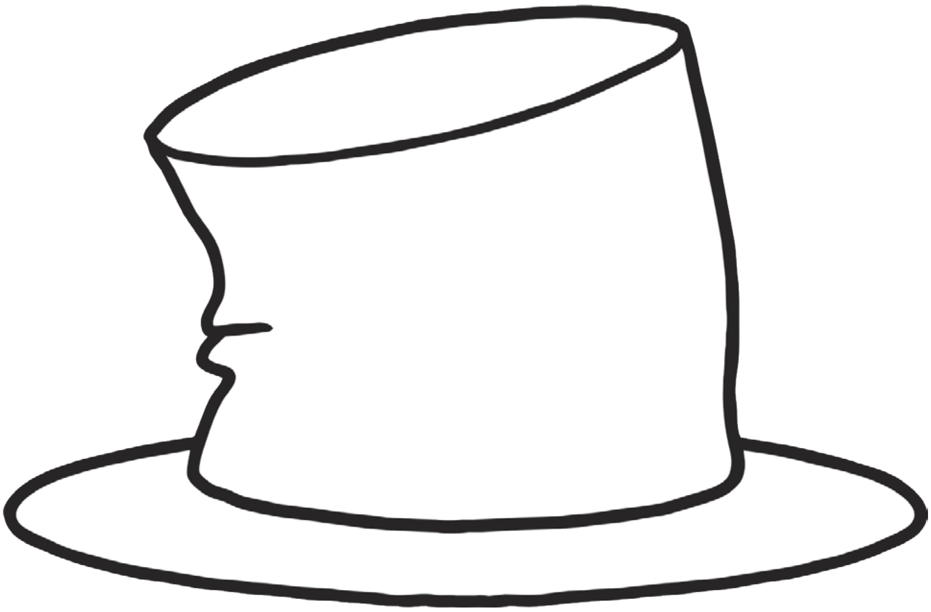
Build a Potato Man!

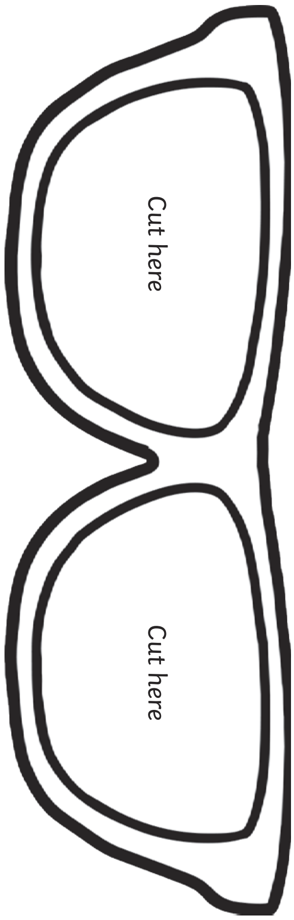


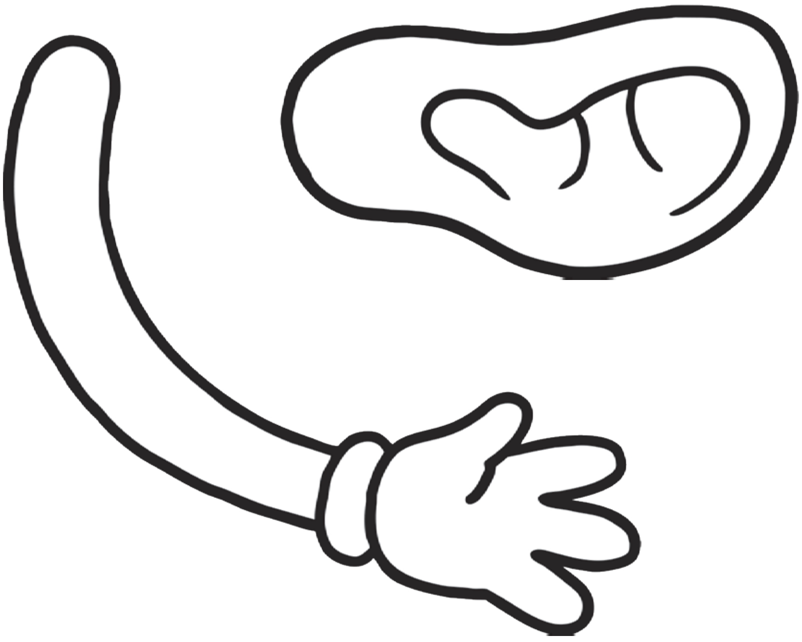







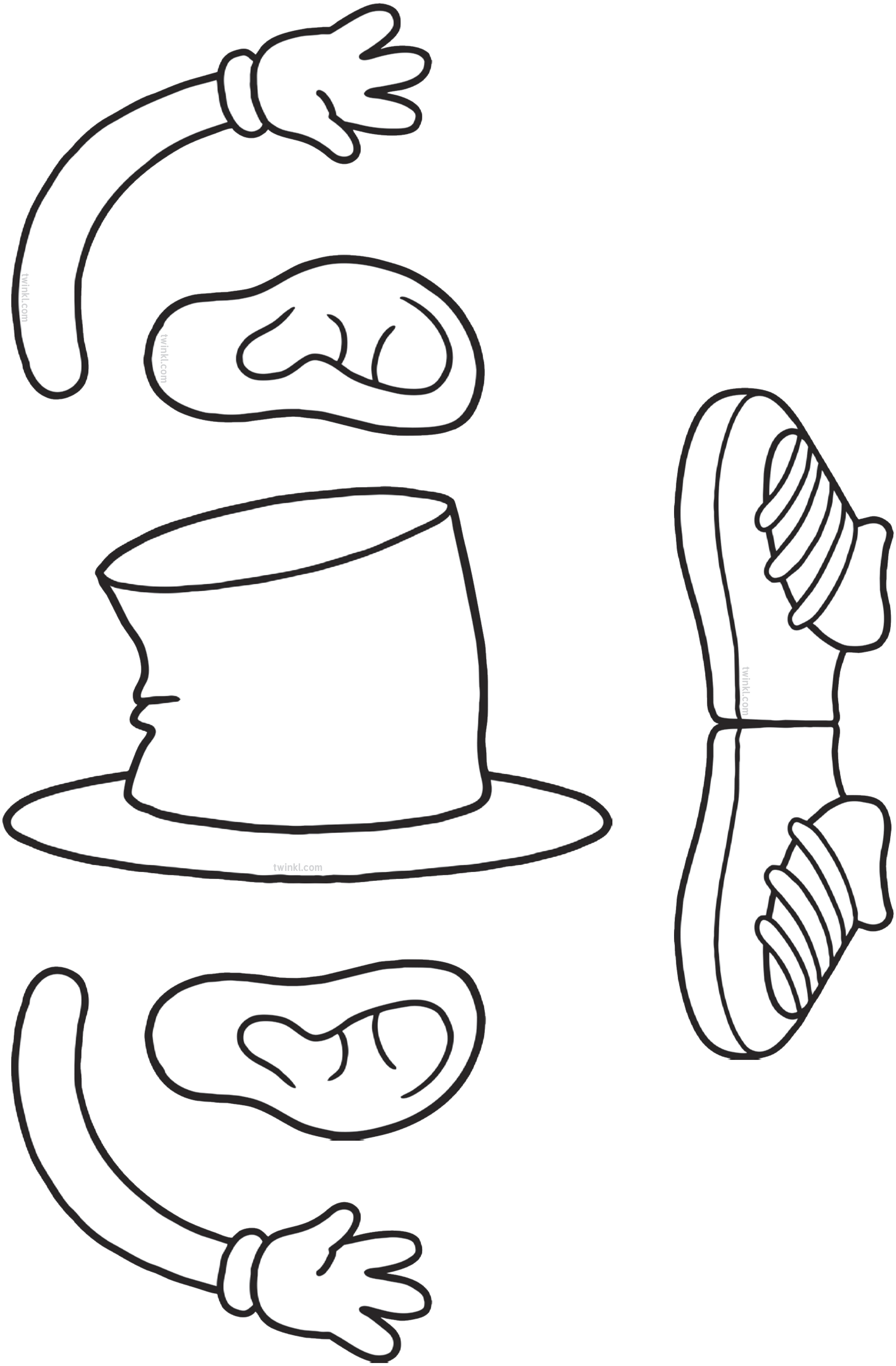




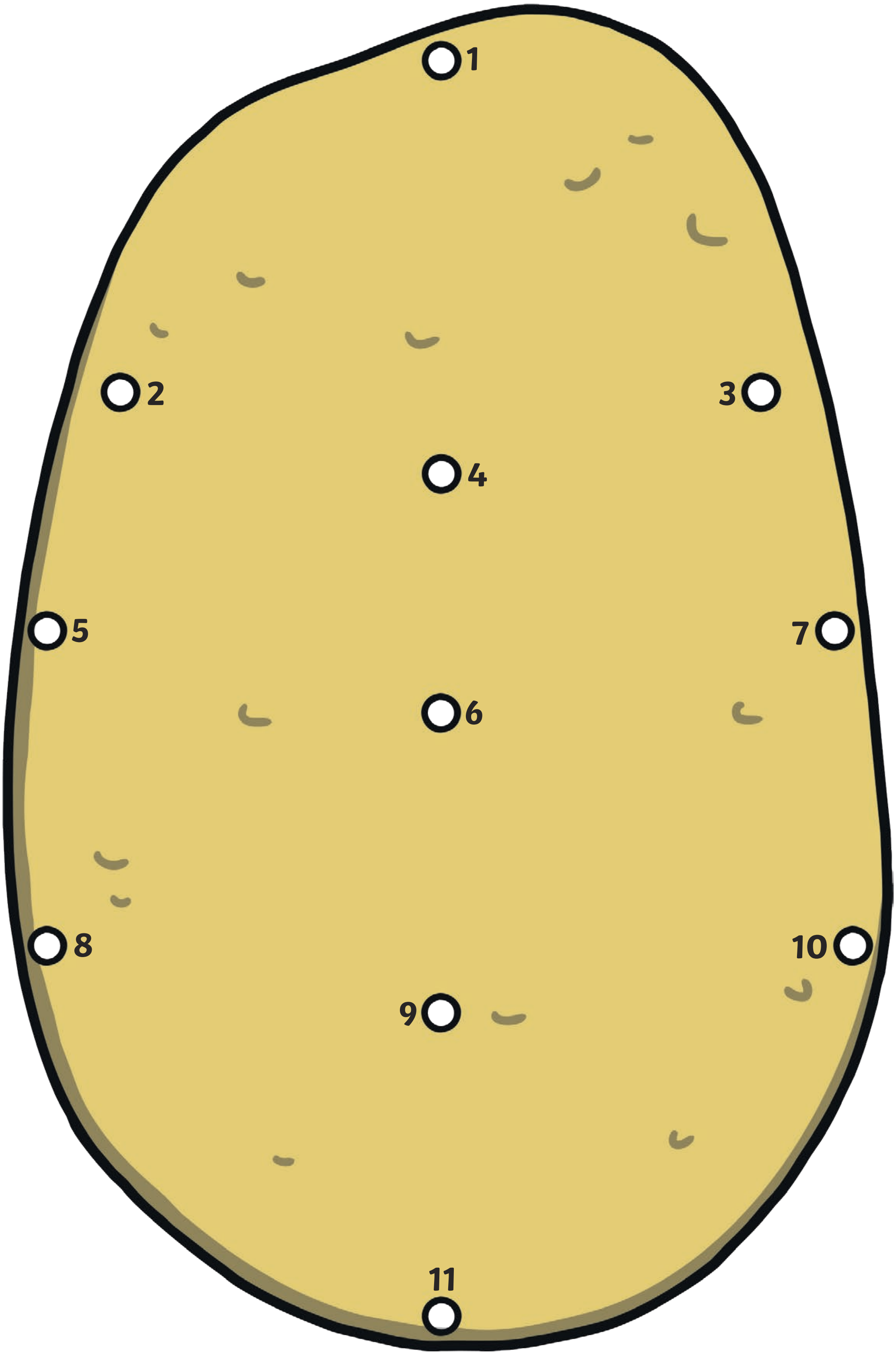


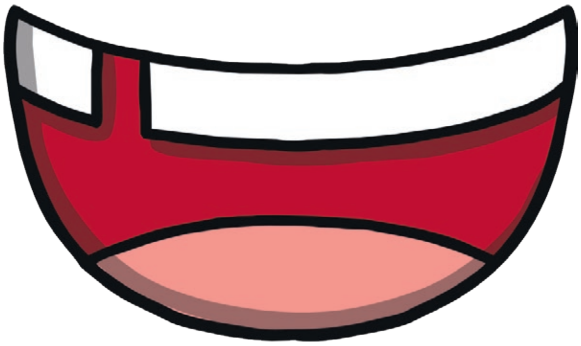

Cut here
Cut here

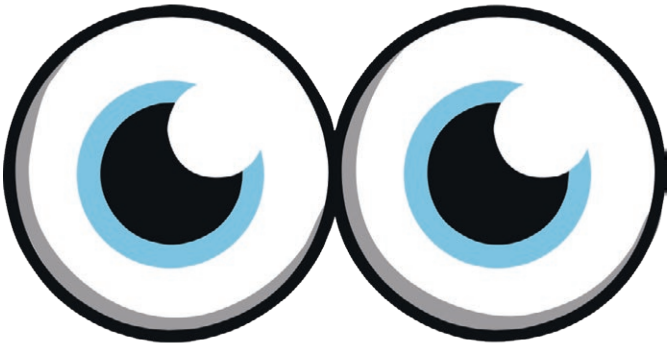


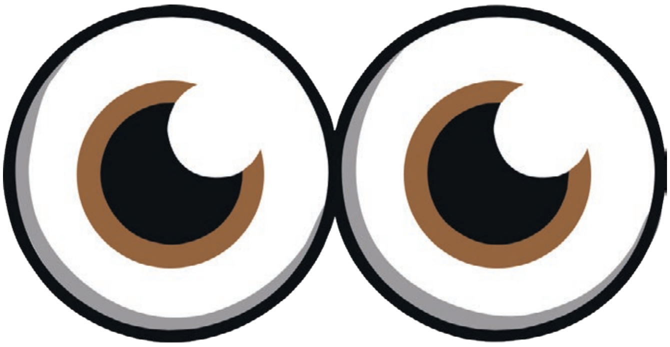


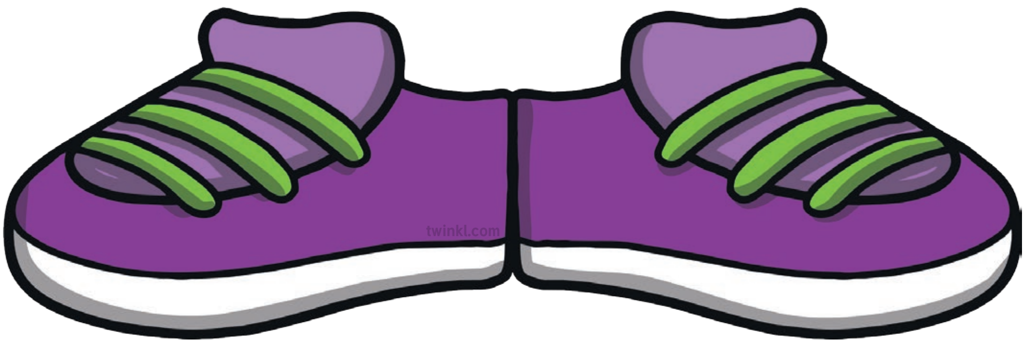
Build a Potato Man!

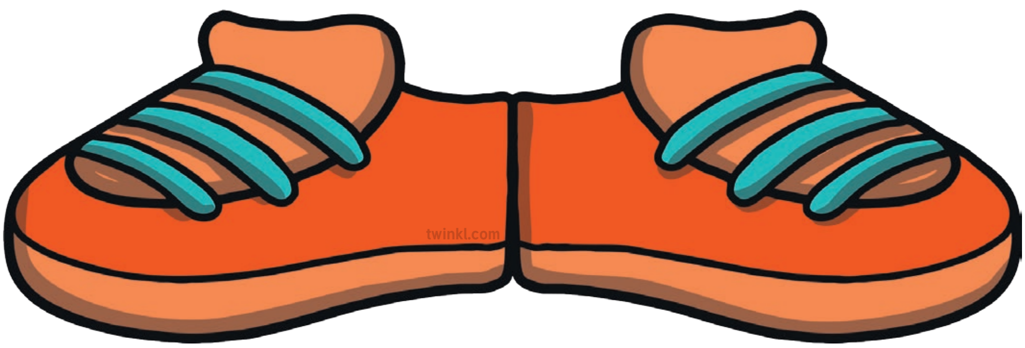


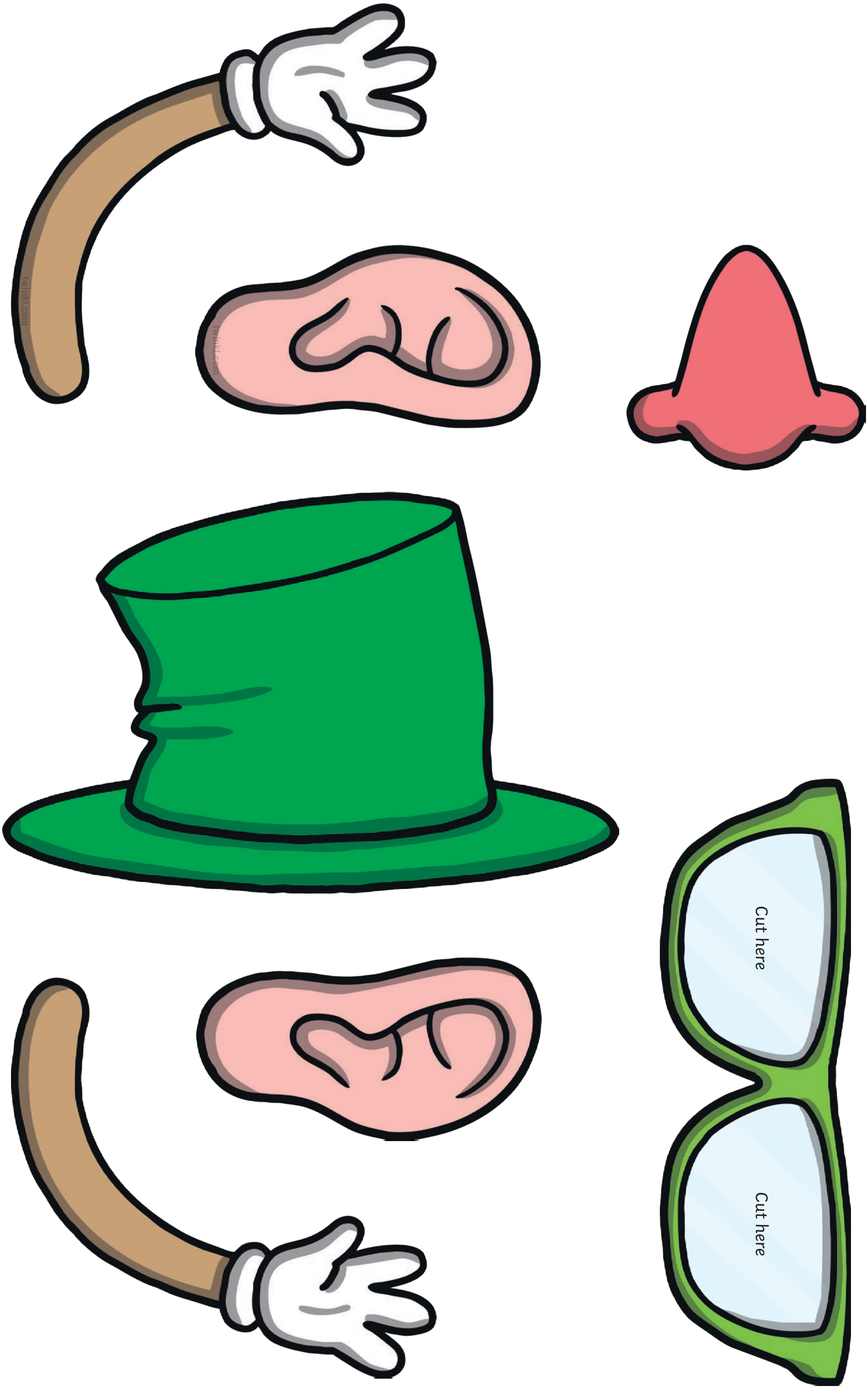


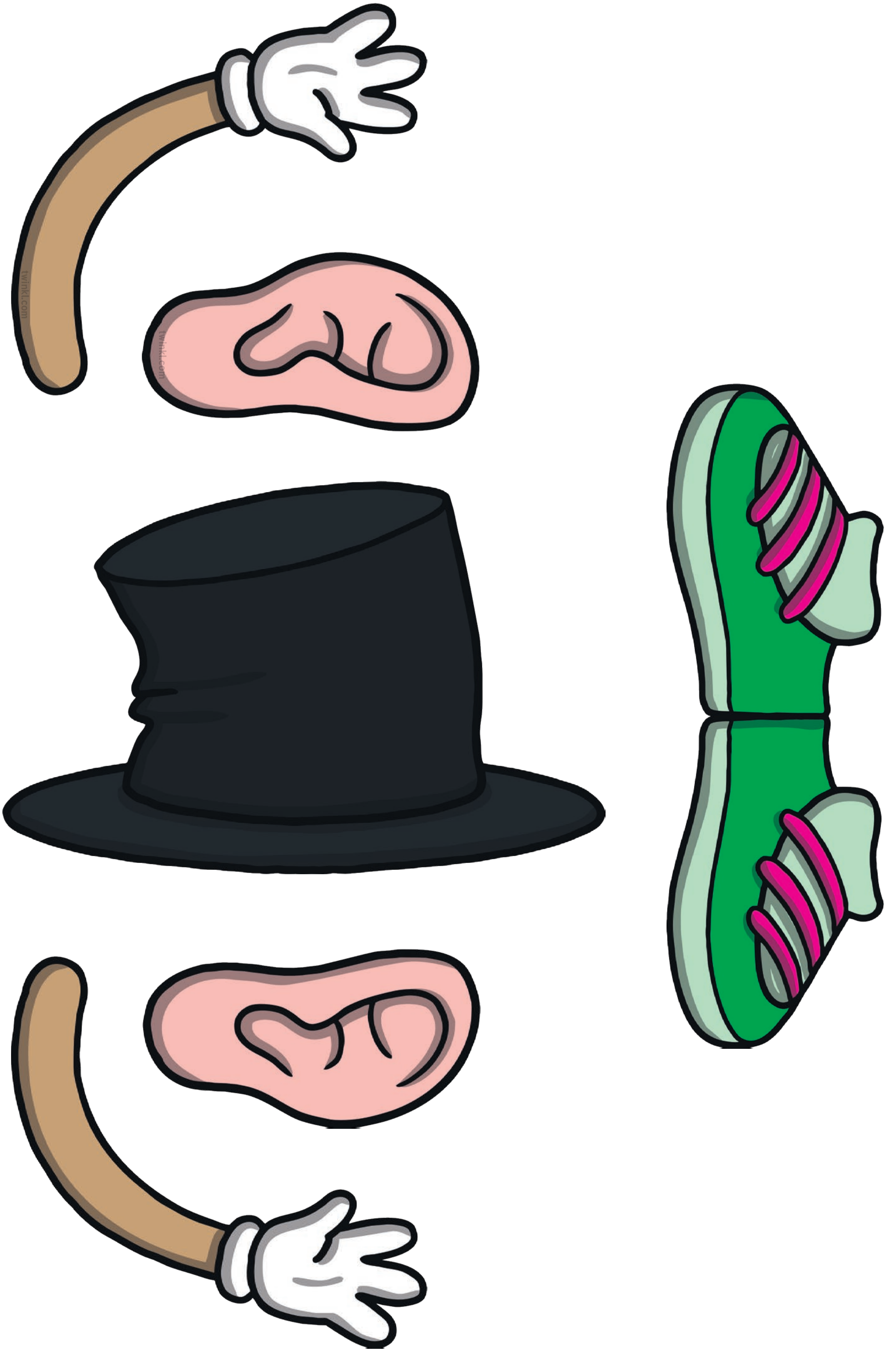








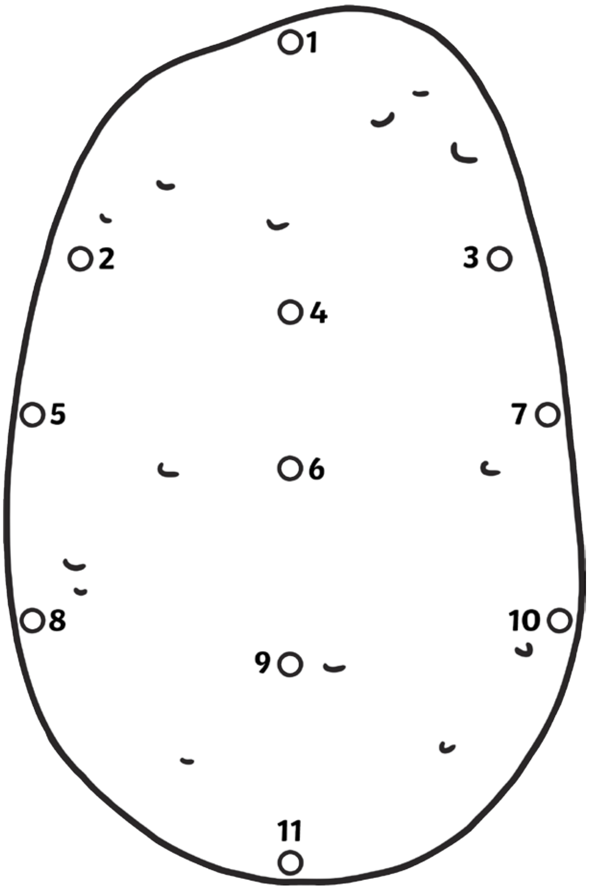

Cut here
Cut here



Potato Man Instructions

Use your Potato Man Picture Card to write instructions for your partner.

Don’t forget to use the right numbered hole!

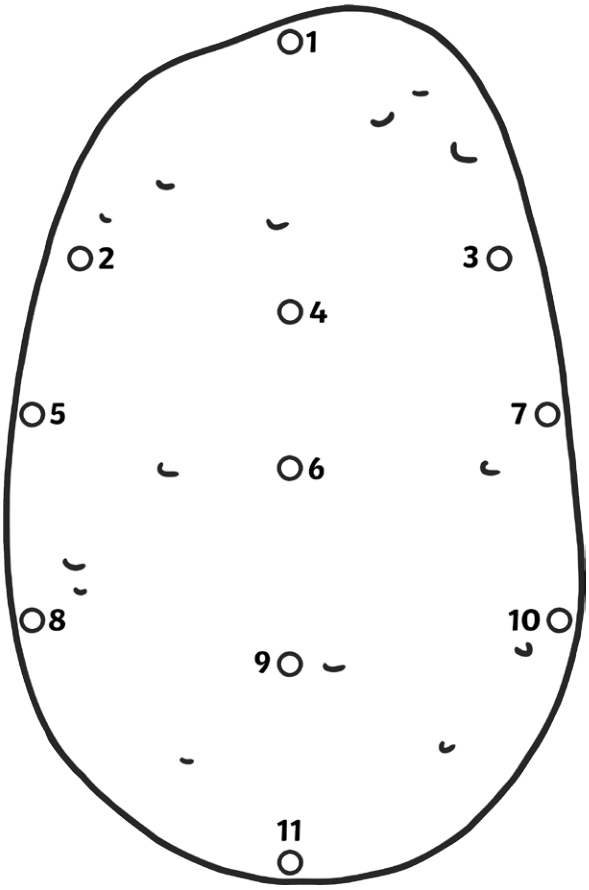


Instructions

Potato Man Instructions

Use your Potato Man Picture Card to write instructions for your partner.

Don't forget to use the right numbered hole and the right coloured parts!



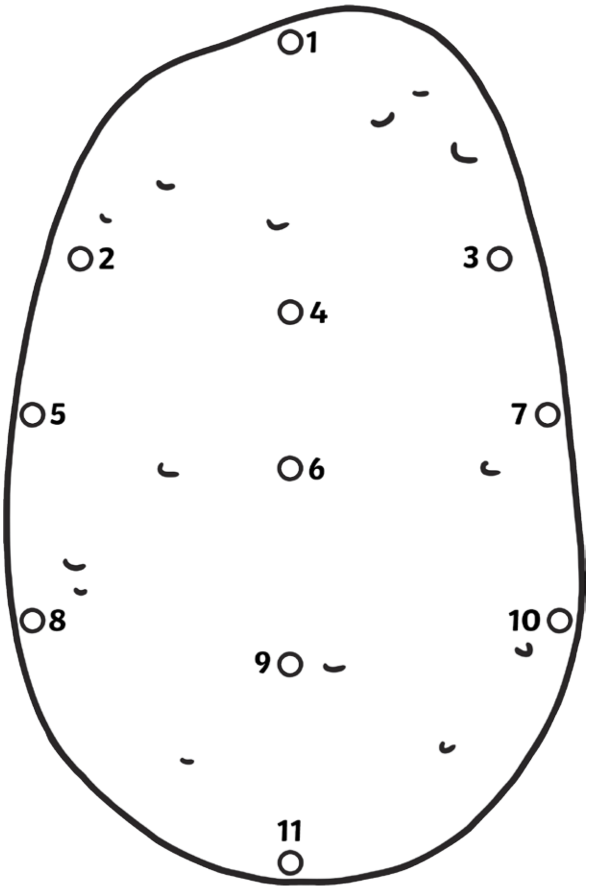
Instructions

Potato Man Instructions

Use your Potato Man Picture Card to write instructions for your partner.

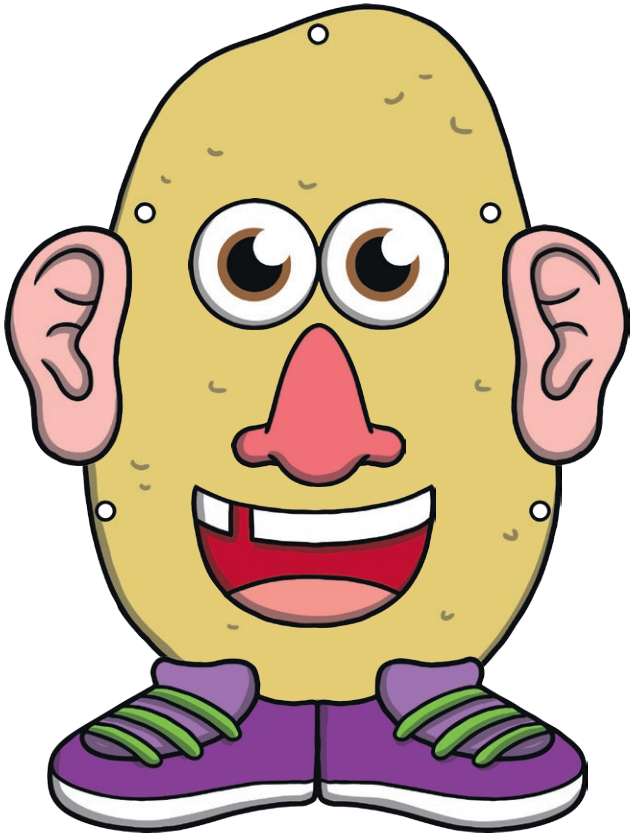
Don't forget to:

- use the right numbered hole;
- add the correct colours;
- put your instructions in the right order.

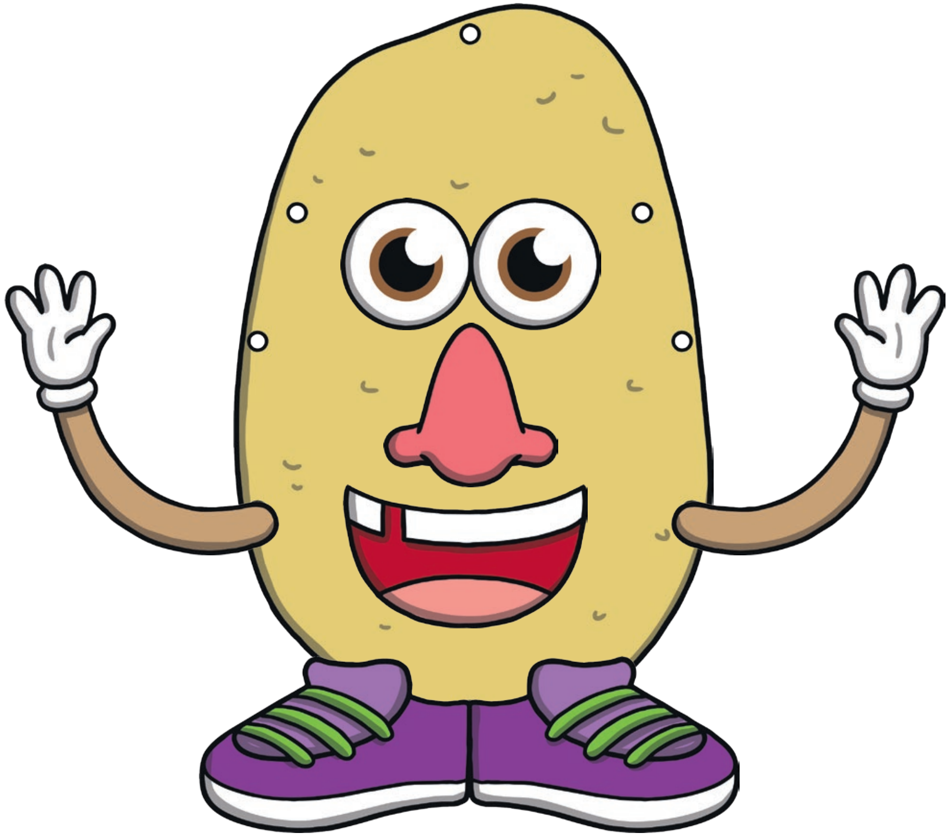


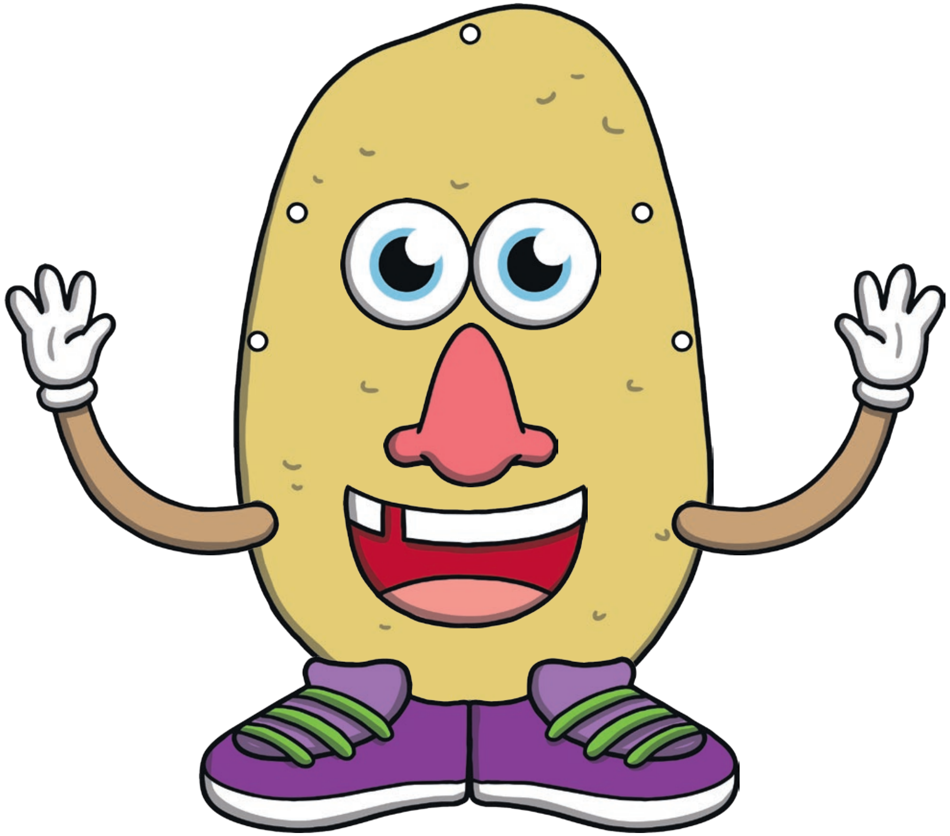
Instructions

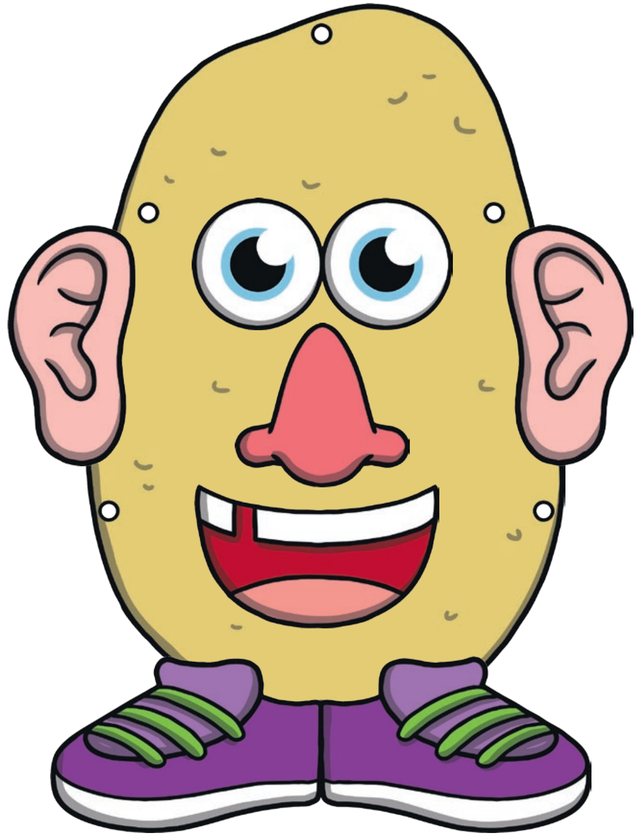
Potato Man Picture Cards



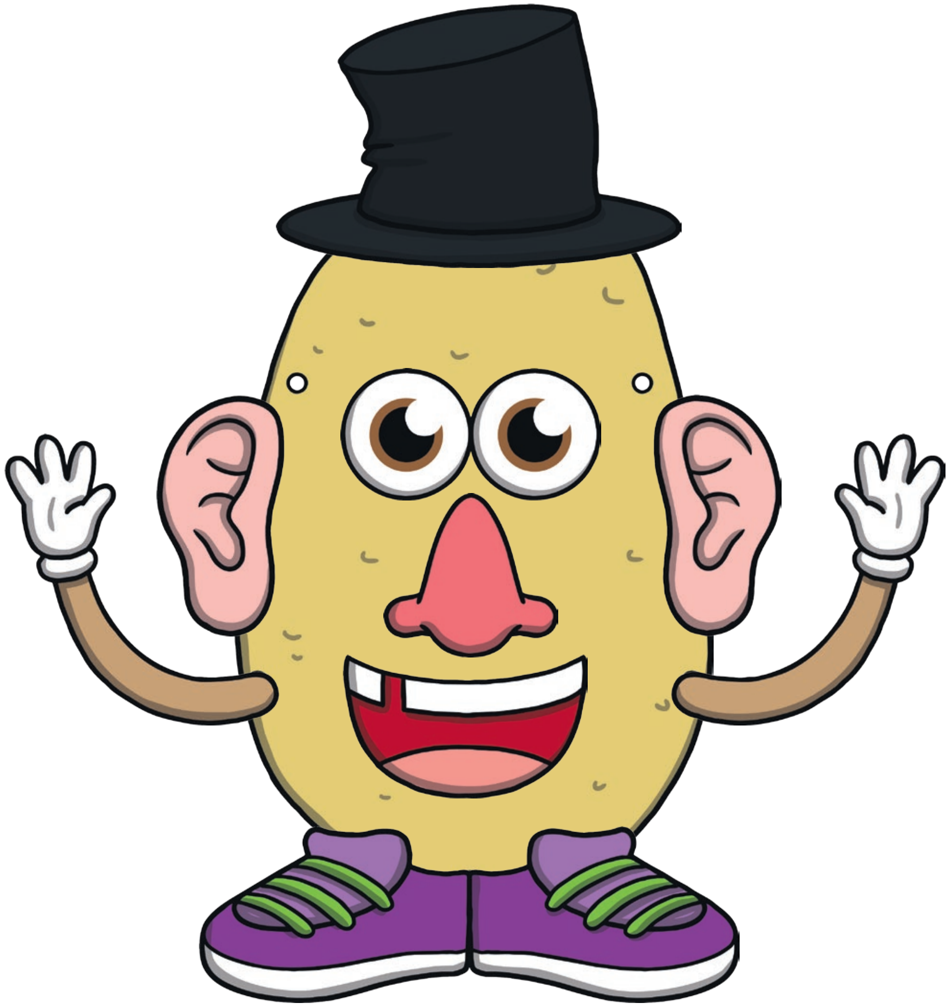
Potato Man Picture Cards



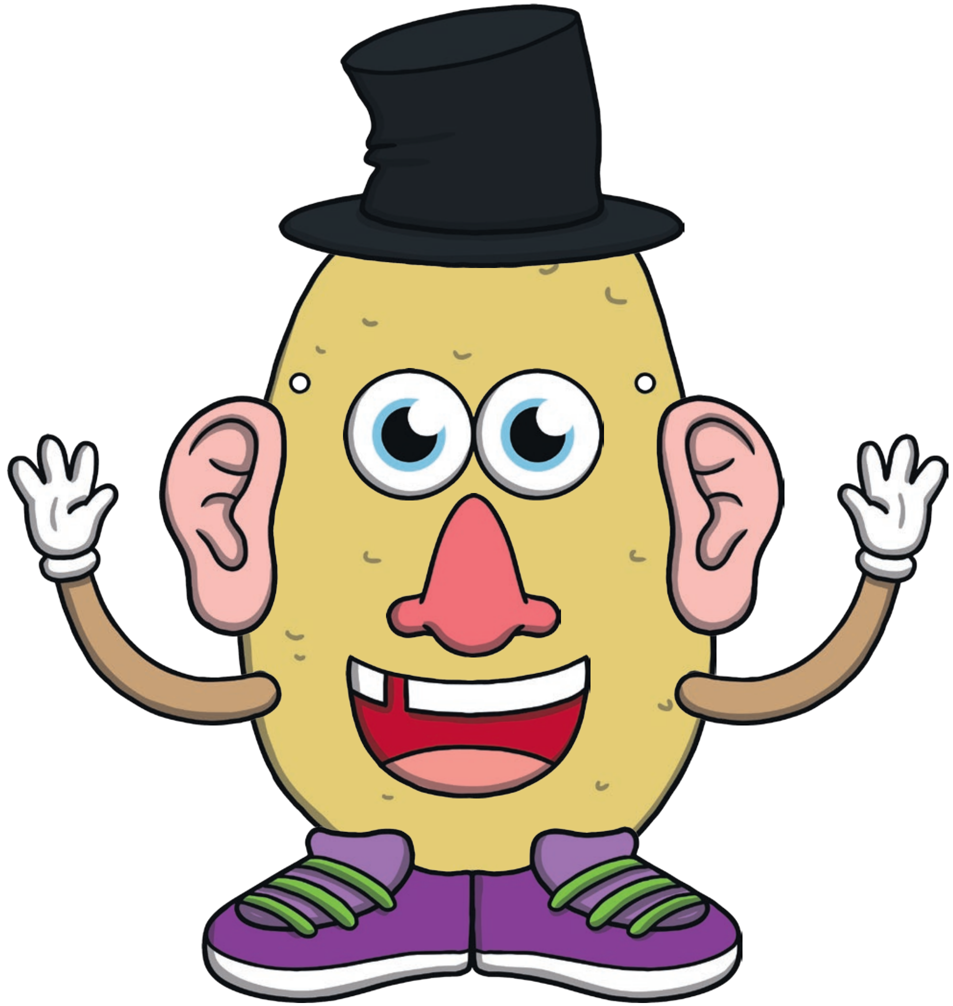




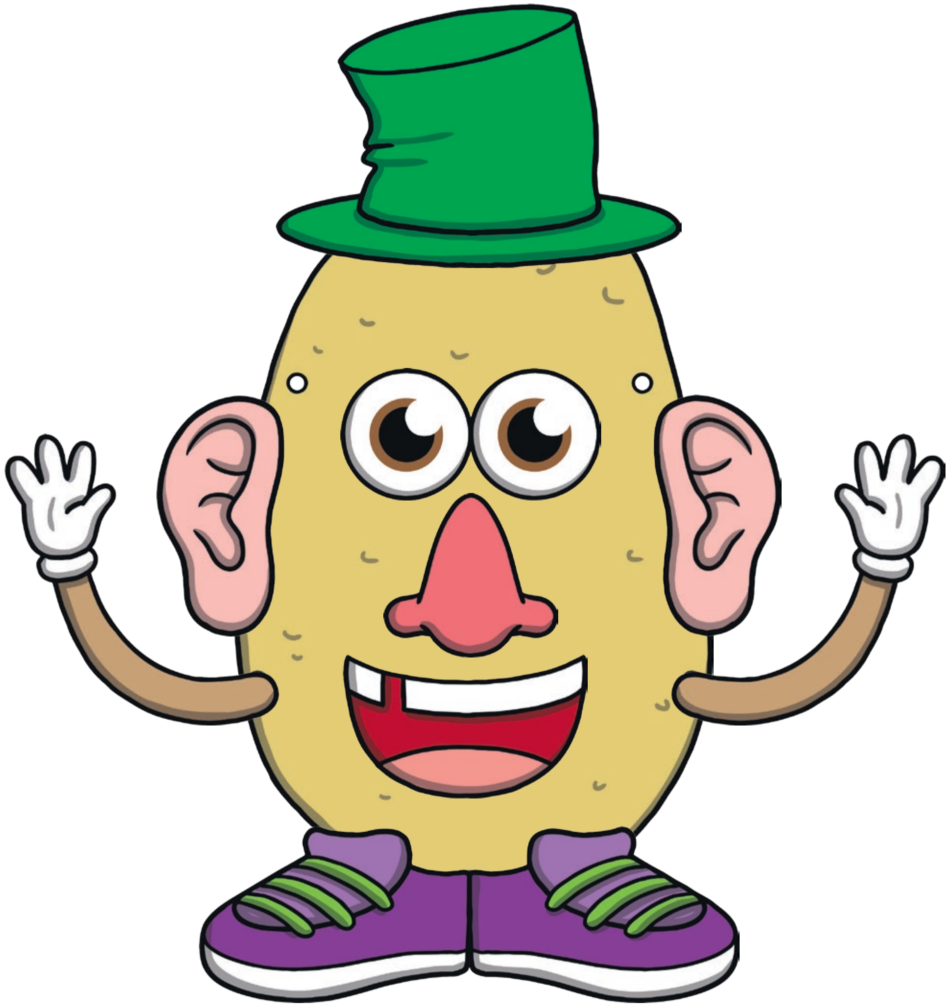
Potato Man Picture Cards



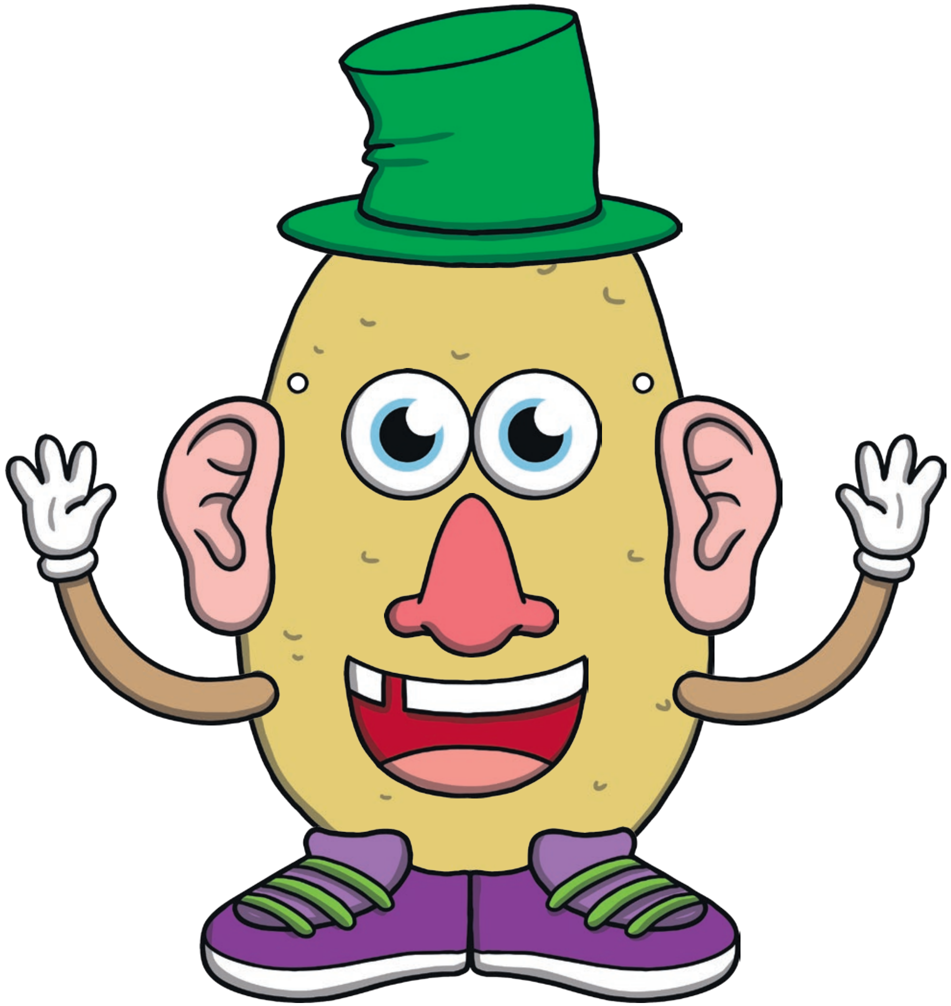
Potato Man Picture Cards

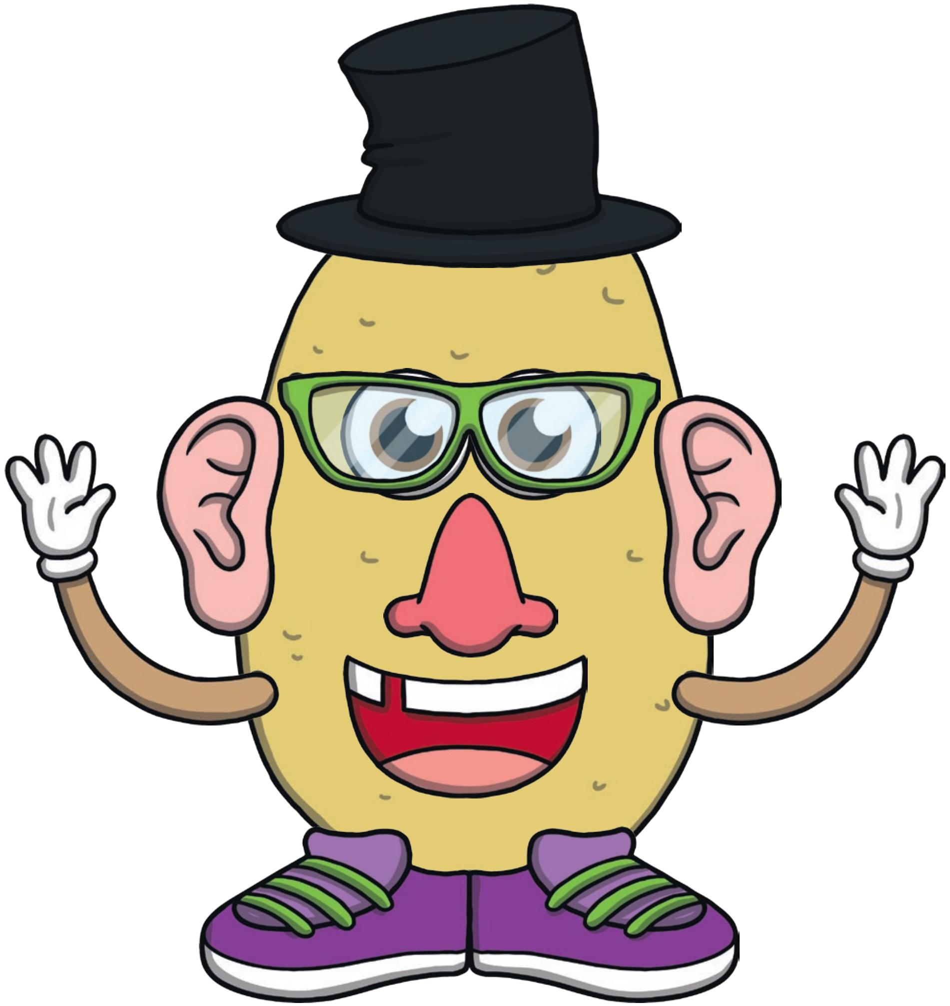


Potato Man Picture Cards

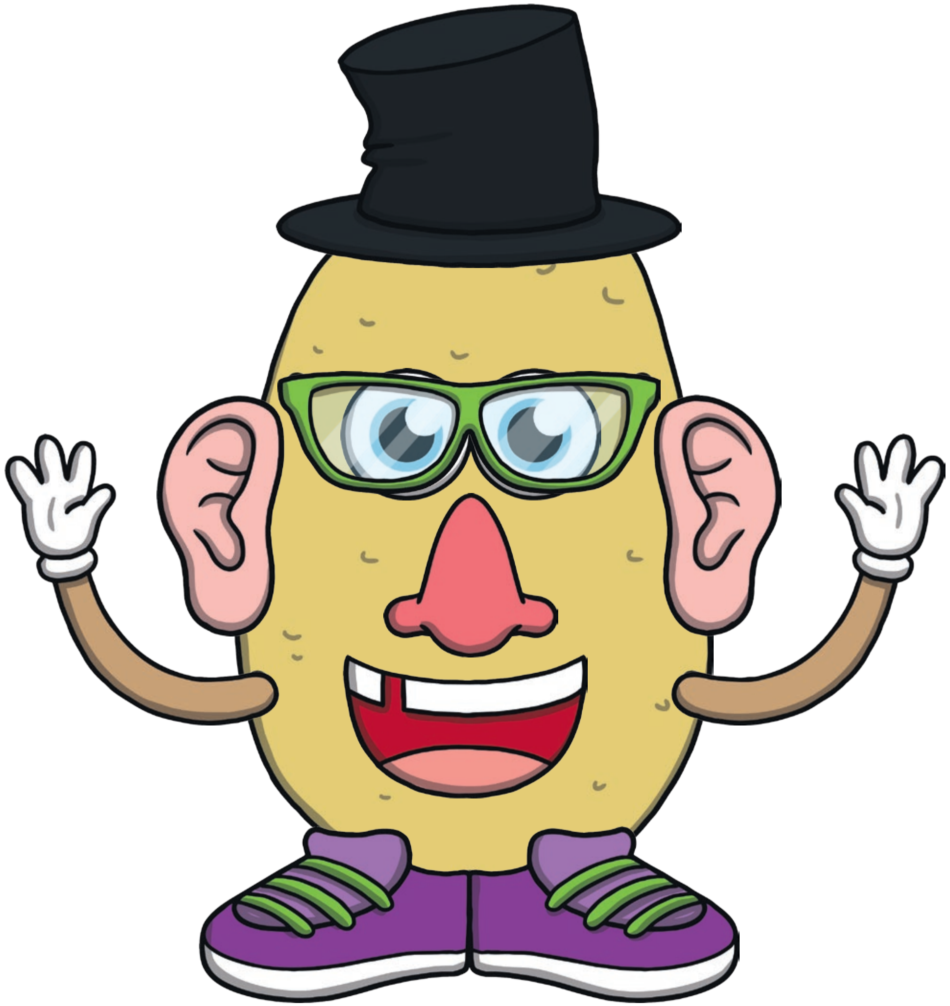


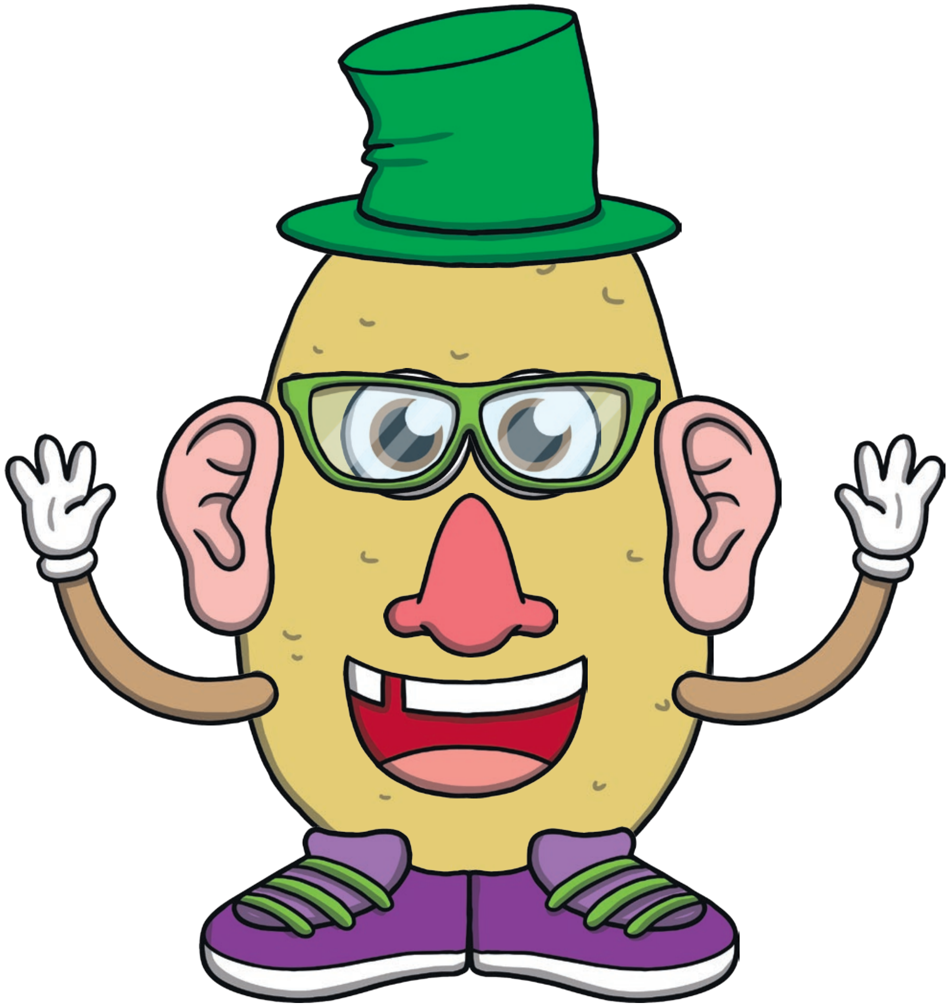
Potato Man Picture Cards



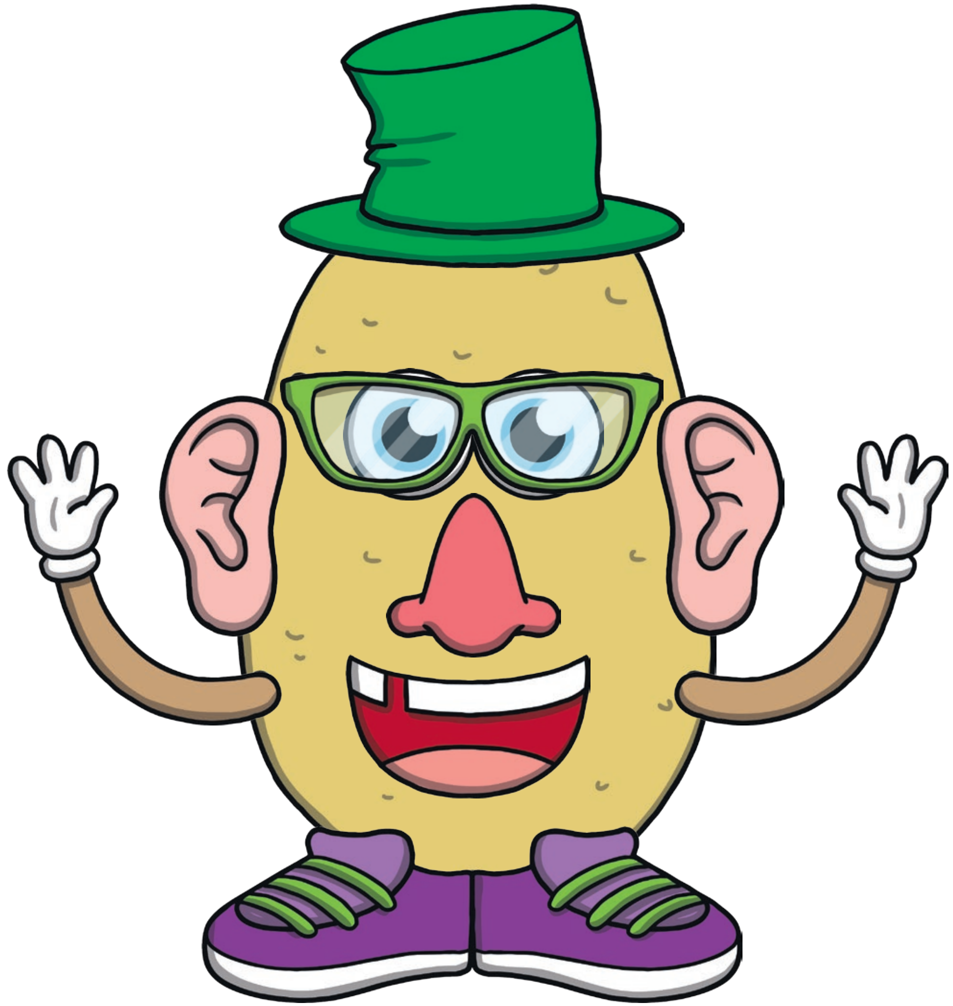


Potato Man Picture Cards





Potato Man Picture Cards



Programming Toys: Potato Man Algorithms

Aim:	**Success Criteria:**	**Resources:**
Understand how [algorithms] are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions in the context of writing detailed instructions to build a face on a potato man toy. I can say why it is important to be precise when writing an algorithm.	I can write and follow detailed instructions. I can see how a product changes when I change the instructions.	**Lesson Pack** 10 building bricks Glue Scissors Flipchart or large whiteboard
	Key/New Words: Algorithm, instruction, detail.	**Preparation:** **Build a Potato Man Activity Sheets** - 1 per child **Differentiated Potato Man Picture Cards** - 1 per child **Differentiated Potato Man Instructions Activity Sheet** - 1 per child

Prior Learning: Children will have been introduced to ordering instructions in lesson 1.

Learning Sequence

	Build an Animal: Invite two children up to the front and give them a small set of building bricks each. Using the **Lesson Presentation**, give the children 1 minute to 'build an animal'. When finished, compare similarities and differences using the **Lesson Presentation**. Remind the children that instructions for completing a task need to be clear, detailed and in the right order. Can children identify how the teacher could have added more detail to the instruction?	
Whole Class	**Potato Man Building:** Using the **Lesson Presentation**, introduce the children to the parts of a potato man toy. Use the **Lesson Presentation** to choose an instruction and click it, allowing the lack of detail to mean that the product will turn out wrong. Use the following slides to choose better instructions so that the potato man turns out correctly. Can children offer detailed instructions? Can the children say which parts must be added in a certain order (eyes before glasses)?	
	Children use the **Differentiated Potato Man Picture Cards** and **Potato Man Instructions Activity Sheet**, writing the instructions for how to build the pictured potato man. ★ Children need to write detailed instructions in each box, including which numbered holes to use. ★★ Children need to write detailed instructions, including which numbered holes to use, and are given more options for pieces. ★★★ Children need to write detailed instructions, including which numbered holes to use, and are given more options for pieces. They will need to add the eyes before the glasses.	
	Did It Work? Once the children have written their instructions, give them to a partner and see if their partner can build the potato man using the **Build a Potato Man Activity Sheet**. Evaluate success using the **Lesson Presentation**.	

Taskit

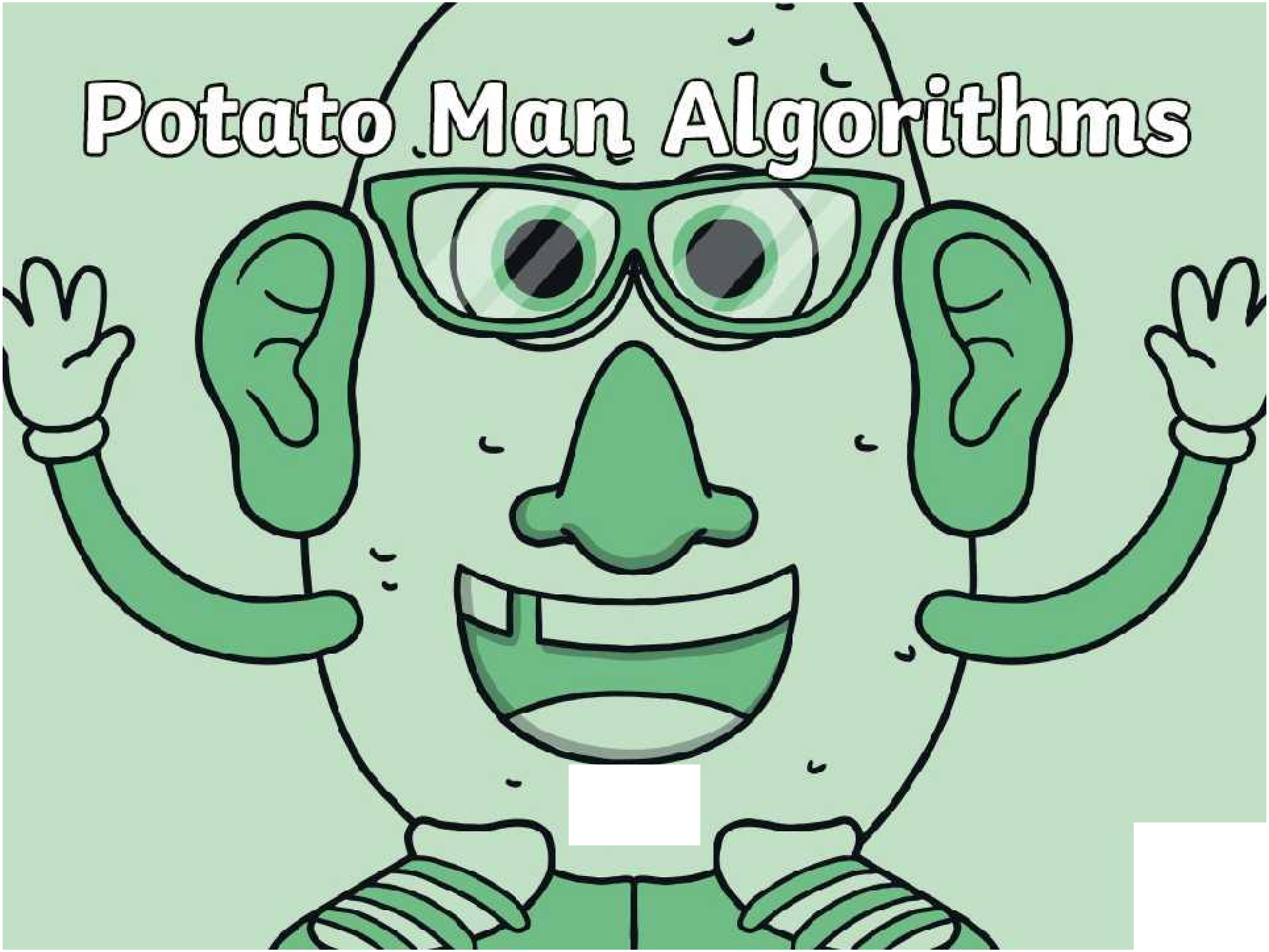
Exploreit: Can children write instructions for a friend to build a potato man online on a painting program? One person should write the instructions, then the other should try to create it without the first person watching.

Playit: Using either the **Build a Potato Man Activity Sheet** or a physical toy, children could explore the different ways of arranging the potato man. Ask whether the eyes always have to go that way around. Ask the children to think about all the different ways of building him; can a computer program work like this too?

Computing

Programming Toys

Potato Man Algorithms



Aim

- I can say why it is important to be precise when writing an algorithm.

Success Criteria

- I can write and follow detailed instructions.
- I can see how a product changes when I change the instructions.

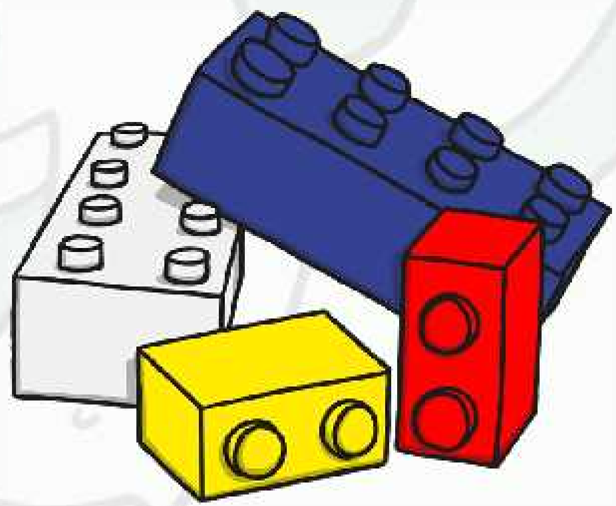
Build an Animal

Use these building bricks to make an animal.

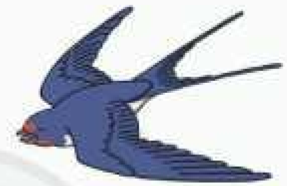
You have 1 minute.

Start timer!

STOP!



Build an Animal



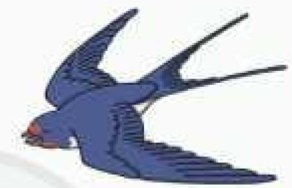
What did you make?

How are your models similar?

How are they different?

Why are they different?

Build an Animal



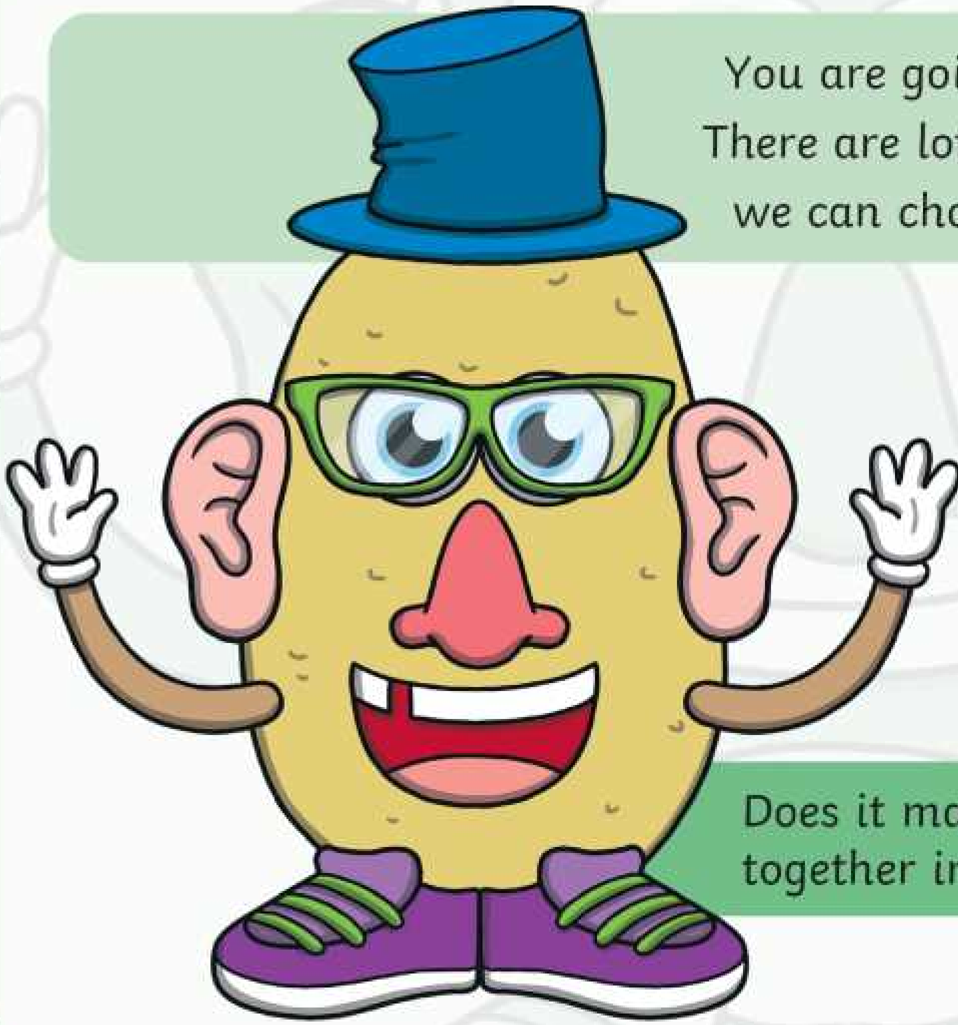
Without detailed instructions in the right order, we can't make exactly the right thing.

How could we make the instructions better, so that you both build the same animal?

Detailed instructions written in the right order are called an **algorithm.**

Potato Man Building

Whole Class

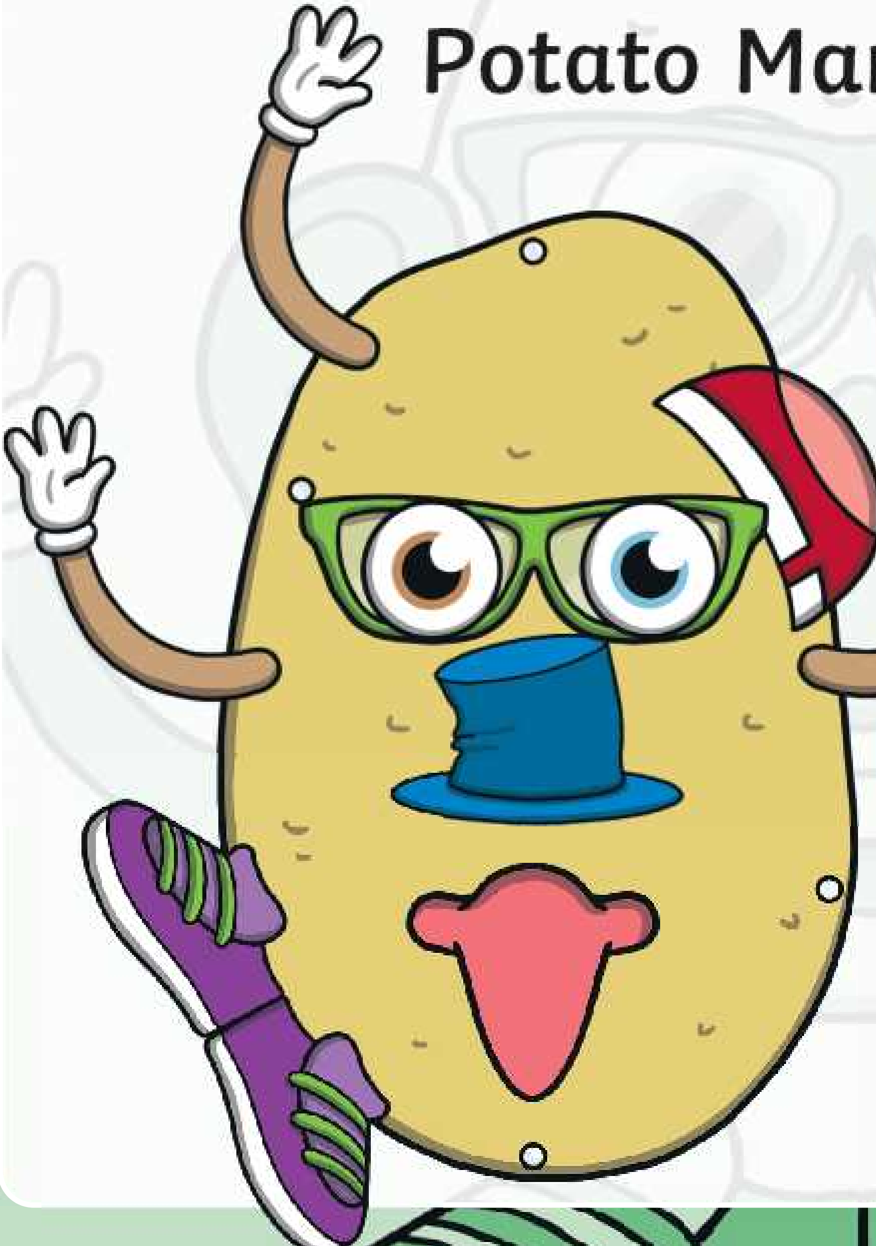


You are going to build a potato man toy. There are lots of different parts of him that we can choose to give him a funny face!

What parts can you see?

Does it matter what order we put him together in? Which bits can't come first?

Potato Man Building



Can you choose the instructions to build the potato man?

Click on each instruction to see what happens.

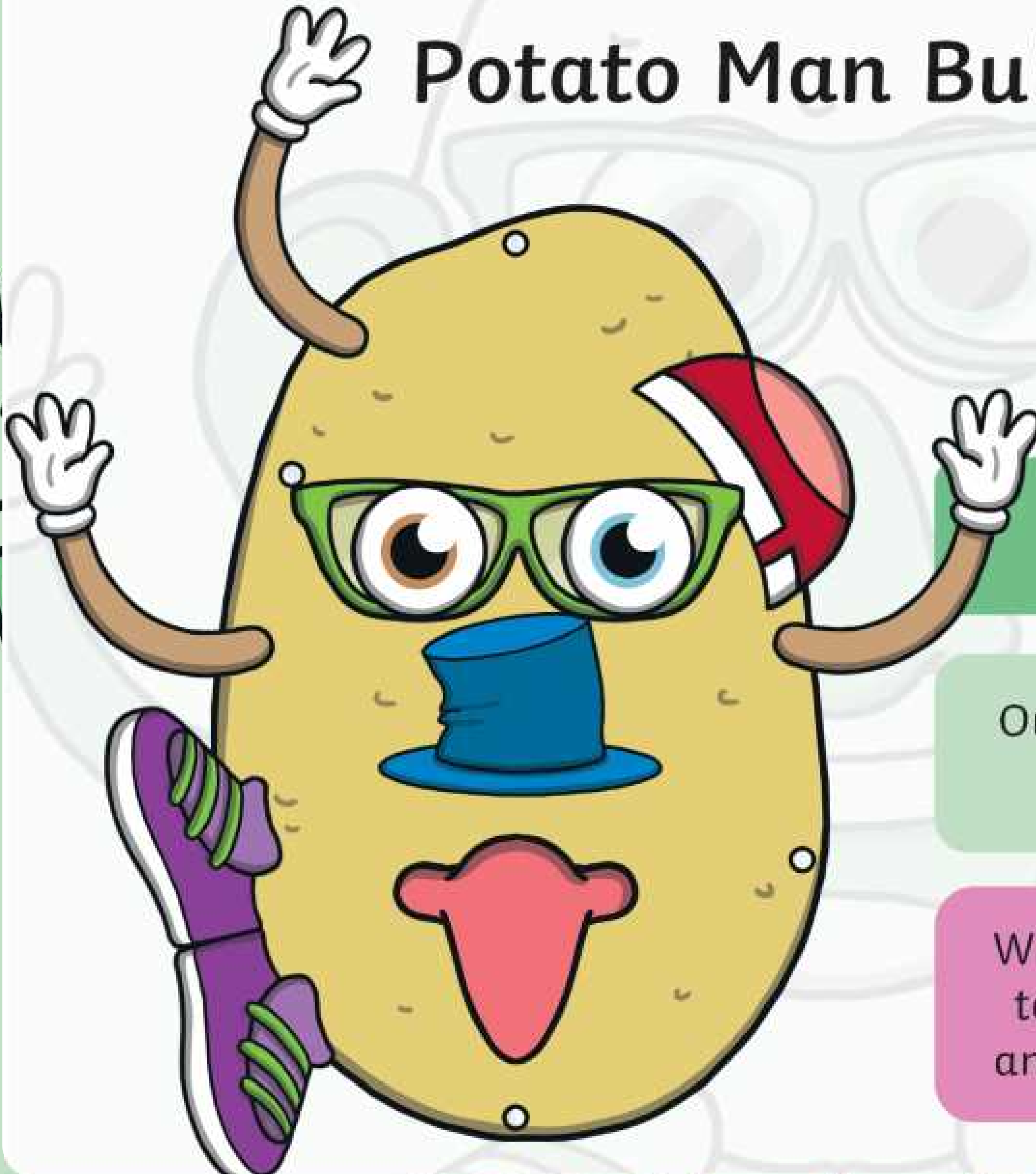
Add arms. | Add glasses.

Add eyes. | Add a hat.

Add a nose. | Add a mouth.

Add feet.

Potato Man Building



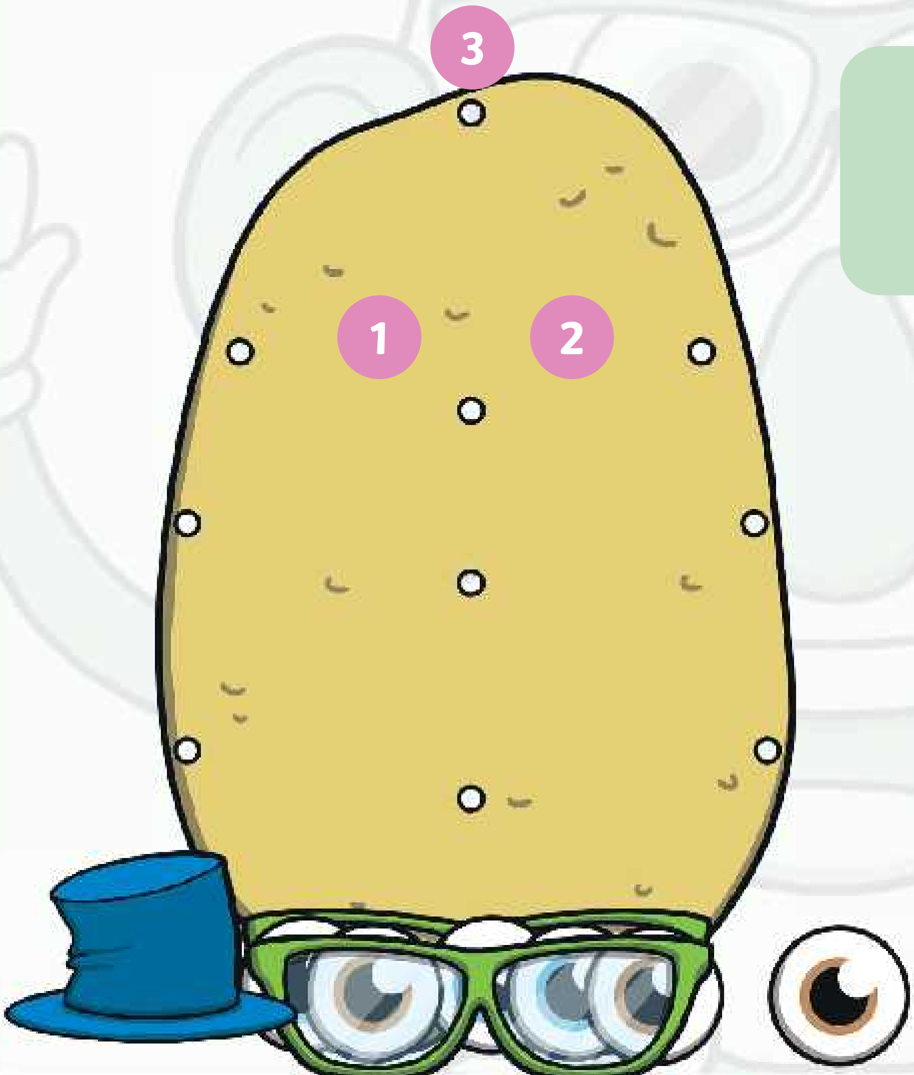
Why did it go wrong?

Our instructions need to be more detailed.

We need to say which parts to use, where to put them and which order to do it in.

Potato Man Building

Whole Class



See if you can choose the most sensible instructions to add the parts to this potato man.

Add brown eyes to holes 1 and 2.

Add a hat the right way up to hole 3.

Add eyes.

Add glasses.

Add a hat.

Did It Work?

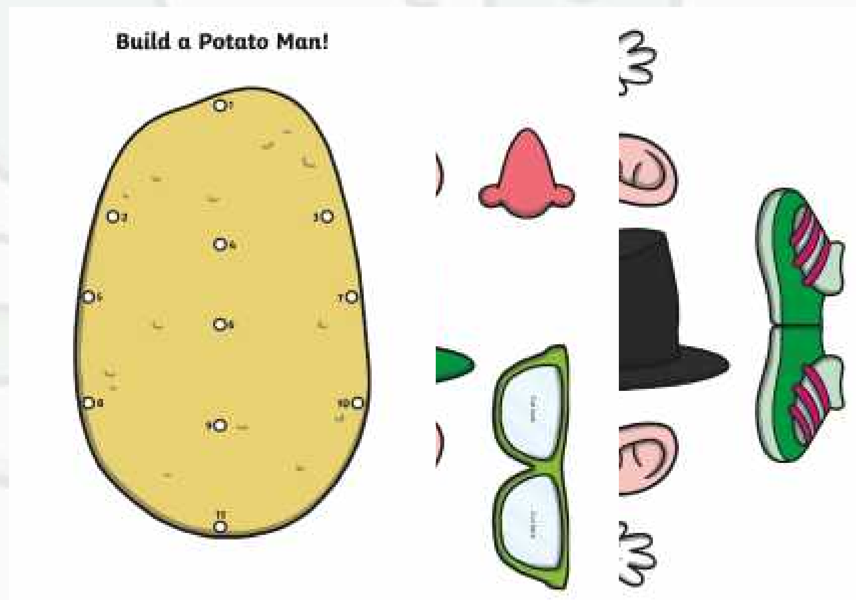
Now give your instructions to your partner.

You need to use the Build a potato man activity sheet to build your partner's potato man!

Next, show your partner the toy they should have built.

Does your toy look like the one in the picture?

Why? Why not?

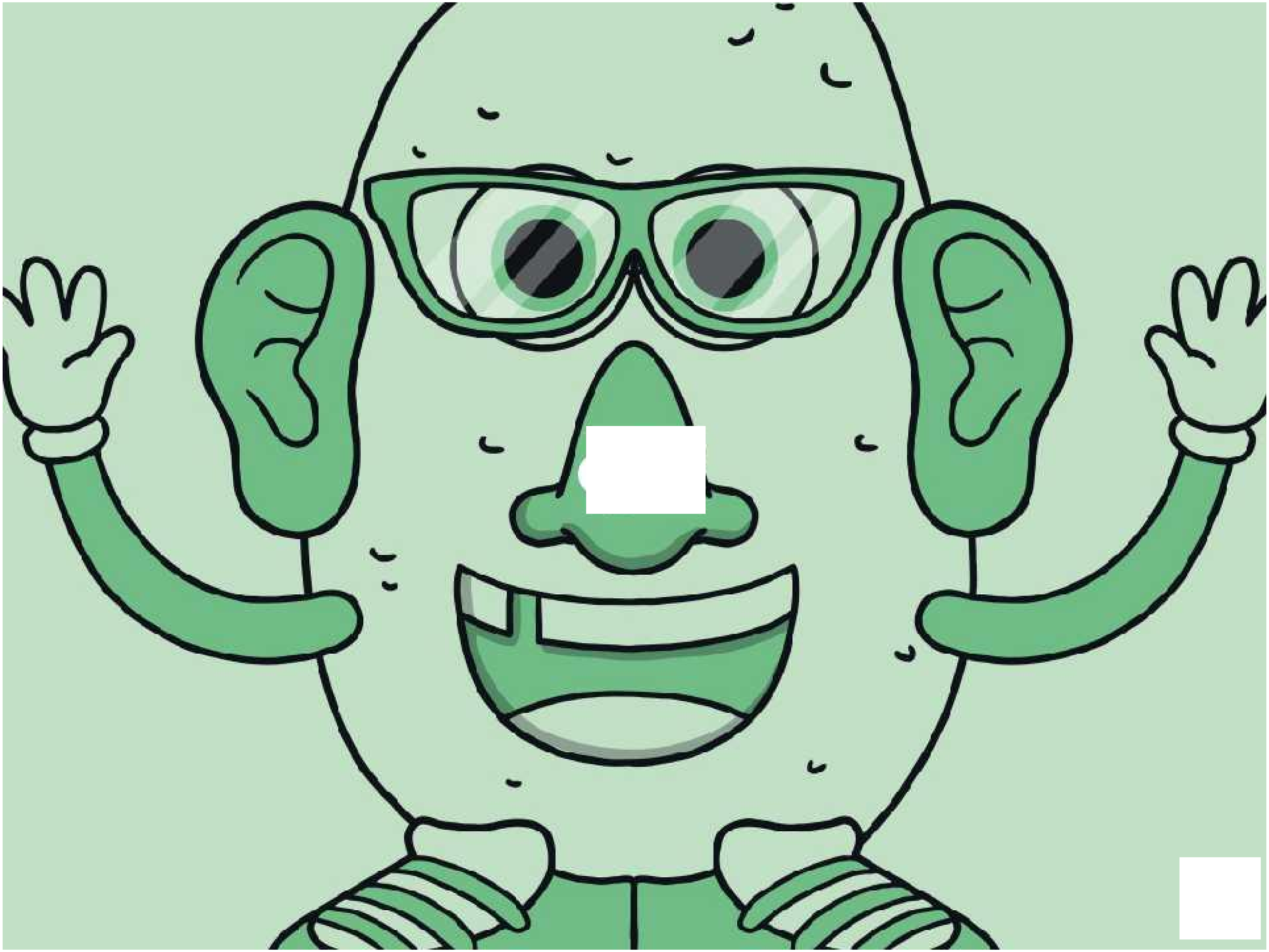


Aim

- I can say why it is important to be precise when writing an algorithm.

Success Criteria

- I can write and follow detailed instructions.
- I can see how a product changes when I change the instructions.



Avery Template: Name Badge Label, 8 per sheet | **Compatible Products**: 15395, 25395, 42395, 45395, 48395, 5395, 8395, 88395, 85395.

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys | Potato Man Algorithms

I can say why it is important to be precise when writing an algorithm.		
I can write and follow detailed instructions.		
I can see how a product changes when I change the instructions.		

Programming Toys: Program a Person

Aim:	**Success Criteria:**	**Resources:**
Understand what algorithms are and that programs execute by following precise and unambiguous instructions. Create and debug simple programs. Children will work within the context of writing instructions to program a person. I can write instructions to program a person like a computer.	I can write step-by-step instructions. I can check my work for mistakes (debug).	**Lesson Pack** Shoes Whiteboards Scissors
	Key/New Words: Algorithm, debug, program, turn, left, right, clockwise, anticlockwise.	**Preparation:** **Program a Person Editable Activity Cards** - 1 per pair, edited and cut up if required

Prior Learning: It would be helpful if children are familiar with directional words (left, right, half/quarter turn, clockwise, anticlockwise).

Learning Sequence

Help Me Put My Shoes On: Show the children a pair of your shoes. Ask them to tell their partner instructions for how you should put them on. Ask some children to say their instructions out loud. Some children will correctly identify that you will need to take off the shoes you are already wearing first. Explain to them using the **Lesson Presentation** what this lesson will be about.

Program the Teacher: Allow the children in pairs to program you. Tell them that you really need to reach something on the other side of the room and bring it back to the front. In pairs, children should think of instructions for you to follow and then ask one of them to say their instructions out loud. Follow their instructions, showing the need for detail, e.g. make it clear that 'walk forward' needs to be followed by 'x number of steps', and that 'turn around' will need a direction and possibly either a 'quarter' or 'half' instruction. Can children give precise instructions that someone else can follow?

Debugging: Explain to the class that if they notice something has gone wrong, they must go back and change the algorithm. Show the next slide on the **Lesson Presentation** and ask the children to identify where the instructions need fixing. Repeat for the following slides. Can children work out the mistakes? Can children correct a mistake?

Program Your Friend! Get the children into pairs of similar ability and tell them that they will take it in turns to be a toy robot! Each pair has a set of **Program a Person Activity Cards** which they must work through. One child should pick a card and attempt to instruct their partner, step by step, to complete the task. Ensure that as children are working, they make a written record of at least one set of instructions that was successful. You may wish to take photographs or videos of the children working together. Can children give precise instructions that someone else can follow?

- ★ Children follow instructions involving moving around the room to fetch objects.
- ★★ Children follow instructions asking them to move in more precise ways and use simple objects.
- ★★★ Children follow instructions including moving and using objects which may have more than one way of interacting with them.

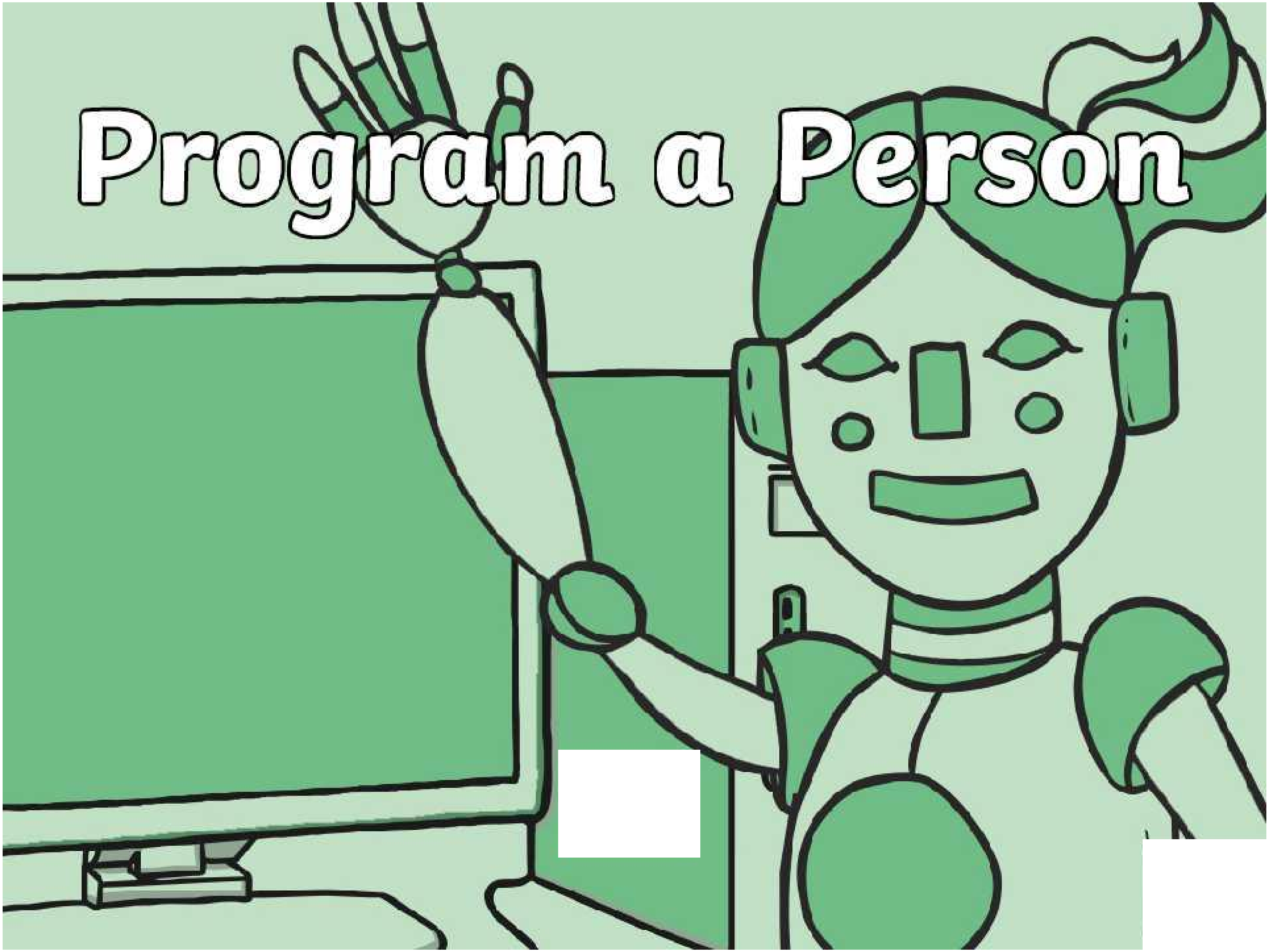
Algorithms Without Words: Ask the children how you could write an algorithm without words. Show them the symbols on the **Lesson Presentation** and ask them to draw which symbols they would have to draw to get you, the teacher, to a different part of the room. Tell them that they will also need to tell you how many steps or turns to take, using a number of arrows. Can children use simple symbols to represent a single action?

Taskit

Challengeit: Can a child, with an adult, direct a friend to complete a task, without them knowing what it is?

Askit: Can you think of a task which could have different algorithms for the same job? For example, how many ways are there to get to our classroom door?

Program a Person



Computing

Programming Toys

Aim

- I can write instructions to program a person like a computer.

Success Criteria

- I can write step-by-step instructions.
- I can check my work for mistakes (debug).

Help Me Put My Shoes On

Whole Class

How do I put these shoes on?

Tell your partner instructions for how I should put my shoes on.

Help Me Put My Shoes On

Did it work?

Did you make sure that I took off my old shoes first?

What would happen if I forgot to take them off first?

Sometimes, a computer can only follow an instruction if something else has already happened.

Can you think of any other tasks that have to be done in a certain order?

Help Me Put My Shoes On

What tasks did you think of?

Think about:

Computers work this way too – everything needs to be done in the right order.

What if you forgot to take off your normal clothes first?

What if you didn't get out a bowl before you poured the milk?

Program the Teacher!

Can you program me?

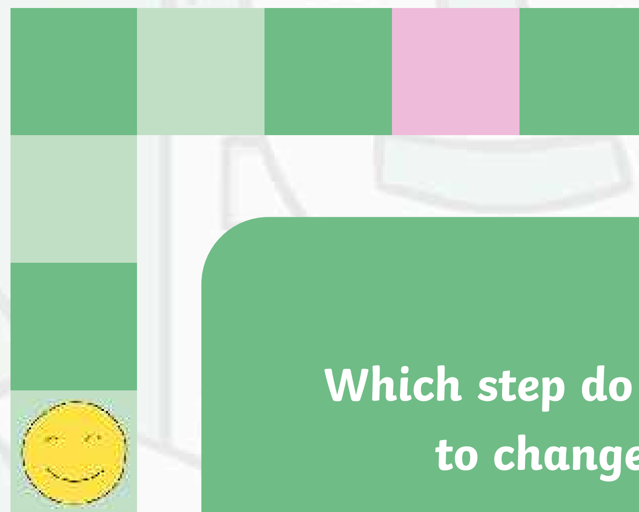
What kind of instruction words will you use?

Debugging

Uh oh! I'm trying to write an algorithm to get the smiley face to the pink square, but I can't seem to get there.

To get to the pink square:

1. Move forward 3 squares.
2. Turn a quarter turn clockwise.
3. Move forward 4 squares.



Which step do I need to change?

Debugging

Uh oh! I'm trying to write an algorithm to get the smiley face to the pink square, but I can't seem to get there.

To get to the pink square:

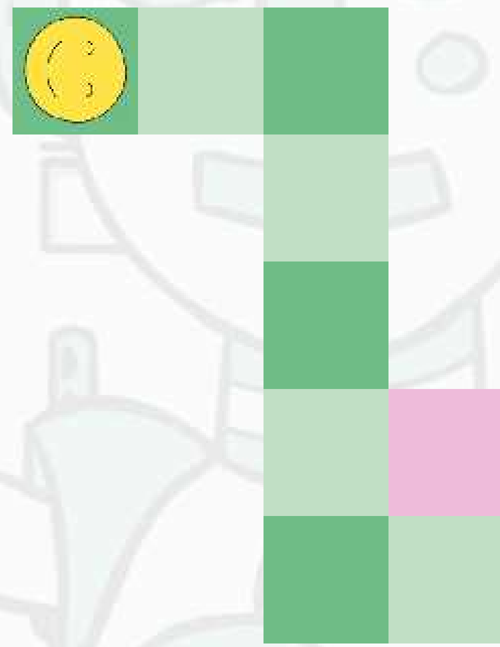
1. Move forward 3 squares.
2. Turn a quarter turn clockwise.
3. Move forward **3** squares.

Which step do I need to change?

Debugging

To get to the pink square:

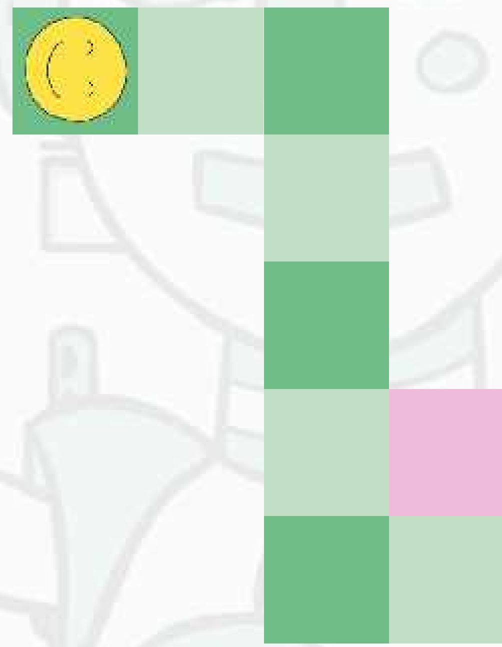
1. Move forwards 2 squares.
2. Turn a quarter turn clockwise.
3. Move forward 4 squares.
4. Turn a quarter turn anticlockwise.
5. Move forward 1 square.



Debugging

To get to the pink square:

1. Move forwards 2 squares.
2. Turn a quarter turn clockwise.
3. Move forward **3** squares.
4. Turn a quarter turn anticlockwise.
5. Move forward 1 square.



Debugging

To get to the pink square:

1. Move forward 4 squares.
2. Turn a quarter turn clockwise.
3. Move forward 1 square.
4. Turn a quarter turn anticlockwise.
5. Move forward 1 square.
6. Turn a quarter turn anticlockwise
7. Move forward 1 square.

Debugging

To get to the pink square:

1. Move forward **3** squares.
2. Turn a quarter turn clockwise.
3. Move forward 1 square.
4. Turn a quarter turn anticlockwise.
5. Move forward **2** squares.
6. Turn a quarter turn anticlockwise
7. Move forward 1 square.

Program Your Friend!

Whole Class

Your friend is now a toy robot! Your job is to get them to do the tasks on the cards.

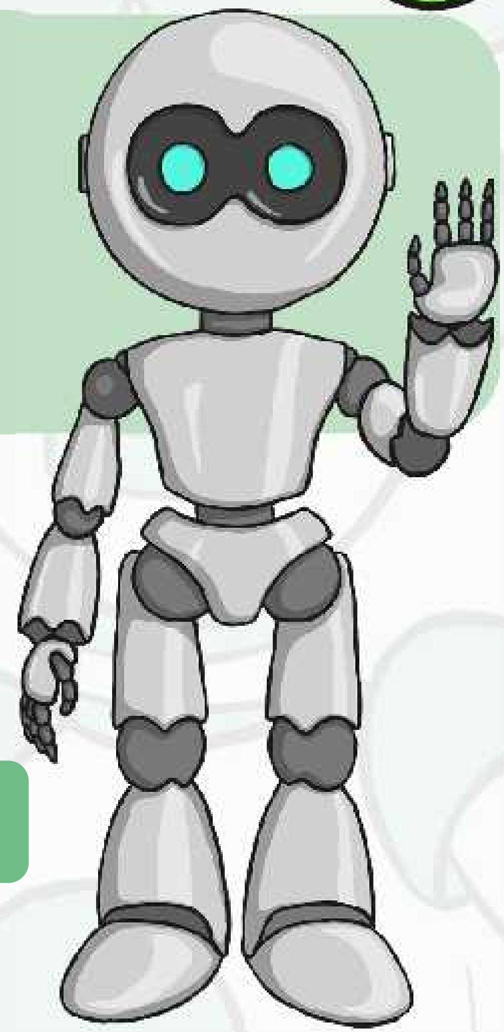
Remember, you have to tell them exactly how to do each step.

Which words will you need to use?

steps turn right sidestep half

left backwards forward quarter

clockwise anticlockwise three-quarters



Algorithms Without Words

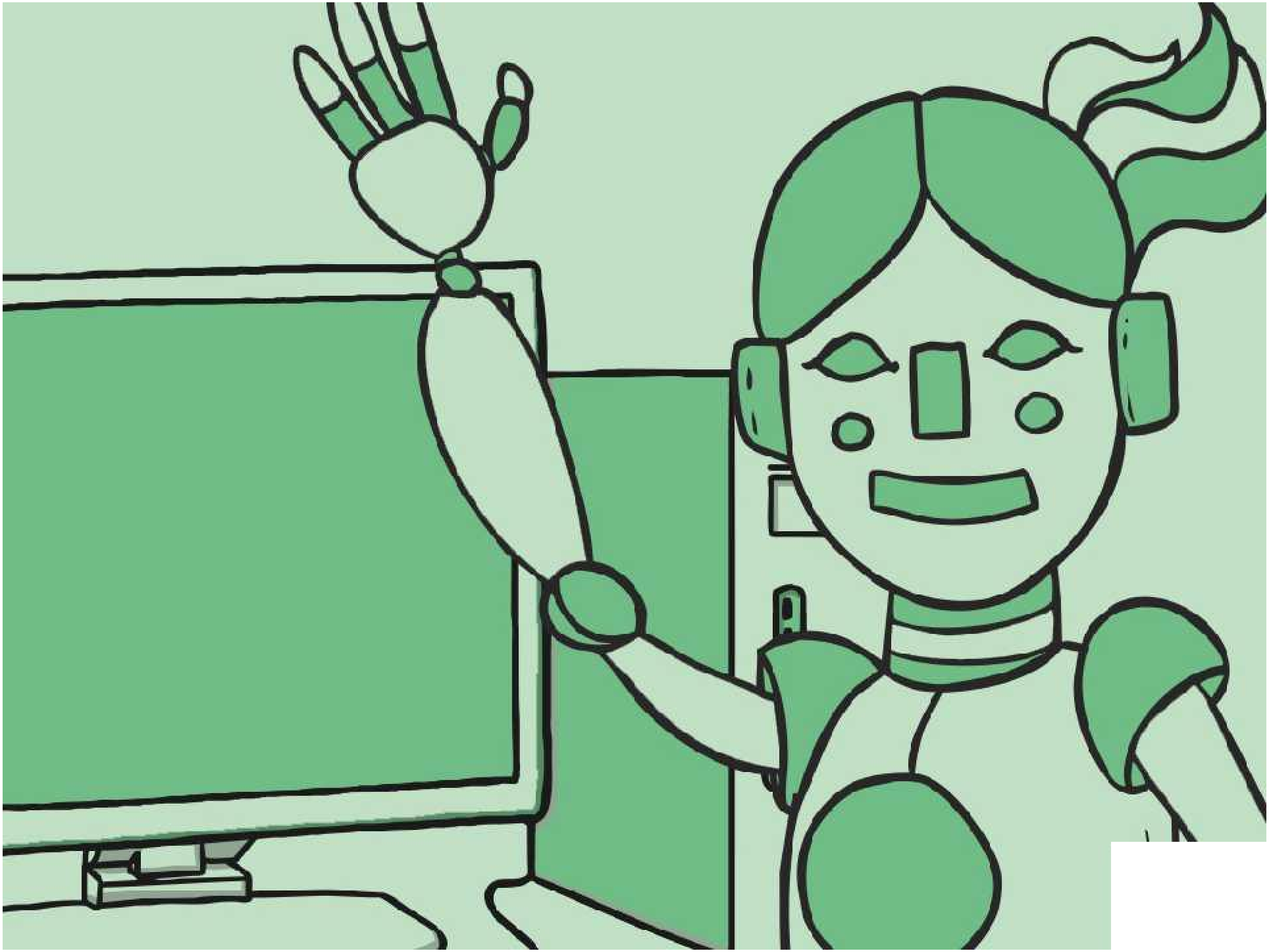
What if we couldn't use words to program our robot? How else could we give it instructions?

Aim

- I can write instructions to program a person like a computer.

Success Criteria

- I can write step-by-step instructions.
- I can check my work for mistakes (debug).



Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Programming Toys | Program a Person

I can write instructions to program a person like a computer.		
I can write step-by-step instructions.		
I can check my work for mistakes (debug).		

Program a Person Cards

Your job is to get your robot partner to:

Walk to a door.	Stand by the teacher's desk.	Sit down on someone else's chair.
Pick up a pencil.	Walk to the bin.	Crouch down next to the window.
Kneel next to a cupboard.	Lie down on the carpet.	Stand with their back to the window.

Program a Person Cards

Your job is to get your robot partner to:

Open a door.	Stand by the teacher's desk with their hands in the air.	Sit down on someone else's chair.
Pick up a pencil.	Pick up a pencil and put it down somewhere else.	Crouch down next to the window.
Jump!	Open a book.	Stand with their back to the window.

Program a Person Cards

Your job is to get your robot partner to:

Open a door.	Stand by the teacher's desk and wave their hands.	Sit down on 3 different chairs.
Draw a circle and a line in the air with their finger.	Pick up a pencil and put it down somewhere else.	Crouch down next to the window.
Jump! Can you do this without saying the word 'jump'?	Bring a book to you and open it.	Stand with their back to the window, holding a ruler.

Program a Person Cards

Your job is to get your robot partner to:

Walk to a door.	Stand by the teacher's desk.	Sit down on someone else's chair.
Pick up a pencil.	Walk to the bin.	Crouch down next to the window.
Kneel next to a cupboard.	Lie down on the carpet.	Stand with their back to the window.

Program a Person Cards

Your job is to get your robot partner to:

Open a door.	Stand by the teacher's desk with their hands in the air.	Sit down on someone else's chair.
Pick up a pencil.	Pick up a pencil and put it down somewhere else.	Crouch down next to the window.
Jump!	Open a book.	Stand with their back to the window.

Program a Person Cards

Your job is to get your robot partner to:

Open a door.	Stand by the teacher's desk and wave their hands.	Sit down on 3 different chairs.
Draw a circle and a line in the air with their finger.	Pick up a pencil and put it down somewhere else.	Crouch down next to the window.
Jump! Can you do this without saying the word 'jump'?	Bring a book to you and open it.	Stand with their back to the window, holding a ruler.

Programming Toys: Toy Shop Part 1

Aim: Understand what algorithms are and that programs execute by following precise and unambiguous instructions; create and debug simple programs in the context of programming a Bee-Bot (or similar programmable toy) to reach a set marker. I can program a Bee-Bot (or similar programmable toy) to move.	**Success Criteria:** I can direct a Bee-Bot (or similar programmable toy) to a toy. I can program a Bee-Bot (or similar programmable toy) using the arrow buttons. **Key/New Words:** Algorithm, debug, program, turn, left, right, sequence, clockwise, anticlockwise.	**Resources:** **Lesson Pack** Bee-Bots (or similar programmable toy) Whiteboards and pens Camera **Preparation:** **Toy Cupboard Programmable Toy Mat** - 1 per pair or group

Prior Learning: Children were introduced to using pictures and symbols to give instructions in the first three lessons of this unit.

Learning Sequence

Symbols: Use the **Lesson Presentation** to introduce the children to a Bee-Bot (or similar programmable toy) and discuss what the different buttons do (and why these symbols have been chosen). Use the following slides to practise choosing the correct buttons to achieve a movement. You may also wish to use a real Bee-Bot (or similar programmable toy) in the classroom and ask children to help you to replicate the movements on the **Lesson Presentation**. If needed, continue to practise as a class, predicting the movement of the Bee-Bot (or similar programmable toy). Can children direct the Bee-Bot (or similar programmable toy) correctly?
Programmable Toy at the Toy Shop: Show children the **Toy Cupboard Programmable Toy Mat**. Show them how one square on the mat is equal to one press of the forward arrow on the Bee-Bot (or similar programmable toy). Complete the practice task on the **Lesson Presentation**.
Toy Shop Task: In pairs or groups, children direct their Bee-Bot (or similar programmable toy) to a toy on the mat. Children pick their own starting challenge level from the **Lesson Presentation** and move through the tasks when they are ready. Take photographs for evidence, if needed. Can children program a Bee-Bot (or similar programmable toy) using the arrow buttons? ★ Children get to chosen toy, pressing and moving one instruction at a time. ★★ Children program more than one step at once, with the goal of programming all instructions in one go. ★★★ Children direct the Bee-Bot (or similar programmable toy) to the toy while avoiding obstacles placed on the mat.
How Did You Do? Discuss and evaluate the children's success using the **Lesson Presentation**.

Taskit

Playit: If you have access to tablets, allow children to freely play using the Bee-Bot (or similar programmable toy) app, free in the app store, to practise programming a single sequence.

Recordit: Children draw the arrows they pressed to reach their toy, and ask their partner to read and program the Bee-Bot (or similar programmable toy).

Computing

Toy Shop Part 1


Toy Shop
Part 1

Aim

- I can program a Bee-Bot (or similar programmable toy) to move.

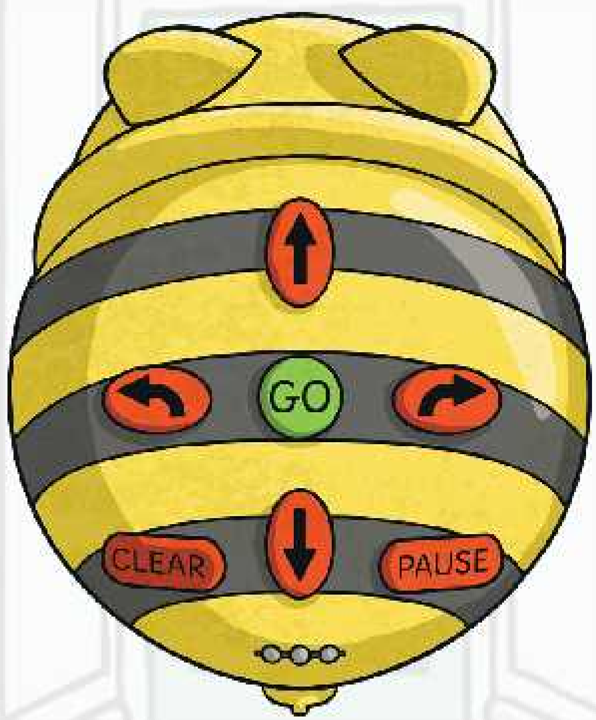
Success Criteria

- I can direct a Bee-Bot (or similar programmable toy) to a toy.
- I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.

Symbols

This is a programmable toy.

Today we will be programming the toy (Bee-Bot or similar programmable toy) to move around a toy shop.

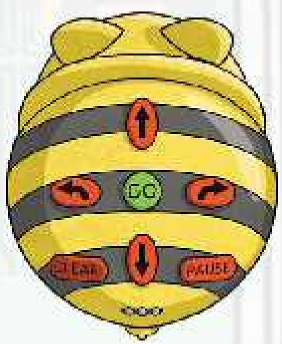


If you make a mistake, press this button and you can start your instructions all over again.

Symbols

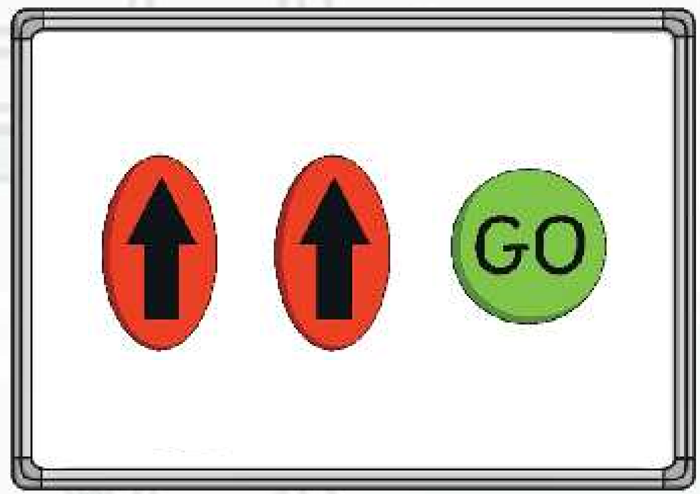
Draw on your whiteboard the buttons that you would have to press to make a Bee-Bot (or similar programmable toy) move like this.

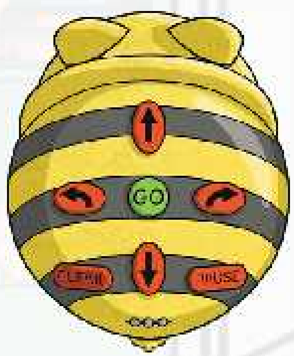
Click “**Go**” to see the programmable toy move.



Symbols

Did you draw these instructions?

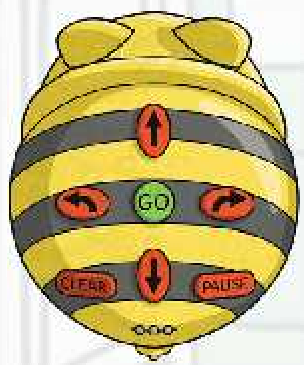




Symbols

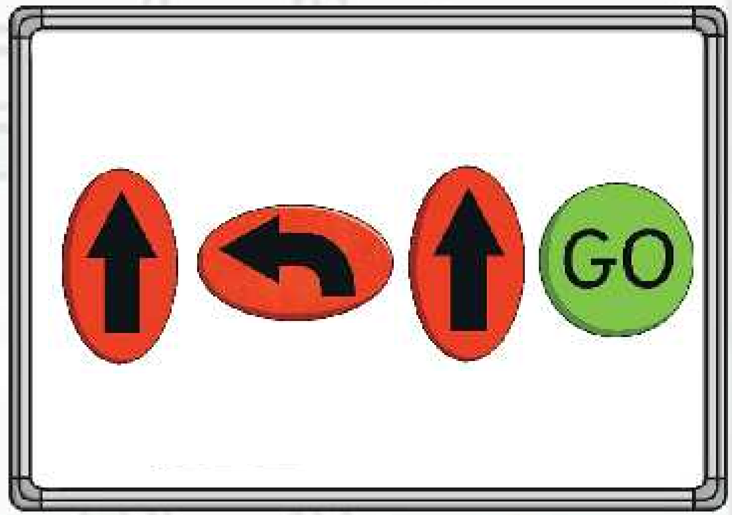
Draw on your whiteboard the buttons that you would have to press to make a Bee-Bot (or similar programmable toy) move like this.

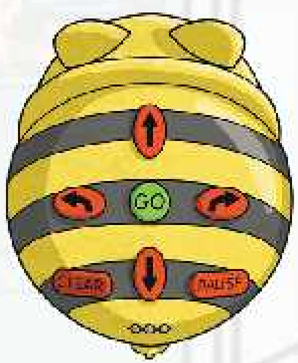
Click “**Go**” to see the programmable toy move.



Symbols

Did you draw these instructions?

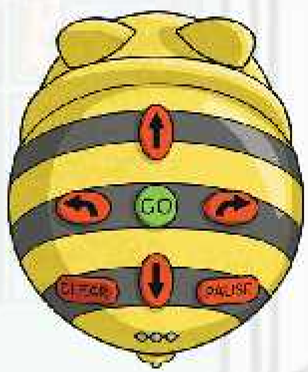




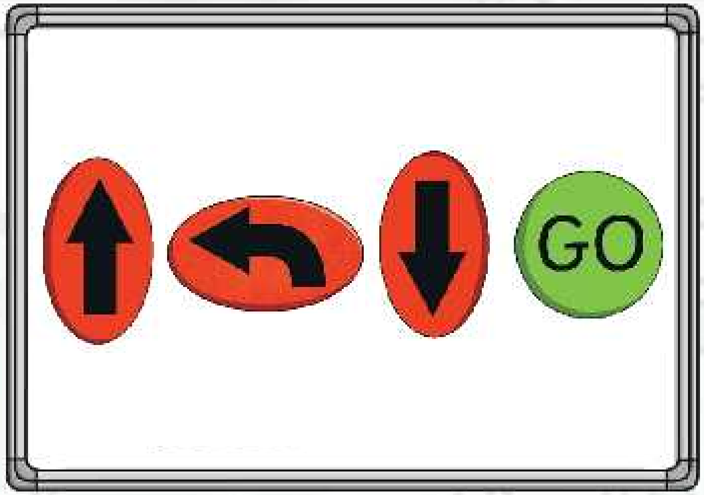
Symbols

Draw on your whiteboard the buttons that you would have to press to make a Bee-Bot (or similar programmable toy) move like this.

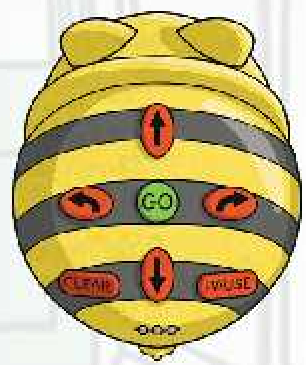
Click "**Go**" to see the programmable toy move.



Symbols

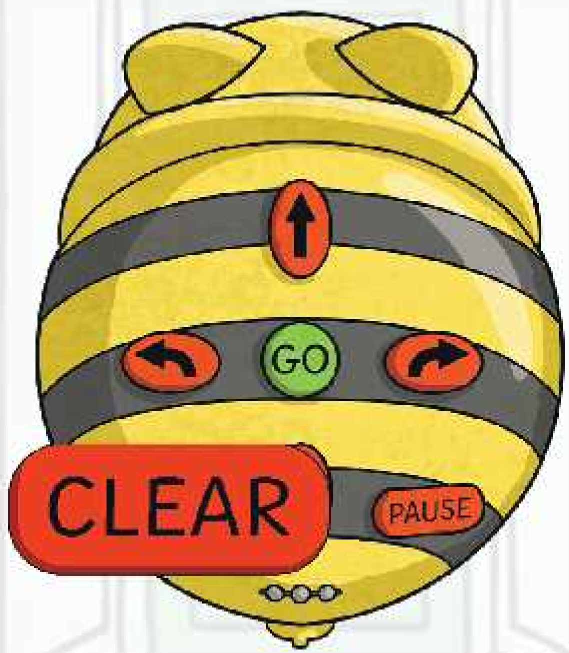


Did you draw these instructions?



Symbols

It's a good idea to press CLEAR before you start programming your toy with instructions, to make sure it hasn't remembered anything from the last time.



Programmable Toy at the Toy Shop

This is the Twinkl Toy Shop.

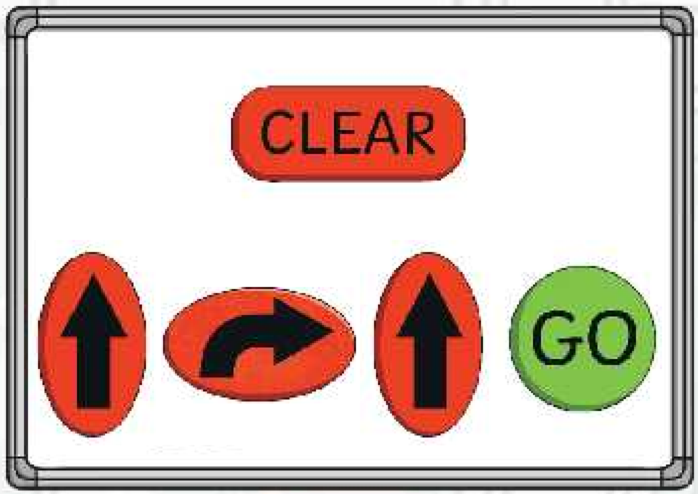
You need to direct your Bee-Bot (or similar programmable toy) to the right toy, starting from the label at the bottom.

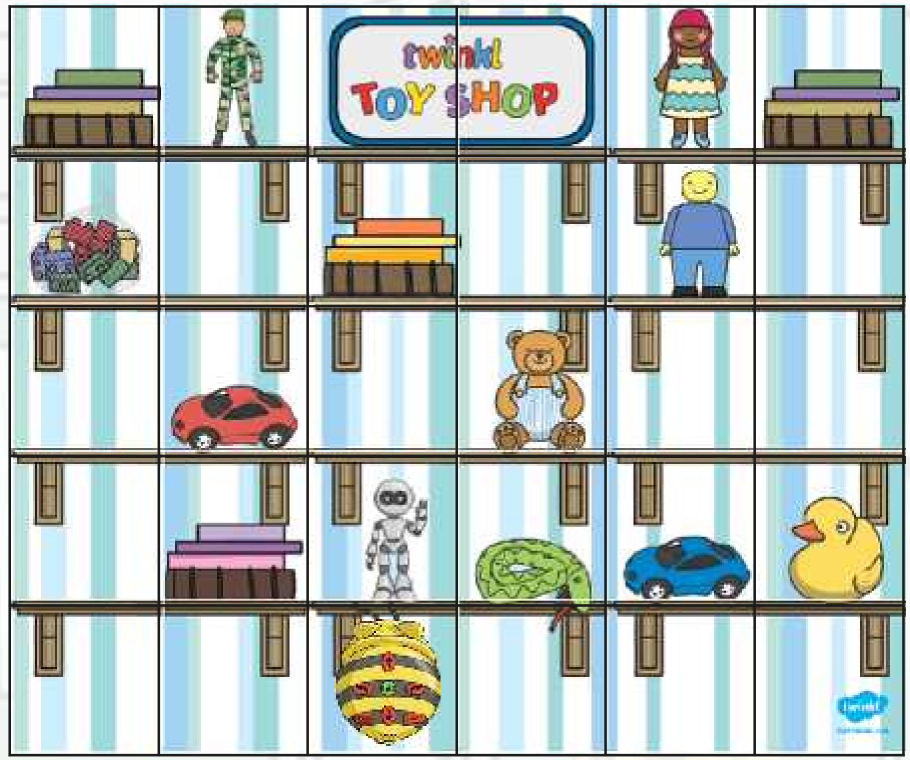
Can you draw on your whiteboard the buttons that you would need to get the Bee-Bot (or similar programmable toy) to the **snake**?

Remember to start here every time.

Programmable Toy at the Toy Shop

Did you draw these instructions?





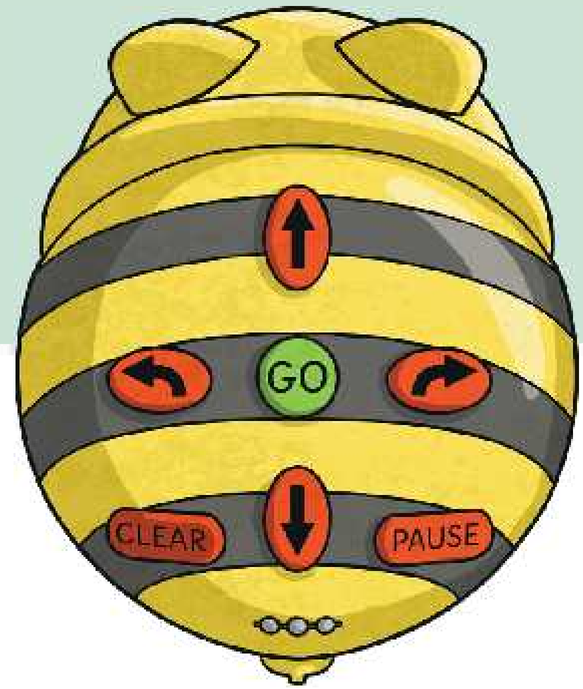
Did you remember to press CLEAR ?

Toy Shop Task

Take it in turns to program your Bee-Bot (or similar programmable toy).

Your partner will choose a toy on the mat.

Start from the label:



Press CLEAR .

Press the buttons for your instructions.

Press GO .

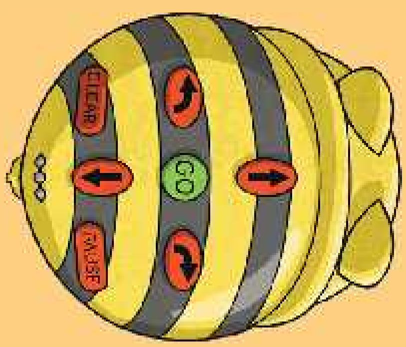
If you make a mistake, press CLEAR and start again.

Make sure everyone gets a turn.

Toy Shop Task

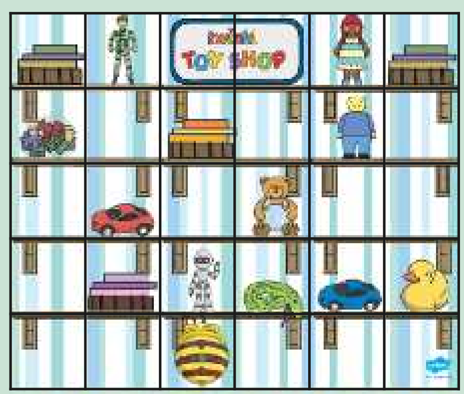
★

Get to the toy. Push one button at a time and press CLEAR.



★★

Get to the toy. Push all of your buttons before you press CLEAR.



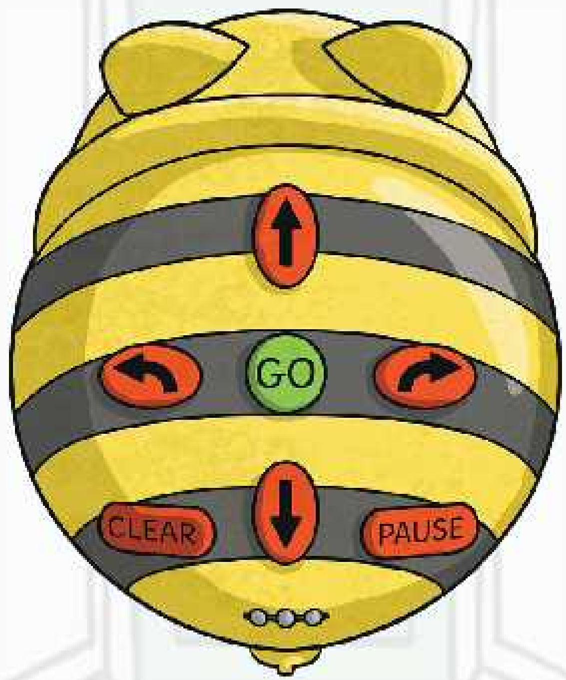
★★★

Put an object on the mat. Can you go around it to get to your toy?

How Did You Do?

Which bit of programming was the trickiest?

How did you fix something if it went wrong?



Aim

- I can program a Bee-Bot (or similar programmable toy) to move.

Success Criteria

- I can direct a Bee-Bot (or similar programmable toy) to a toy.
- I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.


MONOPOLY


TOY SHOP
1
2
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
Here
28
29
30

1

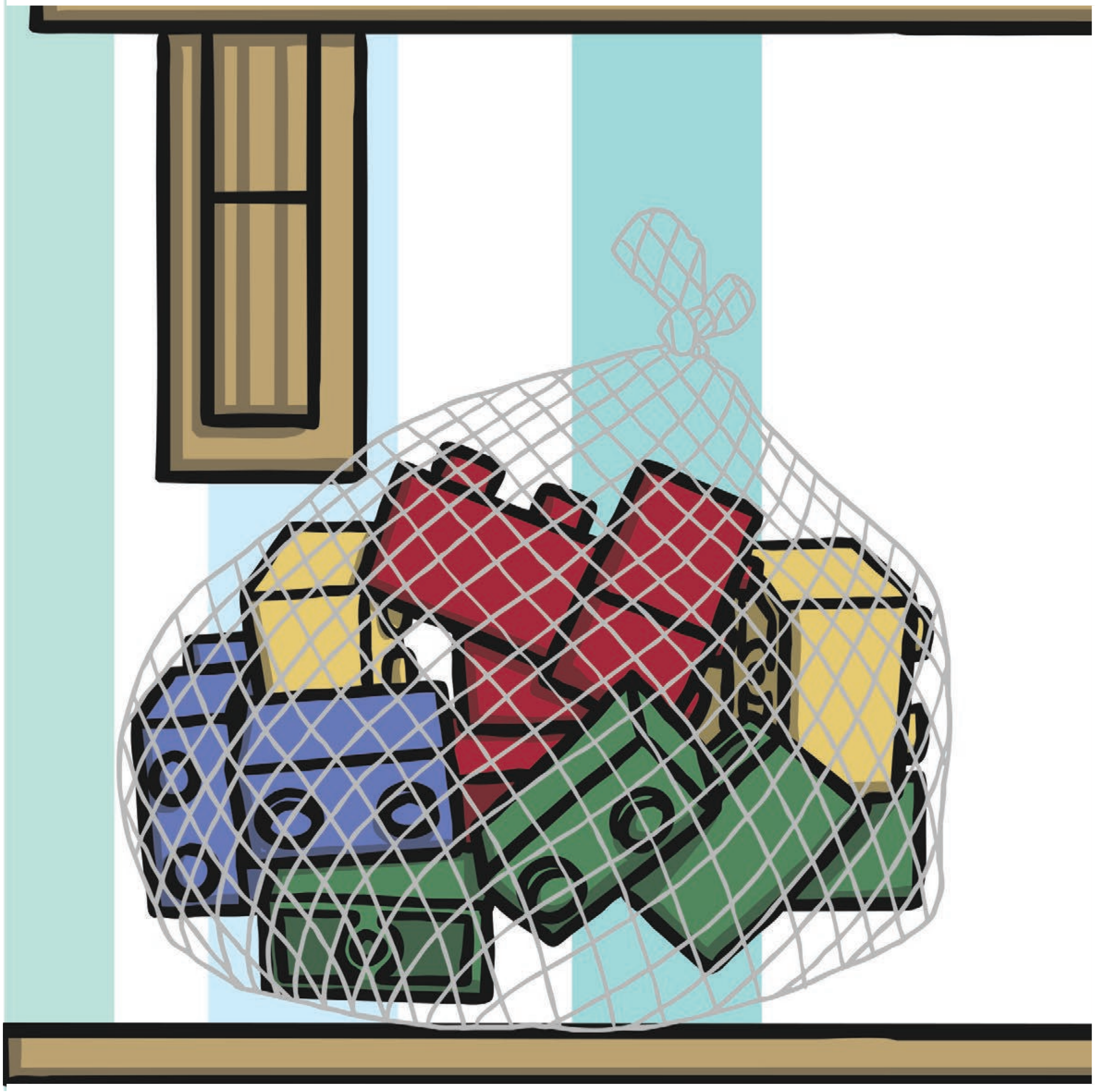
2

3

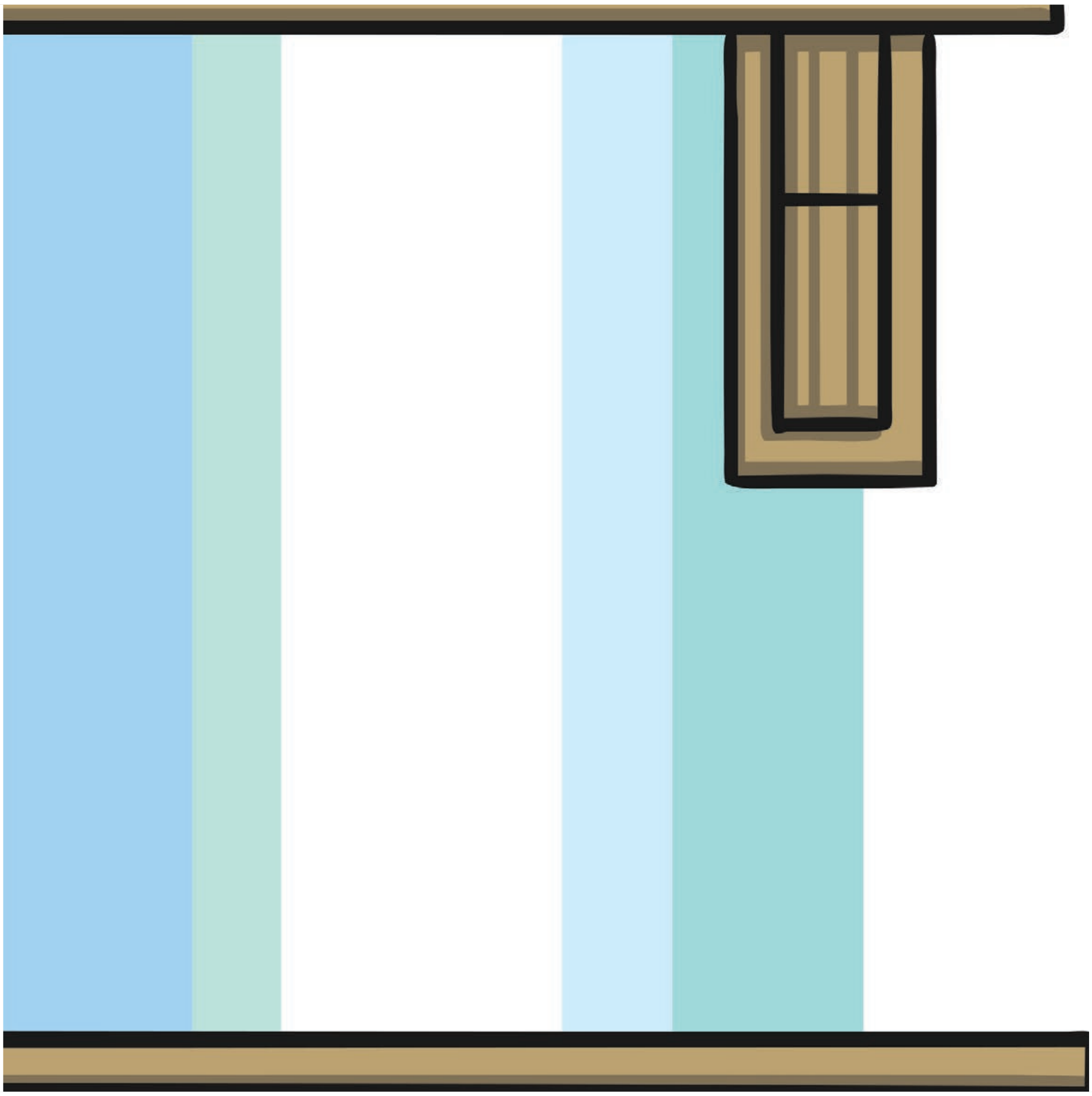
4

5

6

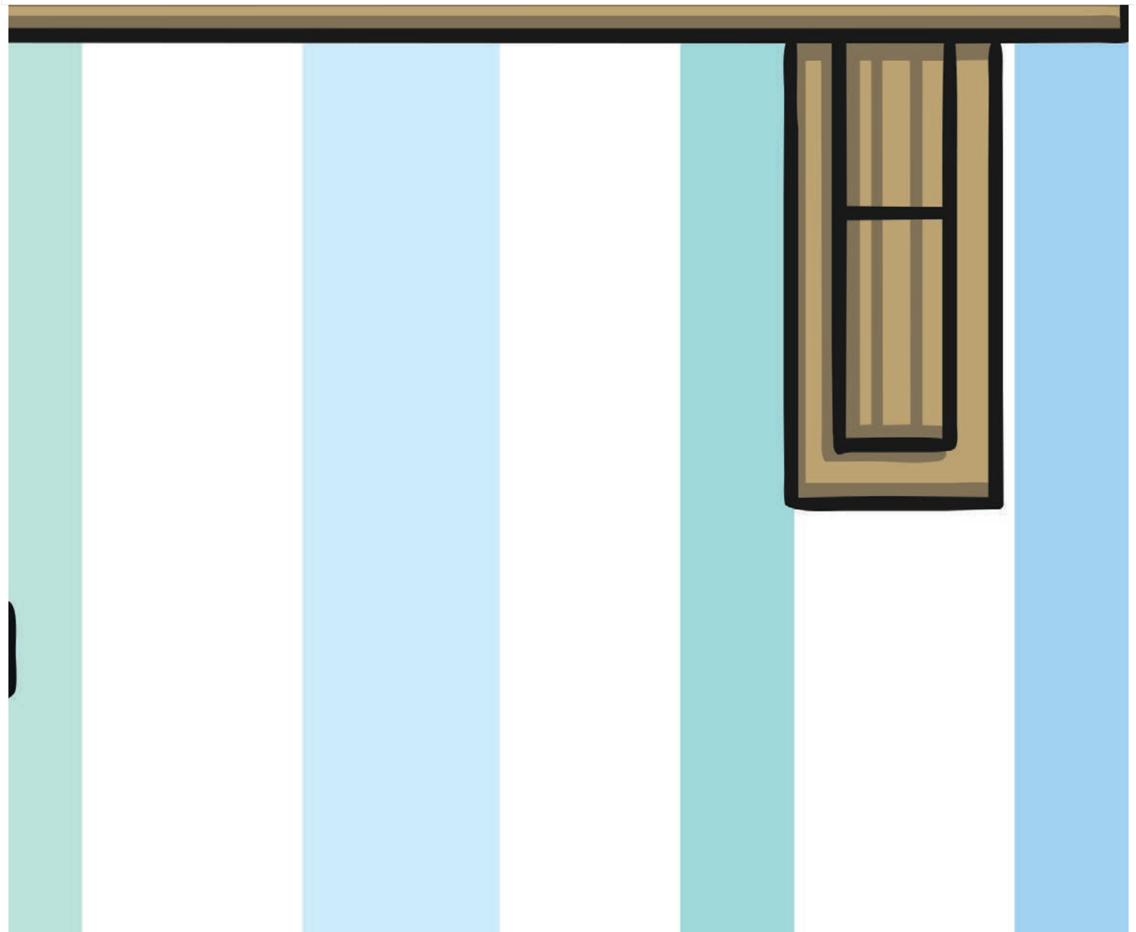


7

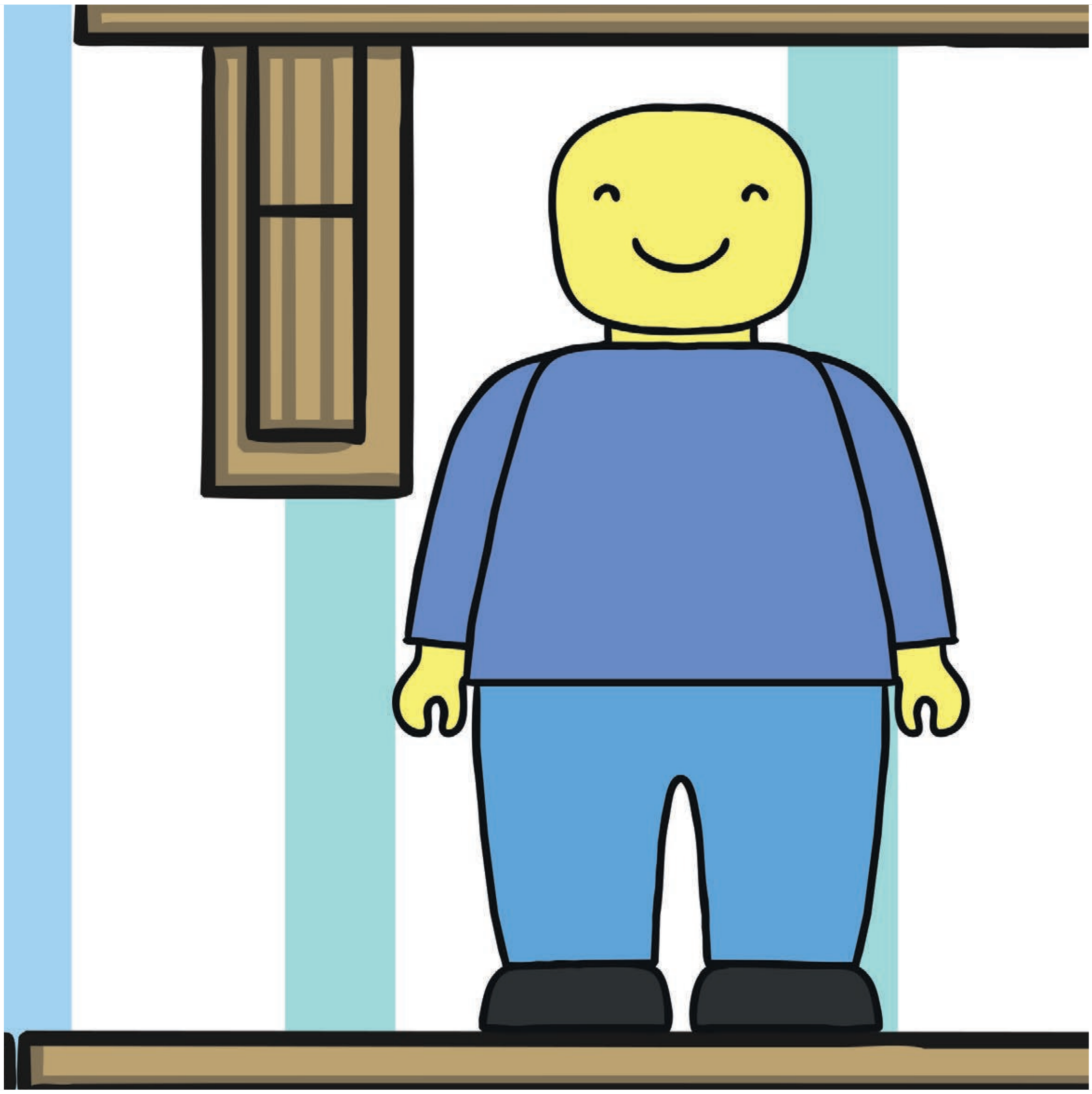


8

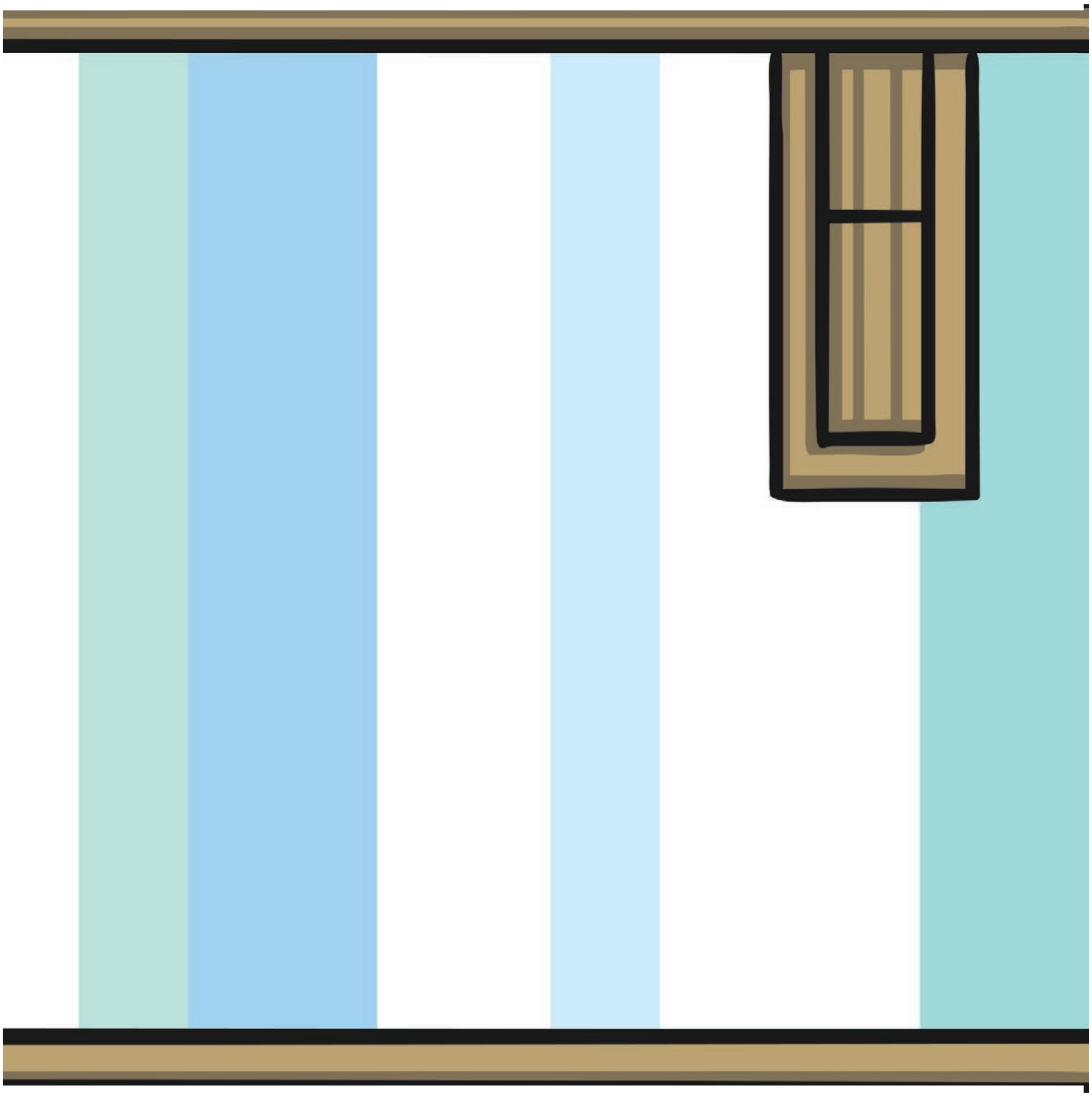
9



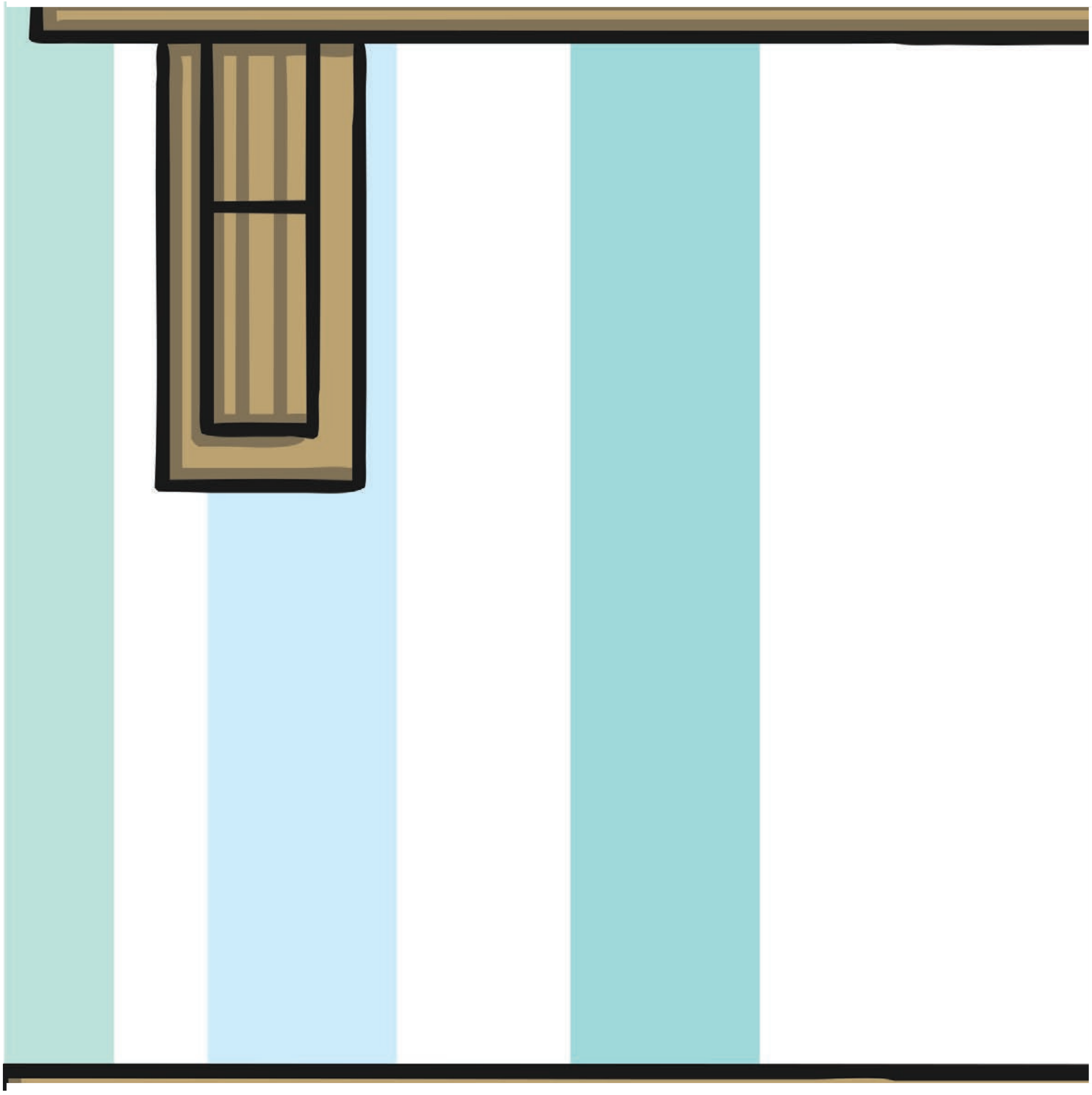
10



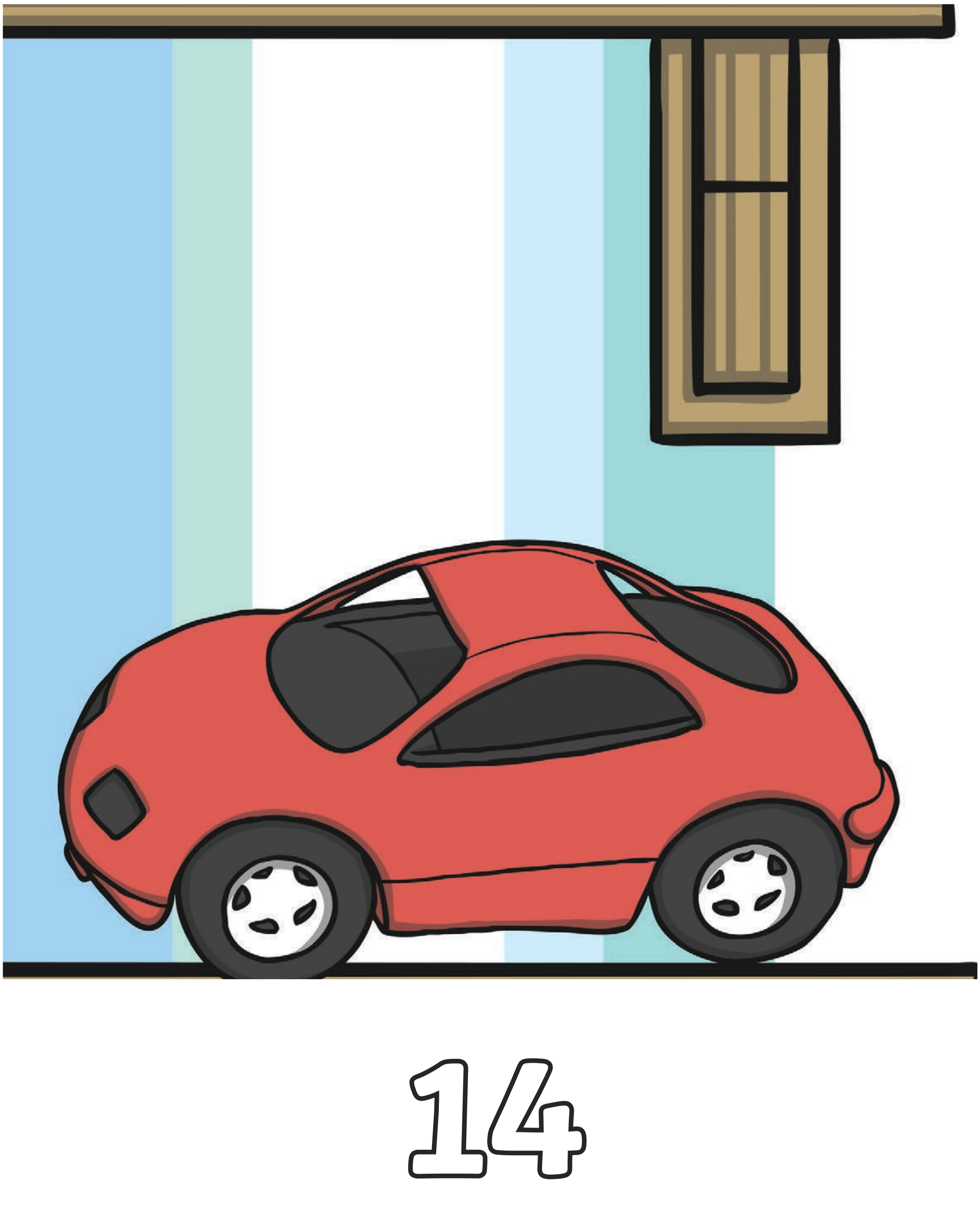
11



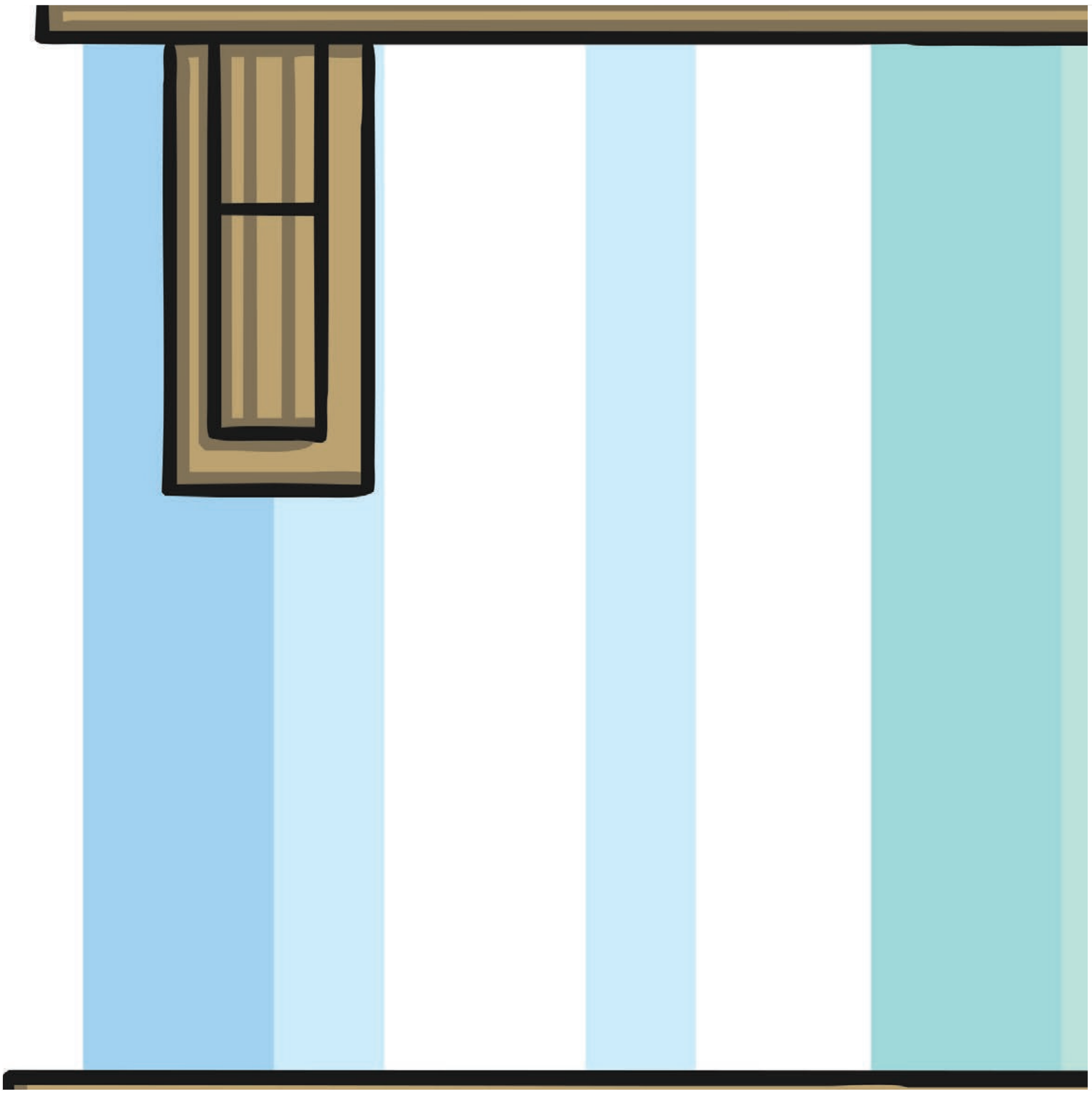
12



13

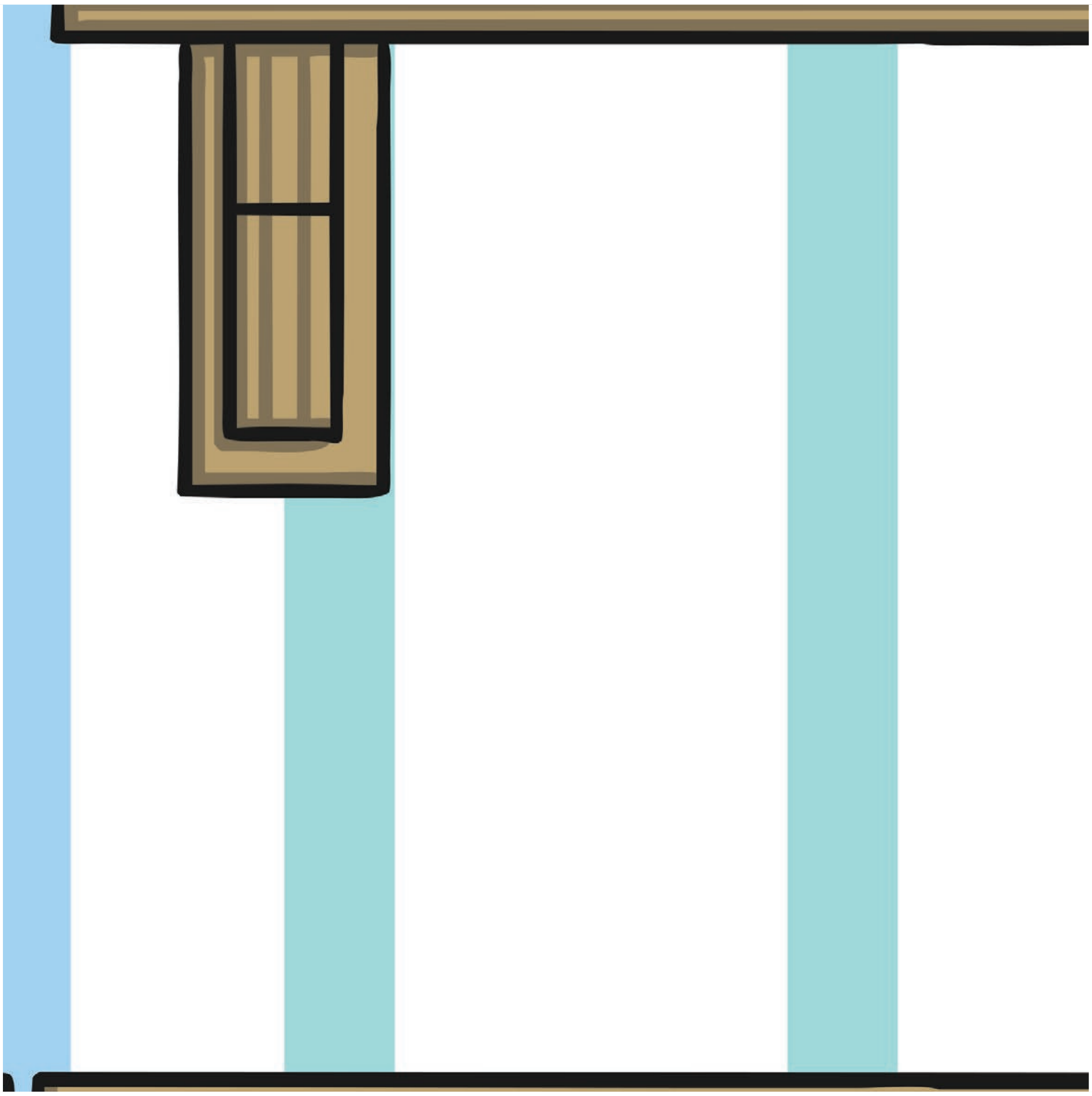


14

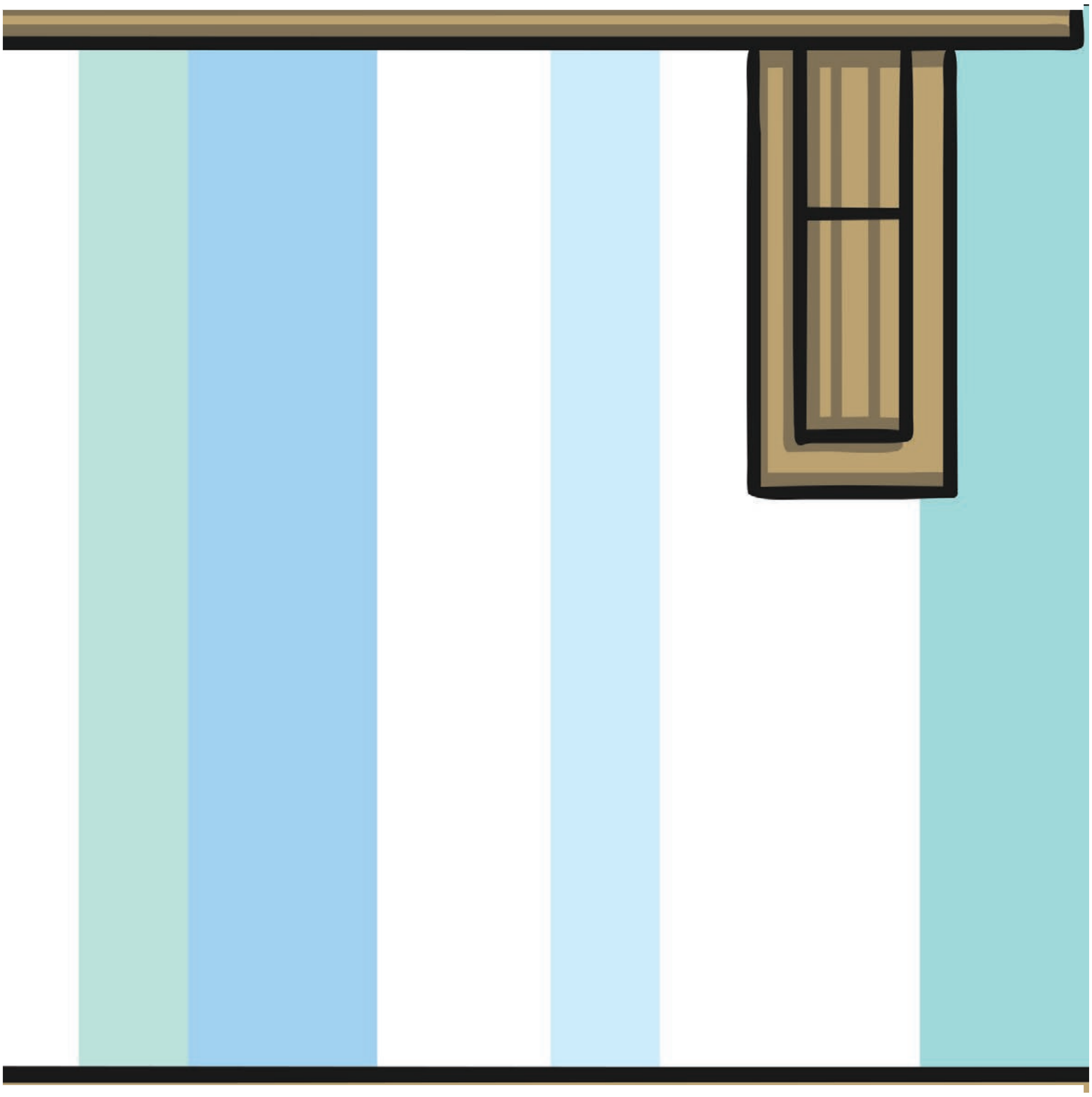


15

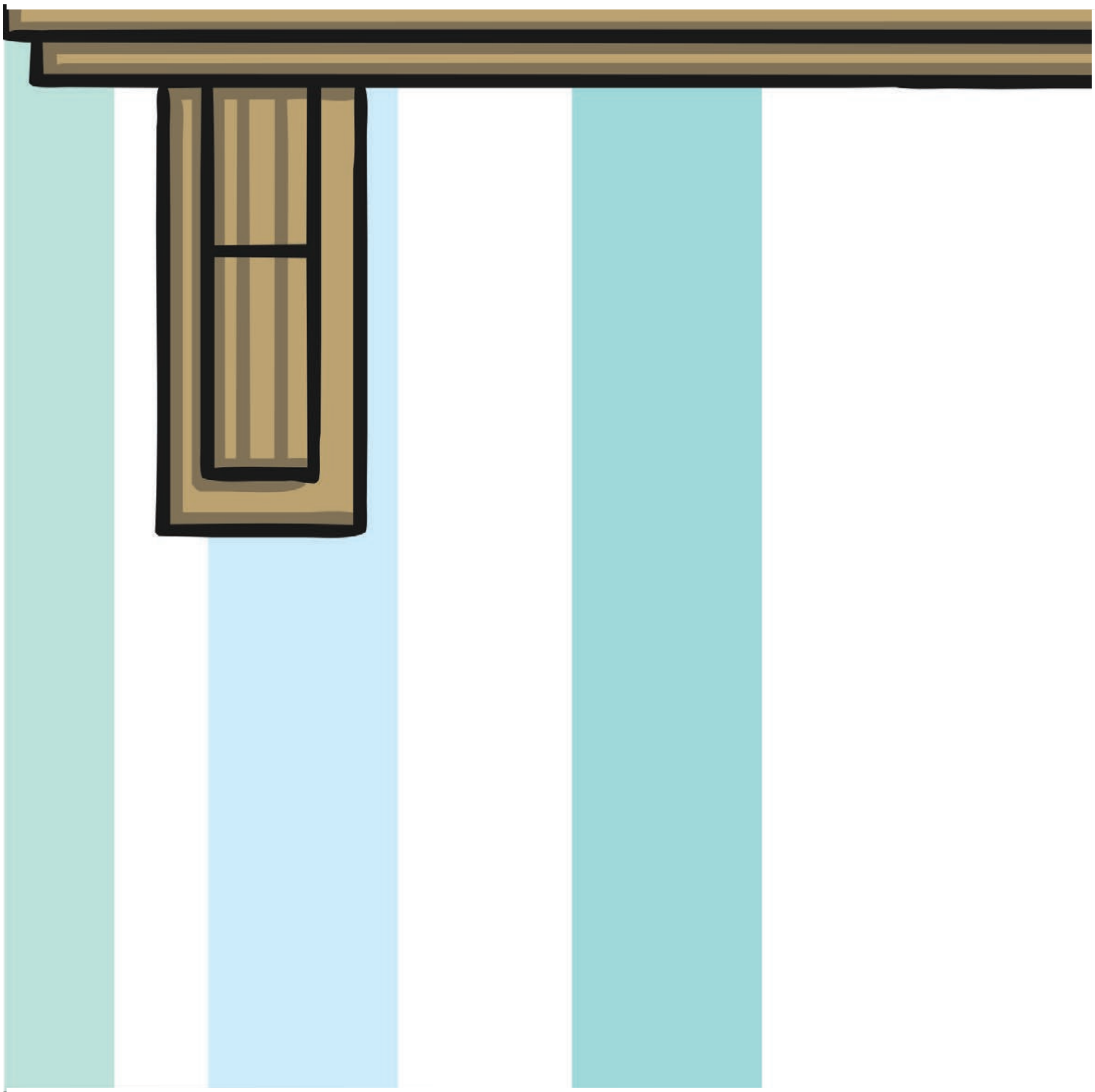
16



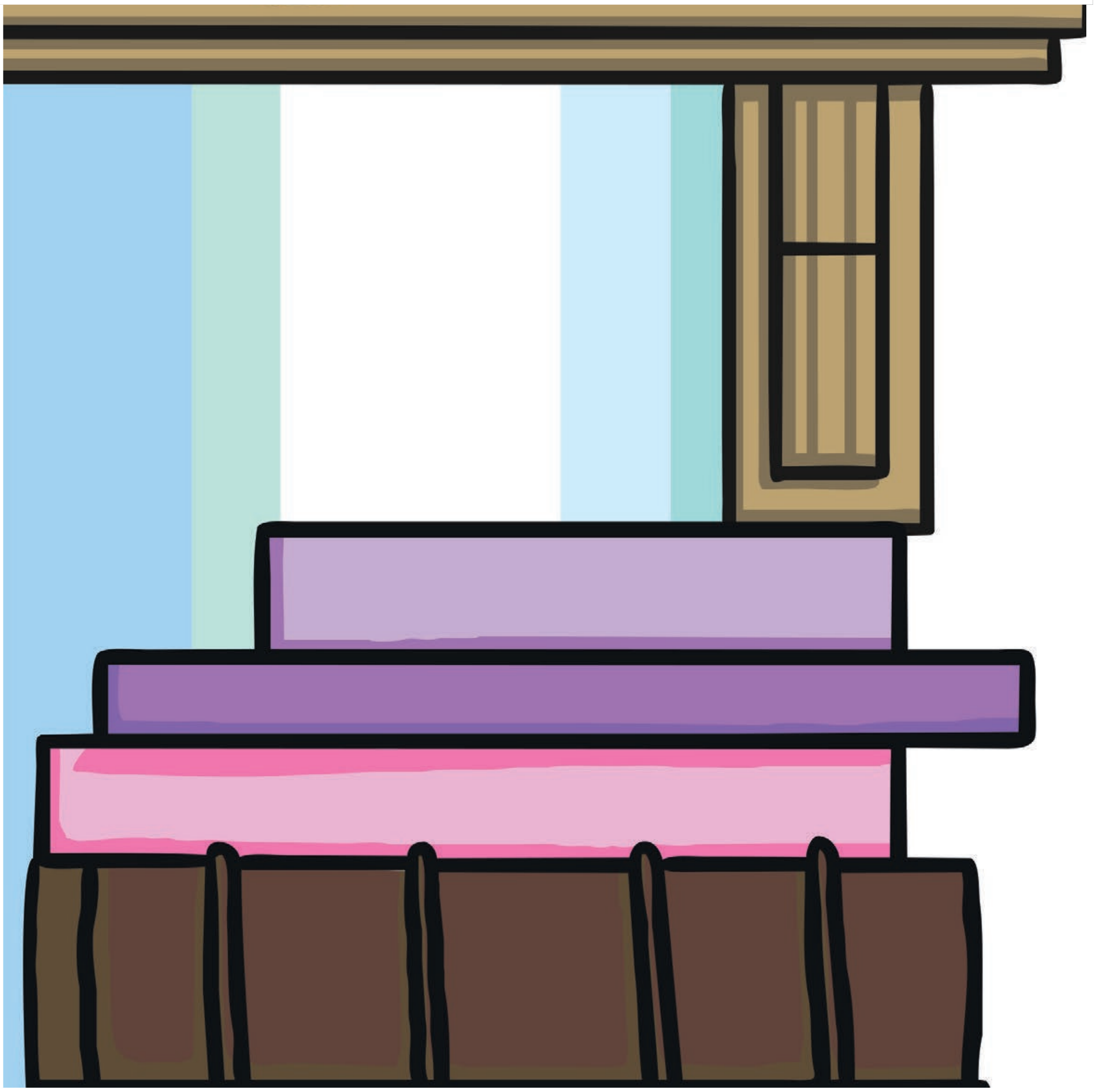
17



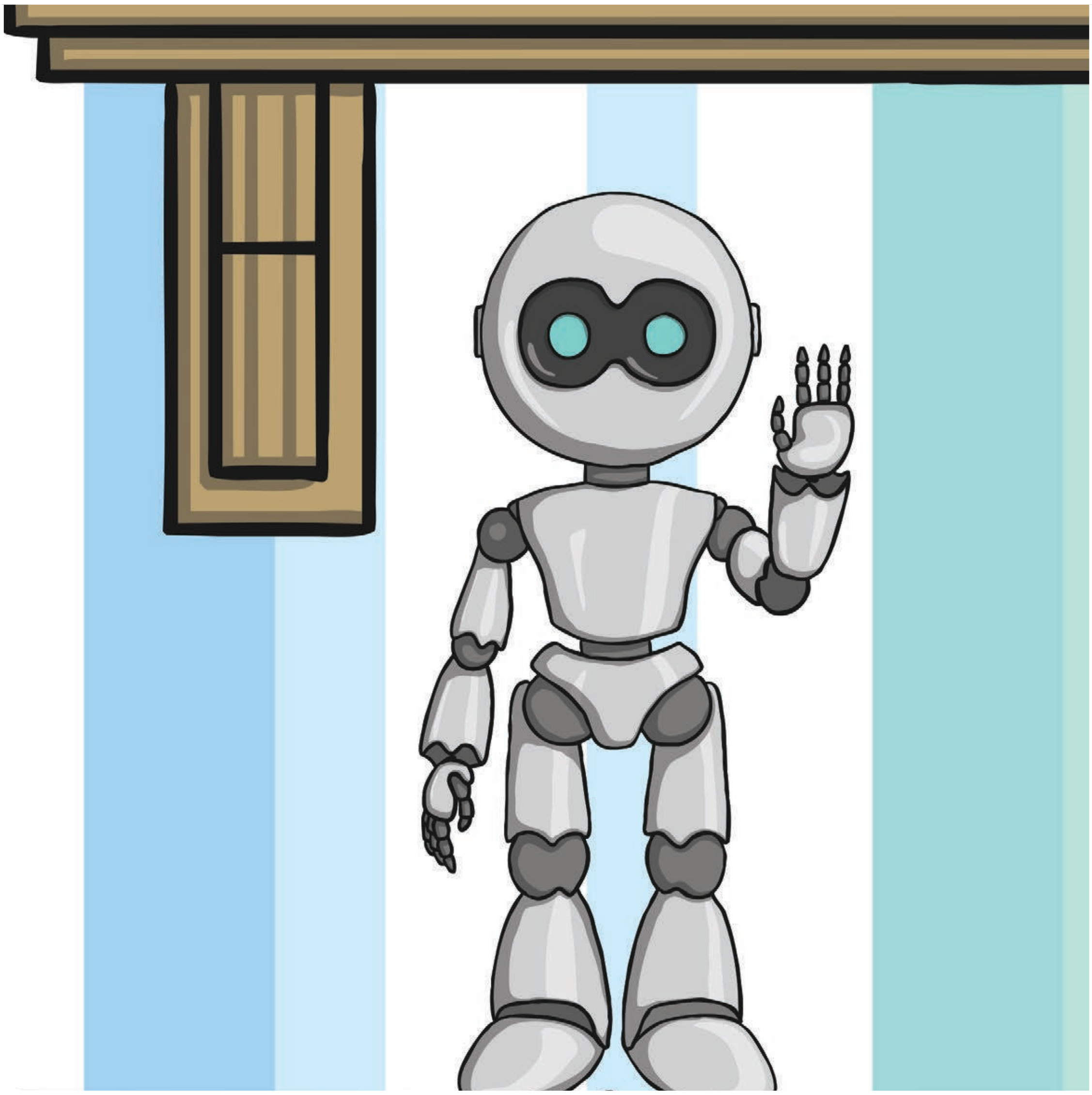
18



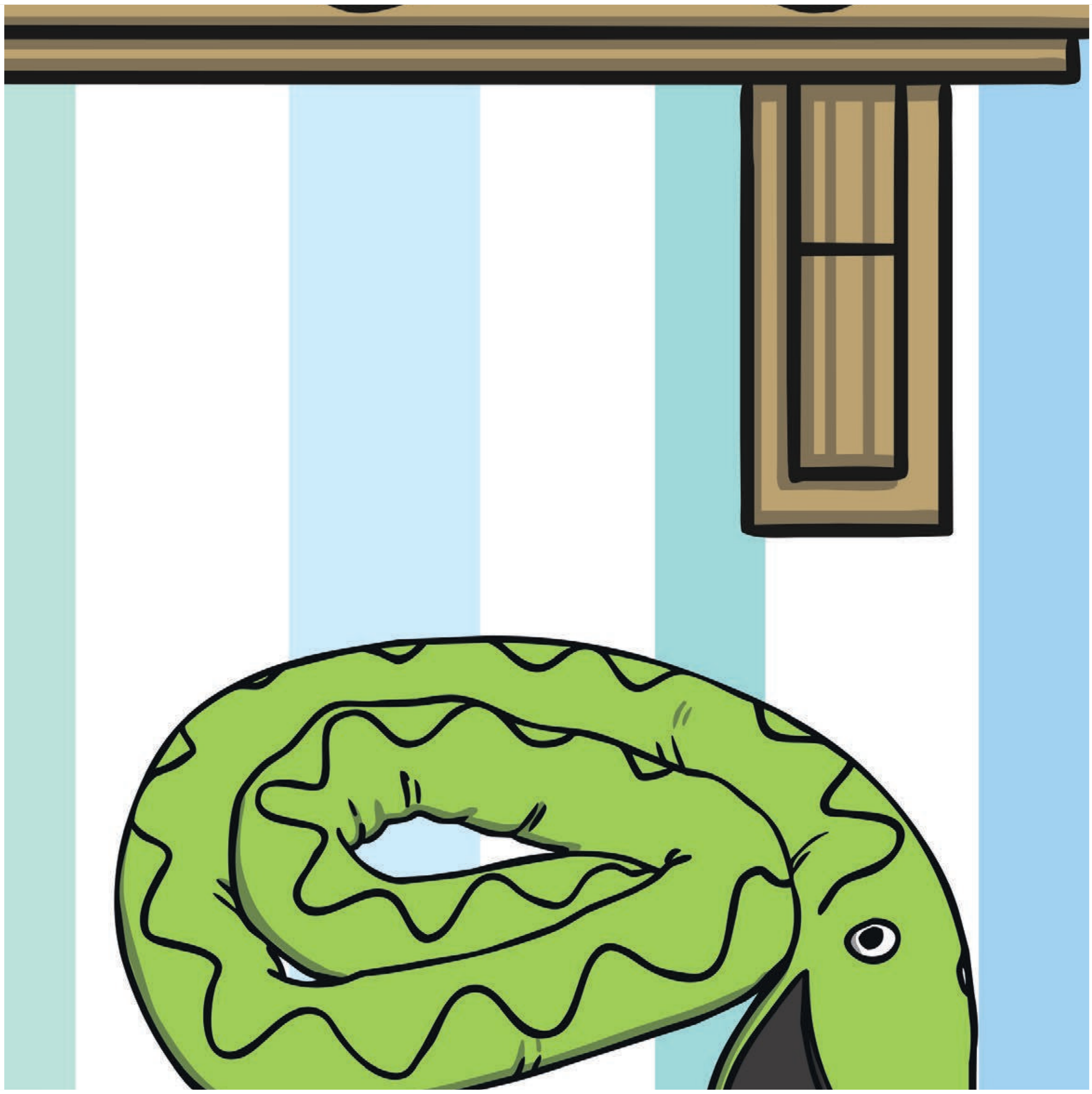
19



20



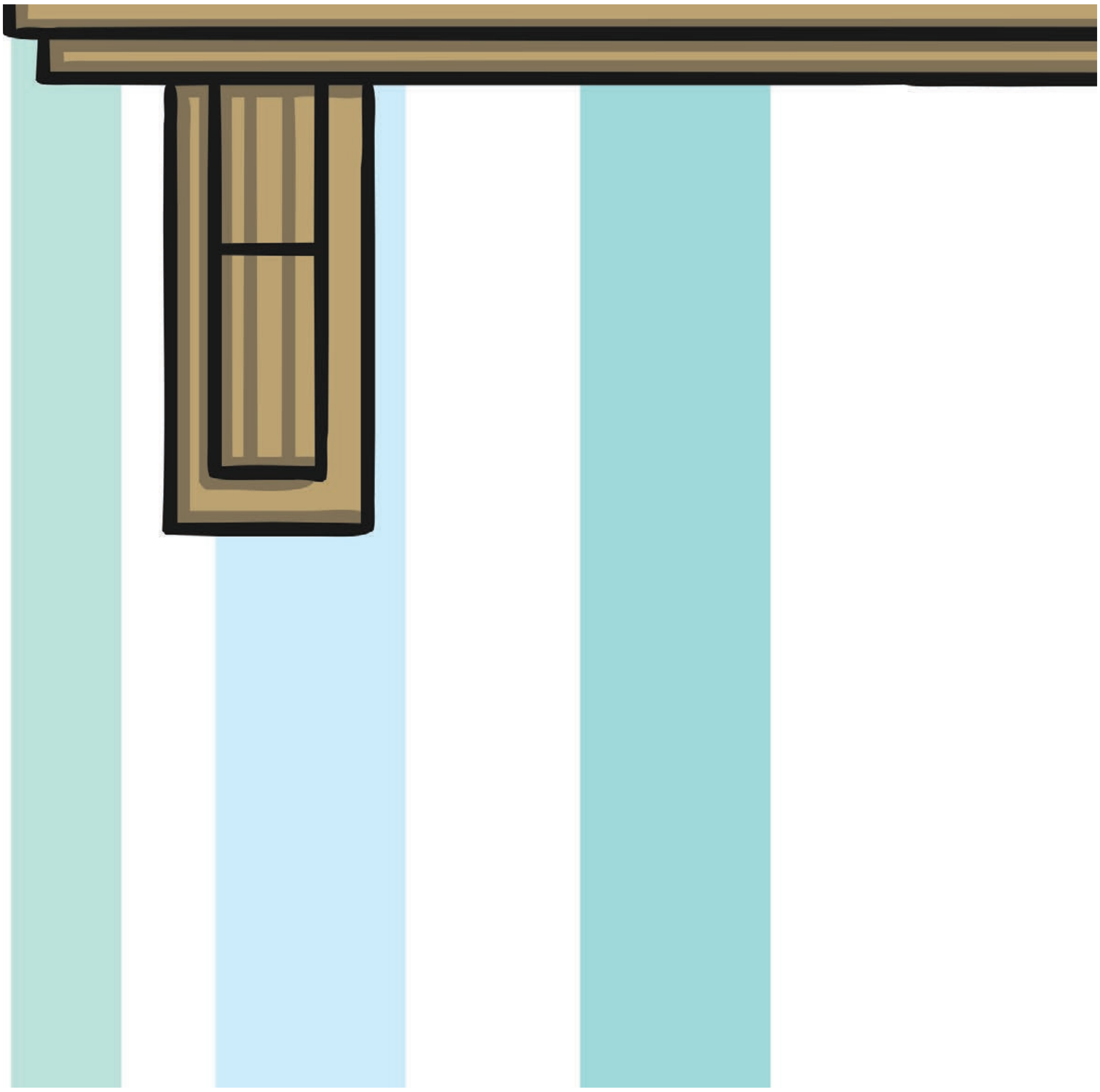
21



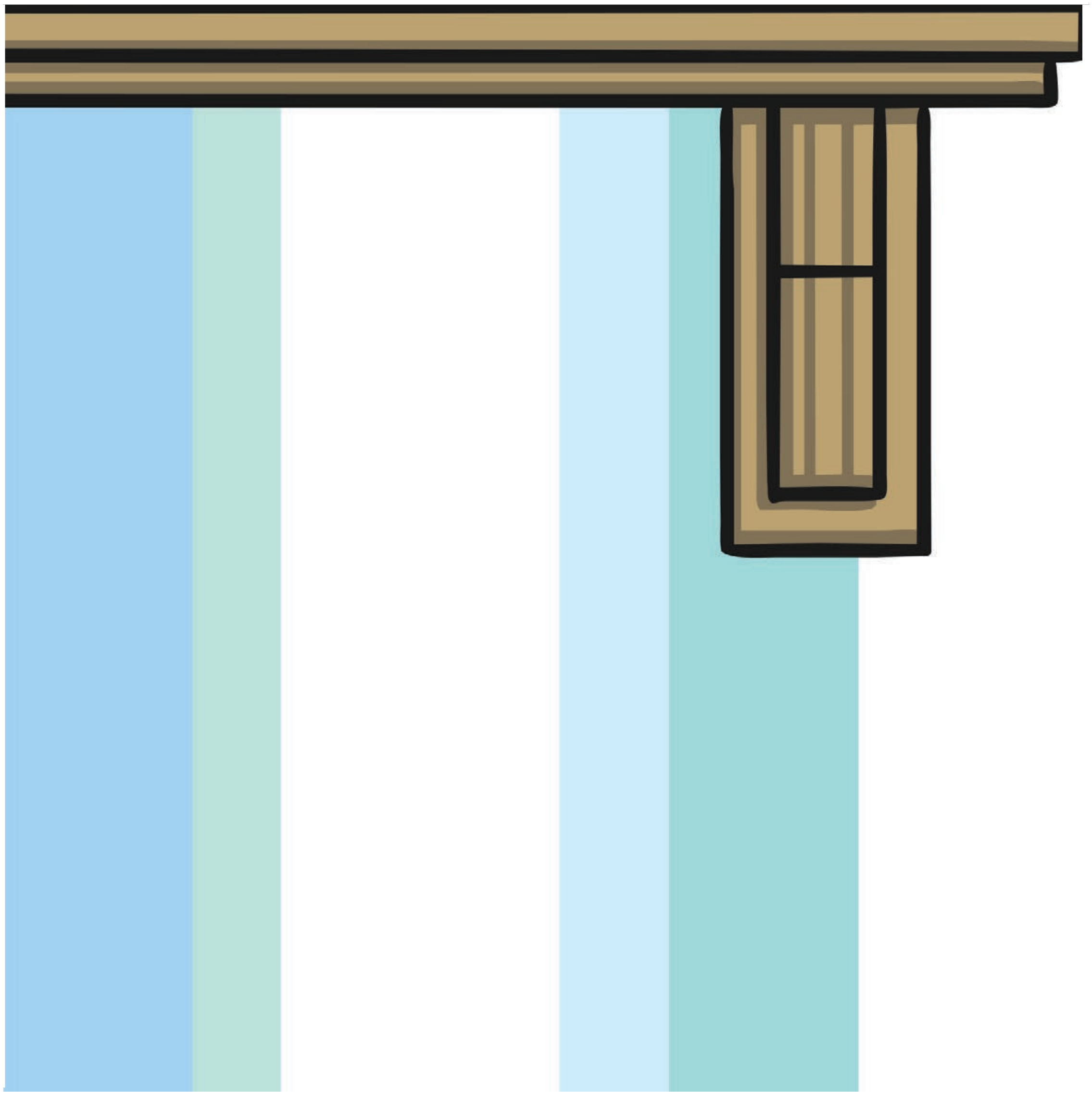
22

23

24

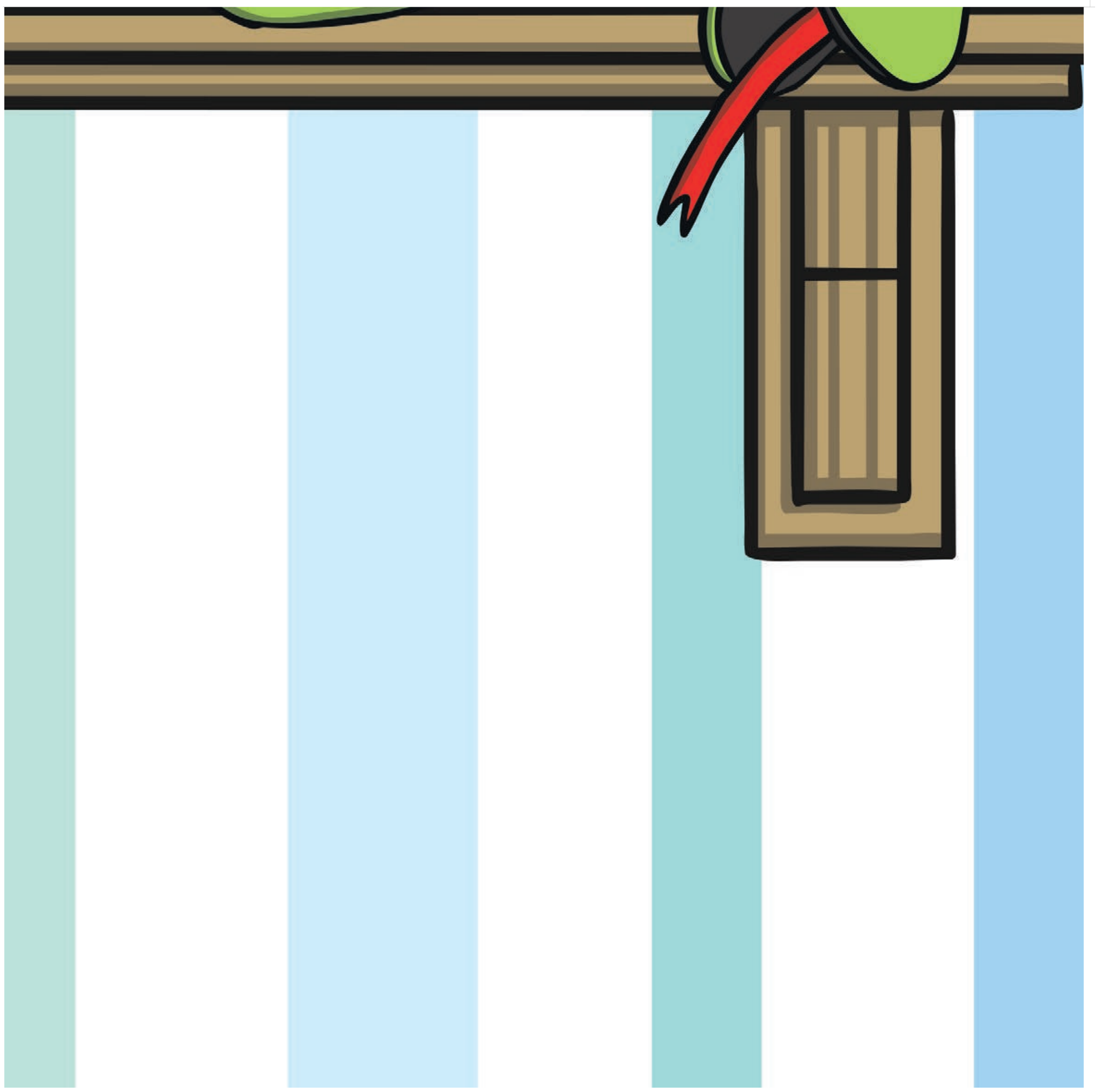


25

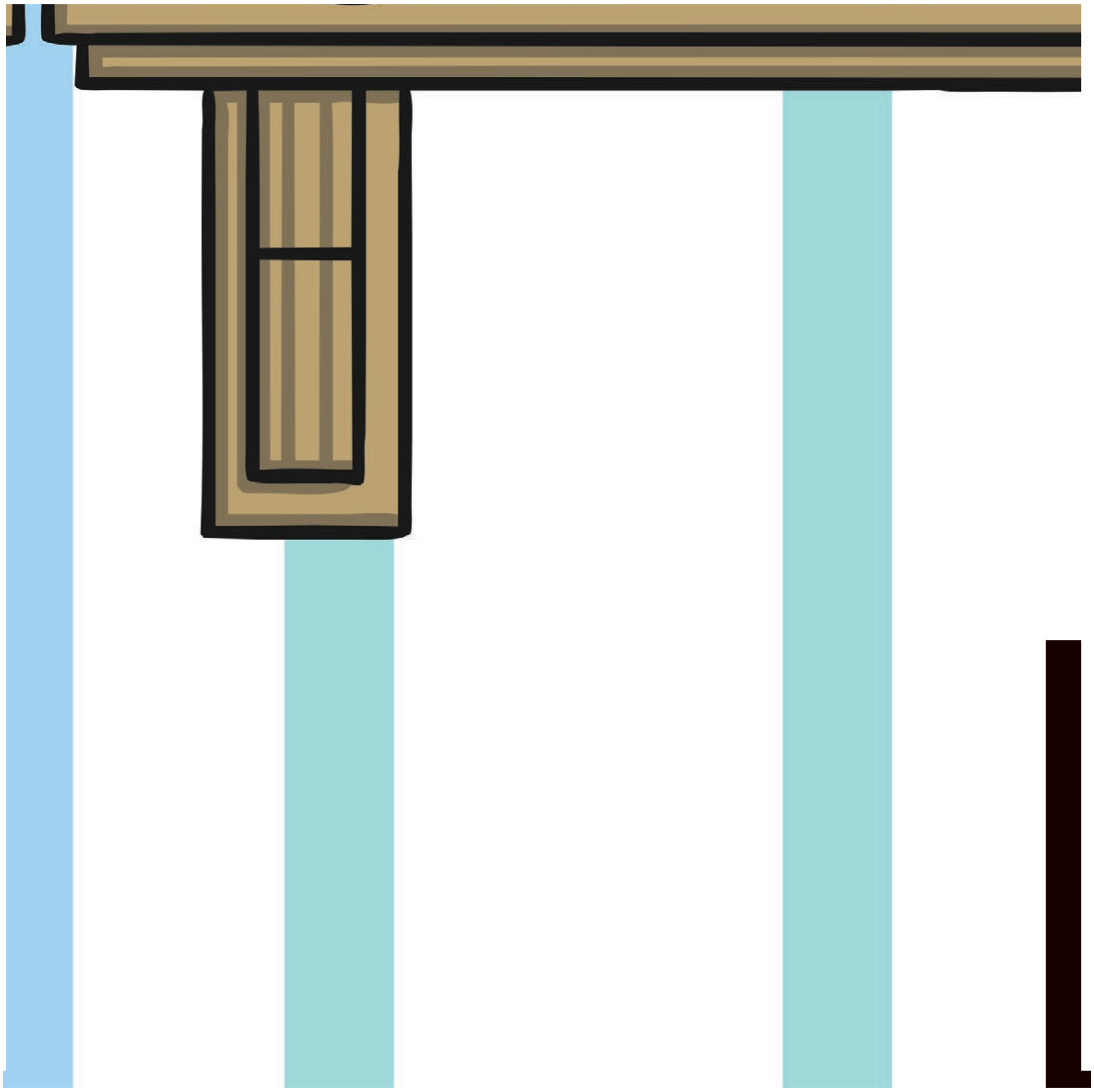


26

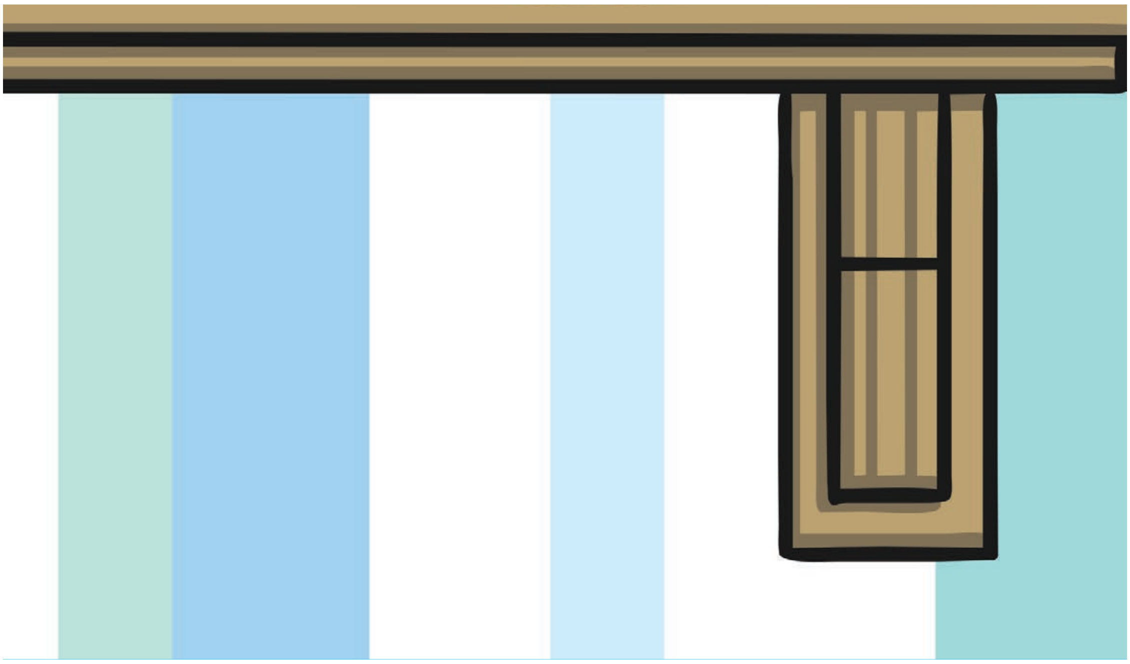
27



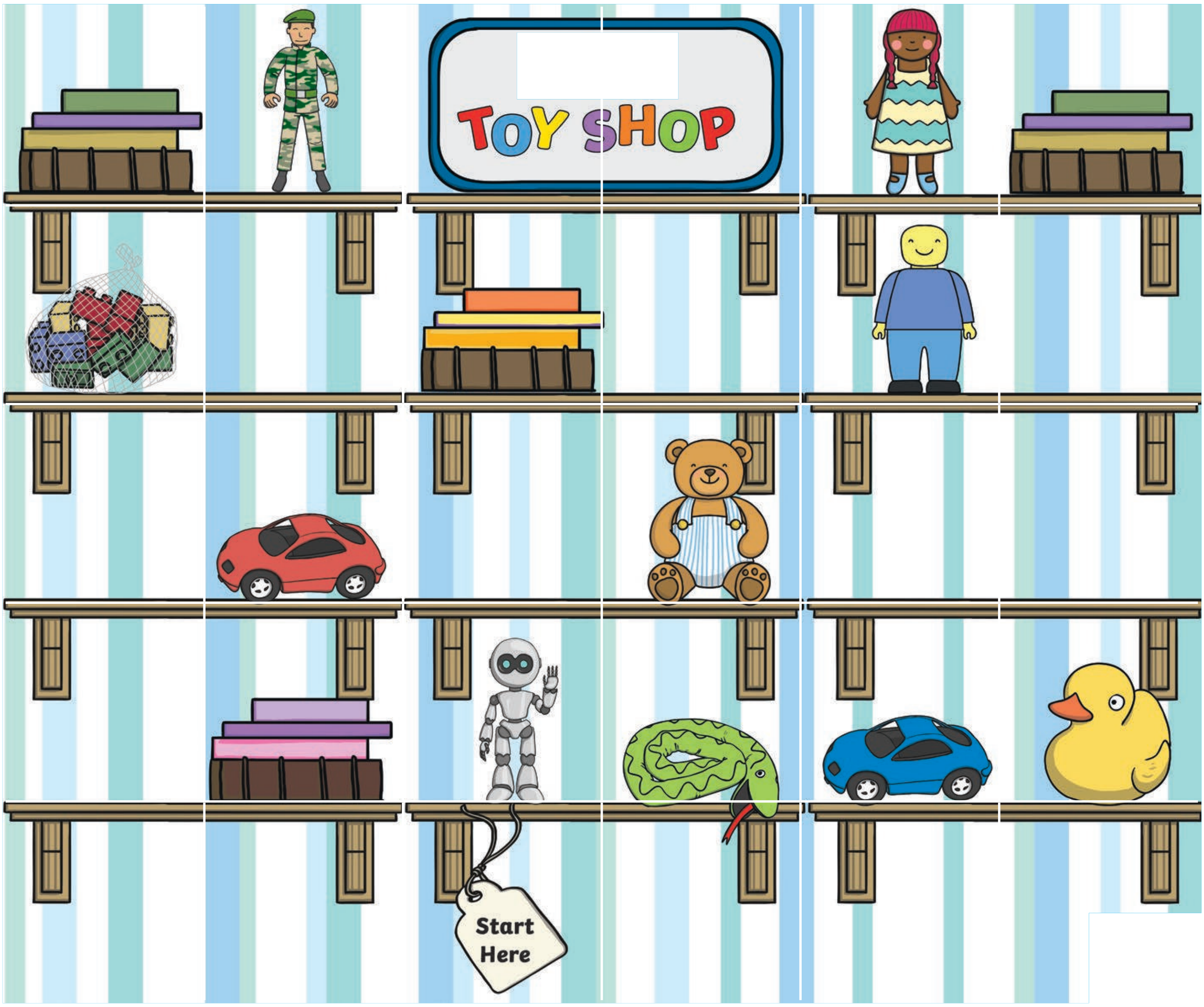
28

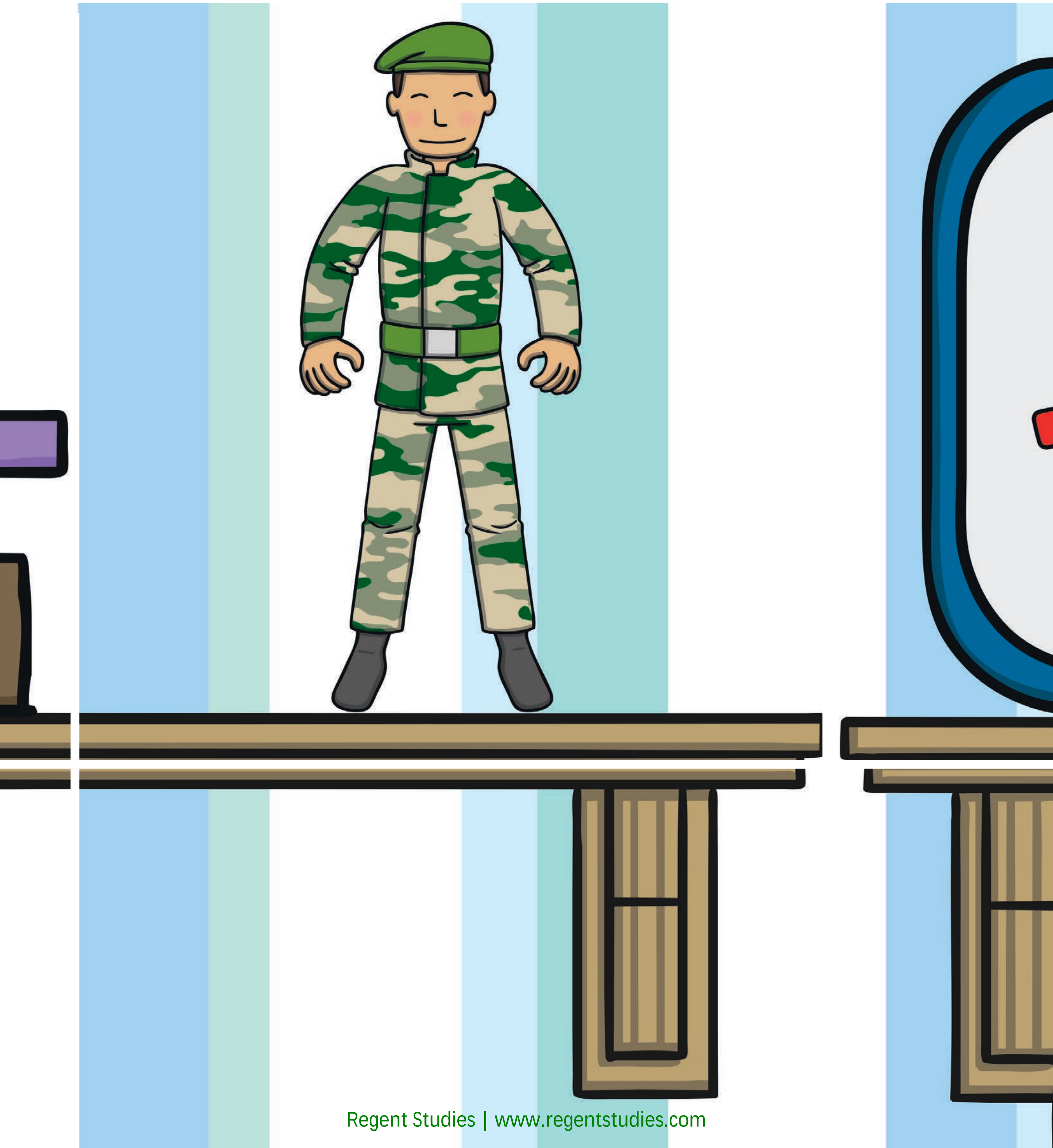


29

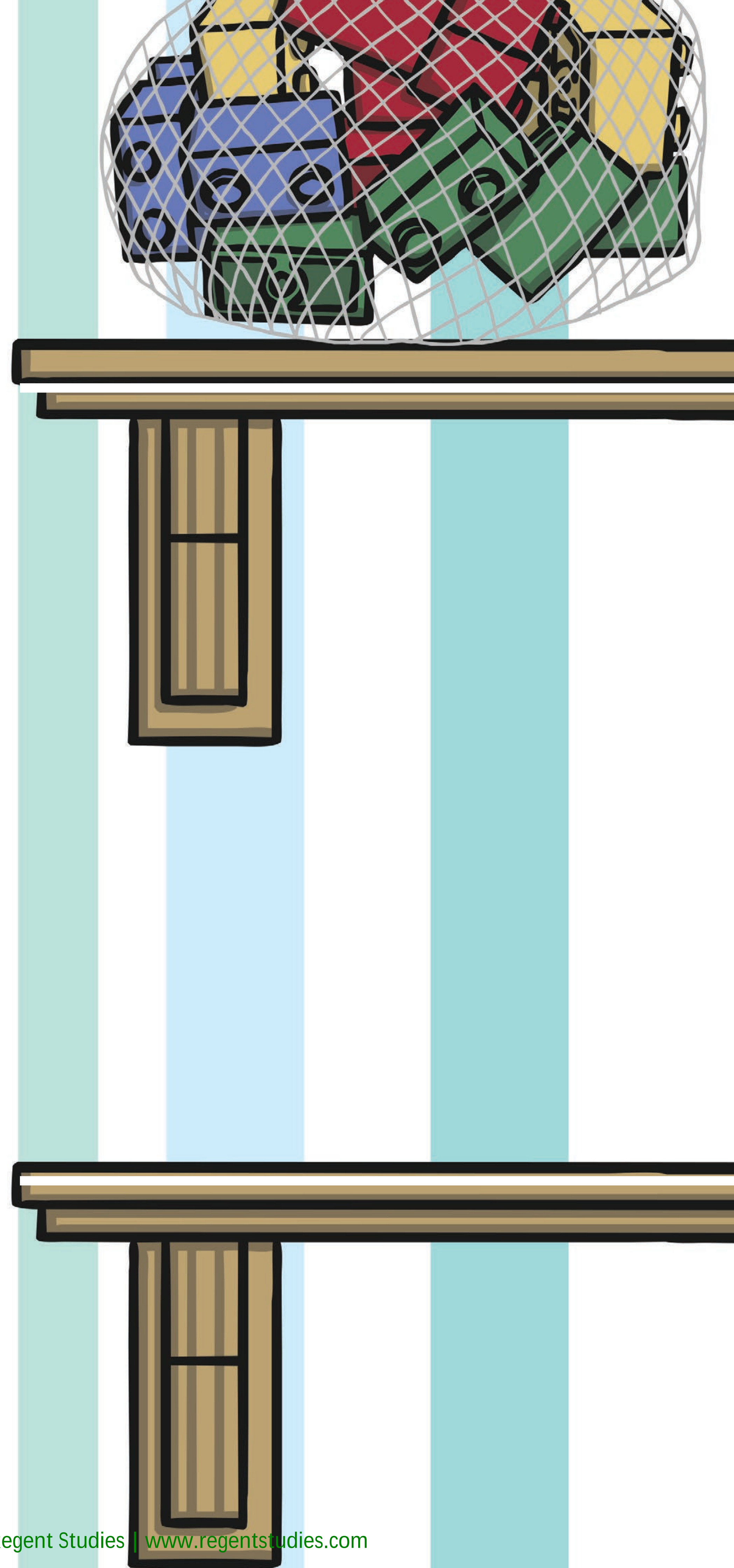


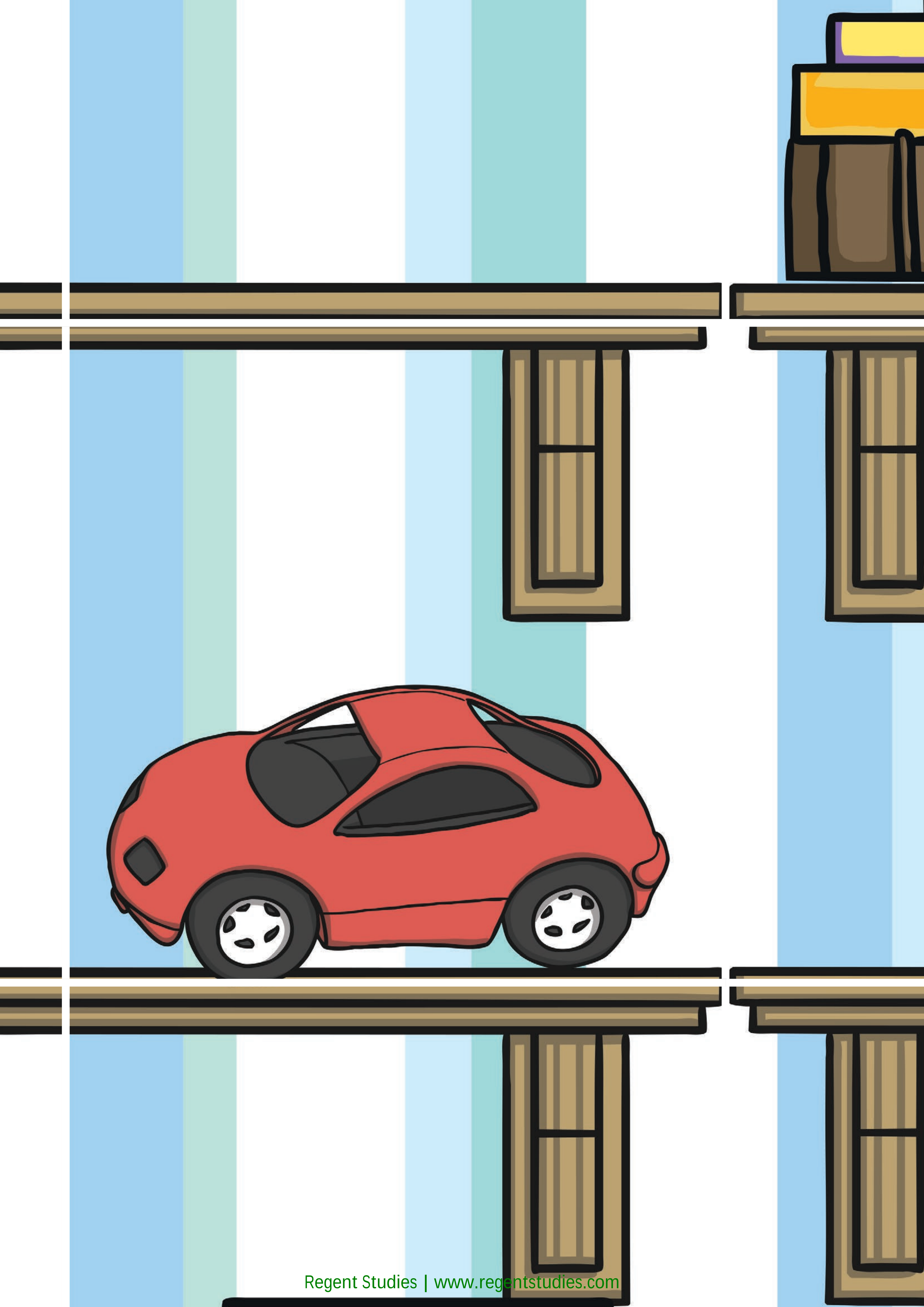
30

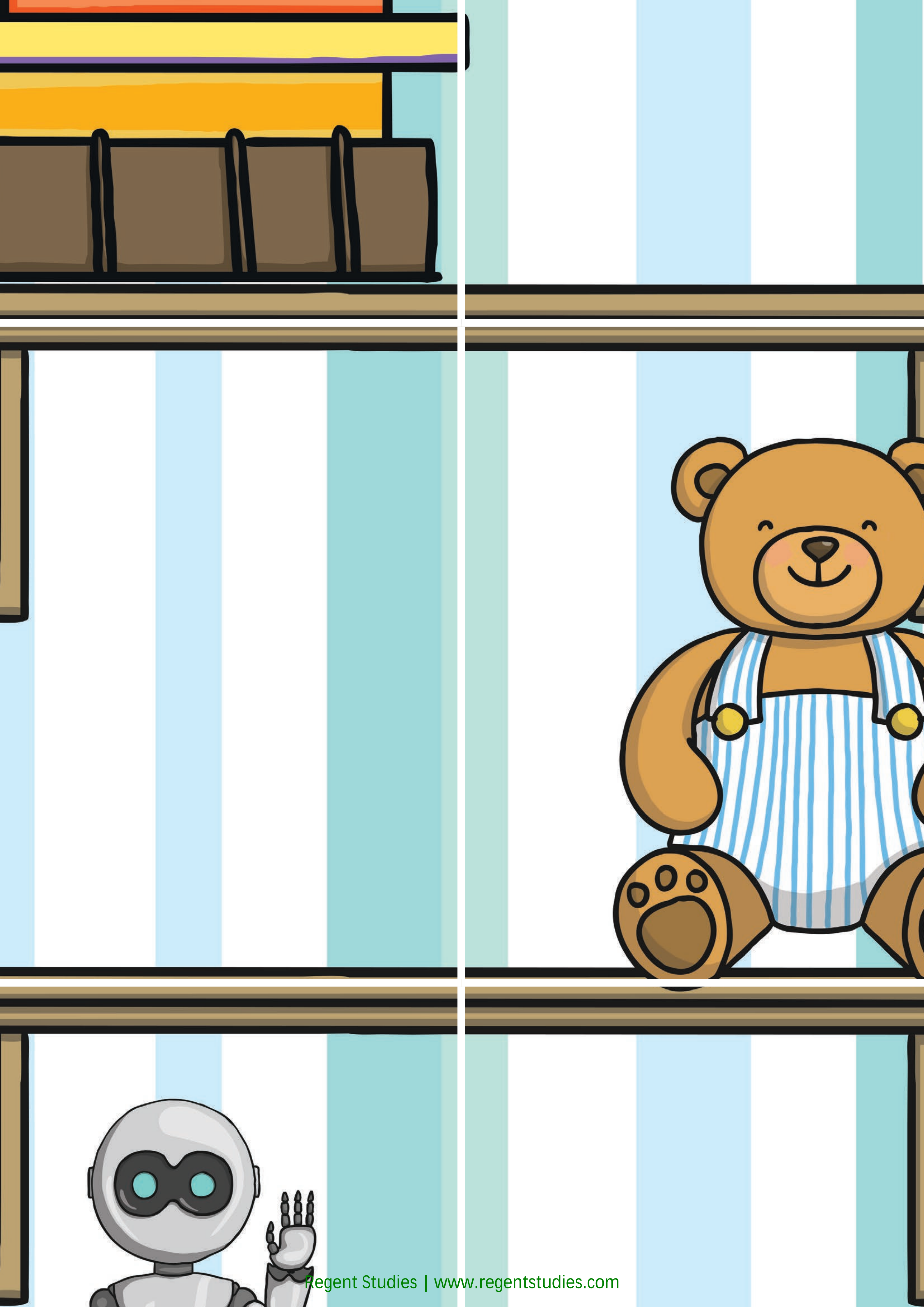

TOY SHOP
Start Here

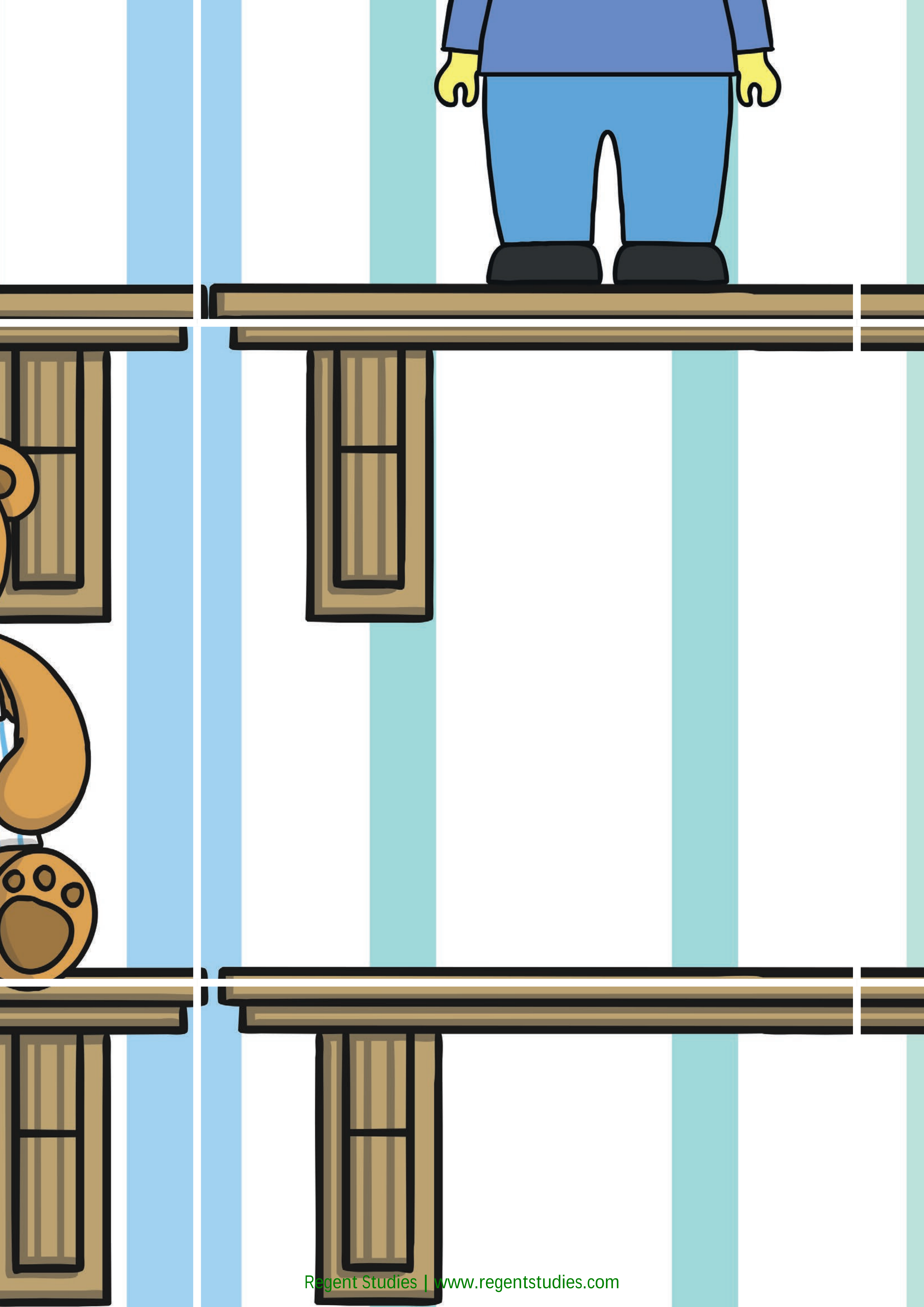


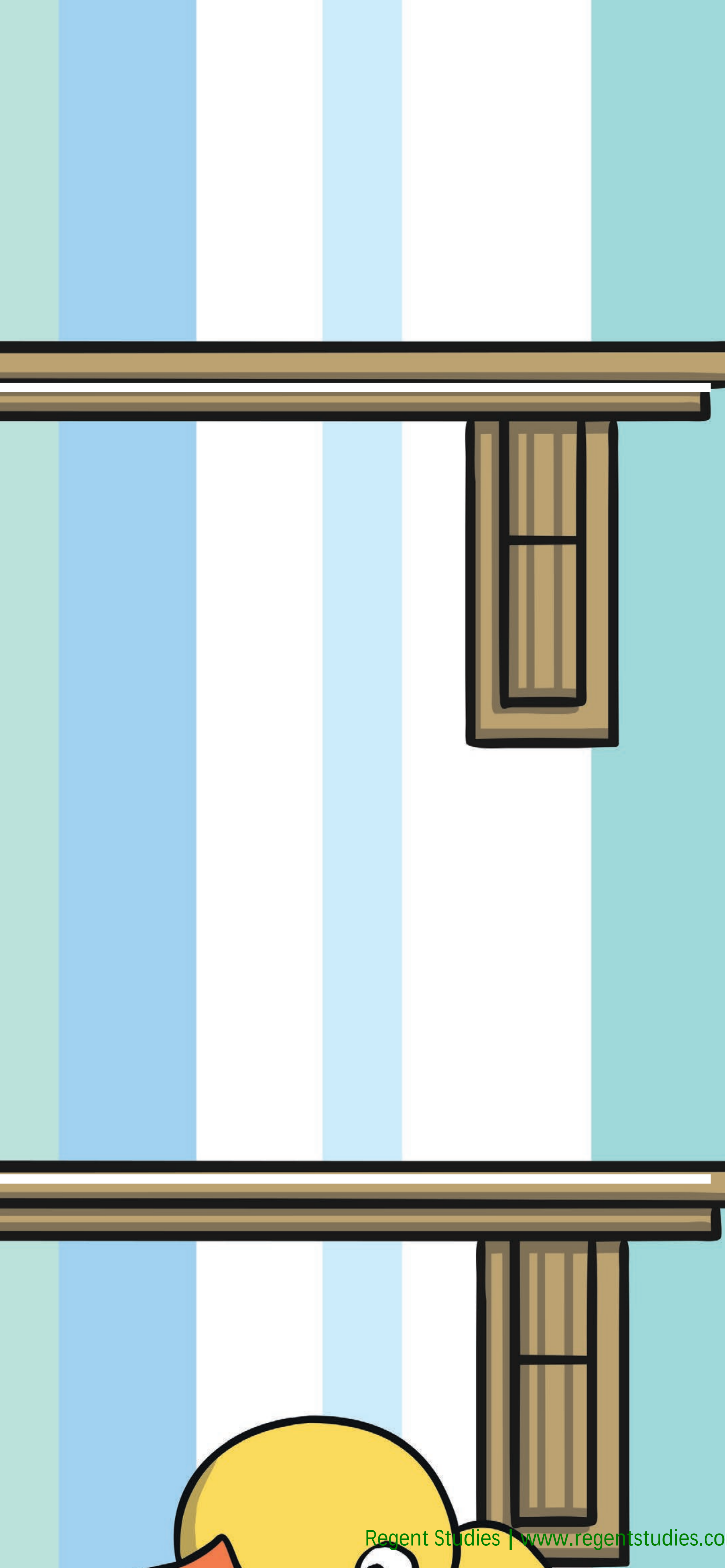

twinkl
TOY SHOP

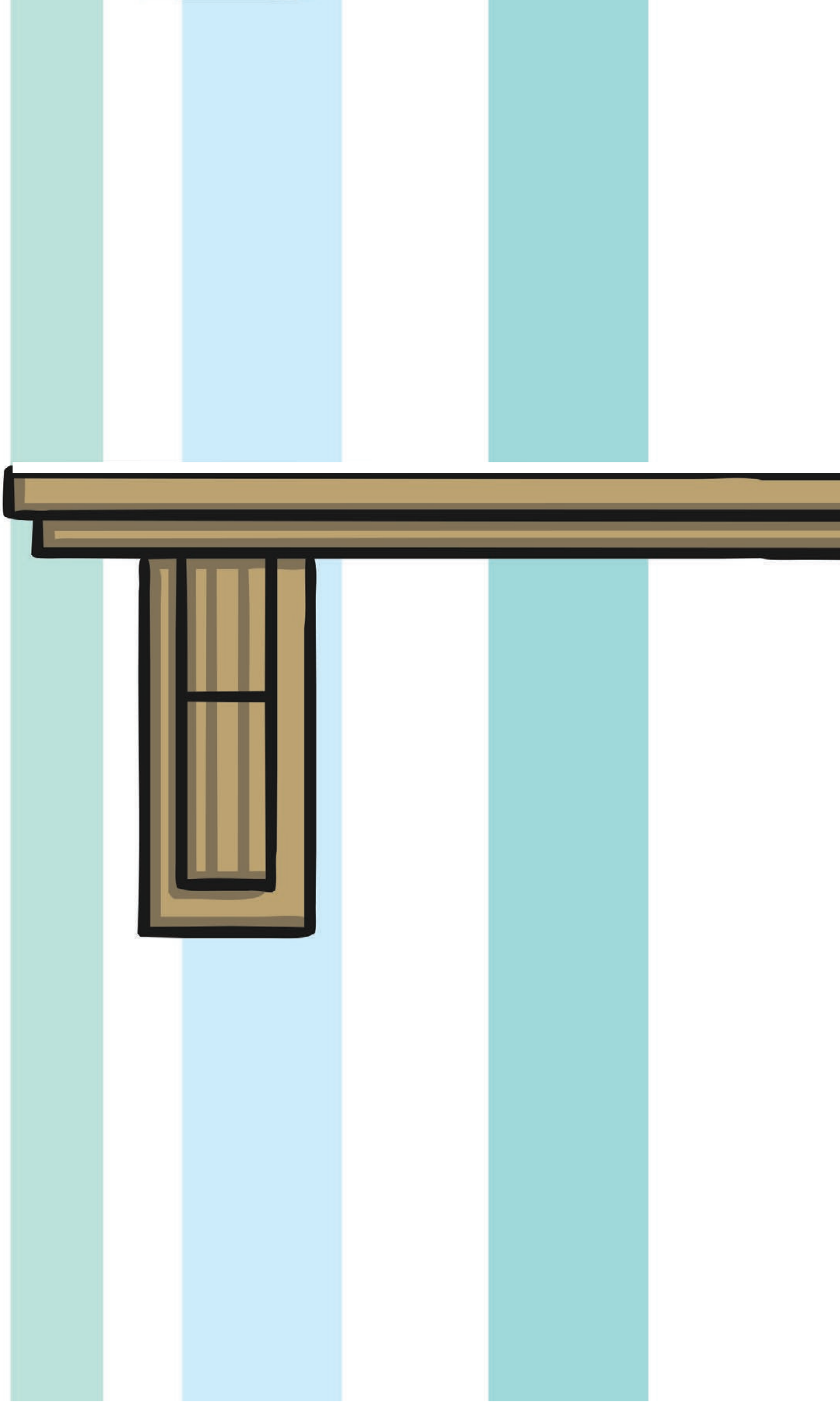


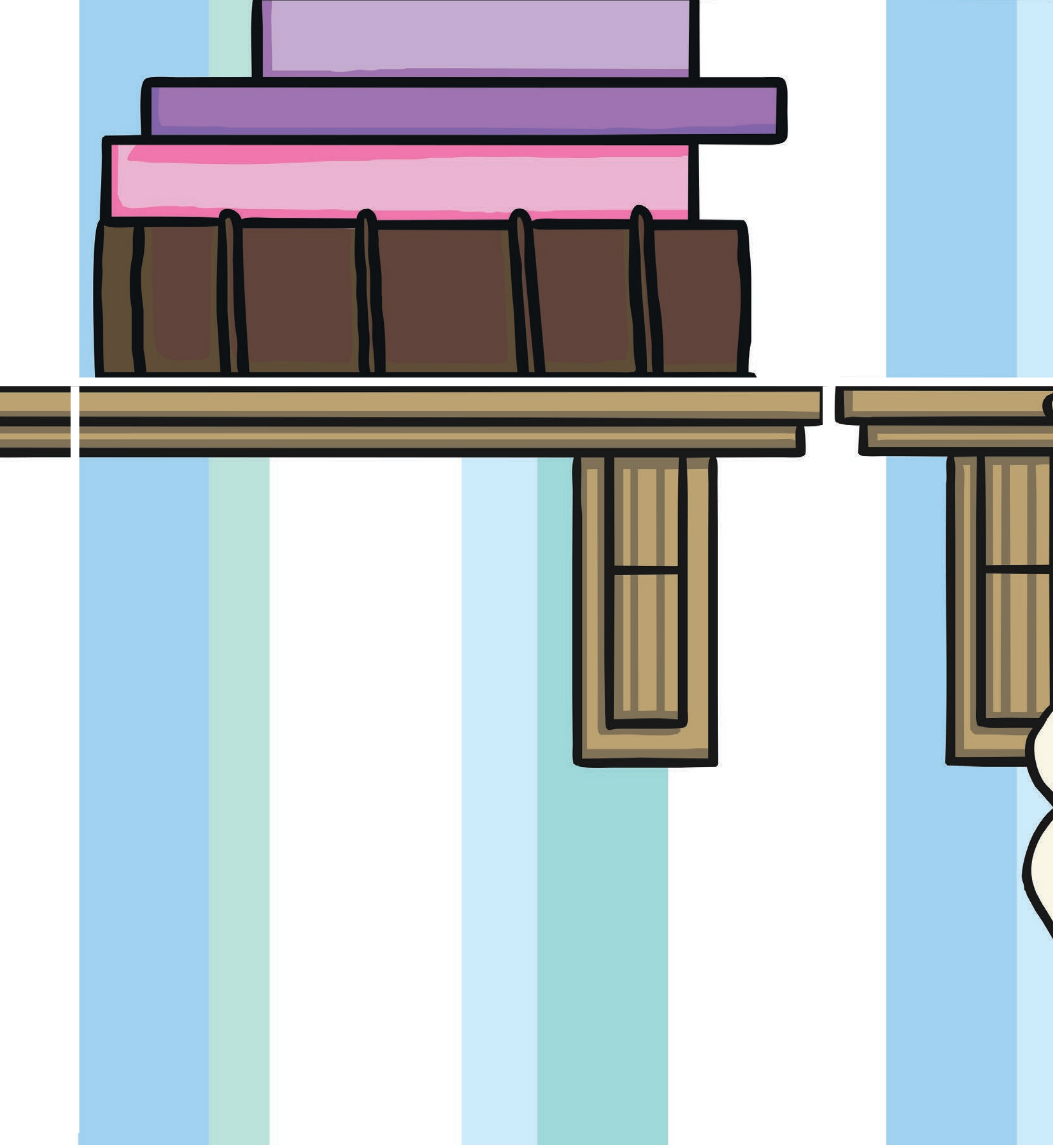


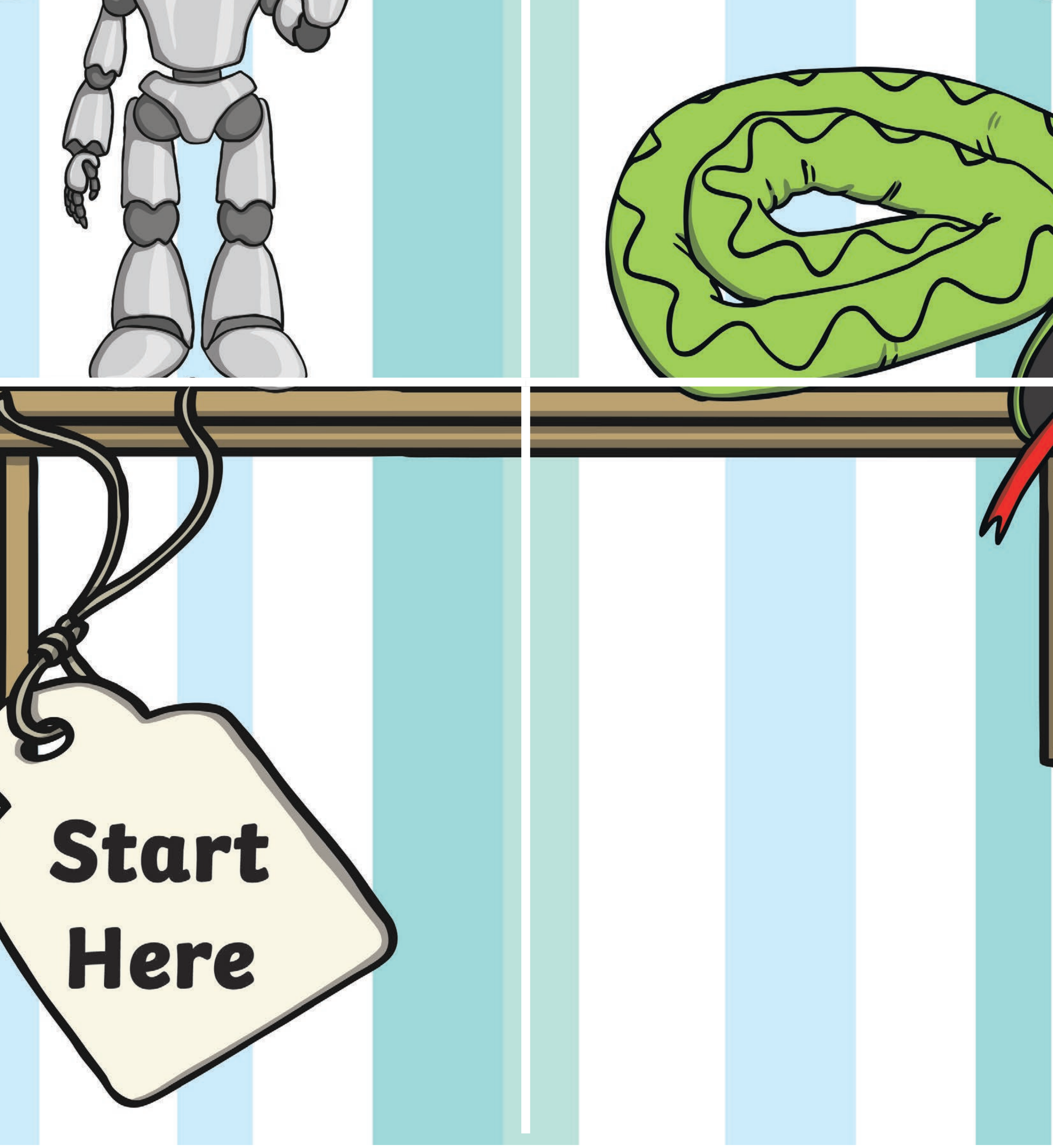


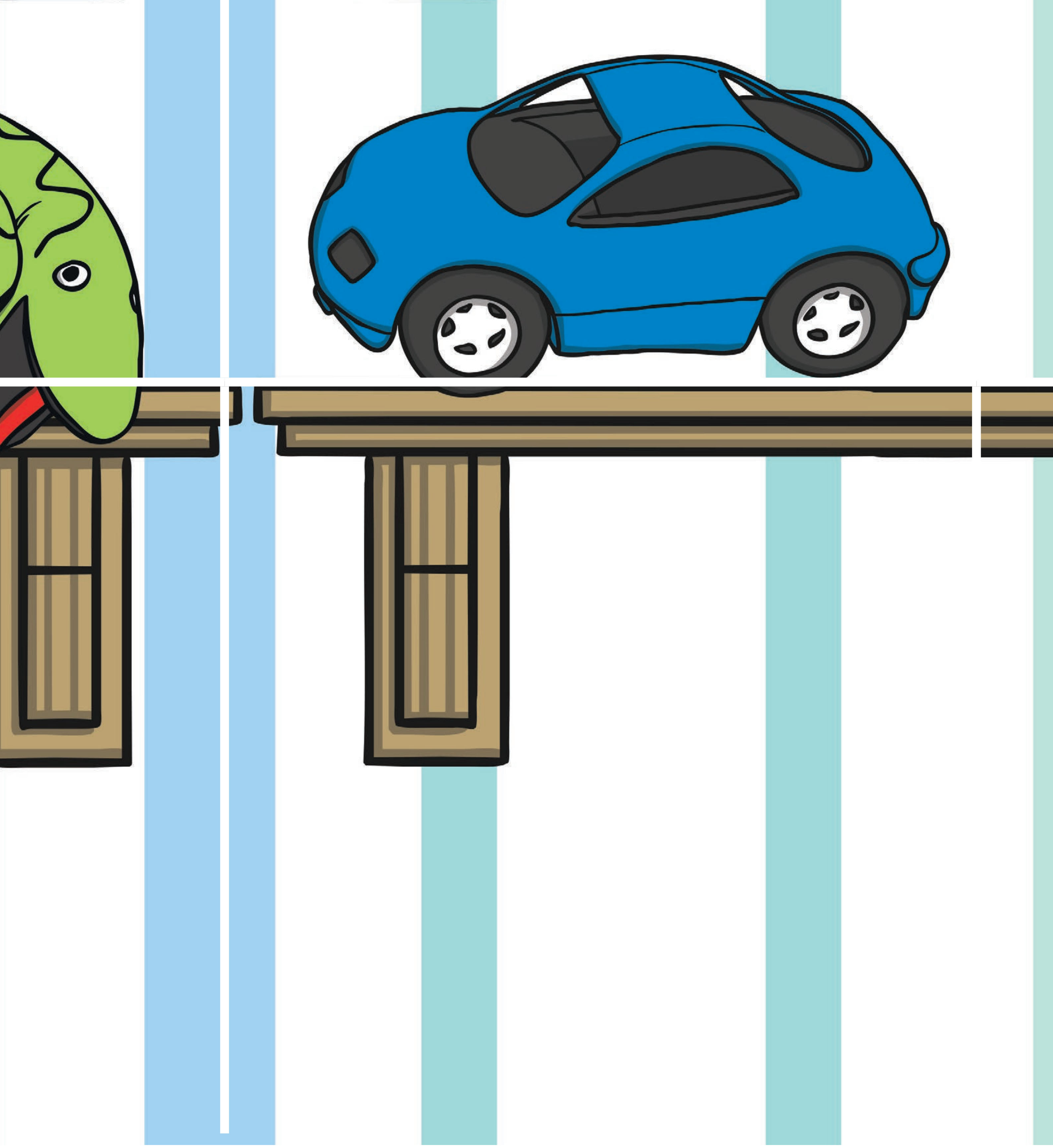


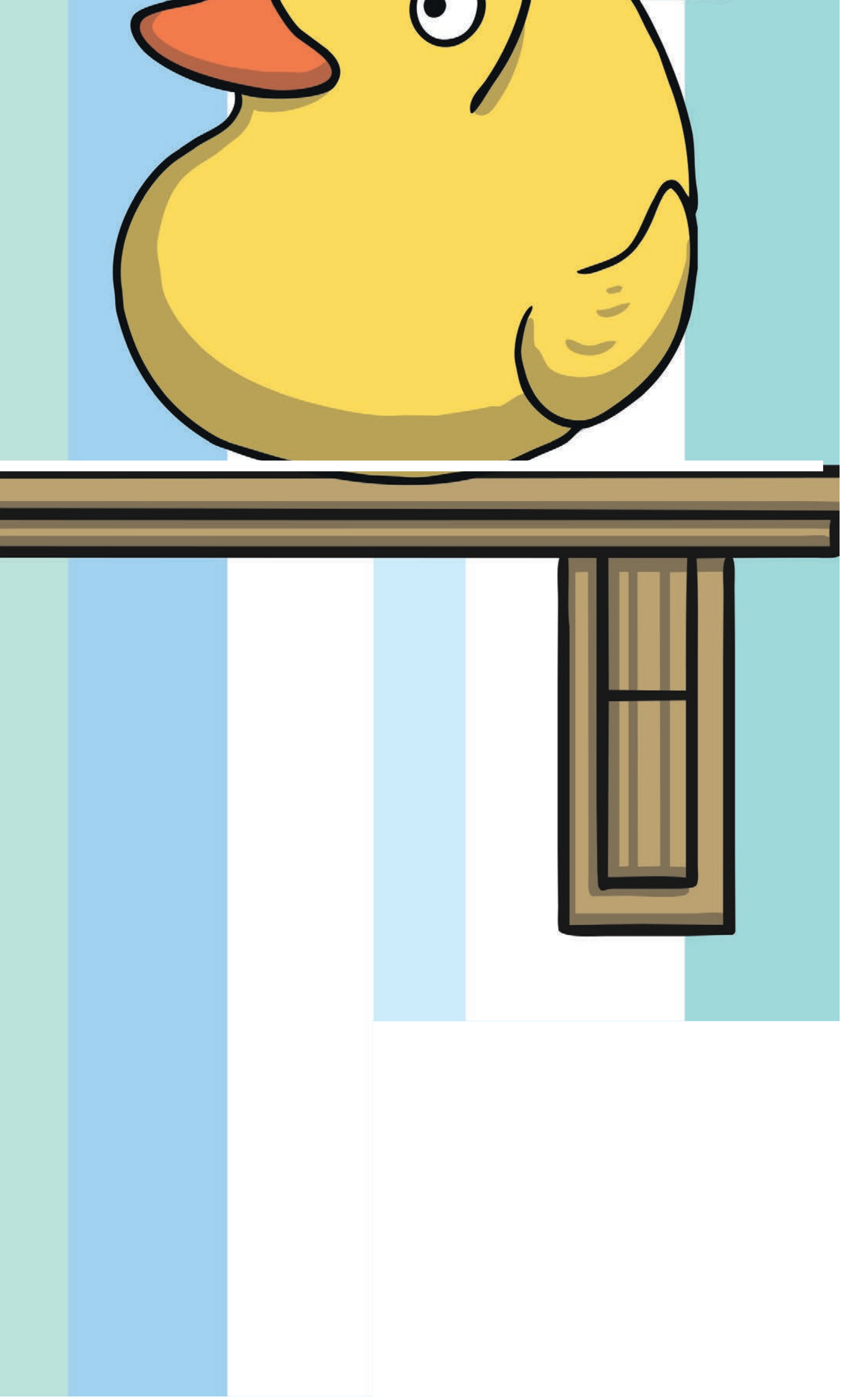







Start
Here





Avery Template: Name Badge Label, 8 per sheet | **Compatible Products**: 15395, 25395, 42395, 45395, 48395, 5395, 8395, 88395, 85395.

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys | Toy Shop Part 1

I can program a Bee-Bot (or similar programmable toy) to move.		
I can direct a Bee-Bot (or similar programmable toy) to a toy.		
I can program a Bee-Bot (or similar programmable toy) using the arrow buttons.		

Programming Toys: Toy Shop Part 2

Aim: Understand what algorithms are and that programs execute by following precise and unambiguous instructions. Create and debug simple programs. Children will work in the context of programming a Bee-Bot (or similar programmable toy) to reach set markers. I can program a sequence to make a Bee-Bot (or similar programmable toy) move.	**Success Criteria:** I can plan and check an algorithm. I can evaluate and improve my sequence (debug). **Key/New Words:** Algorithm, program, debug, sequence.	**Resources:** **Lesson Pack** Bee-Bots (or similar programmable toy) 3-4 toys Whiteboards and pens **Preparation:** **Differentiated Shopping List Activity Sheets** - 1 per child **Toy Shop Mat** - 1 per pair or group

Prior Learning: Children will have learned how to program a Bee-Bot (or similar programmable toy) in lessons 4 and 5 of this unit.

Learning Sequence

Whole Class	**Toy Shop:** Use the **Lesson Presentation** to help the children to plan an algorithm on their whiteboards. Remind them to press 'Clear' and 'Go'. Can the children plan an algorithm before pressing 'Go'?	
	Shopping List: Sit the class in a circle. Use the list on the **Lesson Presentation** and ask children to help you program the route around the toys. Children should use whiteboards to practise drawing a route around the Toy Shop grid, then amend their algorithm to get the Bee-Bot (or similar programmable toy) back to the start.	
	Go Shopping: Children to write an arrow algorithm which would direct the Bee-Bot (or similar programmable toy) through the list on their **Differentiated Shopping List Activity Sheet**. ★ Children direct the Bee-Bot (or similar programmable toy) around the mat to collect the 4 toys pictured. ★★ Children direct the Bee-Bot (or similar programmable toy) around the mat to collect 5 toys, including reversing to pick up the rubber duck. ★★★ Children direct the Bee-Bot (or similar programmable toy) around the mat to collect 5 toys, avoiding obstacles. They then use the backwards button where possible to shorten the sequence.	
	Try It Out: Once they have completed their algorithm, in pairs, children should program the Bee-Bot (or similar programmable toy) to see if it works. If it hasn't worked, can the pairs identify why and debug the sequence?	
Whole Class	**What Did I Buy?** Show the children the algorithm on the **Lesson Presentation**. Can they work out what your Bee-Bot (or similar programmable toy) picked up at the shop?	

Taskit

Listit: Children write a list, in groups, of toys that they think are robots. What kind of algorithms do they think makes it work? (Simpler wording – What instructions has that robot toy been given?) Write them down.

Playit: In a hall or on the playground, children can free play the activities in lessons 1 or 6 on a large scale: either programming each other, or a Bee-Bot (or similar programmable toy) to reach a real toy in a larger area, or to follow a chalk drawn pattern outside.

Computing

Programming Toys


Toy Shop
Part 2

Aim

- I can program a sequence to make a Bee-Bot (or similar programmable toy) move.

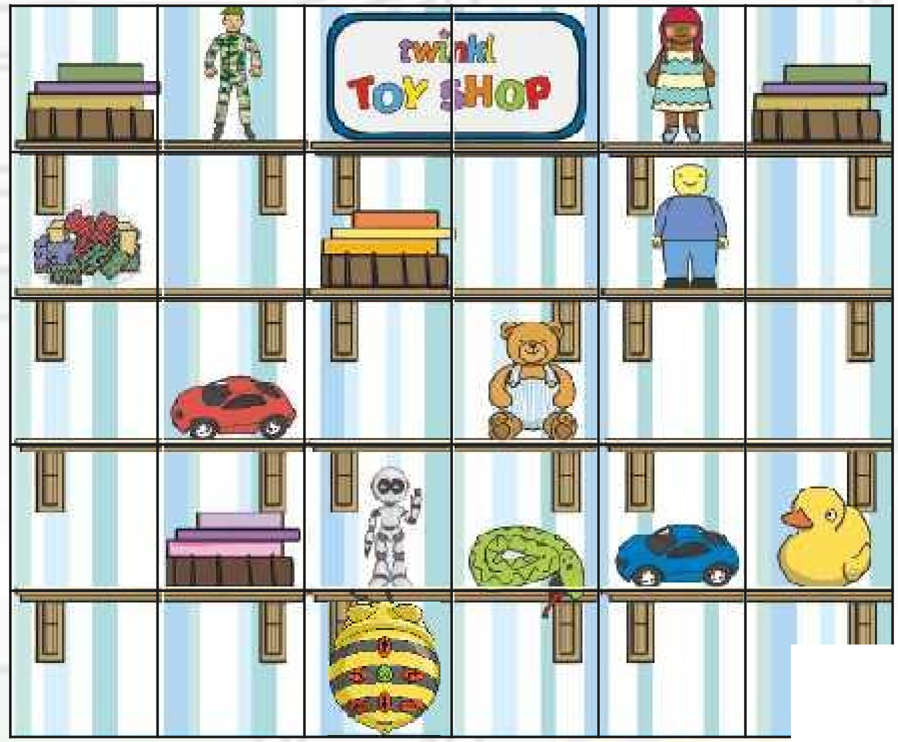
Success Criteria

- I can plan and check an algorithm.
- I can evaluate and improve my sequence.

Toy Shop

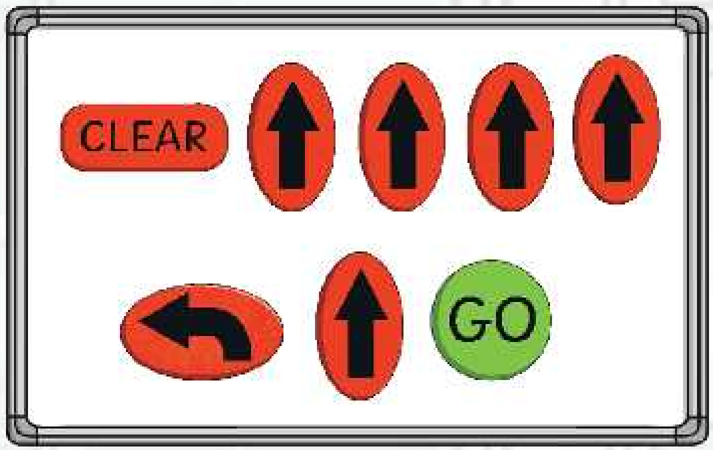
Welcome back to the Twinkl Toy Shop!

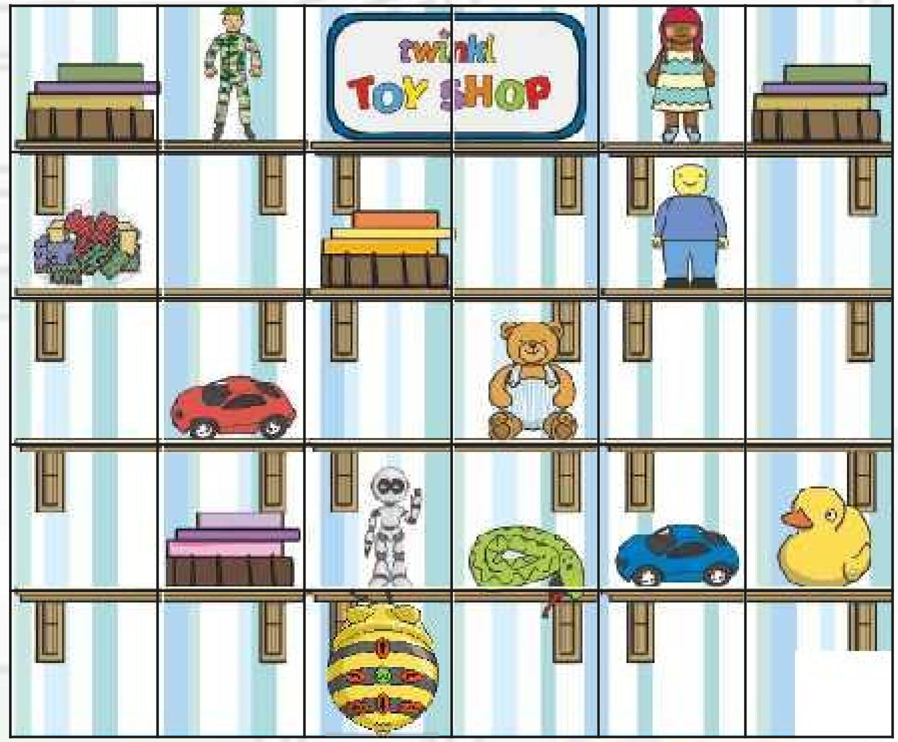
Can you draw on your whiteboard the buttons that you would need to get the programmable toy to the **soldier**?



Bee-Bot at the Toy Shop

Did you draw these instructions?





Did you remember to press CLEAR and GO ?

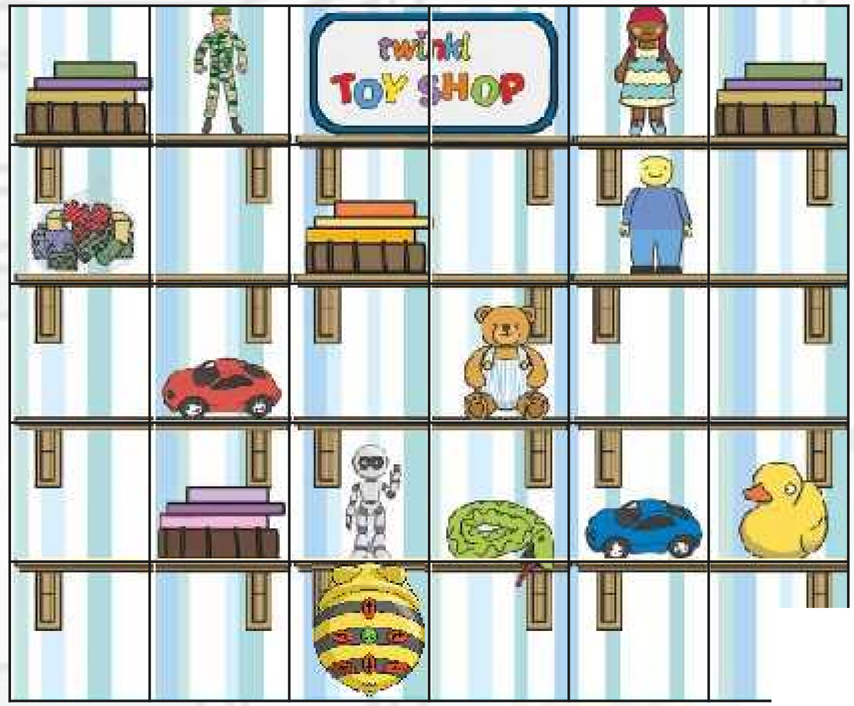
Shopping List

I want to buy:

- a blue car
- a brick man
- a red car
- a toy soldier

I want to get them in this order.

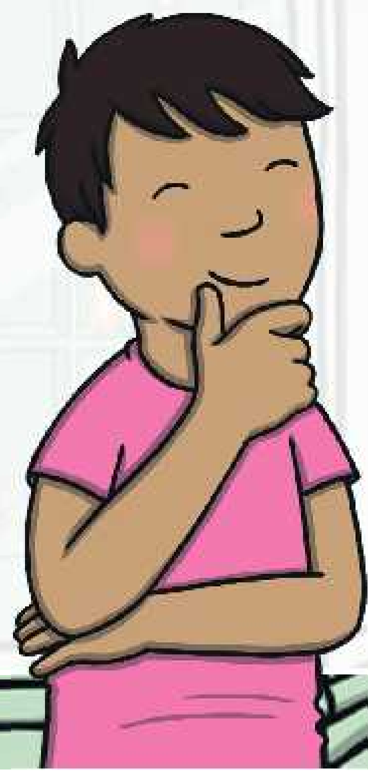
What's the best route for my programmable toy to take?

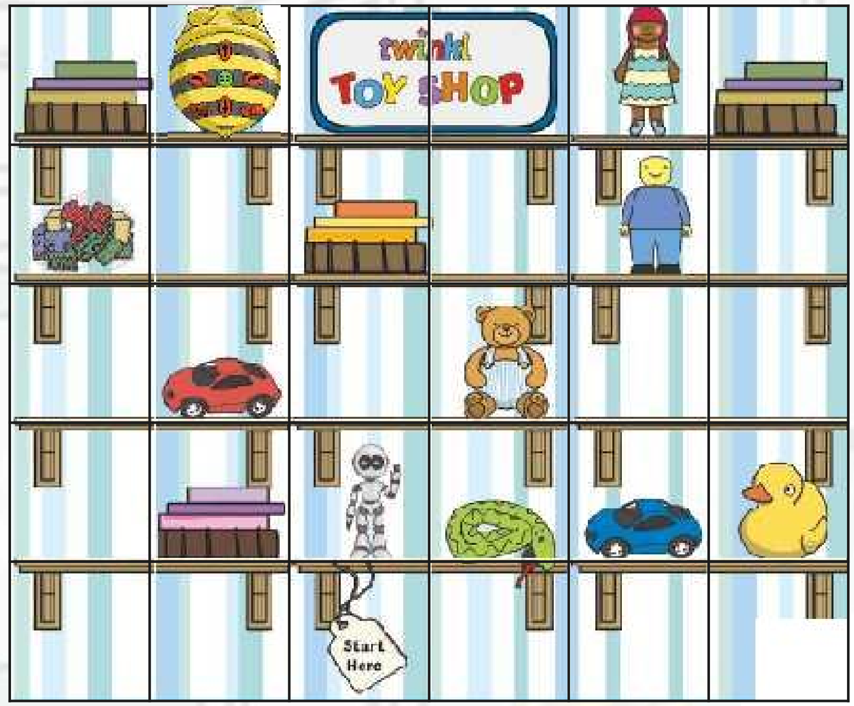


Remember, you can use the button to go back a square.

Shopping List

Now I want to go home.
Can you get the
programmable toy back to
the starting square?

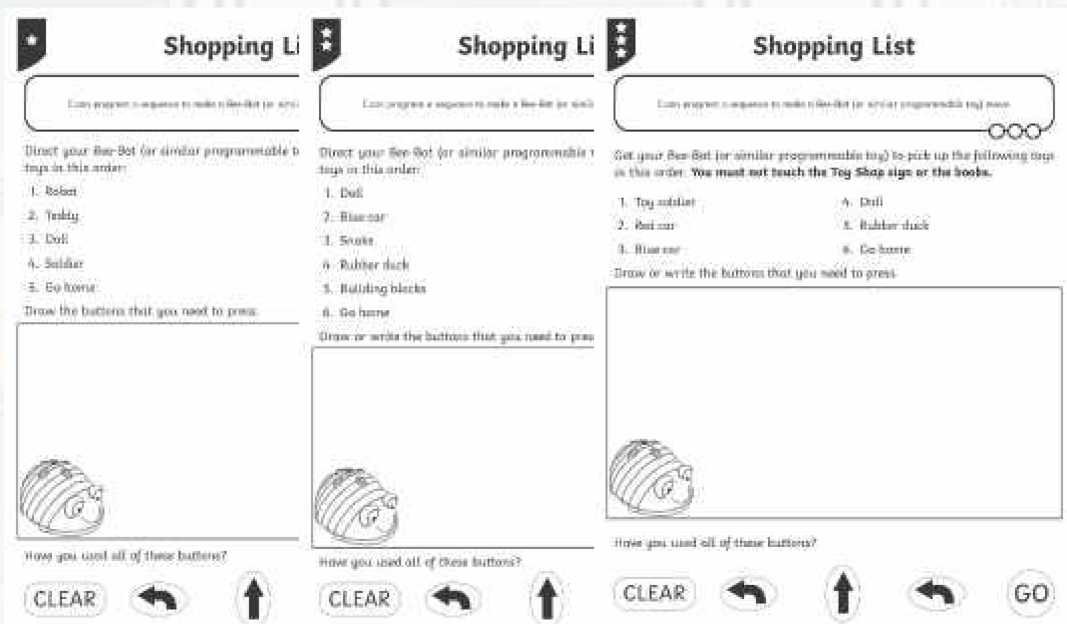




Go Shopping

Now it's time to send your Bee-Bot shopping! Use your Shopping List to direct it around the mat.

Write your algorithm down and check it carefully before you press GO.



Try It Out

Help your partner to program the Bee-Bot (or similar programmable toy) using their algorithm.

Does it work? What **bugs** does it have?

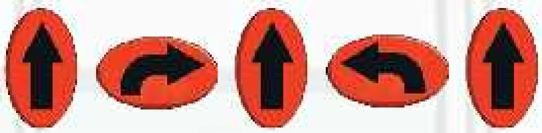
If not, help them to **debug** it!

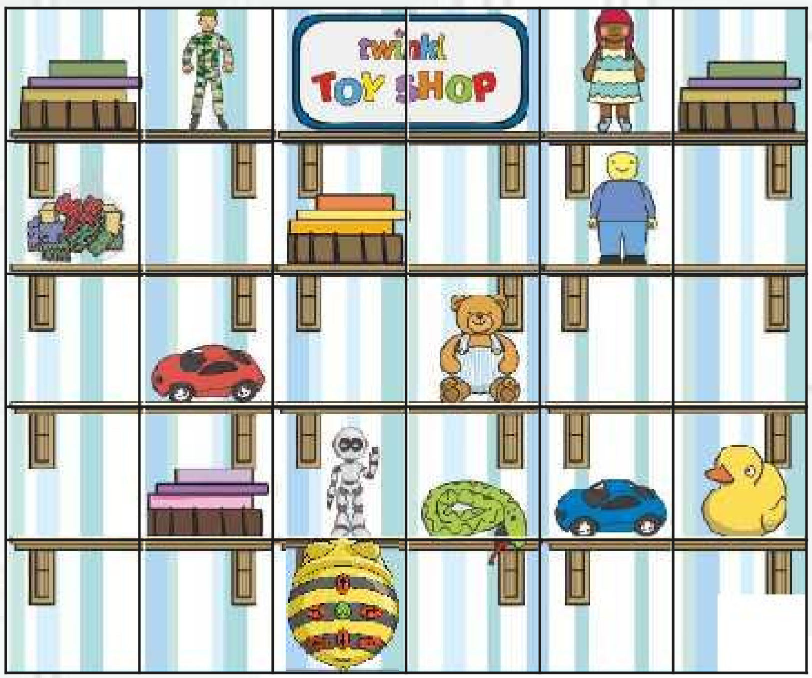
Don't forget to use the CLEAR button when you swap turns!

What Did I Buy?

Whole Class

These are the buttons I pressed:





Which toys did I buy?

robot

snake

teddy

Aim

- I can program a sequence to make a Bee-Bot (or similar programmable toy) move.

Success Criteria

- I can plan and check an algorithm.
- I can evaluate and improve my sequence.

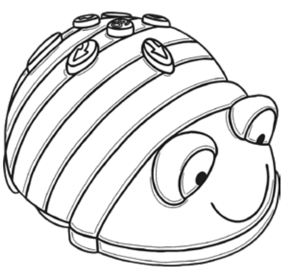
Shopping List

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.

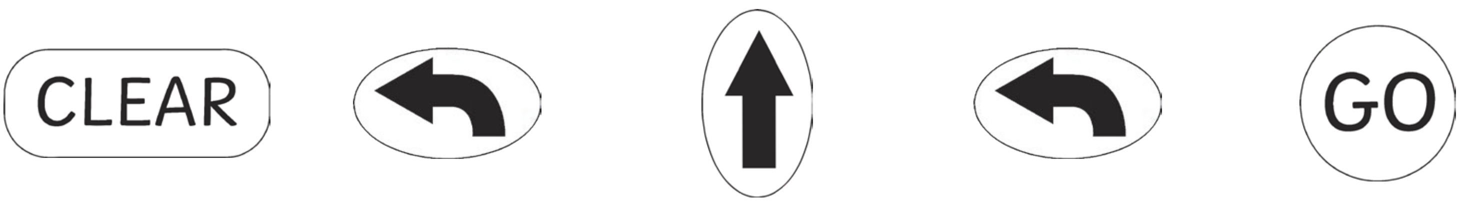
Direct your Bee-Bot (or similar programmable toy) to pick up the following toys in this order:

1. Robot
2. Teddy
3. Doll
4. Soldier
5. Go home

Draw the buttons that you need to press.



Have you used all of these buttons?



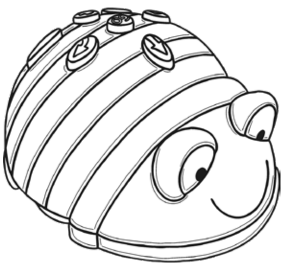
Shopping List

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.

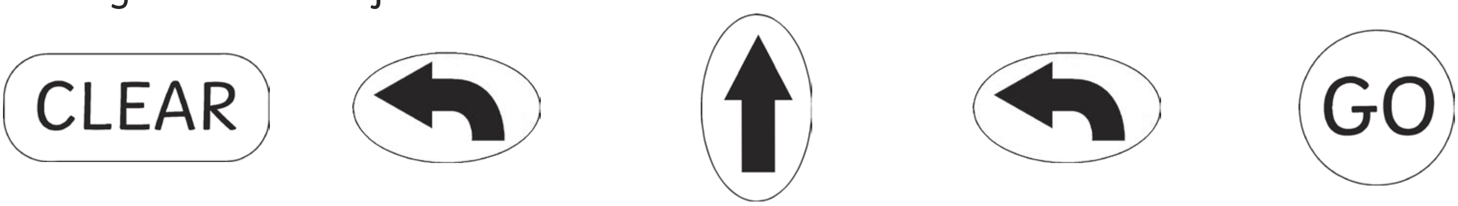
Direct your Bee-Bot (or similar programmable toy) to pick up the following toys in this order:

1. Doll
2. Blue car
3. Snake
4. Rubber duck
5. Building blocks
6. Go home

Draw or write the buttons that you need to press.



Have you used all of these buttons?



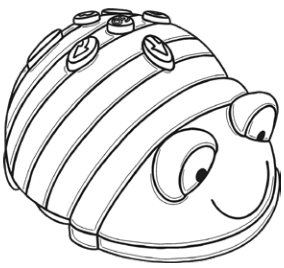
Shopping List

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.

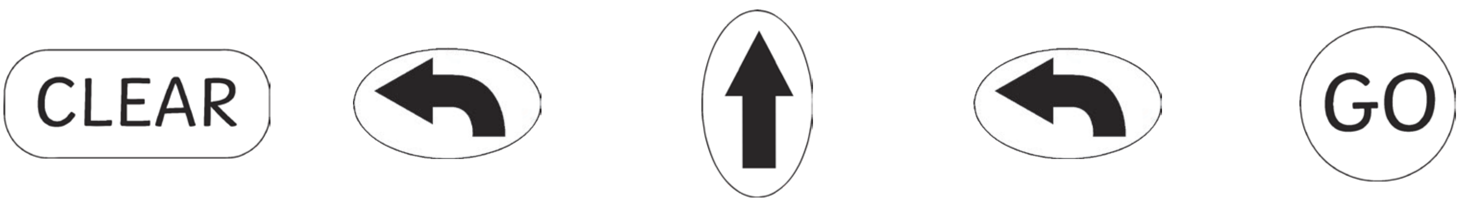
Get your Bee-Bot (or similar programmable toy) to pick up the following toys in this order. **You must not touch the Toy Shop sign or the books.**

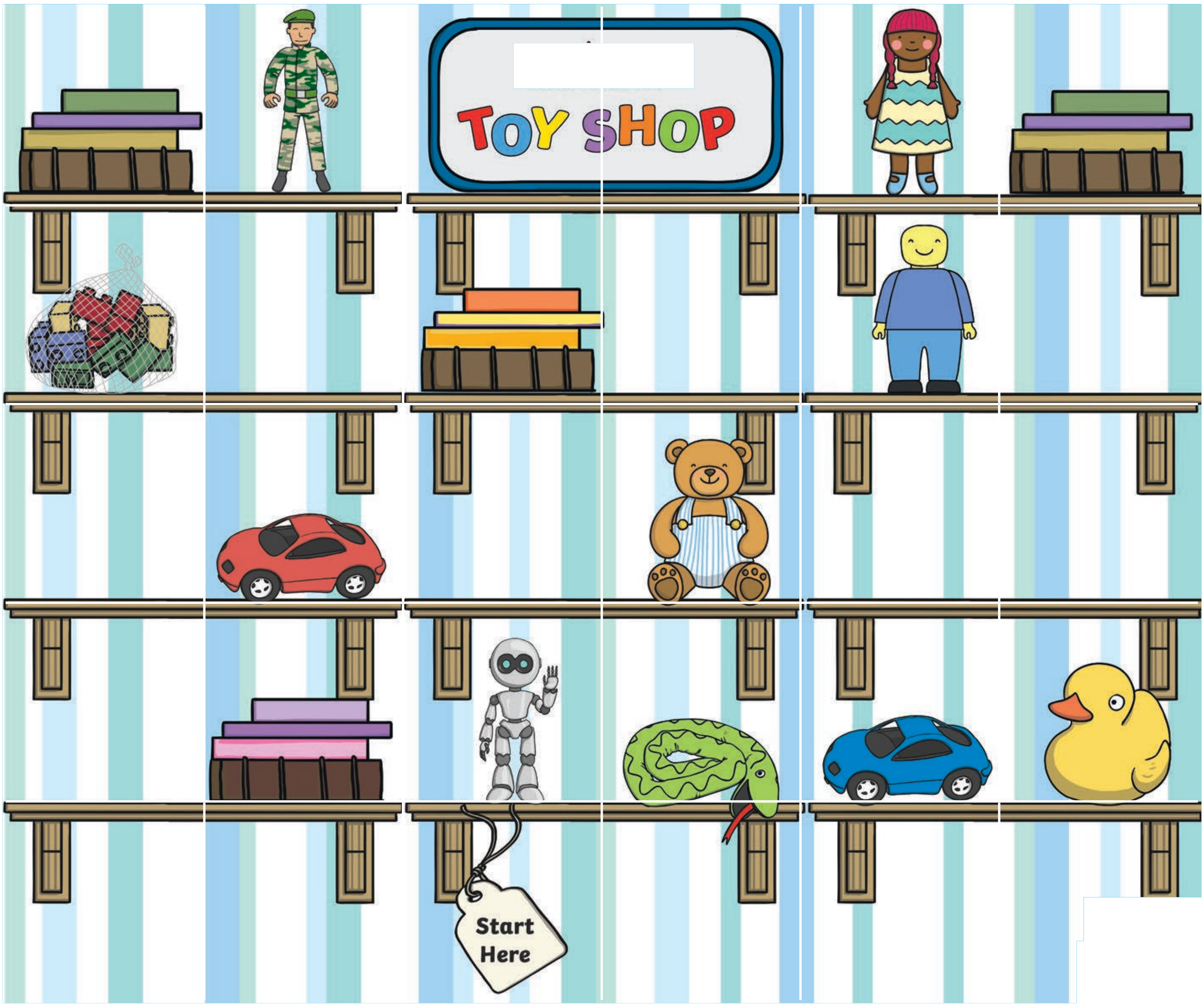
1. Toy soldier
2. Red car
3. Blue car
4. Doll
5. Rubber duck
6. Go home

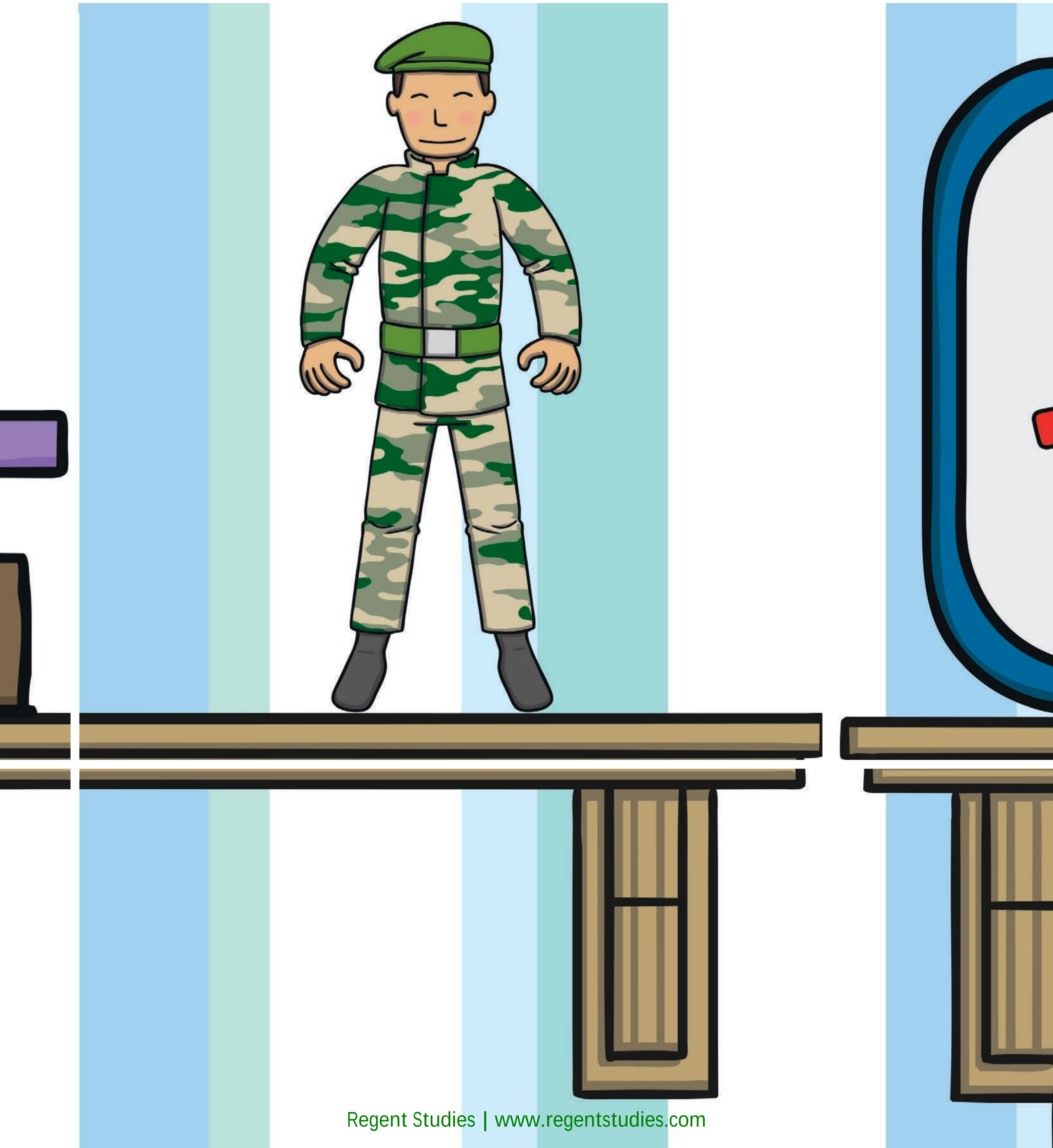
Draw or write the buttons that you need to press.



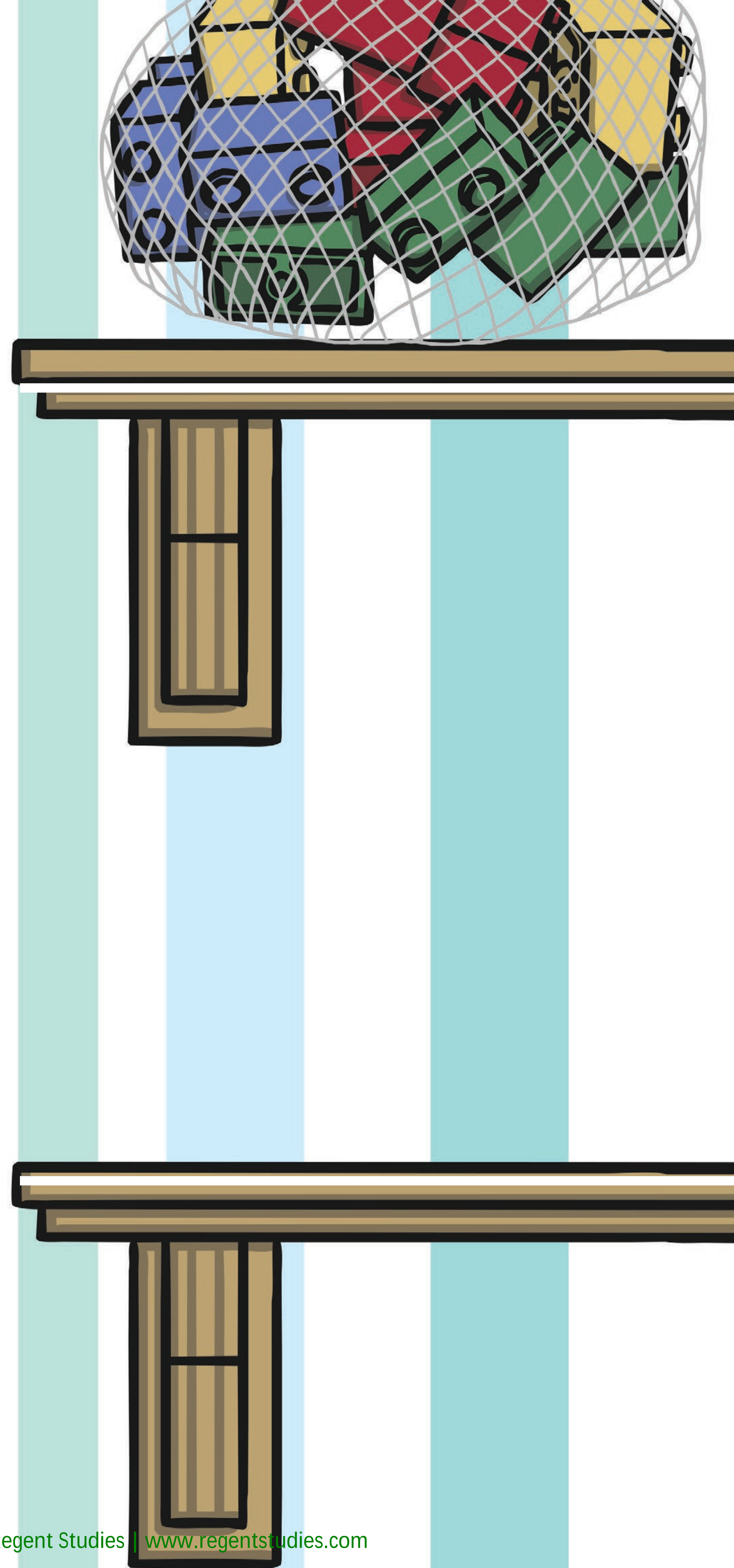
Have you used all of these buttons?

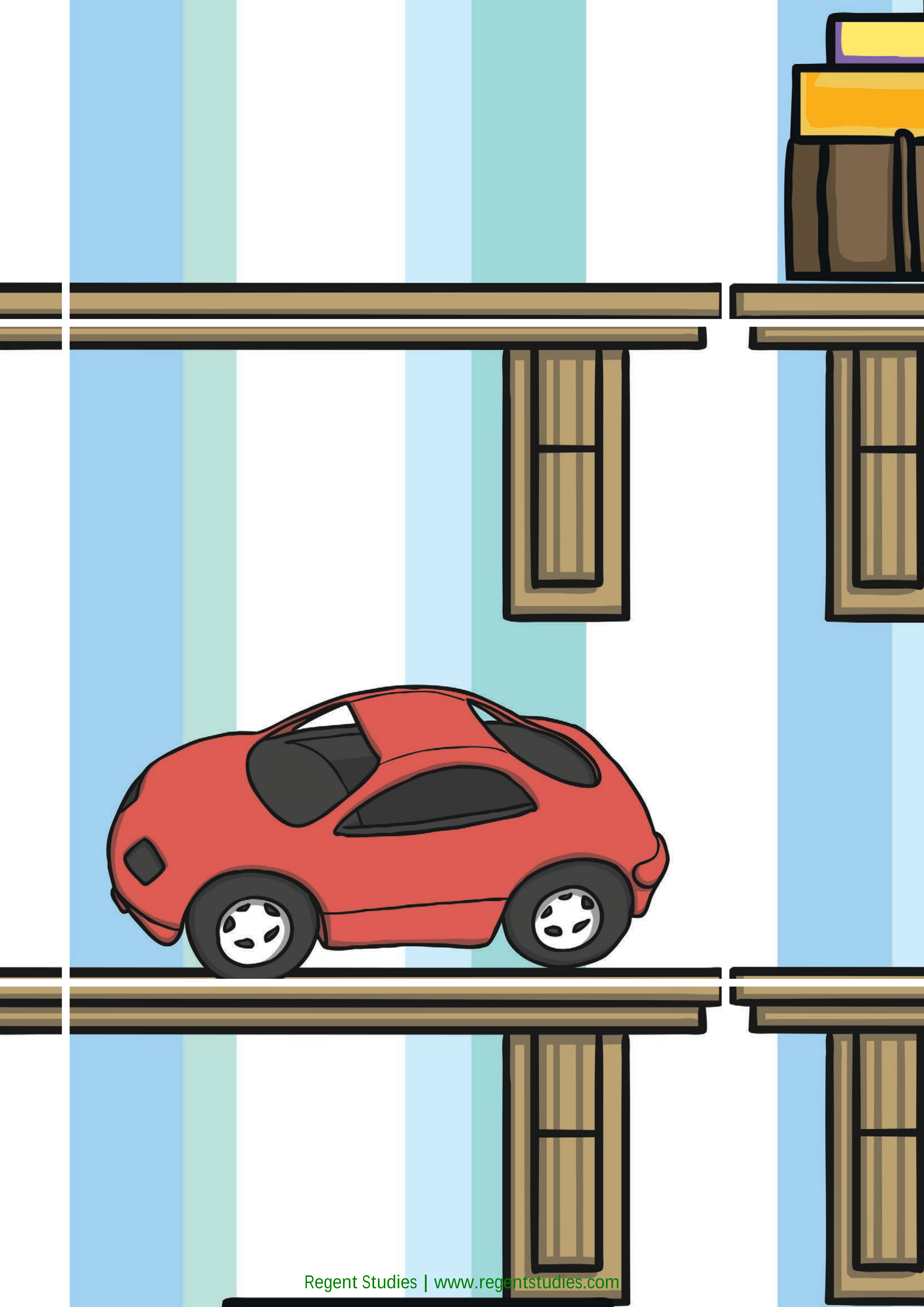


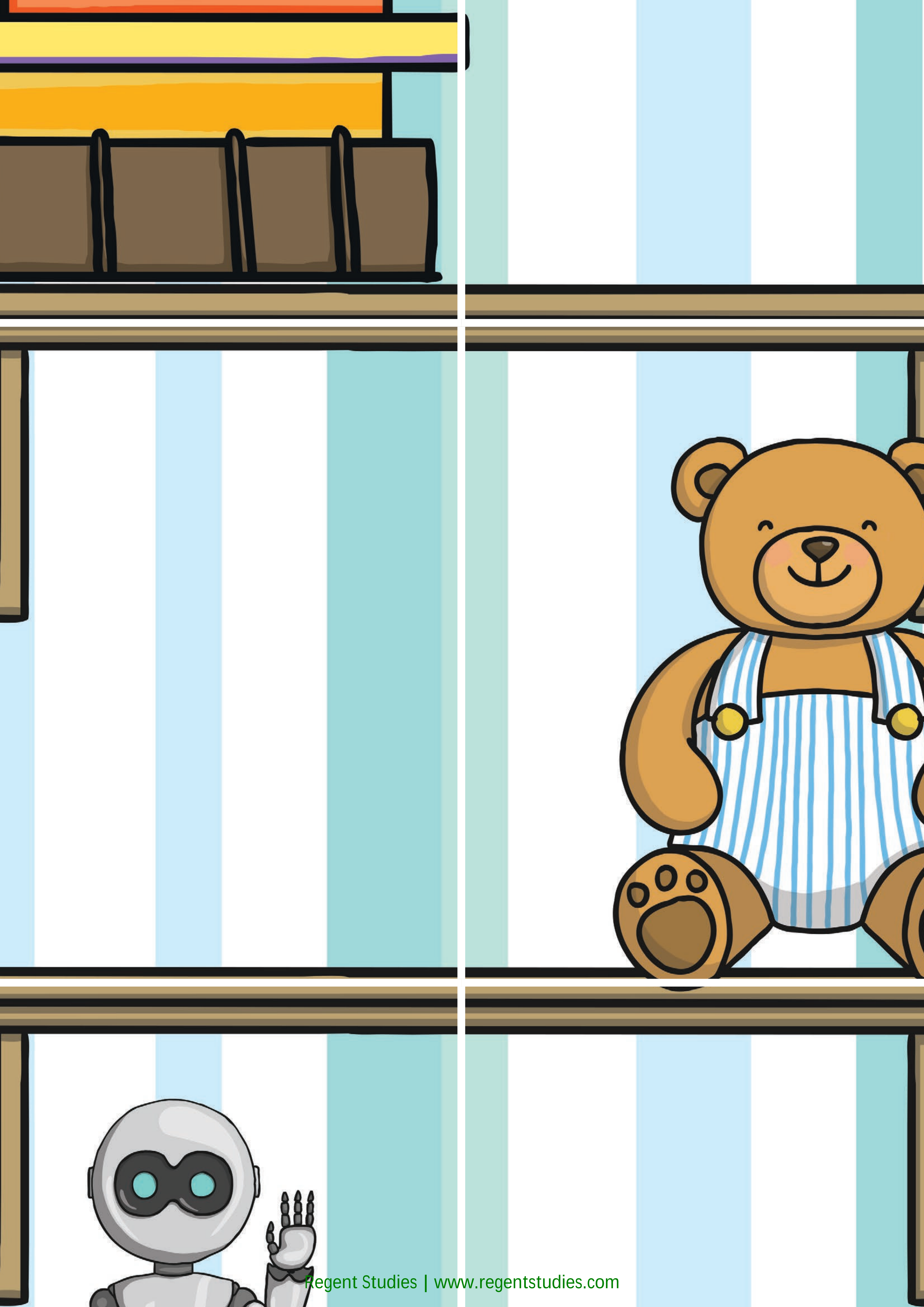

TOY SHOP
Start Here

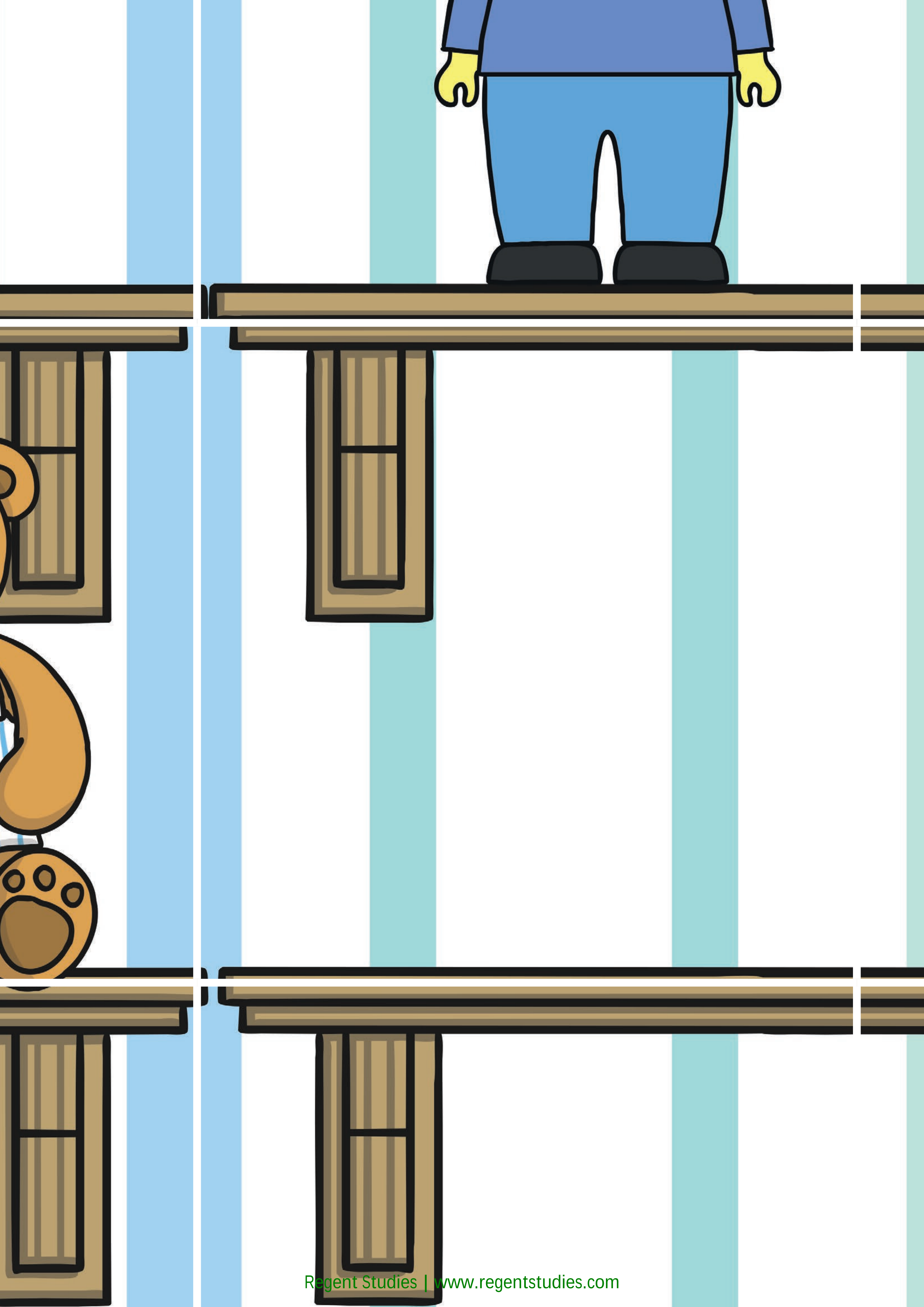


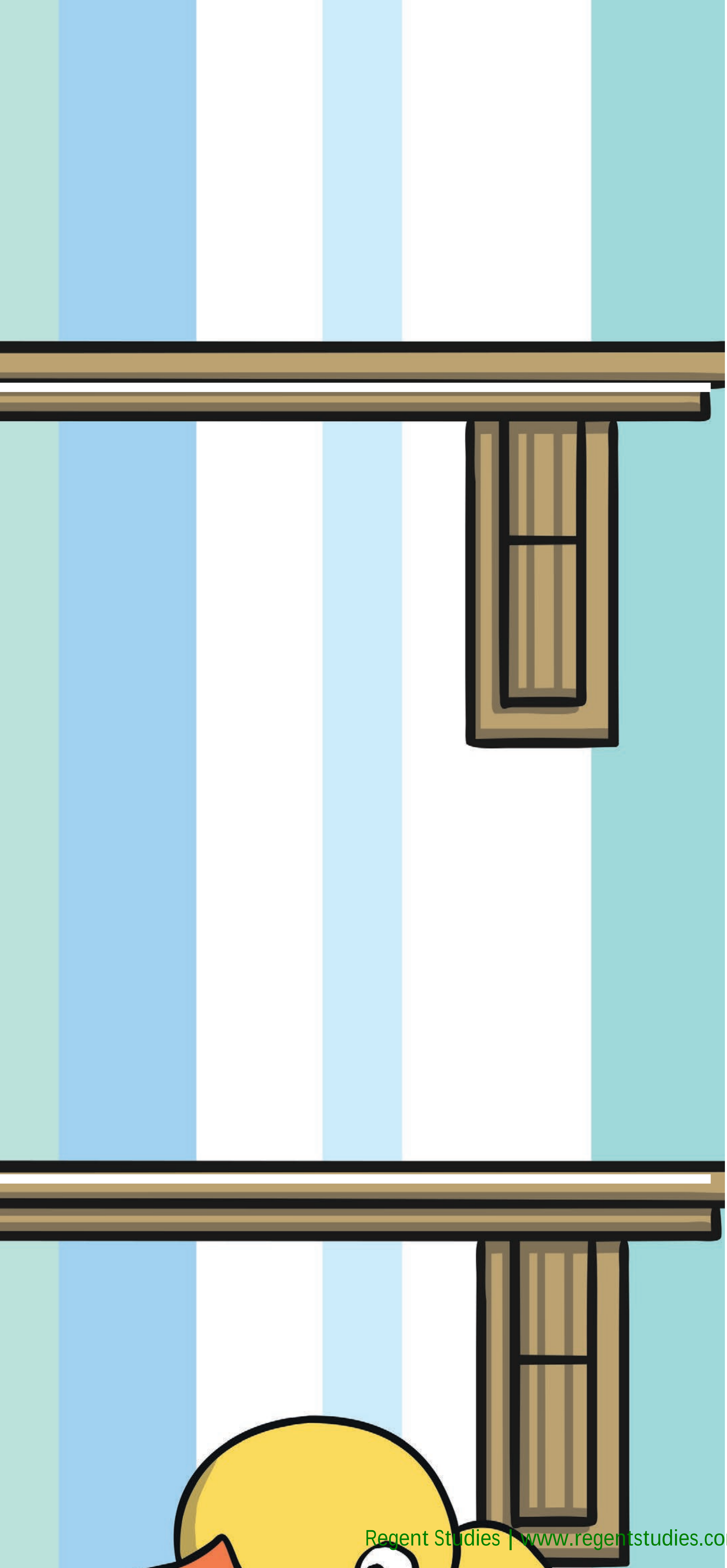

TOY SHOP

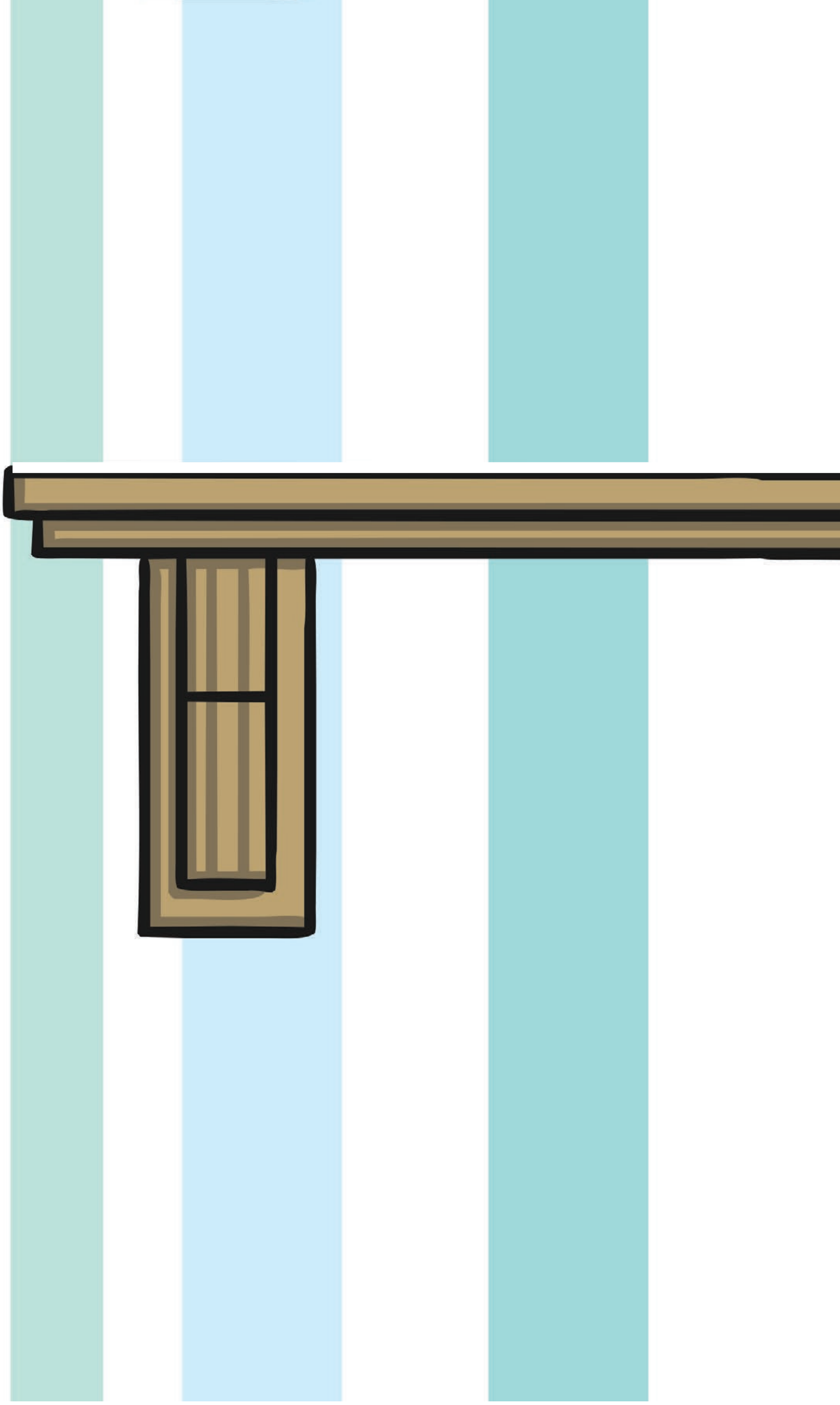


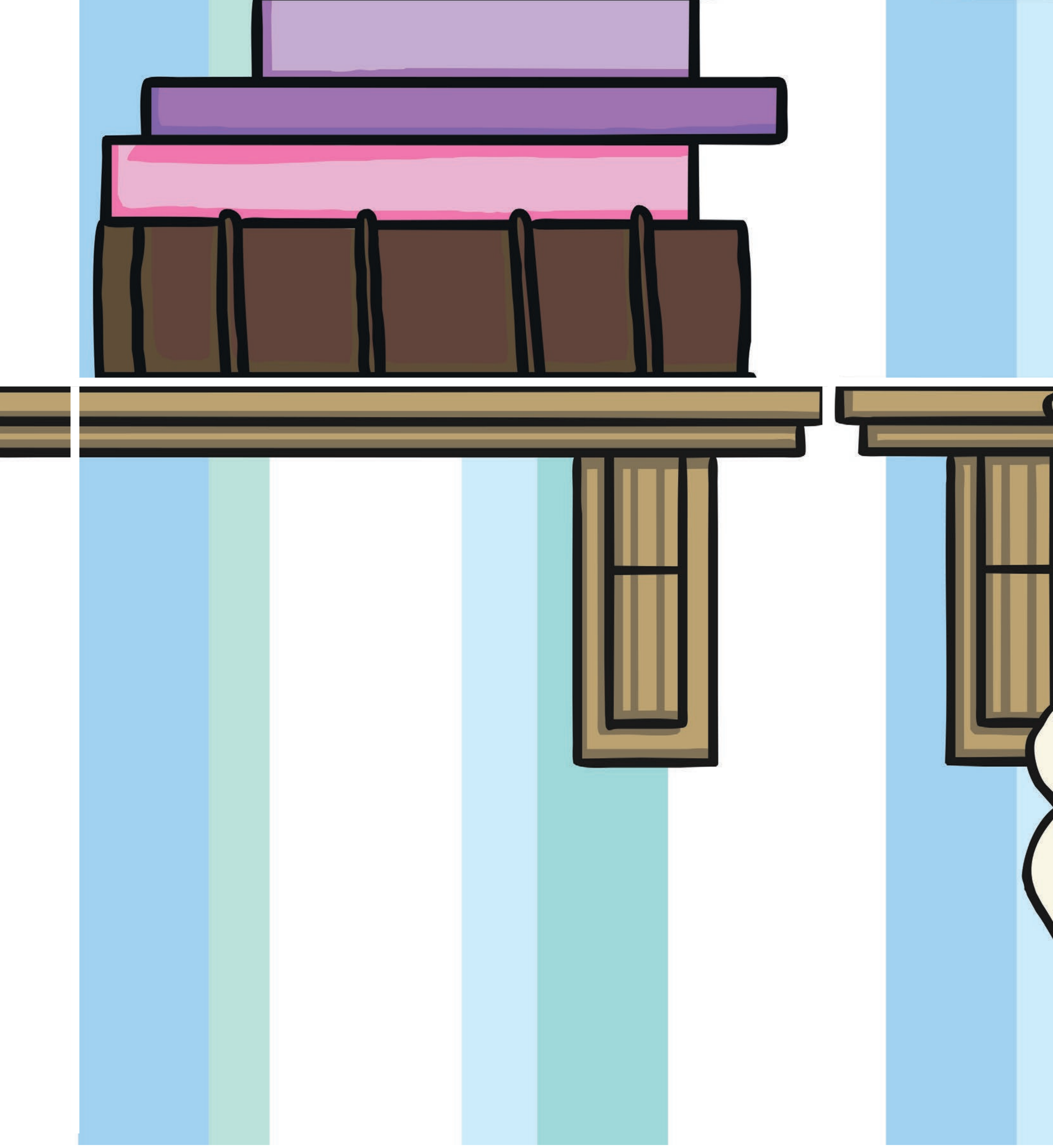


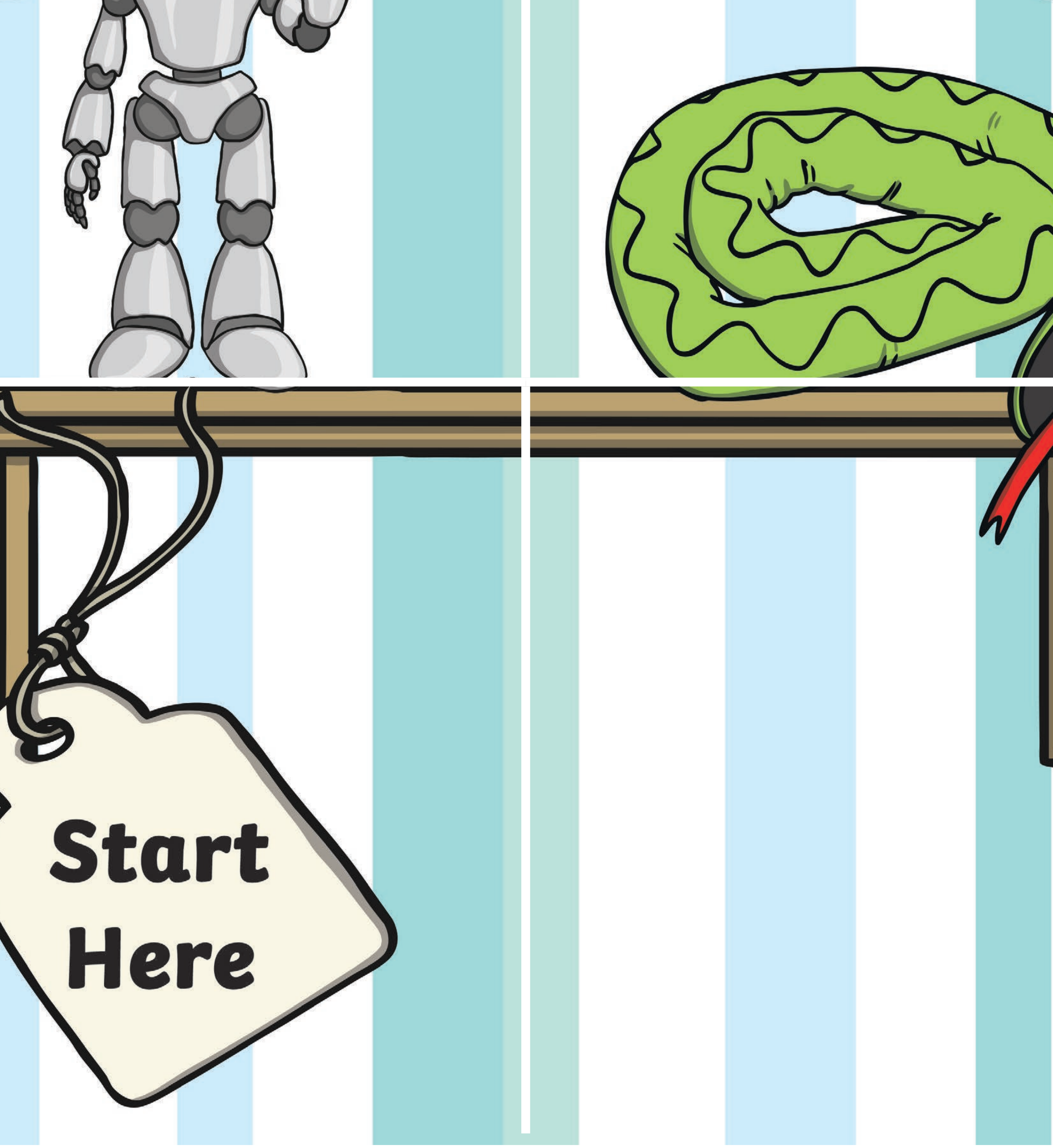


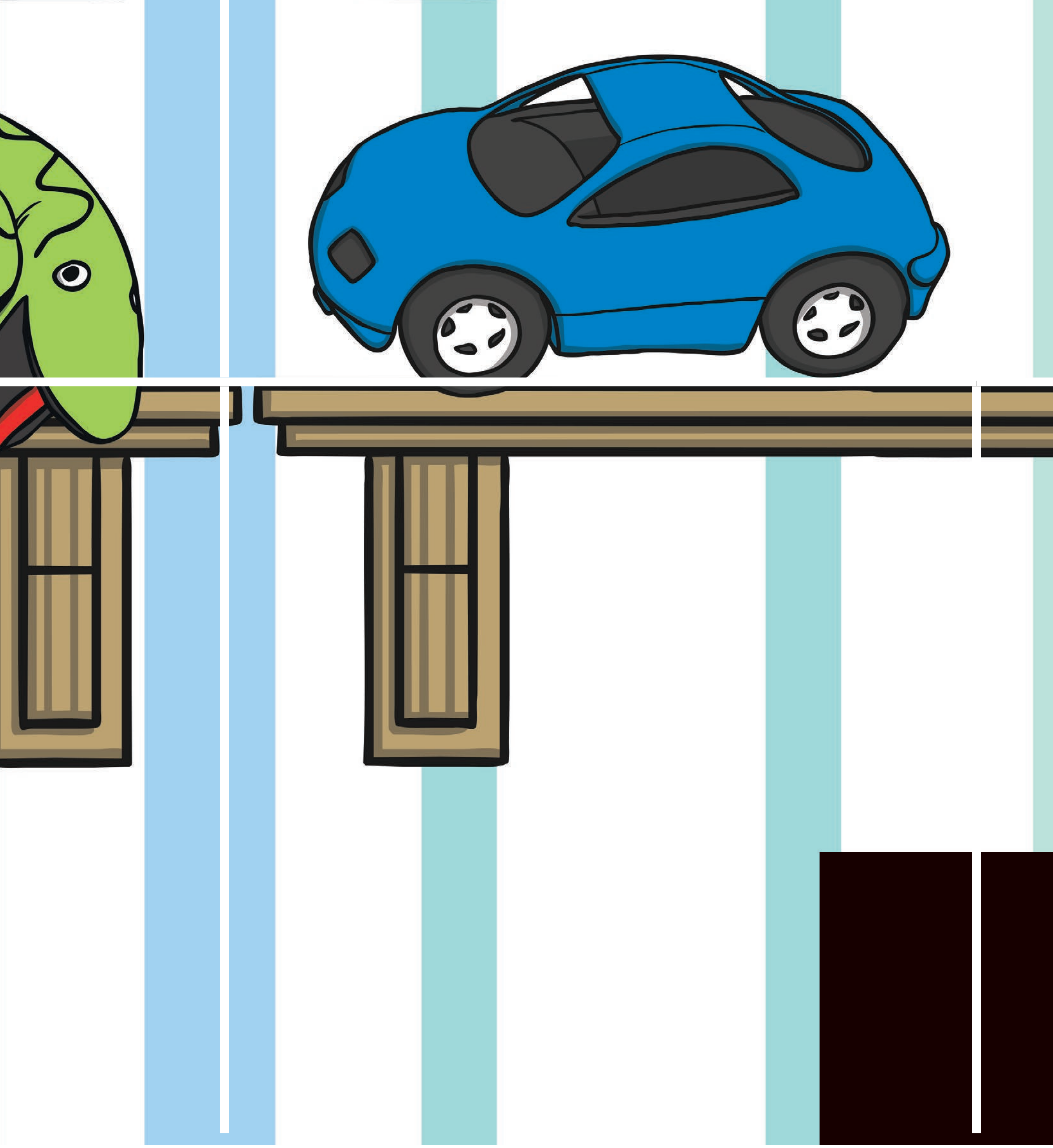


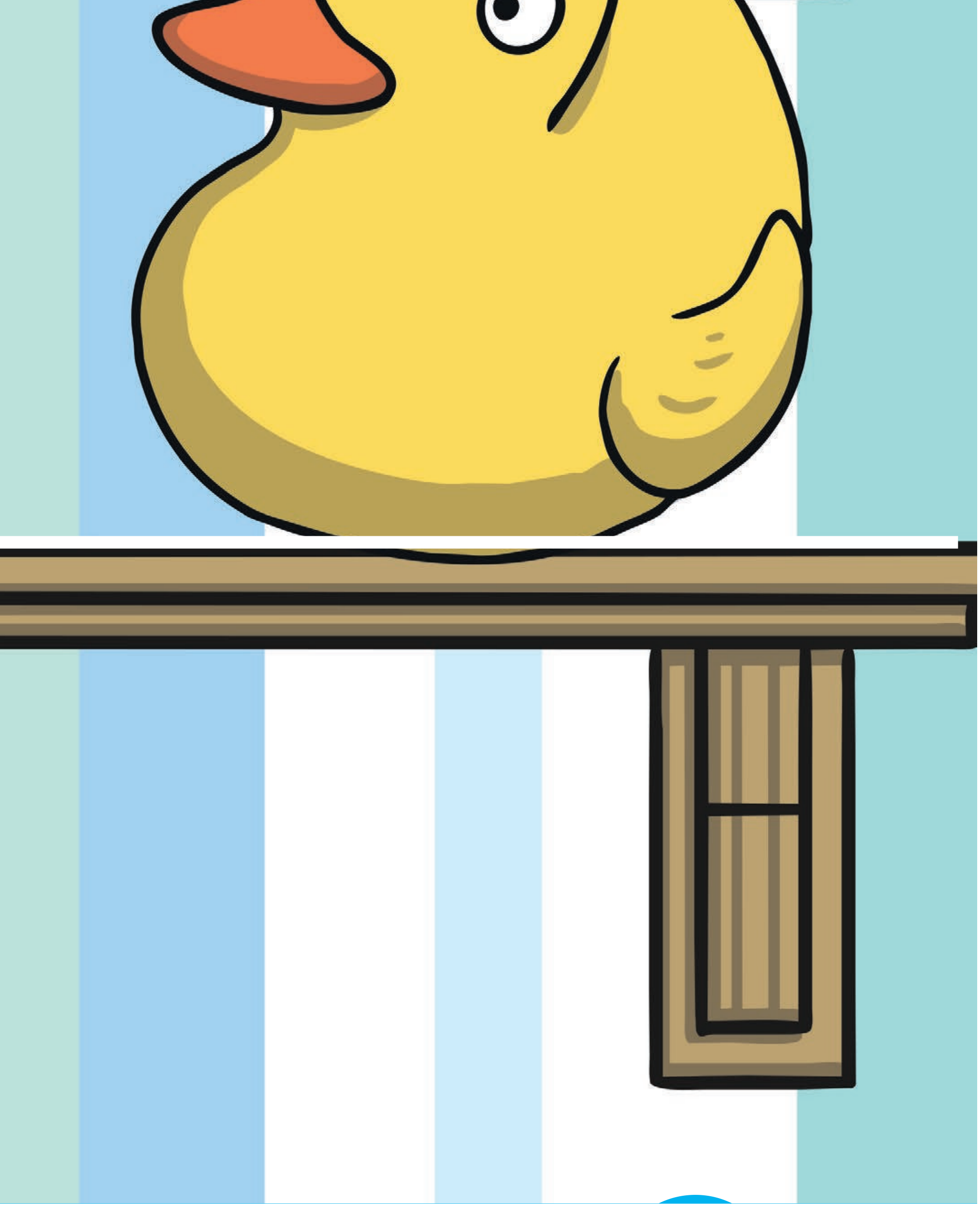







Start
Here





Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys | Toy Shop Part 2

I can program a sequence to make a Bee-Bot (or similar programmable toy) move.		
I can plan and check an algorithm.		
I can evaluate and improve my sequence (debug).		

Programming Toys

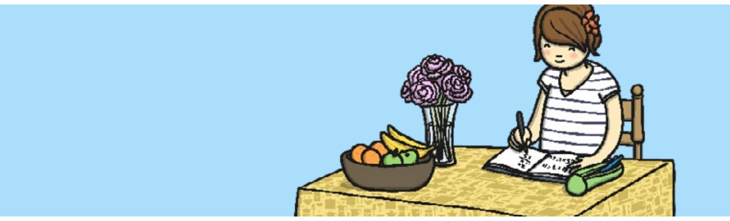
Computing | Year 1 | Unit Overview

Introduction

In this unit about programming toys, children will be introduced to the principles of programming through unplugged tasks and the use of Bee-Bots (or similar programmable toys). They will be introduced to algorithms as a set of step-by-step instructions given to a device, will learn how to debug simple algorithms and how to use logical reasoning to predict how a program will behave.

Health & Safety

When asking the children to use scissors and glue, please take care to provide good supervision. When moving around the classroom or other spaces in school, ensure that children are aware of their surroundings. Ensure that rules are clear for taking photographs on tablet devices and using equipment safely, carefully and respectfully.



Home Learning

Ordering Instructions: In this task, children will be asked to number instructions for how to build a tower from toy bricks in the correct order, and to try explain what will happen if the instructions are in a different, incorrect order.

Using Symbols in Algorithms: In this task, children will be asked to draw arrows in a sequence which will direct a Bee-Bot (or similar programmable toy) to a toy of their choice on a grid.

Assessment Statements

By the end of this unit...

...all children should be able to:

- create step-by-step instructions using pictures;
- write and follow detailed step-by-step instructions;
- direct a Bee-Bot (or similar programmable toy) to a toy;
- program a Bee-Bot (or similar programmable toy), one instruction at a time, using the arrow buttons.

...most children will be able to:

- say what an algorithm is;
- say why it is important to be precise when writing an algorithm;
- check their work for mistakes (debug);
- program a Bee-Bot (or similar programmable toy) using the arrow buttons;
- start their programming sequence again if they need to;
- check their work for mistakes to debug a program;
- plan and check an algorithm.

...some children will be able to:

- see how a product changes when they change the instructions;
- evaluate and improve their sequence (debug).

Lesson Breakdown

1. Building Bricks

Understand that programs execute by following precise and unambiguous instructions.

Create and debug simple programs.

Use technology purposefully to create digital content.

Children will work within the context of following picture instructions for building shapes.

- I can create instructions using pictures.

Resources

- Building bricks - 5 per pair
- Tablets with cameras - 1 per pair

2. Potato Man Algorithms

Understand how [algorithms] are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions in the context of writing detailed instructions to build a face on a potato man toy.

- I can say why it is important to be precise when writing an algorithm.

Resources

- 10 building bricks
- Glue
- Scissors
- Flipchart or large whiteboard

3. Program a Person

Understand what algorithms are and that programs execute by following precise and unambiguous instructions.

Create and debug simple programs.

Children will work within the context of writing instructions to program a person.

- I can write instructions to program a person like a computer.

Resources

- Shoes
- Whiteboards
- Scissors

4. Toy Shop Part 1

Understand what algorithms are and that programs execute by following precise and unambiguous instructions; create and debug simple programs in the context of programming a Bee-Bot (or similar programmable toy) to reach a set marker.

- I can program a Bee-Bot (or similar programmable toy) to move.

Resources

- Bee-Bots (or similar programmable toy)
- Whiteboards and pens
- Camera

5. Debugging Programmable Toys

Create and debug simple programs in the context of fixing incorrect Bee-Bot (or similar programmable toy) instructions.

- I can debug a Bee-Bot (or similar programmable toy).

Resources

- Bee-Bots (or similar programmable toy) - 1 per pair or group
- Whiteboards

6. Toy Shop Part 2

Understand what algorithms are and that programs execute by following precise and unambiguous instructions.

Create and debug simple programs.

Children will work in the context of programming a Bee-Bot (or similar programmable toy) to reach set markers.

- I can program a sequence to make a Bee-Bot (or similar programmable toy) move.

Resources

- Bee-Bots (or similar programmable toy)
- 3-4 toys
- Whiteboards and pens