Programming Turtle Logo and Scratch: Regular Polygons

Aim:	**Success Criteria:**	**Resources:**
Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts. Use sequence, selection, and repetition in programs; work with variables and various forms of input and output. Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs. This unit continues the learning from the Year 2 Turtle Logo units and links well to shape and direction in Maths. I can create and debug algorithms that draw regular polygons.	I can write commands in the correct order. I can write a variable value where required. I can correct any mistakes. I can rotate the turtle angles other than 90˚. I can use calculations as a variable.	**Lesson Pack** Desktop computer /laptop Turtle Logo application (installed or online) Whiteboards and pens or books, pens and pencils for recording.
	Key/New Words: Algorithm, instructions, commands, forward (fd), left (lt), right (rt), move, turn, clear screen (cs), variable, pen up, pen down, calculation.	**Preparation:** None needed

Prior Learning: Children will have created an algorithm using the pen up and pen down commands in lesson 2.

Learning Sequence

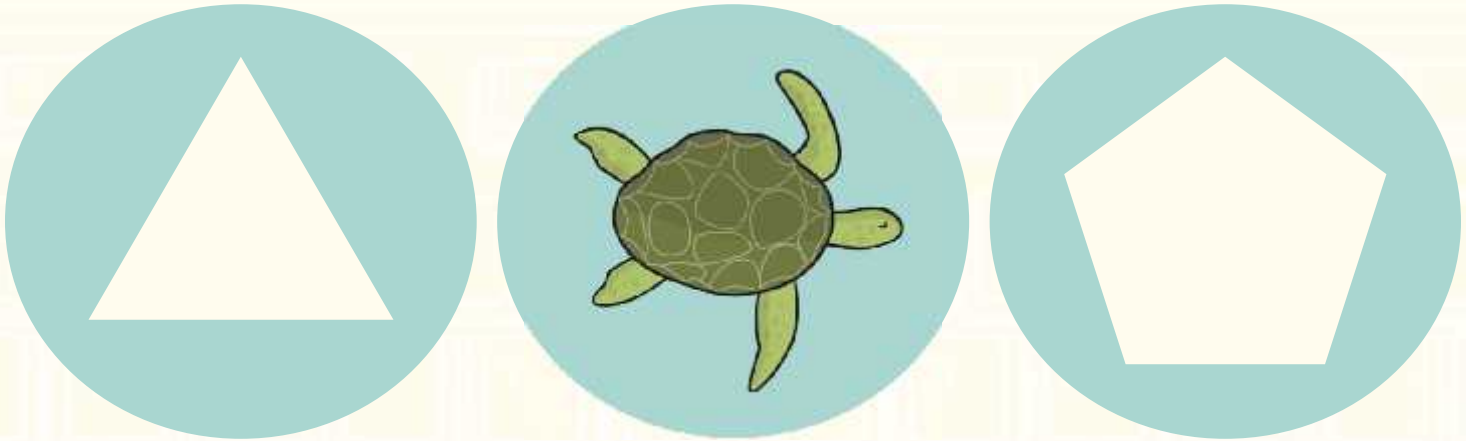
	Write the Algorithm: Ask the children to write algorithms for different shapes and test each one. • Write an algorithm for a square of side 120 and a square of 60 inside. • Write an algorithm for 4 rectangles of sides 30 and 50 with a space of 20 between each. • Write an algorithm for a rectilinear number 8.	
	Turning Angles Other Than 90˚/What Angle to Turn? Introduce the different rotating angles and demonstrate each one. Explain how to work out what angle you would use in your command and work through the example on the **Lesson Presentation.**	
	Drawing Different Polygons: Children use the differentiated **Activity Sheets** to draw different polygons, then either take a snip or screenshot to save their patterns and algorithms. Do you come across any problems with any of the shapes? What happens as you draw regular polygons with more and more sides? What is the most number of sides you can draw a regular polygon with? Children are given step by step support to write the algorithms. Children are given prompts to help them write the algorithms. Children create algorithms to draw regular polygons and answer the questions on the **Activity Sheet.**	
	Let Turtle Logo Work It Out! When sufficient children have drawn a number of polygons and are confident with the algorithm, show how to let Turtle Logo calculate the required angle of turn. *(rt 360/6 is the same as rt 60 because 360/6=60.)*	
	Write Your Own: Children continue the activities above, but may use the option to have Turtle Logo calculate the turn.	
	17 Sides? Can you draw a regular polygon with 17 sides? Ask children to share their answers and explain how they worked out the algorithm.	
	Which Shape Will Be Drawn? Ask children to look at the algorithm, fd 120 rt 120 fd 120 lt 60 fd 120 rt 120 repeat 3[fd 120 rt 60]. Which shape do you think this algorithm will draw? Give children time to think about which solution is correct and then as a class decide on which answer to select.	

Task*it*
***Polygon*it:** Children make algorithms for repeating shapes with spaces between.

Computing

Programming Turtle Logo and Scratch

Regular Polygons



Aim

- I can create and debug algorithms that draw regular polygons.

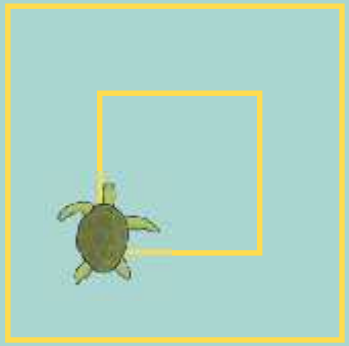
Success Criteria

- I can write commands in the correct order.
- I can write a variable value where required.
- I can correct any mistakes.
- I can rotate the turtle angles other than 90°.
- I can use calculations as a variable.

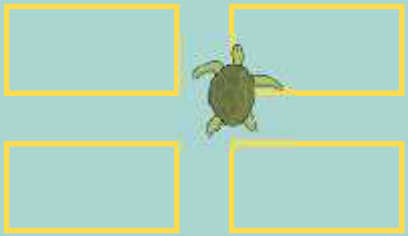
Write the Algorithm

Write algorithms for these different shapes and test them:

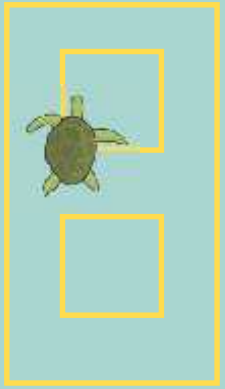
A square of 120 sides with a square of 60 sides inside.



4 rectangles of 30 and 50 sides with a space of 20 between each.



A rectilinear number 8.

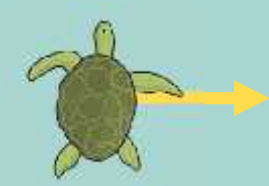


Turning Angles Other Than 90°

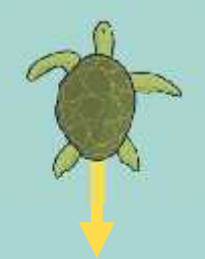
The turtle can be commanded to turn any angle. The angle of the turn is measured in degrees.

rt will turn the turtle an angle of 45° to the right.

90° is a quarter turn.



180° is a half turn.



360° is a full turn.

What Angle To Turn?

To work out the angle to turn for any polygon you need to divide 360 by the number of sides.

In a square there are 4 turns, so the turn is $360 \div 4 = 90$.

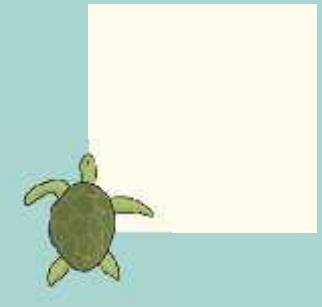
So, for a hexagon, it is $360 \div 6 = 60$.

Drawing Different Polygons

Using a Turtle Logo programme on a computer or tablet, draw different regular polygons.

How will you work out what angle you need to turn?

Remember a whole turn is 360°, and with a square you turned a quarter turn, which is 90°.



Let Turtle Logo Work it Out!

You can write an algorithm that includes a calculation.

So for a square you can make the turn 360/4 (representing 360 ÷ 4)

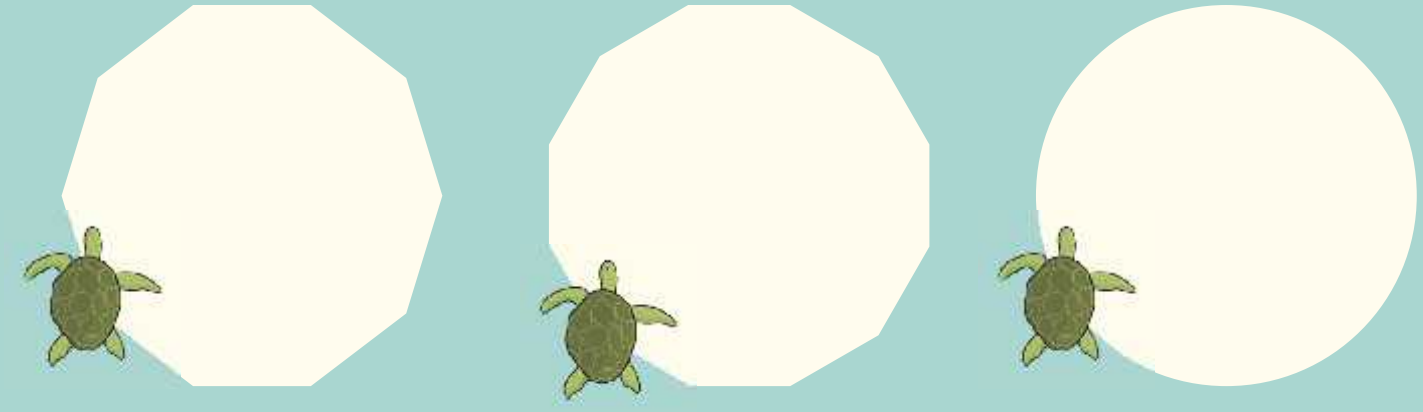
The command will be repeated 4[fd 100 rt 360/4]

So, for a heptagon, it is repeat 7[fd 80 rt 360/7]

What happens when you draw a polygon with many sides?
Try writing an algorithm for a shape with 360 side. What happens?

Write Your Own

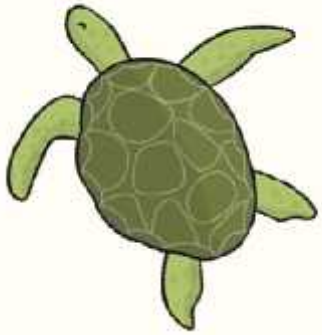
Write your own algorithms that include calculations to draw regular polygons that have a different numbers of sides.



17 Sides?

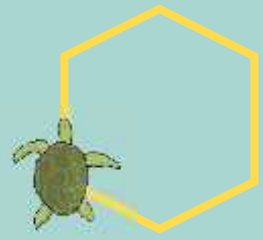
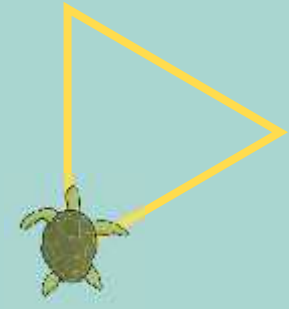
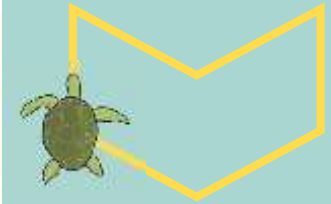
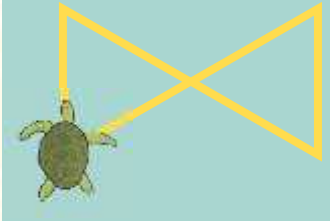
Can you write
the algorithm for a
regular 17 sided shape?

repeat 17[fd 40 rt 360/17]



Which Shape Will Be Drawn?

fd 120 rt 120 fd 120 lt 60 fd 120 rt 120
repeat 3[fd 120 rt 60]

A	B	C	D
			
Incorrect: All turns would be rt 60 or lt 60.	**Incorrect:** All turns would be rt 60 or lt 60.	Correct Well done!	**Incorrect:** Second line would repeat the first line.

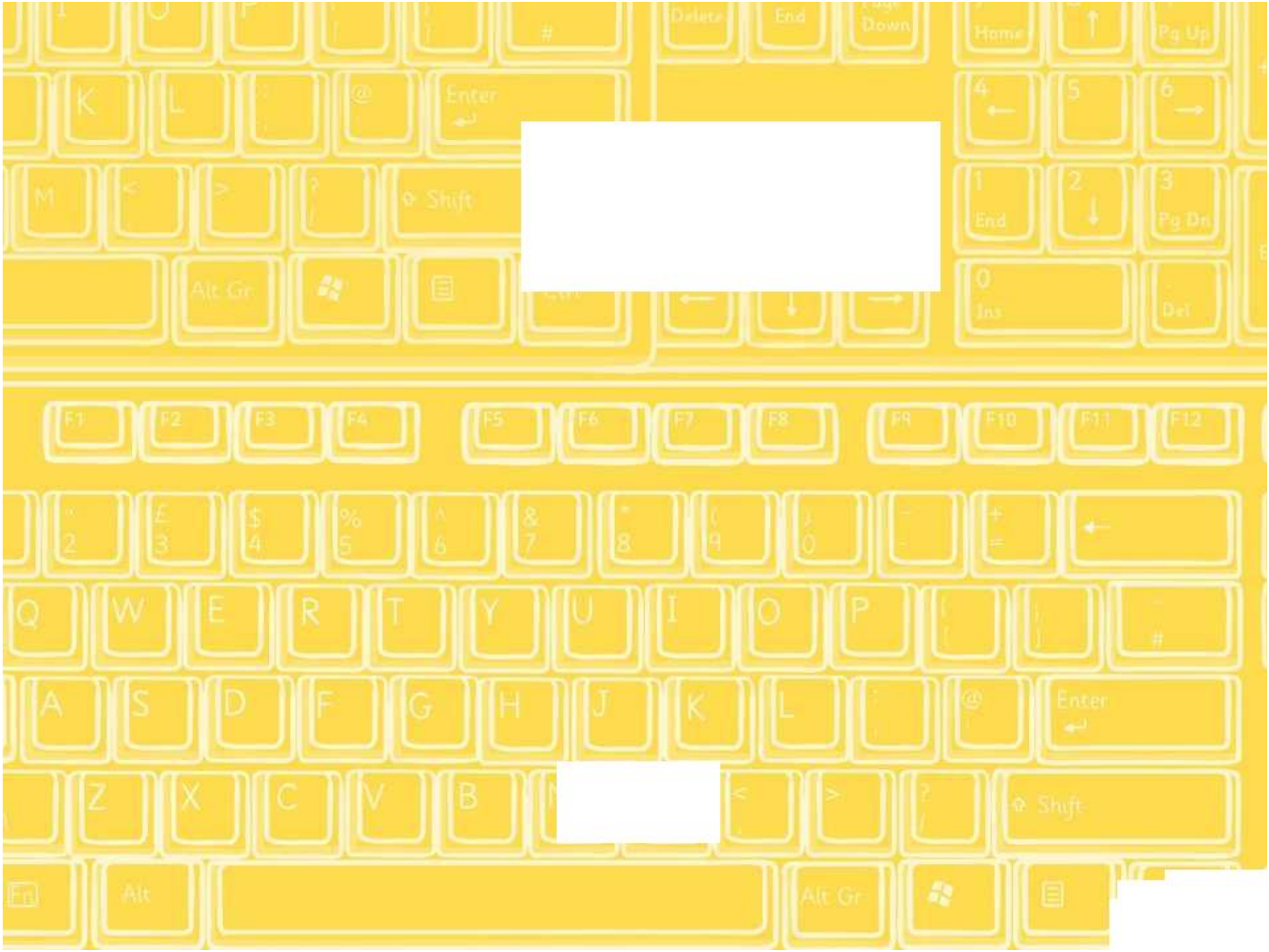
Click on the shape that you think is correct

Aim

- I can create and debug algorithms that draw regular polygons.

Success Criteria

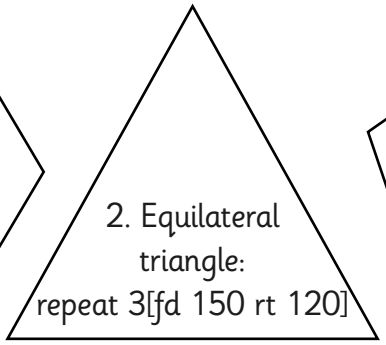
- I can write commands in the correct order.
- I can write a variable value where required.
- I can correct any mistakes.
- I can rotate the turtle angles other than 90°.
- I can use calculations as a variable.

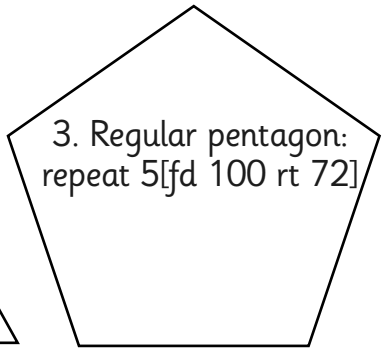


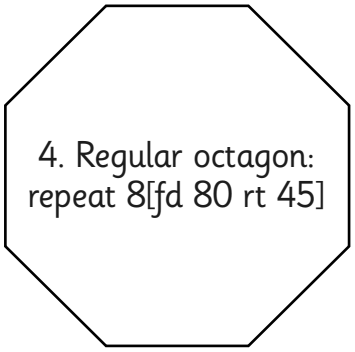
Regular Polygons

Draw the following algorithms in Turtle Logo.
Remember to snip or take a screen shot of your work to save your pictures, patterns and algorithms.

1. Regular hexagon:
repeat 6[fd 100 rt 60]

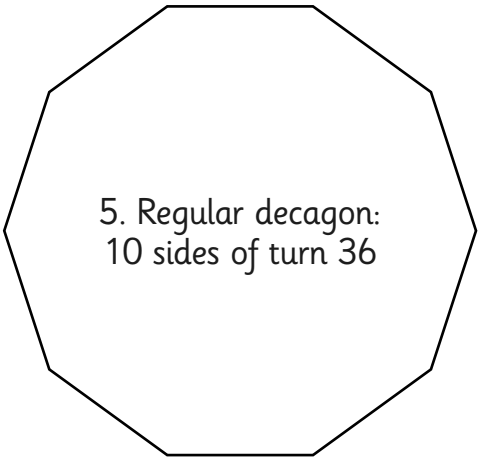


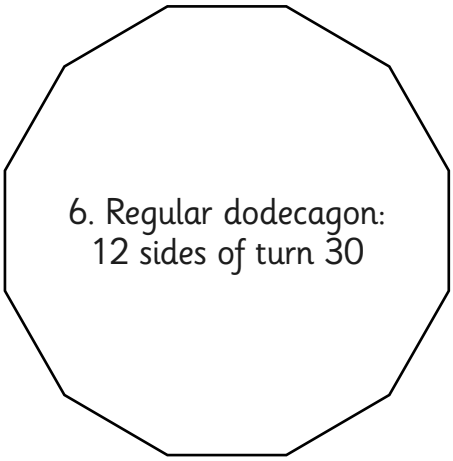


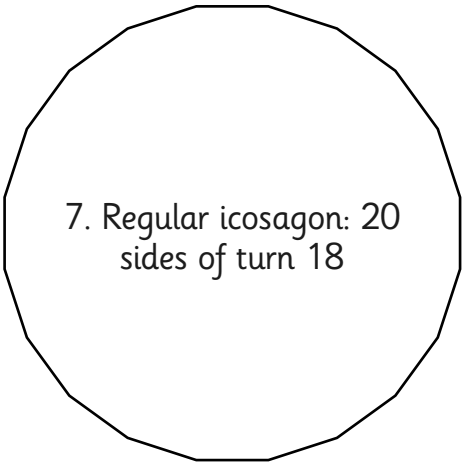


Now try drawing the following algorithms.

Don't forget to change the number of amount of times the algorithm is reapeated and the amount you want the turtle to turn. Take care not to make the sides too long!





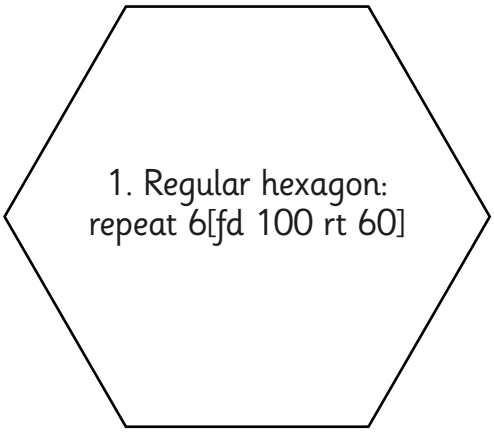


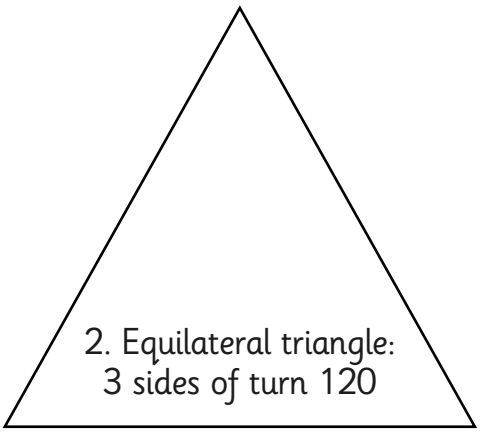
Now try drawing some other regular polygons.

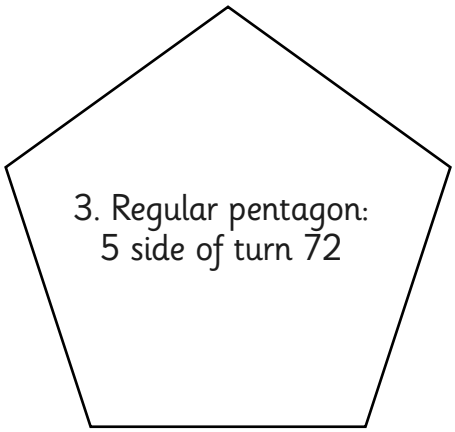
Have you come across any problems with any of the shapes?

Regular Polygons

Draw the following algorithms in Turtle Logo.
Remember to snip or take a screen shot of your work to save your pictures, patterns and algorithms.

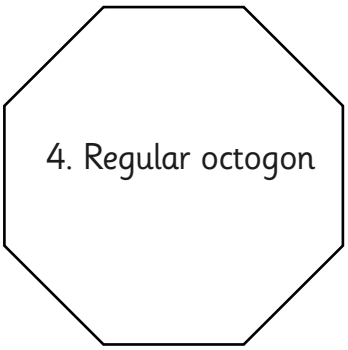


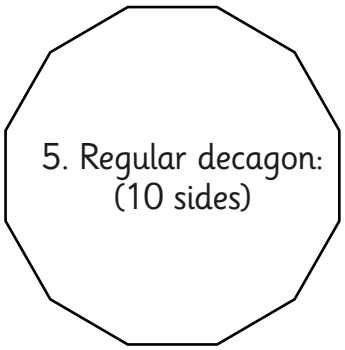


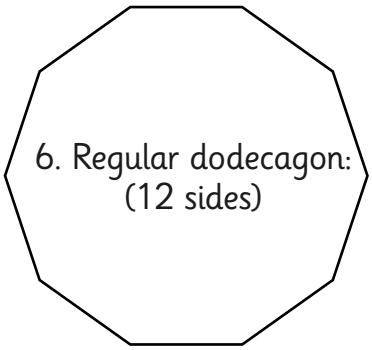


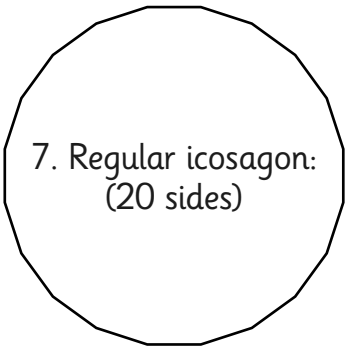
Now try drawing the following algorithms.

Calculate the turn by dividing 360 by the amount of sides the shape has. Take care not to make the sides too long!







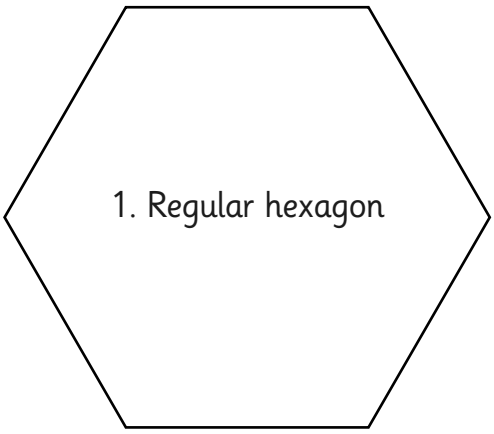


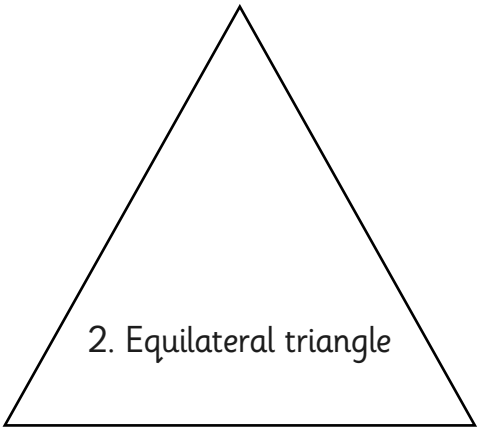
Now try drawing some other regular polygons.

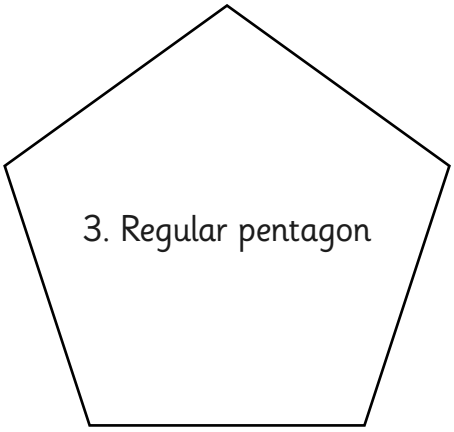
Have you come across any problems with any of the shapes?

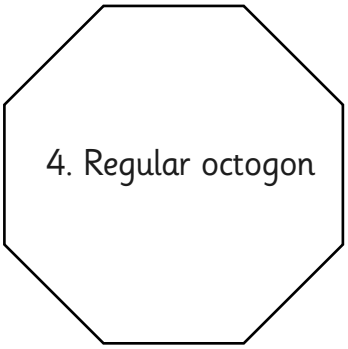
Regular Polygons

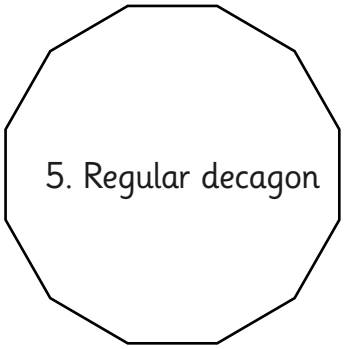
Draw the following algorithms in Turtle Logo. Remember to snip or take a screen shot of your work to save your pictures, patterns and algorithms. Calculate the turn by dividing 360 by the amount of sides the shape has. Take care not to make the sides too long!

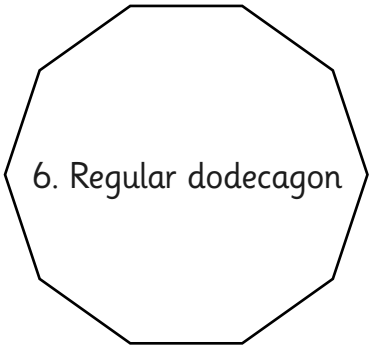


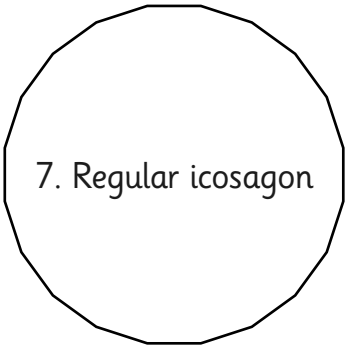












Now try drawing some other regular polygons and answer the questions below:

Have you come across any problems with any of the shapes?

What happens as you draw regular polygons with more and more sides?

What is the maximum number of sides that you can draw a regular polygon with?

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

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		

Programming Turtle Logo and Scratch | Regular Polygons

I can create and debug algorithms that draw regular polygons.		
I can write commands in the correct order.		
I can write a variable value where required.		
I can correct any mistakes.		
I can rotate the turtle angles other than 90°.		
I can use calculations as a variable.		