Summer Term Overview Year 5 and 6 – Computing

Summer Term Book – Mortal Engines		
Topic – Scratch Animated Stories		**Guide Time = 60 mins per lesson**
Assessment:	This unit will be assessed through ongoing formative assessment of skills and knowledge, based on the content taught in each lesson. Children may be quizzed on the VIPs that are covered throughout the unit.	**Very Important Points (VIPs):** • An algorithm is a sequence of instructions or a set of rules that are followed to complete a task. • A backdrop is the 'stage' on which all of sprites' actions are completed. This can be changed. • When an algorithm does not work, or does not do what we expect it to, we must debug it. This means to make changes to fix the problem. • A sprite is an object that performs actions on a Scratch project. There are many to choose from. • A sequence is a set of programming blocks that make actions happen one after another. • An iteration is when a sequence of actions is repeated. • Broadcasting means to send out a signal or a message. **Fat Questions:** How has coding developed over the last century? What impact has coding had on our daily lives? How could coding be used in the future to enhance how technology is used?
Links to prior learning (sequencing) and canon book	In KS1, children will have been introduced to the term 'algorithm' when preparing for programming with Turtle Logo. They will also begin to explore simple algorithms using Scratch and they will start to debug the algorithms they create. In LKS2, children will have used Scratch to create shapes and patterns using algorithms. They will have also created their own question and answer VIP quiz. Specific to this UKS2 unit, they will have learnt how to program a sprite, change and add a sprite and how to turn a sprite.	
Links to other learning (cross fertilisation)	English – poetry is linked to technology (Xbox) Geography – technology allows us to use OS Maps online to give us easier access to maps. This helps us understand land use across both our local area and the rest of the country. DT – children will be designing their own programmable toy this term Thematic Questions: The World Beyond Us What recent developments in technology have allowed us to explore space further? Modern Britain In the technology industry, how have jobs changed since the new millennium? Healthy Bodies & Healthy Minds Coding helps us to become more resilient people. How is this helpful for our futures? The World Around Us How has technology in our world changed since your parents were your age? Culture	

	Travel can be expensive. How does technology allow us to explore other cultures without leaving our homes? Technology in Action Can you find examples of where coding is used in everyday life?	
Links to future learning	(If children are in Year 5) Summer term Cycle A – to use / write a program to control a mechanical robot's movements. As part of the KS3 Computing National Curriculum, children will be taught: • To understand several key algorithms that reflect computational thinking; use logical reasoning to compare the utility of alternative algorithms for the same problem. • To use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures; design and develop modular programs that use procedures or functions. • To understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal].	
Character/Wider Development ('50 things', cultural capital, skills)	Programming encourages skills of resilience, patience and determination. These are skills that are needed for many aspects of life and children will benefit massively from having the opportunity to build on these skills. This, in turn, relates to the 50 Things document for Year 5 and 6. They are required to create their own CV – a document that focuses on the children's skills.	

OVERVIEW OF TEACHING SEQUENCE

Key Facts/Learning	Learning Focus or Key Question	Learning Outcomes (NC)	Key Words/ Vocabulary	Greater Depth/SEND	Misconceptions	Activities and Resources
Lesson 1 An algorithm is a sequence of instructions or a set of rules that are followed to complete a task.	LO: To animate a scene on Scratch	NC: 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	algorithm, sprite, backdrop, script, block, iteration, sequence, debug, repeat, glide	SEND: Children input the script exactly as provided on the activity sheet. May require more adult support, or a more able partner to support them through the process. Children may also just focus on adding one sprite, rather than three. GD: Children use open-ended prompts to write and develop their code. Individual solutions to creating the script may vary, but trial and error should be encouraged. Children may also experiment with adding other sprites and altering the code.	Children may not remember that they must be 'clicked' on the sprite in order to see the code for each sprite. Many children assume their code has been deleted if they cannot see this anymore. This has been included as a teaching point. Many children also assume that this should be 'easy'. Remind children throughout that coding requires resilience and a 'trial and error' approach.	Children will recap on previous learning and vocabulary definitions: algorithm, sprite, backdrop, script, block BBC Bitesize What is an Algorithm Children will explore a non-animated scene on Scratch and discuss how it could possibly be animated. They will then be introduced to three different ways of animating their sprites. Using a set background and characters, children will use the activity sheets to input the code and explore what their algorithms will create. They are encouraged to debug as they go along. **Deepen the moment** Year 5: • Try adding sounds effects for each sprite. • Use the 'go to' block to experiment with moving sprites around the screen. Year 6: • Is there a way you can make the knight turn around and go the other way when he reaches the side of the screen? • Add your own sprite and animate it - remember, it needs to be something we would associate with a haunted house!

		algorithms and programs.				
Lesson 2 Broadcasting means to send out a signal or a message.	LO: To broadcast a message on Scratch	NC: Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems. 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.	algorithm, sprite, backdrop, script, block, iteration, sequence, broadcast, debug	SEND: Children may need the use of the 'broadcast message' block modelled to them specifically, on their piece of work. Children to use very clearly modelled worksheet to support them with the correct blocks to use. They may only be able to create one broadcast message. GD: Children should be encouraged to investigate if they can trigger other actions using the 'broadcast message' block. This could be with the boy and girl sprites, or with additional sprites they have added.	There are a number of different screens to control either 'code', 'costume' or 'sounds'. Children may find this interface quite difficult to use and may think their code has disappeared. This is included as a teaching point in the slides. Many children also assume that this should be 'easy'. Remind children throughout that coding requires resilience and a 'trial and error' approach.	Children will recap what was achieved in the previous session. They will explore the 'broadcast message' blocks. These allow the children to control the timing and order of the sequence of actions their sprites complete. This will be discussed and modelled to them, and they will then explore themselves. They will be encouraged to debug their code as they go along. **Deepen the moment** Year 5: • See if you can animate your boy and girl sprites to react to the 'broadcast message' blocks. • Experiment with costumes for the other sprites (knight, ghost and bat). What can you achieve by doing this? Year 6: • See if you can animate your boy and girl sprites to react to the 'broadcast message' blocks. • If you haven't already, try to add another haunted sprite and animate it. Control the timing and order of these additional sprites using the 'broadcast message' blocks.
Lesson 3 A backdrop is the 'stage' on which the sprites' actions are performed.	LO: To show and hide sprites in Scratch	NC: Design, write and debug programs that accomplish specific goals, including controlling or	algorithm, sprite, backdrop, costume, block, sequence,	SEND: Children may need the lesson content modelled to them again specifically on their piece of work. Children to use very clearly modelled worksheet to	The Scratch user interface may be confusing when they start to add other backdrops, as this is accessed via clicking on the backdrop. This is	Children will recap what has been covered so far and what they have been able to achieve in their animated story. They will progress to using the 'show' and 'hide' blocks to control when the bat, knight and ghost are seen in the Haunted Castle.

This can be changed.		simulating physical systems. 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.	show, hide, debug	support them with the correct blocks to use. They may only manage to show and hide one sprite rather than all of them. Adding a new backdrop may continue into the next lesson. GD: Children may progress to adding a third backdrop which enhances the story further. This could include the introduction of new sprites, sound effects and costume changes.	included as a teaching point in the slides. Continue to remind children that computing coding requires a lot of patience and resilience. They may not be successful first time and the code they write may need to be constantly debugged.	Following this, the children will learn how to use the 'show' and 'hide' blocks to add another backdrop, which advances the story. They will be encouraged to debug their code as they go along. **Deepen the moment** Year 5: • (Part 1) Swap your device with a partner. Does it work well, or does it need debugging? Can you give some advice? • (Part 2) Add blocks for each of your two main characters to get them to say something before they enter the castle. Year 6: • (Part 1) Swap your device with a partner. Does it work well, or does it need debugging? Can you give some advice? • (Part 2) Add blocks for each of your two main characters to get them to say something before they enter the castle. • (Part 2) See if you can add another backdrop following the castle, which advances the story further.
Lesson 4 A sequence is a set of programming blocks that make actions happen one after another.	LO: To sequence a story on Scratch	NC: Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems. Use sequence, selection, and repetition in programs; work with variables	algorithm, sprite, backdrop, costume, block, sequence, show, hide, broadcast, debug	SEND: Children may still be on the stage of adding one new backdrop. Encourage them to focus on this particular backdrop and add simple code for the sprites, for example getting the sprite to say something. GD: Encourage GD children to support their peers who may be struggling with the	Continue to remind children that computing coding requires a lot of patience and resilience. Their plans may not be successful and they may have to make some changes.	Children will recap what has been covered in the unit of work so far: moving sprites, controlling the timing of events, showing and hiding sprites and adding new backdrops. They will then begin to plan how they will advance the story from here, using a planning sheet. Finally, the children will be given time to put these plans into action by adding new backdrops, adding speech for the characters and applying their learning from the past 4 lessons. They will be encouraged to debug their code as they go along.

		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.		debugging process. Can they successfully coach them through it?		**Deepen the moment** Year 5: • In your books, answer the following questions in full sentences: - What has been most challenging about this Scratch project and why? - What skills do you think you have learnt? Year 6: • In your books, answer the following questions in full sentences: - What has been most challenging about this Scratch project and why? - What have you enjoyed about this Scratch project and why? - What skills do you think you have learnt? - Give an example of where you had to debug and how you fixed it.

Context (big picture learning):
Coding is a very technical skill that requires resilience, patience and determination. Many of us do not know how our smartphones, laptops and video games run, but by giving the children basic coding knowledge, we can open their eyes to the possibilities of coding and what it means for the present as well as the future. Whilst coding is a technical skill, it also requires creativity. Instead of simply playing a video game or using an app, they can imagine making their own game or have a vision for what their own app might look like. Without computing programmers being creative, we would be without many technological advances that we have seen to date. Computer coding empowers children to break problems down into smaller steps and look for solutions through research and collaboration. Communication is an essential skill throughout school, work, and life. People who can clearly communicate complex ideas in simple terms tend to be successful in different industries and occupations.

Link to resources:
Week 1
Week 2
Week 3
Week 4

Knowledge Organiser – Computing (Scratch Animated Stories)

FAT Questions:

How has coding developed over the last century?

What impact has coding had on our daily lives?

How could coding be used in the future to enhance how technology is used?

Did You Know?

The first computer weighed more than 27 tons!

About 90% of the world's currency only exists on computers.

The first known computer programmer was a woman named Ada Lovelace.

The parts for the modern computer were first invented in 1833, with the first computer being built 120 years later.

Modern computers can do billions of calculations in a second.

Most programs are written using a programming language, like C, C++ and Java.

VIPs and Key Vocabulary

- An **algorithm** is a sequence of instructions or a set of rules that are followed to complete a task.
- A **backdrop** is the 'stage' on which all of sprites' actions are completed. This can be changed.
- When an algorithm does not work, or does not do what we expect it to, we must **debug** it. This means to make changes to fix the problem.
- A **sprite** is an object that performs actions on a Scratch project. There are many to choose from.
- A **sequence** is a set of programming blocks that make actions happen one after another.
- An **iteration** is when a sequence of actions is repeated.
- **Broadcasting** means to send out a signal or a message.

Computing - Curriculum Intent

Our Computing lessons prepare you for your future by giving you the opportunities to gain knowledge and develop skills that will equip you for an ever-changing digital world.

Key People

Here are some key people who helped shape how we use technology today. Without them, life would be very different!

Bill Gates

Ada Lovelace

Alan Turing

Mark Zuckerberg

Sir Tim Berners-Lee