



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



PROGRAMMING B – KS1			
Concept	EYFS	YEAR ONE	YEAR TWO
Content (NC with local adaptations)	Make an animal costume for Bee-bot – explore how to use a Bee-bot.	Introduction to animation Designing and programming the movement of a character on screen to tell stories.	Introduction to quizzes Designing algorithms and programs that use events to trigger sequences of code to make an interactive quiz.
Why here?/ Why now?	Continuing to learn that things only move if they are given instructions in a program. This prepares for KS1 programming unit.	To start to understand that we can program things to move objects such as Sprites.	Continues to develop on the Year 1 Scratch Junior unit 'Programming B - Programming animations. To begin to understand sequences of commands have outcomes and make better predictions.
Declarative (factual) Knowledge Children will know that...	-instructions tell us what to do. -things only move if we give instructions in a program. -instructions must be followed in a specific order. This is called a sequence of commands.	-programming is when we make a set of instructions for a computer to follow. -Scratch Jr is a program that we can use in order to code our own stories and animations. It involves sprites (characters on the screen). -we use algorithms (a set of instructions to perform a task) to program the sprite to do different things. -a sprite is a character. The main sprite in Scratch Jr is a cat called Scratch. -the house symbol takes you 'home' to their project screen. -start blocks are yellow/end blocks are red. -a programmed set of instructions is written as a code that makes the sprite perform actions we want it to do.	-programming is when we make a set of instructions for a computer to follow. -Scratch Jr is a program that we can use in order to code our own stories and animations. It involves sprites (characters on the screen). -we use algorithms (a set of instructions to perform a task) to program the sprite to do different things. -a programmed set of instructions is written in code. -different codes can achieve different outcomes. -an outcome is something that happens as a result of us doing something, e.g., we can mix and cook ingredients to make an outcome of food like a cake. In Scratch Jr a sequence of



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



		<ul style="list-style-type: none"> -a code can be written in pictures and words. -instructions can contain more than one sequence. -a command is one instruction. -commands can be repeated. -a command can be repeated to make a Sprite move further. 	<p>commands is followed, and this results in an outcome.</p> <ul style="list-style-type: none"> -in a quiz in Scratch Jr, an outcome occurs when the sprite is clicked. -a programmed quiz needs multiple sprites. -algorithms have to be planned first before they are entered.
<p>Procedural Knowledge to perform tasks.</p> <p>Children will know how to....</p>	<ul style="list-style-type: none"> - listen to instructions. - follow a 2-part sequence. - order a simple sentence. - give one simple instruction for a Beebot (forward, backward, left, right). 	<ul style="list-style-type: none"> -start a new project by clicking on the + . -drag programming blocks into the programming area. -click the block in the programming area makes the sprite perform on the stage. -add a background. -start, run and see what happens at the end of the program. -stack blocks together side by side to create a program made up of sequences. -sequence commands to achieve an outcome. -give a Sprite up to 4 commands. -delete (remove a block from the program by dragging it back to the Blocks Palette. -make something happen more than once by changing the number underneath the block. -run the code (animation) by tapping the full screen icon and then the green flag. -debug the algorithm or program, if the animation does not work correctly first time. 	<p>To program sequences:</p> <ul style="list-style-type: none"> -plan what questions to ask the user, e.g., who lives here? -program each sprite with a command sequence so that they know if they get the answer right or not when clicking on the sprite. -plan and create a simple quiz in Scratch Jr whereby the user clicks on a sprite to select an answer. -add and program sprites so that there are multiple sprites for the user to select from. -add new sprites by choosing the + option. -start a new project by clicking on the + . -drag programming blocks into the programming area. -spot errors in coding and correct them. -design algorithms and programs that use events to trigger sequences of code to make an interactive quiz.
<p>Vocabulary</p>	<p>Image, object, button, see, beebot, move, forward, backward, on, off, sequence, instructions</p>	<p>Scratch Jr, Forwards, backwards, turn, clear, go, commands, instructions, plan, algorithm, program, sprite, blocks, debug.</p>	<p>Sequence, command, program, run, program, start, outcome, predict, blocks, sprite, algorithm, design, sequence, actions, design, modify, change, match, compare, design, debug, program, features, evaluate.</p>



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



<p>Digital Safety Links and knowledge</p> <p>Children will know how to...</p>	<p>- be careful with technology and handle it safely because it is easily breakable.</p> <p>-Bee-bots should only be used on the floor so they don't fall and break.</p>	<p>-only use kind words and pictures when using a program or online platform.</p> <p>-talk to a trusted adult if they see something on screen that makes them feel uncomfortable, upset or worried.</p>	<p>-talk to a trusted adult if they see something on screen that makes them feel uncomfortable, upset or worried.</p>
---	--	---	---



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



PROGRAMMING B – KS2				
Concept	YEAR THREE	YEAR FOUR	YEAR FIVE	YEAR SIX
Content (NC with local adaptations)	Events and actions Writing algorithms and programs that use a range of events to trigger sequences of actions.	Repetition in games Using a block-based programming language to explore count-controlled and infinite loops when creating a game.	Selection in quizzes Exploring selection in programming to design and code an interactive quiz.	Sensing movement Introduction to Micro:bits.
Why here?/ Why now?	To continuing to develop their programming skills taught in KS1 and become familiar with the technical vocabulary used on Scratch.	This unit extends pupils' understanding of sequences and repetition following on from Y3.	This unit combines the elements previously taught and progresses to develop pupil's knowledge of selection by revisiting how conditions can be used in programs and then learning how the If... Then... Else structure can be used to select different outcomes depending on whether a condition is true or false.	This is the final unit of programming where pupils will revisit and bring together elements of all the four programming constructs: sequence from Year 3, repetition from Year 4, selection from Year 5, and variables (introduced in Year 6 – 'Programming A') to overlearn and help keep these in their long-term memory.
Declarative (factual) Knowledge Children will know that...	-an algorithm is a set of instructions to perform a task. -programming is a set of instructions for a computer to follow. -Scratch is a website/app that lets us code our own stories and animations.	-the repetition code simplifies your algorithm. -algorithms can be shortened to make them more efficient. -loops within code avoids repeating commands. -an infinite loop is a sequence of instructions that is on an endless repeat and doesn't stop.	-in programming, conditions are statements that control how a computer performs actions based on certain criteria. -conditions allow programs to make decisions and perform different actions depending on whether a condition is true or false.	-micro:bits are pocket-sized computers that perform different actions based on programs written on computer software. -micro:bits have an LED light display, buttons, sensors and input/output features that we can program. -the rear parts of a micro:bit are: temperature sensor, compass,



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



	<ul style="list-style-type: none"> -the Block Palette contains all the different blocks (commands) which control the animation. -the Code Area is where the blocks are placed to create the program. -Stage with Sprite is where the output of the program is presented. -the sprite is the character. -Event blocks are coloured yellow and are used to sense different events that happen (e.g., when an icon is clicked, or key is pressed). -Event blocks are needed for every project. -action blocks include 'motion' blocks (coloured blue), 'sound' (pink), 'looks' (purple). -Action blocks make the Sprite move, make sounds and change appearance when the event is triggered. 	<ul style="list-style-type: none"> - count-controlled loop repeats a block of code a specific number of times. 	<ul style="list-style-type: none"> -in programming, selection is the process of making a decision that determines the path a program takes next. -'if/then/else' refers to a conditional statement that allows your code to perform different blocks of code depending on whether a certain condition is true or false. 	<ul style="list-style-type: none"> accelerometer (detects movement), radio (communicates with other micro:bits and devices), USB port (connects device to computer), Reset button, battery socket (to power away from a computer) and processor (the 'brain' of the device). -input/output pins connect the micro:bit to other devices. -programs are downloaded to the micro:bit. -micro:bits have a range of input sensors that can trigger different codes to run. -output devices on micro:bits can be programmed to display words, pictures and numbers. -micro:bit will only run the code that has been downloaded. If edited it needs downloading again. -the emulator gives a simulation for testing the code. -a step counter keeps track of how many times a specific action or process has been performed.
<p>Procedural Knowledge to perform tasks.</p>	<ul style="list-style-type: none"> -plan and trial a program of algorithms that includes a range of events that will trigger a sequence of actions (e.g., makes the sprite move, 	<ul style="list-style-type: none"> -shorten an algorithm to make it more efficient. -use loops/ repetition within code to avoid repeating commands. 	<ul style="list-style-type: none"> -identify conditions in a program. -modify conditions in a program. 	<ul style="list-style-type: none"> -change variables on an existing project -develop programs to update the variable by using the micro:bit's



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



<p>Children will know how to....</p>	<p>makes a sound, changes appearance). -use event and action command blocks to make the Sprite react. -debug an algorithm if it does not work first time.</p>	<p>-use the Repeat Block by selecting 'code' and then 'control' blocks (orange) to find the repeat block. Place it around the command blocks they want to repeat. The number of times something is repeated can be typed into the white area. -control the number of loops of a command with the number typed into the repeat block. The forever block makes a command continue infinitely (forever). -code multi-step programs to follow a simple, logical sequence. -identify an error within an algorithm and correct it. -use count-controlled and infinite loops to create a programmed game.</p>	<p>-use the 'if/then/else' structure in algorithms and programs to identify two responses to a yes/no question. -design a storyboard of questions with outcomes for both correct and incorrect answers. -create an algorithm that uses selection to direct the flow of a program. -run the program to test if it correctly uses selection to control the outcomes and debug where needed. -evaluate completed programs and identify ways to improve them.</p>	<p>accelerometer which senses motion. -connect a micro-bit to a computer using a USB cable by locating the downloaded file, copy into the micro:bit drive and run the program. -display a variable after it has been updated or in response to an input. -use the compass to design and create a program to display direction in which the micro:bit is pointing before turning it into a navigational device. - make a micro:bit step counter to design the algorithm and appearance, and to program the flow of the project. -test and debug a code and look for ways to improve the sensitivity of the step counter.</p>
<p>Vocabulary</p>	<p>Motion, event, sprite, algorithm, logic, resize, algorithm, move, extension block, pen up, set up, event, action, debugging, errors, setup, design, code, set up, test, debug.</p>	<p>Scratch, programming, sprite, blocks, code, loop, repeat, value, block, repeat, forever, infinite loop, count-controlled loop, costume, repetition, animate, costume, duplicate, modify, design, sprite, algorithm, debug, refine, evaluate</p>	<p>Selection, condition, if, true, false, count controlled loop, outcomes, conditional statement algorithm, program, debug, input, program, implement, design, test, run, program, setup, selection, share, evaluate, constructive.</p>	<p>Sensing, Micro:bit, step counter, transfer, environment, selection, determine, sense, controllable device, algorithm, bug, emulator, USB, interactive, project, conditions.</p>



LINCOLN ANGLICAN
ACADEMY TRUST
DIOCESE OF LINCOLN



<p>Digital Safety Links and knowledge</p> <p>Children will know how to...</p>	<p>-conduct themselves appropriately when using online platforms for communication.</p> <p>-talk to a trusted adult if they see something online that makes them feel uncomfortable, upset or worried.</p>	<p>-conduct themselves appropriately when using online platforms for communication.</p> <p>-talk to a trusted adult if they see something online that makes them feel uncomfortable, upset or worried.</p>	<p>-demonstrate safe and respectful use of a range of different devices and online services.</p> <p>-be mindful of how actions and words online can impact others.</p>	<p>-demonstrate safe and respectful use of a range of different devices and online services.</p>
--	--	--	--	--