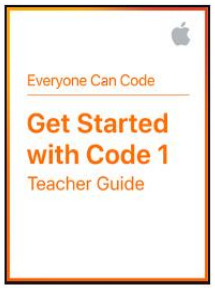
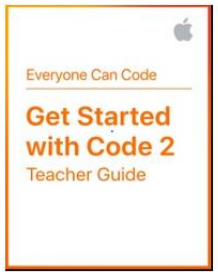
Coding Curriculum knowledge progression

Problem Solving and Logical Thinking

FS	KS1	3/4	5	6
Teacher Guide	Teacher Guide	Teacher Guide	Teacher Guide	Teacher Guide
				
Lesson sequence iPad tinkering Lessons not specifically on the iPad	**Lesson sequence** Year 1 1, 2, 3 Year 2 1, 2, 3, 4	**Lesson sequence** <u>Year 3</u> Code 1 – 4, 6 Code 2 – 1, 2, 3, 4 <u>Year 4</u> Code 1 – 7 Code 2 – 1, 2, 4, 5, 6, 8	**Lesson sequence** 7 8 9 10	**Lesson sequence** 1 2 3 5
Apps 	**Apps** Year 2 	**Apps** 	**Apps** 	**Apps**

Everyone Can Code: A Primary Progression for Programming		
All good coding lessons involve computational thinking skills		
Logical Reasoning This is the process of applying rules to problem solving. This involves algorithms as a set of steps to follow to solve a problem.	Pattern Spotting Pattern spotting involves finding the similarity or patterns among small decomposed problems that can solve more complex problems efficiently.	Décomposition Breaking down complex problems into more manageable parts.
Debugging This is essential to logical reasoning. Debugging is the ability to identify mistakes and fix them.	Evaluating This is the process that allows us to make sure the solution has done the job. It is designed to make you think about how it could be improved and what you did well.	Tinkering Builds on computational thinking and blends creativity, interest and knowledge. Tinkering is the ability to explore programs on your own and at your own pace.
Strands for Progression		
Sequencing Giving commands - KS1 Function - Year 34 Variables - Year 56	Repeat Loops Understanding loop- KS1 Solving problems with loops 3/4 Coding loops 5/6	Event Handling Understanding sequences - KS1 Using event handling to create a game-3/4 Read and write code 5/6
Conditional Statements (if, then, else) Understand actions and consequences - KS1 Using IF statements 3/4 Writing conditional code 5/6	Tinkering Box island, Bee Bots, Box island - KS1 Box Island, Tynker, Code academy -3/4 Swift playground, Scratch, Hour of code 5/6	

	FS	1	2	3	4	5	6
Sequencing	Give commands such as forwards, left and right.	Follow a given sequence including forwards, left and right turns and backwards.	Sequence commands including forwards, back and turns more efficiently using repeat loops.	Use decomposition to break the sequence in to manageable steps.	Sequence commands in Tynker blocks. Use abstraction as a way of making it easier to think about problems Understand how functions help us think more efficiently.	To read code in Swift Code blocks repeat loops and event handling selection. Be able to assess success of given instructions and identify and correct any errors that occur.	To sequence using written Swift Code. To read and write Swift code using: Repeat loops Functions Event handling Selection Variables Be able to evaluate the effectiveness of an algorithm written by their peers in class.
Lessons		Get Started With Code 1 Use Codespark: The Foo's Lesson 1 2 3	Get Started With Code 1 Use Tynker (regular blocks) Lesson 1 2 3	Get Started With Code 2 Tynker (regular blocks) Lesson 1 2 4	Get Started With Code 2 Tynker (Swift blocks) Lesson 1 2 4 5 6	Get Started With Code 2: 8 9 10	Puzzles Lesson 1 , 2, 3

	FS	1	2	3	4	5	6
Repeat Loops	Recognise repeating patterns.	Understand that some instructions need to be repeated eg walk forward 5 spaces.	Understand what a loop is. Understand that a loop can make code more efficient. Code with loops.	Understand that a loop is powerful. Identify loops in everyday life. Solve coding problems with loops.	Solve coding problems with loops. Understand that loops are away of handling conditions that are the same.	Understand nested loops are an efficient way of handling actions that contain other repeating actions. Solve coding problems while using loops and nested loops.	Understand what a FOR loop does. Learn how to code loops.
Lessons			Get started with code 1 Lesson 4	Get started with code 1 Lesson 4 Get started with code 2 Lesson 3	Get started with code 2 Lesson 3, 8	Get started with code 2 Lesson 8	Puzzles For loops lesson 3
Event Handling	Know that when I press go the sequence will run.	Know that when a key is pressed the character will move.	Know that when arrow keys are pressed, direction is determined.	Be able to create an animation or game.	Be able to create an animation or game using parallelism.	Be able to read code in Swift blocks to program events.	Be able to write code in Swift to program events.
Lessons		Get Started With Code 1 Lesson 1 2 3	Get Started With Code 1 Lesson 1 2 3	Get Started With Code 2 Lesson 1 2 4	Get Started With Code 2 Lessons 1 2 4 5 6	Get Started With Code 2 Lessons 8 9 10	Puzzles Lesson 1 2 3

	FS	1	2	3	4	5	6
Conditional Statements **If, then, else**	Understand that one action ca n change the next action. Example: If you do this then that will happen.	Understand consequences of actions and that changing your action can alter the outcome.	Identify which parts of a sequence are flexible.	Understand that an event is an action that causes something to happen. Recognise that events give us options in coding - they cause things to happen if an event occurs.	Understand that we can make actions occur only under certain circumstances. Use the IF statements in everyday life and coding. Coding using IF statements.	Understand that conditional statements are a way of handling different situations. Identify where we use conditional statements in real life. Solve coding problems using conditional statements.	Identify how conditional code applies to everyday life. Understand that conditional statements allow programmers to see logic in their code and are usually written as IF statements.
Lessons			Get started with code 1 Lesson 3	Get started with code 1 Lesson 6	Get started with code 1 Lesson 7	Get started with code 2 Lesson 7	Puzzles Conditional code lesson 5
Tinkering **Apps to apply skills on**	Bee-Bots Bee-Bots app	Box island Bee-Bots Bee Bots app	Box island Tynker Code sparks academy	Hour of code Tynker Box island Code sparks academy	Hour of code Tynker Scratch Code sparks academy	Hour of code Swift playgrounds Scratch	Swift playgrounds Hour of code. Scratch