

Everyone Can Code: A Primary Progression for Programming

All good coding lessons involve computational thinking skills

<p><u>Logical Reasoning</u></p> <p>Logic helps us to establish and check facts and make predictions.</p>	<p><u>Pattern Spotting</u></p> <p>Spotting patterns and similarities to help make predictions, create rules and solve other problems.</p>	<p><u>Decomposition</u></p> <p>Breaking down a problem or system into its parts.</p>
<p><u>Debugging</u></p> <p>Finding and fixing errors.</p>	<p><u>Evaluating</u></p> <p>Making judgements based on different factors such as design criteria or user needs.</p>	<p><u>Tinkering</u></p> <p>Experimenting and playing. Practising and applying skills.</p>
<p>Strands for Progression</p>		
<p><u>Sequencing</u></p> <p>Giving commands - KS1 Function - Year 3/4 Variables - Year 5/6</p>	<p><u>Repeat Loops</u></p> <p>Understanding and using loops- KS1 Using for loops- Year 3/4 Nested loops- Year 5/6</p>	<p><u>Event Handling</u></p> <p>Arrow keys to determine direction- KS1 Create animation or game- Year 3/4 Program events in Swift code- Year 5/6</p>
<p><u>Conditional Statements (if, then, else)</u></p> <p>If statements- KS1 If and else-if on Tynker- Year 3/4 If, else-if and conditions on Swift- Year 5/6</p>	<p><u>Tinkering</u></p> <p>All skills learnt within the year group curriculum are practised, applied and evaluated during these opportunities.</p>	

	FS	1	2	3	4	5	6
Sequencing	Give commands 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 Swift Code 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 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	Use beebots to give directions.	Get Started With Code 1 Use Codespark: The Foos 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: Lessons 8 9 10 Tynker (Swift blocks) Puzzles	Puzzles Lesson 1 , 2, 3

	FS	1	2	3	4	5	6
Repeat Loops	Understand that doing the same thing again is repeating.	Understand what a loop is. Code with loops	Identify where it can make an instruction more efficient. Understand why a loop is powerful.	Identify loops in everyday life. Solve coding puzzles using loops.	Understand how loops help us think more efficiently. Use Forloops (repeat a set of instructions for a certain number of times). Build more efficient algorithms using loops.	Understand while loops as a way to handle conditions that stay the same. Understand nested loops (for actions that contain other repeating loops). Solve puzzles using nested and while loops. Identify nested and while loops in everyday life.	Code using for loops in swift code. Code using while loops in swift code. Demonstrate the use of loops in everyday situations.
Lessons	Repeat instructions on beebots (e.g. press forward 4 times).	Get started with code 1. Lesson 4 (you can do it over and over) Page 34. Use codespark	Get started with code 1. Lesson 4 (you can do it over and over) Page 34. Use Tynker	Get started with code 2. Lesson 3 (think in circles) Page 26. Use Tynker (regular blocks).	Get started with code 2. Lesson 3 (think in circles) Page 26. Use Tynker (swift blocks).	Get started with code 2. Lesson 8 (while and nested loops) Page 57. Use Tynker (swift blocks).	Puzzles page 62 (lesson 3). Page 184- while loops (lesson 9).
Event Handling	Know that when I press go the sequence will run.	Know that a key is pressed the character will move.	Know that then 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	Using beebots. Opening apps on ipads.	Get Started With Code 1 Lesson 1 2 3 Use codespark	Get Started With Code 1 Lesson 1 2 3 Tynker (normal blocks)	Get Started With Code 2 Lesson 1 2 4 Tynker (normal blocks)	Get Started With Code 2 Lessons 1 2 4 5 6 Tynker (swift blocks)	Get Started With Code 2 Lessons 8 9 10 Tynker (swift blocks)	Puzzles Lesson 1 2 3

	FS	1	2	3	4	5	6
Conditional Statements	<p>Know that actions have an outcome. Start to say what will happen if I.....</p> <p>When it is playtime we will.</p>	<p>Understand that we can make actions occur only at a certain time (if something is in place).</p> <p>Use if statements.</p>	<p>Use if statements in everyday life and in coding.</p> <p>Understand how to use if statements in coding.</p>	<p>Identify some situations in real life where we use conditional statements.</p> <p>Use conditional statements in coding.</p> <p>Use if statements.</p> <p>Use else-if statements.</p>	<p>Understand conditional statements as a way of handling different situations.</p> <p>Understand how to use if and else-if statements in code.</p> <p>Give examples of else-if and if statements in real life.</p>	<p>Use if and else-if statements in swift code blocks.</p>	<p>Describe what conditionals are.</p> <p>Code using conditions</p> <p>Loop conditional code.</p>
Lessons	<p>Encourage use of conditional language in the classroom.</p> <p>If, when, then.</p> <p>E.g. If I press this arrow on the beebot what will happen?</p>	<p>Get started with code 1</p> <p>Page 53 – lesson 7</p> <p>Use codespark</p>	<p>Get started with code 1</p> <p>Page 53 – lesson 7</p> <p>Use Tynker</p>	<p>Get started with code 2.</p> <p>Page 50- Lesson 7</p> <p>Use Tynker (normal blocks)</p>	<p>Get started with code 2.</p> <p>Page 50- Lesson 7</p> <p>Use Tynker (Swift blocks)</p>	<p>Get started with code 2.</p> <p>Page 50- Lesson 7</p>	<p>Puzzles</p> <p>Page 107- Lesson 5.</p> <p>Use swift playground</p>

Other resources where coding activities can be found. Once children have learnt the skills they need opportunities to apply these in different situations and ways and on different coding platforms.

<http://code-it.co.uk/csplanning.html> - lots of ideas for tinkering on here.

Code.org for progressively more challenging opportunities to practice key coding concepts.

Barefoot computing

Beebots

Other codespark and Tynker levels

Scratch website