

## Introduction to Coding



Key Stage	English national curriculum – the computer science strand	Rapid Router coding vocabulary	Progression through teaching resources
Key Stage 1 Ages 5–7	<ul style="list-style-type: none"> <li>Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions</li> <li>Create and debug simple programs</li> <li>Use logical reasoning to predict the behaviour of simple programs</li> <li>Use technology purposefully to create, organise, store, manipulate and retrieve digital content</li> <li>Recognise common uses of technology beyond school</li> </ul>	<ul style="list-style-type: none"> <li>Algorithm</li> <li>Program</li> <li>Debug</li> </ul> <p><b>Sequence instructions:</b></p> <ul style="list-style-type: none"> <li>Move forwards</li> <li>Turn left</li> <li>Turn right</li> </ul>	<p><b>Rapid Router levels 1 to 16:</b></p> <ul style="list-style-type: none"> <li>Creating algorithms for physical movement</li> <li>Controlling van on screen app using movement commands</li> <li>Working out the shortest route to a destination</li> </ul>
Key Stage 2 Ages 7–11	<ul style="list-style-type: none"> <li>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</li> <li>Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs</li> <li>Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals</li> </ul>	<ul style="list-style-type: none"> <li>Algorithm</li> <li>Program</li> <li>Debug</li> </ul> <p><b>Sequence instructions:</b></p> <ul style="list-style-type: none"> <li>Move forwards</li> <li>Turn left</li> <li>Turn right</li> </ul> <p><b>Repetition:</b></p> <ul style="list-style-type: none"> <li>Repeat x times</li> </ul>	<p><b>Rapid Router levels 17 to 28:</b></p> <ul style="list-style-type: none"> <li>Understanding the repeat function</li> <li>Creating and evaluating their own challenges and programs using the code skills learnt</li> </ul>
Key Stage 3 (Lower) Ages 11–12	<i>Continued overleaf</i>	<i>Continued overleaf</i>	<i>Continued overleaf</i>

## Introduction to Coding



Key Stage	English national curriculum – the computer science strand	Rapid Router Python coding vocabulary	Progression through teaching resources
Key Stage 3 (Lower) Ages 11 –12	<ul style="list-style-type: none"> <li>Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;</li> <li>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]</li> <li>Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems</li> <li>Understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits</li> <li>Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users</li> <li>create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability</li> <li>Understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.</li> </ul>	<ul style="list-style-type: none"> <li>Algorithm</li> <li>Program</li> <li>Debug</li> </ul> <p><b>Sequence instructions:</b></p> <ul style="list-style-type: none"> <li>Move forwards</li> <li>Turn left</li> <li>Turn right</li> <li>Wait</li> </ul> <p><b>Repetition:</b></p> <ul style="list-style-type: none"> <li>Repeat x times</li> <li>Repeat until</li> </ul> <p><b>Selection:</b></p> <ul style="list-style-type: none"> <li>If... do...</li> <li>If... else if...</li> </ul>	<p><b>Rapid Router levels 19 to 28 (recap):</b></p> <ul style="list-style-type: none"> <li>Understanding the repeat function (recap)</li> <li>Creating and evaluating their own challenges and programs using the code skills learnt</li> </ul> <p><b>Rapid Router levels 29 to 43:</b></p> <ul style="list-style-type: none"> <li>Use the core programming commands appropriately in a visual language</li> <li>Understand the repeat while command</li> <li>Decompose the programming task into smaller parts</li> </ul>

## Introduction to Coding



Key Stage	English national curriculum – the computer science strand	Rapid Router Python coding vocabulary	Progression through teaching resources
<p>Key Stage 3 (Intermediate)</p> <p>Ages 12–13</p>	<ul style="list-style-type: none"> <li>Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;</li> <li>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]</li> <li>Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems</li> <li>Understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits</li> <li>Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users</li> <li>create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability</li> <li>Understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.</li> </ul>	<ul style="list-style-type: none"> <li>Algorithm</li> <li>Program</li> <li>Debug</li> </ul> <p><b>Sequence instructions:</b></p> <ul style="list-style-type: none"> <li>Move forwards</li> <li>Turn left</li> <li>Turn right</li> <li>Wait</li> </ul> <p><b>Repetition:</b></p> <ul style="list-style-type: none"> <li>Repeat x times</li> <li>Repeat until</li> </ul> <p><b>Selection:</b></p> <ul style="list-style-type: none"> <li>If... do...</li> <li>If... else if...</li> </ul> <p><b>Variables:</b></p> <ul style="list-style-type: none"> <li>Traffic lights are red/green</li> </ul>	<p><b>Rapid Router Levels 44–67:</b></p> <ul style="list-style-type: none"> <li>Use the core programming commands appropriately in a visual language</li> <li>Understand the repeat while command</li> <li>Decompose the programming task into smaller parts</li> <li>Identify sections of code which can be used several times and write a procedure for that section</li> <li>Use repeat loops within procedure</li> </ul>

## Introduction to Coding



Key Stage	English national curriculum – the computer science strand	Rapid Router Python coding vocabulary	Progression through teaching resources
Key Stage 3 (Upper) Ages 13-14	<ul style="list-style-type: none"> <li>Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;</li> <li>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]</li> <li>Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems</li> <li>Understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits</li> <li>Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users</li> <li>create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability</li> <li>Understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.</li> </ul>	<ul style="list-style-type: none"> <li>Algorithm</li> <li>Program</li> <li>Debug</li> </ul> <p><b>Sequence instructions:</b></p> <ul style="list-style-type: none"> <li>v.move_forwards()</li> <li>v.turn_left ()</li> <li>v.turn_right ()</li> <li>v.wait ()</li> </ul> <p><b>Repetition:</b></p> <ul style="list-style-type: none"> <li>for count in range (3)</li> </ul> <p>Selection:</p> <ul style="list-style-type: none"> <li>if ..... ():</li> <li>elif ..... ():</li> <li>else :</li> </ul> <p><b>Procedures:</b></p> <ul style="list-style-type: none"> <li>Define – def procname ():</li> <li>Call – procname()</li> </ul> <p><b>Variables:</b></p> <ul style="list-style-type: none"> <li>length = 10</li> <li>length = length +5</li> </ul>	<p><b>Rapid Router Levels 68–109:</b></p> <ul style="list-style-type: none"> <li>Develop an initial understanding of Python as a text based language</li> <li>Understand that Python has precise syntax</li> <li>Identify characteristics of Python and compare this with Blockly</li> <li>Use and understand the movement instructions in Python code</li> <li>Use and understand repeat loops in Python (for count in range (n))</li> <li>Create the core program in visual Blockly and understand it in Python code</li> <li>Understand how the syntax of selection statements works in Python</li> <li>Understand Python while, if, elif, else commands</li> <li>Analyse how procedures work in Python (extension)</li> <li>Write code in Python without the support of Blockly</li> <li>Write simple programs in Python using code for simple movement e.g. v.move_forwards()</li> <li>Use the print command in Python (not available in Blockly)</li> <li>Debug their Python programs, demonstrating an understanding of the appropriate syntax</li> <li>Use indents correctly in Python</li> <li>Use the Repeat loop ... for count in range (n):</li> <li>Design and write programs independently in Python using repetition and selection:</li> <li>for count in range (n): and while, if, elif, else</li> </ul>