

Introduction to Coding



Key Stage	New 2014 Computing 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"> Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions Create and debug simple programs Use logical reasoning to predict the behaviour of simple programs Use technology purposefully to create, organise, store, manipulate and retrieve digital content Recognise common uses of technology beyond school 	<ul style="list-style-type: none"> Algorithm Program Debug <p>Sequence instructions:</p> <ul style="list-style-type: none"> Move forwards Turn left Turn right <p>Repetition:</p> <ul style="list-style-type: none"> Repeat x times 	<p>Rapid Router levels 1 to 28:</p> <ul style="list-style-type: none"> Creating algorithms for physical movement Controlling van on screen app using movement commands Working out the shortest route to a destination Understanding the repeat function Creating and evaluating their own challenges and programs
Lower Key Stage 2 Ages 7–9	<ul style="list-style-type: none"> 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 algorithms and programs 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 	<ul style="list-style-type: none"> Algorithm Program Debug <p>Sequence instructions:</p> <ul style="list-style-type: none"> Move forwards Turn left Turn right Wait <p>Repetition:</p> <ul style="list-style-type: none"> Repeat x times Repeat until <p>Selection:</p> <ul style="list-style-type: none"> If... do... If... else if... <p>Variables:</p> <ul style="list-style-type: none"> Traffic lights are red/green 	<p>Rapid Router levels 29 to 48:</p> <ul style="list-style-type: none"> Understanding the repeat until function Using the if statement to decide on movement along the route Beginning to understand the else if extension to the if statement Using variables in the context of the colour of traffic lights Creating and evaluating their own challenges and programs using the code skills learnt

Introduction to Coding



Key Stage	New 2014 Computing curriculum – the computer science strand	Rapid Router Python coding vocabulary	Progression through teaching resources
<p>Upper Key stage 2</p> <p>Ages 10 –11</p>	<ul style="list-style-type: none"> 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 algorithms and programs Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration 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 can accomplish given goals, including collecting, analysing, evaluating and presenting data and information 	<ul style="list-style-type: none"> Algorithm Program Debug <p>Sequence instructions:</p> <ul style="list-style-type: none"> v.move_forwards() v.turn_left () v.turn_right () v.wait () <p>Repetition:</p> <ul style="list-style-type: none"> for count in range (3) <p>Selection:</p> <ul style="list-style-type: none"> if (): elif (): else : <p>Procedures:</p> <ul style="list-style-type: none"> Define – def procname (): Call – procname() <p>Variables:</p> <ul style="list-style-type: none"> length = 10 length = length +5 	<p>Rapid Router Levels 51–109:</p> <ul style="list-style-type: none"> Use the core programming commands appropriately in a visual language Understand the repeat while command Decompose the programming task into smaller parts Identify sections of code which can be used several times and write a procedure for that section Use repeat loops within procedures Develop an initial understanding of Python as a text based language Understand that Python has precise syntax Identify characteristics of Python and compare this with Blockly Use and understand the movement instructions in Python code Use and understand repeat loops in Python (for count in range (n)) Create the core program in visual Blockly and understand it in Python code Understand how the syntax of selection statements works in Python Understand Python while, if, elif, else commands Analyse how procedures work in Python (extension) Write code in Python without the support of Blockly Write simple programs in Python using code for simple movement e.g. v.move_forwards() Use the print command in Python (not available in Blockly) Debug their Python programs, demonstrating an understanding of the appropriate syntax Use indents correctly in Python Use the Repeat loop ... for count in range (n): Design and write programs independently in Python using repetition and selection: <ul style="list-style-type: none"> for count in range (n): and while, if, elif, else

Introduction to Coding



Key Stage	New 2014 Computing curriculum – the computer science strand	Rapid Router Python coding vocabulary	Progression through teaching resources (continued)
Upper Key stage 2 Ages 10 –11			<p>Rapid Router Levels 51–109:</p> <ul style="list-style-type: none"> • Use comments in Python to explain how the program works <p>Extension:</p> <ul style="list-style-type: none"> • Define new procedures in Python (also called functions) • Design and write programs independently in Python using repetition and selection: • for count in range (n): • and while, if, elif, else • Debug their Python programs, demonstrating an understanding of the appropriate syntax • Use indents correctly in Python • Create and increment variables • Use comments in Python to explain their programming