<table>
<thead>
<tr>
<th>Key Stage</th>
<th>English national curriculum – the computer science strand</th>
<th>Rapid Router coding vocabulary</th>
<th>Progression through teaching resources</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Key Stage 1</strong>&lt;br&gt;Ages 5–7</td>
<td>• Understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions&lt;br&gt;• Create and debug simple programs&lt;br&gt;• Use logical reasoning to predict the behaviour of simple programs&lt;br&gt;• Use technology purposefully to create, organise, store, manipulate and retrieve digital content&lt;br&gt;• Recognise common uses of technology beyond school</td>
<td>• Algorithm&lt;br&gt;• Program&lt;br&gt;• Debug&lt;br&gt;<strong>Sequence instructions:</strong>&lt;br&gt;• Move forwards&lt;br&gt;• Turn left&lt;br&gt;• Turn right</td>
<td>Rapid Router levels 1 to 16:&lt;br&gt;• Creating algorithms for physical movement&lt;br&gt;• Controlling van on screen app using movement commands&lt;br&gt;• Working out the shortest route to a destination</td>
</tr>
<tr>
<td><strong>Key Stage 2</strong>&lt;br&gt;Ages 7–11</td>
<td>• 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&lt;br&gt;• Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs&lt;br&gt;• 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</td>
<td>• Algorithm&lt;br&gt;• Program&lt;br&gt;• Debug&lt;br&gt;<strong>Sequence instructions:</strong>&lt;br&gt;• Move forwards&lt;br&gt;• Turn left&lt;br&gt;• Turn right&lt;br&gt;<strong>Repetition:</strong>&lt;br&gt;• Repeat x times</td>
<td>Rapid Router levels 17 to 28:&lt;br&gt;• Understanding the repeat function&lt;br&gt;• Creating and evaluating their own challenges and programs using the code skills learnt</td>
</tr>
<tr>
<td><strong>Key Stage 3</strong>&lt;br&gt;(Lower)&lt;br&gt;Ages 11–12</td>
<td>Continued overleaf</td>
<td>Continued overleaf</td>
<td>Continued overleaf</td>
</tr>
</tbody>
</table>
# Key Stage 3 (Lower)
Ages 11 – 12

- Understand simple Boolean logic (e.g., AND, OR and NOT) and some of its uses in circuits and programming;
- Understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers (e.g., binary addition, and conversion between binary and decimal);
- Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems;
- 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;
- 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;
- Create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability;
- 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.

## English national curriculum – the computer science strand

**Rapid Router Python coding vocabulary**

- **Sequence instructions:**
  - Move forwards
  - Turn left
  - Turn right
  - Wait
- **Repetition:**
  - Repeat x times
  - Repeat until
- **Selection:**
  - if... do...
  - if... else if...

**Rapid Router levels 19 to 28 (recap):**
- Understanding the repeat function (recap)
- Creating and evaluating their own challenges and programs using the code skills learnt

**Rapid Router levels 29 to 43:**
- Use the core programming commands appropriately in a visual language
- Understand the repeat while command
- Decompose the programming task into smaller parts

---

*www.codeforlife.education*
### Key Stage 3 (Intermediate)  
**Ages 12–13**

- Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;
- 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);
- Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems;
- Understand how instructions are stored and executed within a computer system, and understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits;
- 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;
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability;
- 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.

### Rapid Router Python coding vocabulary

- Algorithm
- Program
- Debug

### Progression through teaching resources

**Rapid Router Levels 44–67:**

- 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 procedure.

---

<table>
<thead>
<tr>
<th>Key Stage</th>
<th>English national curriculum – the computer science strand</th>
<th>Rapid Router Python coding vocabulary</th>
<th>Progression through teaching resources</th>
</tr>
</thead>
</table>
| Key Stage 3 (Intermediate)  
**Ages 12–13** | - Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;  
- 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);  
- Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems;  
- Understand how instructions are stored and executed within a computer system, and understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits;  
- 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;  
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability;  
- 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. | - Algorithm  
- Program  
- Debug | - 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 procedure. |
<table>
<thead>
<tr>
<th>Key Stage</th>
<th>English national curriculum – the computer science strand</th>
<th>Rapid Router Python coding vocabulary</th>
<th>Progression through teaching resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Stage 3</td>
<td>• Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;</td>
<td>• Algorithm</td>
<td>Rapid Router Levels 68–109:</td>
</tr>
<tr>
<td>(Upper) Ages 13-14</td>
<td>• 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)</td>
<td>• Program</td>
<td>• Develop an initial understanding of Python as a text based language</td>
</tr>
<tr>
<td></td>
<td>• Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems</td>
<td>• Debug</td>
<td>• Understand that Python has precise syntax</td>
</tr>
<tr>
<td></td>
<td>• 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</td>
<td>Sequence instructions:</td>
<td>• Identify characteristics of Python and compare this with Blockly</td>
</tr>
<tr>
<td></td>
<td>• 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</td>
<td>• v.move_forwards()</td>
<td>• Use and understand the movement instructions in Python code</td>
</tr>
<tr>
<td></td>
<td>• create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability</td>
<td>• v.turn_left()</td>
<td>• Use and understand repeat loops in Python (for count in range (n))</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
<td>• v.turn_right()</td>
<td>• Create the core program in visual Blockly and understand it in Python code</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• v.wait()</td>
<td>• Understand how the syntax of selection statements works in Python</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Repetition:</td>
<td>• Understand Python while, if, elif, else commands</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• for count in range (3)</td>
<td>• Analyse how procedures work in Python (extension)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Selection:</td>
<td>• Write code in Python without the support of Blockly</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• if ... ():</td>
<td>• Write simple programs in Python using code for simple movement e.g. v.move_forwards()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• elif ... ():</td>
<td>• Use the print command in Python (not available in Blockly)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• else :</td>
<td>• Debug their Python programs, demonstrating an understanding of the appropriate syntax</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Procedures:</td>
<td>• Use indents correctly in Python</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Define – def procname():</td>
<td>• Use the Repeat loop … for count in range (n):</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Call – procname()</td>
<td>• Design and write programs independently in Python using repetition and selection:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Variables:</td>
<td>• for count in range (n): and while, if, elif, else</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• length = 10</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• length = length +5</td>
<td></td>
</tr>
</tbody>
</table>