

|                         |   |                 |   |   |                        |                 |  |                      |                        |  |   |                      |   |                 |   |  |                        |                 |   |                      |
|-------------------------|---|-----------------|---|---|------------------------|-----------------|--|----------------------|------------------------|--|---|----------------------|---|-----------------|---|--|------------------------|-----------------|---|----------------------|
| <b>Computing Year 8</b> | <b>Curriculum intent:</b> The year 8 curriculum builds on the achievements of year 7 and consolidates that understanding through the use of interleaving the key concepts listed below and tasks to scaffold the students' learning experiences. As in year 7, new concepts and techniques are acquired through a combination of teacher led tasks, class discussions and practical activities. These are designed to increase the student understanding of the concepts, allowing them to apply their new found understanding in a variety of different scenarios related to the concepts being taught in an engaging way. This approach not only allows for co-ordinated efforts of students to be encouraged but also makes use of numerous open ended tasks that allow all students to stretch their understanding of these concepts as far as they are able. |                 |   |   |                        |                 |  |                      |                        |  |   |                      |   |                 |   |  |                        |                 |   |                      |
| <b>Topic</b>            | <b>HTML</b>   |                 |   | <b>Python</b>   |                        |                 | <b>MicroBit</b>  |                      |                        | <b>Networks</b>  |   |                      | <b>Project</b>  |                 |   | <b>Cryptography</b>  |                        |                 |   |                      |
| <b>Interleaving</b>     | Key knowledge from previously studied topics  |                 |   | Key knowledge from previously studied topics  |                        |                 | Key knowledge from previously studied topics   |                      |                        | Key knowledge from previously studied topics   |   |                      | Key knowledge from previously studied topics  |                 |   | Key knowledge from previously studied topics   |                        |                 |   |                      |
| <b>Knowledge</b>        | Students will learn how to use and combine multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users. They will create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability.  |                 |   | Students will learn how to solve a variety of computational problems; make appropriate use of data structures; design and develop modular programs that use procedures and functions. They will understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem.   |                        |                 | Students will extend learning of MicroBit programming from year 7 to include knowledge of physical programming and the ways in which we can use a microcontroller in interact with the physical world.                     |                      |                        | Students will learn the hardware and software components that make up computer systems, and how they communicate with one another and with other systems. They will learn a range of ways to use technology safely, respectfully, responsibly and securely.  |   |                      | Students will study several key algorithms that reflect computational thinking (for example, ones for sorting and searching); use logical reasoning to compare the utility of alternative algorithms for the same problem.  |                 |   | Students will learn 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.  |                        |                 |   |                      |
| <b>Understanding</b>    | Students will be able to: write HTML code to create a simple web page and display it in a browser.<br>Use a range of HTML tags to create well laid out web pages<br>Write CSS code to define the styles of different parts of a web page<br>Add enhancements or additional features to the original basic design<br>Construct a good-looking, well-formatted interactive website that is suitable for its intended audience   |                 |   | Students will be able to: run simple Python programs in Interactive and Script mode.<br>Write pseudocode to outline the steps in an algorithm prior to coding.<br>Write an error-free, well-documented program involving selection and iteration.<br>Describe how a binary search is carried out.<br>Devise their own algorithms to solve reasonably complex problems, e.g. a binary search.<br>Test and debug their programs, and correct both syntax and logic errors |                        |                 | Students will be able to: demonstrate understanding of the topic through the creation of a number of practical projects both individually and in groups covering topics such as LEDs, music, radio, sensors and Bluetooth. |                      |                        | Students will be able to: define the meaning of the terms "domain name", http protocol.<br>Explain the basic principle of packet switching.<br>Explain the meaning and significance of bandwidth.<br>Explain what is meant by buffering and why it is used.<br>Design a network layout for their school, using icons to represent server, hub, switch, router, Internet, workstation, printer. |   |                      | Students will be able to: select the most suitable type of loop (for or while) for a given problem.<br>Use counters correctly in conjunction with for loops.<br>Create a list and append or change elements of the list.<br>Explain the advantages of functions for reusable sections of program code.<br>Devise their own functions to create a modular program. |                 |   | Students will be able to: demonstrate knowledge of some of the dangers of putting personal data on social networking sites & ways of protecting online identity.<br>Identify some of the signs of fraudulent emails and respond appropriately.<br>Recognise fraudulent emails and protect themselves. effectively from unwittingly giving personal information (e.g. account numbers and passwords) or otherwise being defrauded.<br>Describe the effects on individuals and companies of illegally downloading copyright material, e.g. music, images and movies. |                        |                 |   |                      |
| <b>Skills</b>           | Computational Thinking  | Problem Solving | Analysis, Evaluation and Implementation | Technical Vocabulary  | Computational Thinking | Problem Solving | Analysis, Evaluation and Implementation  | Technical Vocabulary | Computational Thinking | Problem Solving  | Analysis, Evaluation and Implementation | Technical Vocabulary | Computational Thinking  | Problem Solving | Analysis, Evaluation and Implementation | Technical Vocabulary   | Computational Thinking | Problem Solving | Analysis, Evaluation and Implementation | Technical Vocabulary |
| <b>Assessment</b>       | Pupils will put evidence of their final website in an Assessment Portfolio. They will also answer questions on HTML, CSS and web design principles to demonstrate understanding.  |                 |   | Pupils will write and run a program and submit the code and screenshots of the program running in a learning Portfolio.   |                        |                 | Pupils will write and run a program and submit the code and screenshots of the program running in an Assessment Portfolio.   |                      |                        | Pupils will sit an end of unit test as their final assessment.   |   |                      | Pupils will sit an end of unit test as their final assessment.  |                 |   | Pupils will sit an end of unit test as their final assessment.   |                        |                 |   |                      |