

KS3 Curriculum Computing – Long term overview

The national curriculum for computing aims to ensure that all pupils:

- can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- are responsible, competent, confident and creative users of information and communication technology.

Year 7	Focus / Topic	Knowledge & Skills	Assessment
Aut 1 7.0	Digital Literacy	Details of our IT learning resources, usernames and passwords.	Verification of class and home learning tasks.
Aut 1 7.1	Using media – Gaining support for a cause	Creating a digital product for a real-world cause	Evaluation of work against a rubric and summative assessment.
.Aut 2 7.2	Networks from semaphores to the Internet	Recognising networking hardware and explaining how networking components are used for communication.	Summative assessment.
Spr 1 7.3	Programming essentials in Scratch – part I	Applying the programming constructs of sequence, selection, and iteration in Scratch.	Evaluation of work against a rubric.
Spr 2 7.4	Spreadsheets	Sorting and filtering data and using formulas and functions in spreadsheet software.	Summative assessment.
Sum 1 7.5	Programming essentials in Scratch – part II	Using subroutines to decompose a problem that incorporates lists in Scratch.	Summative assessment.
Sum 2 7.6	Clear messaging in digital media	Combining the use of digital tools and online collaboration to produce media	Evaluation of work against a rubric.

Year 8	Focus / Topic	Knowledge & Skills	Assessment
Aut 1 8.1	Media - Vector graphics	Creating vector graphics through objects, layering, and path manipulation.	Evaluation of work against a rubric and summative assessment.
Aut 2 8.2	Layers of computing systems	Exploring the fundamental elements that make up a computer system.	Summative assessment.
Spr 1 8.3	Developing for the web	Using HTML and CSS to create webpages.	Summative assessment.
Spr 2 8.4	Representations: from clay to silicon	Representing numbers and text using binary digits.	Summative assessment.
Sum 1 8.5	Mobile app development	Using event-driven programming to create an online gaming app.	Evaluation of work against a rubric and summative assessment.
Sum 2 8.6	Introduction to Python programming	Applying the programming constructs of sequence, selection, and iteration in Python.	Summative assessment.

Year 9	Focus / Topic	Knowledge & Skills	Assessment
Aut 1 9.1	Python programming with sequences of data	Manipulating strings and lists. Creating a programming project.	Summative assessment.
Aut 2 9.2	Media – Animations	Creating 3D animations through object manipulation, and tweaking and adjusting lighting and camera angles.	Self-assess against a set of skills, and peer-assessment on completion.
Spr 1 9.3	Data science	Using data to investigate problems and make real-world changes.	Evaluation of work against a rubric and summative assessment.
Spr 2 9.4	Representations – going audiovisual	Representing images and sound using binary digits.	Summative assessment.

Year 9	Focus / Topic	Knowledge & Skills	Assessment
Sum 1 9.5	Introduction of cybersecurity	Identifying how users and organisations can protect themselves from cyberattacks.	Summative assessment.
Sum 2 9.6	Applying programming skills with physical computing	Sensing and controlling with the micro:bit.	Evaluation of work against a rubric and summative assessment.

National Curriculum Coverage	7.1	7.2	7.3	7.4	7.5	7.6	8.1	8.2	8.3	8.4	8.5	8.6	9.1	9.2	9.3	9.4	9.5	9.6
1. Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems				✓			✓				✓	✓	✓					✓
2. 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			✓		✓		✓				✓	✓	✓					✓
3. Use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions			✓		✓							✓	✓					✓

National Curriculum Coverage	7.1	7.2	7.3	7.4	7.5	7.6	8.1	8.2	8.3	8.4	8.5	8.6	9.1	9.2	9.3	9.4	9.5	9.6
4. 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]			✓		✓			✓										
5. Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems		✓						✓										
6. 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								✓		✓		✓	✓			✓		✓
7. 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				✓		✓									✓			
8. Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	✓		✓		✓	✓	✓		✓		✓			✓				

National Curriculum Coverage	7.1	7.2	7.3	7.4	7.5	7.6	8.1	8.2	8.3	8.4	8.5	8.6	9.1	9.2	9.3	9.4	9.5	9.6
9. 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.	✓																✓	