

Programme of study for Year 7 Computer Science and IT

Autumn (1 ^{term}) Topic	Autumn (2 ^{term}) Topic	Spring (1 ^{term}) Topic	Spring (2 ^{Term}) Topic	Summer (1 ^{term}) Topic	Summer (2 ^{term}) Topic
Using School Systems and Internet Safety (6) Getting operational with school IT systems – logging in, navigation and file handling – before learning to recognise personal information, spot online risks and evaluate websites critically.	Word Basics and PowerPoint (6) Developing core word-processing skills – drafting, formatting and editing – then applying the same toolbar instincts to build a first interactive presentation in PowerPoint.	Modelling Data with Excel (6) Building spreadsheet skills from entering and formatting data through to formulas, functions and charts, before applying these skills to analyse real-life data.	Computer Hardware and Software (6) Exploring the components inside a computer, how data moves between them, and the difference between hardware, software, and system vs application software.	Algorithms and Programming with Flowol (6) Learning what an algorithm is and creating flowchart-based programs in Flowol to control simulations of real-life automatic systems.	Programming in Scratch (6) Building block-based programming skills in Scratch – from sprites and events through to variables – culminating in a final project and a first look at text-based code.
Skills: Logging in to and navigating school systems confidently. Saving, naming and managing files. Recognising personal information and identifying online risks. Evaluating the reliability of a website.	Skills: Navigating Word and formatting a document (fonts, spell-check, tables). Drafting and editing written work. Building a presentation and adding animations and transitions in PowerPoint.	Skills: Entering and formatting data in a spreadsheet. Writing formulas and using functions (e.g. SUM, AVERAGE). Creating charts to visualise data. Spotting and exploring patterns in data.	Skills: Identifying and describing the role of computer hardware components. Distinguishing between hardware and software. Classifying system software and application software.	Skills: Analysing a problem and creating an algorithmic solution to solve it. Reading and writing flowcharts. Using sequence, selection and iteration.	Skills: Sequencing and using loops (repeat blocks). Using selection (if-then decisions). Storing and using data with variables. Comparing block code to its text-based equivalent.
Key Learning Outcomes: Use school systems independently, including login, navigation and basic file handling. Explain what personal information is and why it needs protecting online. Identify common online risks and how to respond to them safely.	Key Learning Outcomes: Navigate Word's toolbar to create, format and edit a document. Use spell-check and tables to refine written work. Build a presentation in PowerPoint and add animations/transitions to make it interactive.	Key Learning Outcomes: Navigate a spreadsheet and enter and format data. Write basic formulas and use functions such as SUM and AVERAGE. Create charts to visualise data. Analyse real-life data and identify patterns.	Key Learning Outcomes: Recognise, name and describe the function of key hardware components (e.g. CPU, RAM, storage). Understand how data moves between components.	Key Learning Outcomes: Understand how information flows through a computer system using algorithms and flowcharts. Build, test and refine flowchart-based programs in Flowol, including decisions (selection).	Key Learning Outcomes: Use Scratch's interface to control sprites and respond to events. Apply sequence, selection and iteration using block code, linking back to the flowchart work from earlier this term.

Autumn (1 ^{term}) Topic	Autumn (2 ^{term}) Topic	Spring (1 ^{term}) Topic	Spring (2 ^{Term}) Topic	Summer (1 ^{term}) Topic	Summer (2 ^{term}) Topic
Practise typing skills.	Practise typing skills.	Practise spreadsheet skills using free online Excel exercises.	Learn about the history of computing.	Practise designing flowcharts for everyday processes (e.g. making a cup of tea).	Explore online tutorials for Scratch.
Reading / High Quality Text: Evaluating a website – Tree Octopus task https://zapatopi.net/treeoctopus/ Literacy: Read the information website – take notes. Students learn about evaluating information and how to tell fact from fiction on the web.	Reading / High Quality Text: Basic tasks in Word – Microsoft Support https://support.microsoft.com/en-gb/office/basic-tasks-in-word-87b3243c-b0bf-4a29-82aa-09a681999fdc Basic tasks for creating a PowerPoint presentation – Microsoft Support https://support.microsoft.com/en-us/office/basic-tasks-for-creating-a-powerpoint-presentation-efbbc1cd-c5f1-4264-b48e-c8a7b0334e36 Literacy: Technical terms related to word processing and presenting.	Reading / High Quality Text: Guide and tutorial on how to use Excel https://www.w3schools.com/excel/index.php Literacy: Technical terms related to Excel.	Reading / High Quality Text: Understanding Computers – BBC Bitesize KS3 Computer Science https://www.bbc.co.uk/bitesize/subjects/zvc9q6f Literacy: Technical terms related to computer hardware and software.	Reading / High Quality Text: Flowol – flowchart control software https://flowol.com/ Algorithms – BBC Bitesize KS3 Computer Science https://www.bbc.co.uk/bitesize/guides/zpp49j6/review/3 Literacy: Technical terms related to algorithms and flowcharts (sequence, selection, iteration).	Reading / High Quality Text: Guide on getting started with Scratch, from dragging out your first blocks of code to creating your own sprites https://sip.scratch.mit.edu/scratchathome/ Literacy: Technical terms related to programming in Scratch.
Numeracy: Validation – time and date (login records).	Numeracy: Font size, tables, and using a timer for animations and slide transitions.	Numeracy: Data analysis – applying mathematical concepts in real-world scenarios, using calculations and exploring patterns.	Numeracy: Data storage units – comparing capacities (bytes, KB, MB, GB).	Numeracy: Counting and analysing algorithm steps; use of iteration.	Numeracy: Numerical interaction – measuring and adjusting sprite movement, scoring or timing, simple calculations, and Boolean logic (and, or, not).

Autumn (1 ^{term}) Topic	Autumn (2 ^{term}) Topic	Spring (1 ^{term}) Topic	Spring (2 ^{Term}) Topic	Summer (1 ^{term}) Topic	Summer (2 ^{term}) Topic
Enrichment / opportunities to develop cultural capital (including careers, WRL and SMSC): Participate in coding challenges online. Attend tech-related conferences or workshops.					