

Programme of study for Year 7 Computer Science

Autumn (1 st term) Topic	Autumn (2 nd term) Topic	Spring (1 st term) Topic	Spring (2 nd Term) Topic	Summer (1 st term) Topic	Summer (2 nd term) Topic
<p>Computer Components (1) Components of a computer (CPU, RAM, ROM, HDD, SSD, NIC, Motherboard).</p> <p>Input/Output devices. (1) Keyboard, mouse, camera, microphone, printer, 3D printer, speakers, VDU.</p> <p>Storage devices (1) HDD, SSD, USB, Cloud, CD, DVD</p>	<p>Algorithms (1) Using algorithms to solve problems. Using Flowcharts to represent algorithms.</p> <p>Programming:(2) Introduction to programming algorithm solutions using Flowol. Use simulations of real life automatic computer systems</p>	<p>Programming (3) Introduction to a text-based language (Python Turtle) Sequencing, iteration & procedures.</p>	<p>Programming Basics constructs :(3)</p> <ul style="list-style-type: none"> - Variables - Operators =-/*, - Data types string, real, integer, Boolean, character. - inputs and casting - Sequencing 	<p>Create Vector Graphics Understand how vector graphics differ from bitmap images. Create a logo for a company or Organisation</p>	<p>Programming:(3) Scratch: Learning the basics to become familiar with a different programming language and block code</p>
<p>Skills: Recognise, name and describe the roles of computer hardware components and input, output and storage devices.</p>	<p>Skills: To analyse a problem and create an algorithmic solution to solve it. To translate an algorithm into a working program using Flowol. Debug errors in a program.</p>	<p>Skills: Computational thinking and coding. Basic python syntax, turtle graphics commands, loops, creative problem-solving for visual programming.</p>	<p>Skills: Computational thinking and logical thinking to solve problems step –by –step. Algorithms, variables, loops, input/output, debugging, and problem solving through coding.</p>	<p>Skills: Critical thinking, creativity problem-solving. Manipulate shapes in a vector art package using layers.</p>	<p>Skills: Logical thinking and develop problem-solving skills by designing programs that respond dynamically to user input. Sequencing, variables, selection and count controlled iteration.</p>
<p>Key Learning Outcomes: Recognise and describe the purpose of computer components.</p> <p>Identify and provide examples of input, output, and storage devices.</p>	<p>Key Learning Outcomes: Identify the flowchart symbols and understand the process of each stage.</p> <p>Solve problems using step-by-step algorithms.</p>	<p>Key Learning Outcomes: Grasp programming concepts.</p> <p>Develop a foundational understanding of python syntax, data types and variables.</p>	<p>Key Learning Outcomes: Identify and use basic programming syntax in python to write simple programs.</p> <p>Defining and using variables to store and manipulate data.</p>	<p>Key Learning Outcomes: Create simple vector shapes such as lines, rectangles, circles and polygons.</p> <p>Edit vector paths and apply colour to the objects.</p>	<p>Key Learning Outcomes: Construct sequences of code.</p> <p>Creating, modifying, and control sprites and the stage within the Scratch environment.</p>

Explain how a CPU works, including input, processing, and output functions.	Create a program to control a sequence with variables.	Apply the concepts to create programs for drawings and designs.	Ability to create and apply conditional statements for decision making programs.	Understand the use of layers to manage the vector elements.	Grasp the concept of algorithms and how to break down tasks into logical steps.
Term 1 Evidence to cover: Computer components, Input Output devices, Algorithms and Flowcharts.		Term 2 Evidence to cover: interpretation of code, data types and operators		Term 3 Evidence to cover: Creation of a graphic, Creating coded animation	
Rationale for sequence: To provide an understanding of the elements in a computer system. For student to comprehend the technology they use daily.	Rationale for sequence: After studying computer systems, we now develop problem-solving skills and logical thinking. This helps students tackle complex tasks in programming.	Rationale for sequence: Introduce programming in an engaging and visual way, encourage logical thinking, creativity, and the development of essential coding skills.	Rationale for sequence: Building on the programming concepts explored last term, we learn the skills used to make more complex programs and being able to use python operators and constructs confidently.	Rationale for sequence: The logo will in Year 8 website creation for visual content to enhance students' digital literacy. Graphic can also be produced for scratch project.	Rationale for sequence: Using a block-based coding language to create an animation. This is to engage students in coding by making it fun and interactive.
Home – Learning: Write a list of computers that can be found in a home. Explore the history of computing.	Home – Learning: Design your own Flowchart: Think of the sequence of instructions for a task. For example, it could be something done in the kitchen, part of playing a game or how to use an electronic device.	Home – Learning: Online Python Turtle tutorials and challenges on Codeacademy.	Home – Learning: Python coding challenges on W3Schools.	Home – Learning: Design a vector-based artwork to promote a product.	Home – Learning: Online tutorials on Scratch.
Reading / High Quality Text: How tech is reinventing healthcare https://www.wired.co.uk/article/future-of-health Literacy:	Reading / High Quality Text: How Search Algorithms Are Changing the Course of Mathematics https://getpocket.com/explore/item/how-search-algorithms-are-changing-the-course-of-	Reading / High Quality Text: Guide on how to use python turtle https://realpython.com/beginners-guide-python-turtle/	Reading / High Quality Text: Guide and tutorial on how to use python https://www.w3schools.com/python/ Literacy:	Reading / High Quality Text: What is vector art: https://www.adobe.com/uk/creativecloud/illustration/discover/vector-art.html	Reading / High Quality Text: Guide on from dragging out your first blocks of code to creating your own sprites

<p>Watch a video and compose a tweet explaining how computers function</p> <p>Technical terms related to computer systems.</p>	<p>mathematics?utm_source=pocket-newtab-global-en-GB</p> <p>Literacy: Technical terms related to program flow and flowcharts.</p> <p>Write algorithm explanations.</p>	<p>Literacy: Technical terms related to programming and python turtle.</p> <p>Document code with comments.</p>	<p>Technical terms related to programming in python</p> <p>Document code with comments.</p>	<p>Literacy: Technical terms related to vector art.</p>	<p>https://sip.scratch.mit.edu/scratchathome/</p> <p>Literacy: Technical terms related to programming with scratch.</p>
<p>Numeracy: Storage capacities</p>	<p>Numeracy: Count and analyse algorithm steps. Use of iteration</p>	<p>Numeracy: Basic geometry Dimensions and Angles</p>	<p>Numeracy: Basic calculations: Addition, subtraction, multiplication, and division. Integer, float</p>	<p>Numeracy: Numerical dimensions. Measure and adjust proportions in your graphics.</p>	<p>Numeracy: Numerical interaction – scoring or timing. Simple calculations. Boolean Logic(and, or, and not)</p>
<p>Enrichment / opportunities to develop cultural capital (including careers, WRL and SMSC): Participate in coding challenges online. Attend tech-related conferences or workshops.</p>					