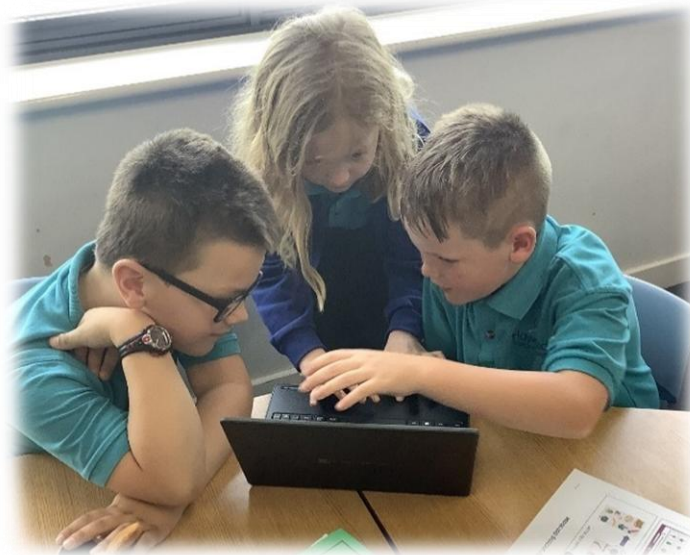




Cabot
Learning
Federation

I am a computer scientist

I am a computer scientist. I study computers and computing, including their theoretical and algorithmic foundations, hardware and software, and their uses for processing information. As a computer scientist, I am developing innovative ways of exploring information, and what I am able to do with it: enhancing my own thinking skills through algorithms, logic, visualisation, precision, and abstraction. I understand that computational thinking involves and illuminates other disciplines and the collaboration of these are vital, when addressing the advances in technology, and the inventive ways to use it. I am interested in pushing the boundaries of invention and transformation, using technology to creatively solve problems around me.



	Key Concepts	Contexts
<p>What is computing?</p> <p>Algorithms have been used for around 4000 years, with evidence in Sumerian tablets displaying examples of division. People have used tools, such as an abacus, to help with logical thinking for 2000 years. Over time, these tools and algorithms have developed in complexity and the first modern computer was designed by Charles Babbage in the 1820s and 30s.</p> <p>Slow and massive machines have developed into more efficient and compact devices we know today. The versatile nature of this technology means that it is in constant use and is ever changing but in essence runs on similar mathematical principals as the original machines. The study and use of these machines and their algorithms is computer science.</p>	<p>Computer components and resultant hardware</p> <p>Computers are electronic devices used for storing or processing data (information). They can be used by inputting, storing, and outputting data. To do this, computer programs are written by coders (programmers) and can be very sophisticated. Therefore, computers do not think and cannot make decisions by themselves, rather the software (programs to do a specific job) enables users to interact to give (input) and receive (output) data.</p>	<p>Y1: Using computers Y2: How computers work and the internet Y3: Global computers Y4: Networks and data Y5: Computing & Security Y6: Compatibility and design</p>
	<p>Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.</p> <p>To be digitally literate is to have access to a broad range of practices and cultural resources that you are able to apply to digital tools. It is the ability to make and share meaning in different modes and formats; to create, collaborate and communicate effectively and to understand how and when digital technologies can best be used to support these processes.</p>	<p>Y1: Digital Painting and Digital Writing Y2: Digital Photography and Digital Music Y3: Stop Frame Animation and Desktop Publishing Y4: Audio Production and Photo Editing Y5: Video Production and Vector Graphics Introduction Y6: Web Page Creation and 3-D Modelling</p>
	<p>Computational Thinking</p> <ul style="list-style-type: none"> • Understanding and using algorithms: a sequence of specific, ordered instructions to complete a task • De-bugging: finding and resolving bugs (defects or problems that prevent the correct operation) • Abstraction: removing or filtering out all unnecessary information to focus on what we need to solve a problem • Decomposition: breaking down a problem into smaller, easier to solve parts • Pattern Recognition: recognising patterns within outcomes 	<p>Y1: Introduction to algorithms Y2: Knowing and exploring algorithms Y3: Programming Y4: Being a Programme Engineer Y5: Complex Programming with conditions and patterns Y6: Mastering Programming</p>
	<p>Technology should be used safely, respectfully, and responsibly.</p> <p>Electronic safety (e-safety) incorporates:</p> <ul style="list-style-type: none"> • Safe, respectful, responsible use of technology • Identifying dangers • Know where or who to contact if they have concerns 	<p>E-safety is explicitly taught within the CLF Citizenship/PSHE curriculum. Opportunities to develop pupils e-safety are expected to be referred to within the teaching of Computer Scientists.</p>

Key Concept	Key Knowledge and Vocabulary		Skills: Techniques & Application	
Computer components and resultant hardware	Using computers <ul style="list-style-type: none"> Identify technology e.g. traffic lights, pass entry systems, Identify a computer and its main parts e.g. monitor/screen, keyboard, central processing unit, mouse/navigator, speakers Computers can do nothing, unless instructed by a user Data is collected and can be stored on a computer A network is two or more computers that are linked 		Using computers <ul style="list-style-type: none"> Turning devices on/off safely Using the equipment in a safe and respectful manner Troubleshoot the computer/device for errors e.g. checklist for start up: monitor on, computer on 	
Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.	Key Knowledge	Skills/Learning Checkpoints	Vocabulary	Ideas for Great Teaching
	<ul style="list-style-type: none"> Explain what different freehand tools do Recognise computers can be used to create art Recognise tools can be adjusted to suit need Compare computer painting with using brushes 	<ul style="list-style-type: none"> Use freehand tools for painting Use shape and line tools Choose appropriate shapes and colours Choose appropriate paint tools Use correct colour and brush tools Spot differences between painting on a computer and paper 	paint program, tool, paintbrush, erase, fill, undo, shape tools, line tool, fill tool, undo tool, colour, brush style, brush size, pictures, painting, computers	NCCE Creating media – digital painting https://teachcomputing.org/curriculum/key-stage-1/creating-media-digital-painting Hardware/software requirements: Microsoft Paint or paintz.app
<ul style="list-style-type: none"> Recognise that a keyboard is used to enter text Recognise that shift key changes output Recognise that text can be changed Recognise that appearance of text can change 	<ul style="list-style-type: none"> Use a computer to write Add and remove text Type capital letters; use bold, italic and underline Dragging and clicking and changing font Undo changes to text Explain differences between typing and writing 	word processor, keyboard, keys, letters, type, numbers, space, backspace, text cursor, capital letters, toolbar, bold, italic, underline, mouse, select, font, undo, redo, format, compare, typing, writing.	NCCE Creating media – digital writing https://teachcomputing.org/curriculum/key-stage-1/creating-media-digital-writing Hardware/software requirements: Word processing software (eg: Microsoft Word)	
Computational Thinking: <ul style="list-style-type: none"> Understanding and using algorithms De-bugging Abstraction Decomposition Pattern recognition 	Key Knowledge	Skills/Learning Checkpoints	Vocabulary	Ideas for Great Teaching
	Introduction to algorithms Algorithms: <ul style="list-style-type: none"> An algorithm is a series of set of instructions Know the sequence of an algorithm is important De-bugging Debugging is identifying and removing errors from an algorithm. Abstraction: <ul style="list-style-type: none"> Abstraction is taking out instructions you do not need in an algorithm Decomposition: <ul style="list-style-type: none"> Decomposition is solving a problem in smaller parts to make it easier Pattern & Recognition: <ul style="list-style-type: none"> Know which parts of instructions/algorithms are similar and different 	I can predict, match and run an algorithm. I can debug my program. I can streamline my algorithm. I can program my algorithm into manageable parts. I can identify similar and different algorithms.	<ul style="list-style-type: none"> Command - A single instruction that can be used in a program to control a computer Computer - A programmable machine that accepts and processes inputs and produces outputs (input, process, output; IPO) Algorithm - A precise set of ordered steps that can be followed by a human and implemented on a computer to achieve a task Program - A set of ordered commands that can be run by a computer to complete a task Programming - the process of writing commands (coding) Run - To action the commands in a program Sprite - a 2-D image in a programme Reset - to start again/to clear any errors 	NCCE Programming A – A robot https://teachcomputing.org/curriculum/key-stage-1/programming-a-moving-a-robot NCCE Programming B -Programming animations https://teachcomputing.org/curriculum/key-stage-1/programming-b-introduction-to-animation Hardware/software requirements: Bee-bot robots Scratch Jr
Technology should be used safely, respectfully, and responsibly.	Electronic safety (e-safety) incorporates: <ul style="list-style-type: none"> Safe, respectful, responsible use of technology Identifying dangers Know where or who to contact if they have concerns 			

Key Concepts, Knowledge, Vocabulary and Skills – Computer scientists: Year 2

Key Concept	Key Knowledge and Vocabulary		Skills: Techniques & Application	
Computer components and resultant hardware	How computers work and the internet <ul style="list-style-type: none"> Know that computers have no intelligence and that computers can do nothing unless a program is run Know the purpose of a network and why they are computers are networked. Recognise different storage tools, e.g. <i>memory sticks, external hard drives, cable connections i.e. USB</i> The internet is a global network of computers, including online devices and servers The World Wide Web is the collection of public webpages stored on the internet Some websites are blocked or have limited access because of network administration permissions e.g. parental and school controls Explain the roles within a computer system for hardware and application software: <ul style="list-style-type: none"> Computer memory and processing speed Size of files and storage available There are operating systems and these may look differently Understand where files are stored and saved Be aware that websites and videos are files that are shared from one digital device to another. Know that computers collect data from various input devices, including sensors and application software., e.g. digital camera, mouse roller, gate/motion sensors, temperature and weather sensors 		How computers work and the internet <ul style="list-style-type: none"> Troubleshoot the computer/device to identify possible errors such as connections to networks, e.g. symbols for wi-fi, online/offline, battery power Seek technical help for an error message e.g. “cannot connect to the server” Correctly insert and remove different storage tools to ensure data in not corrupted <p>Maintain a safe and organised working environment e.g. keep liquids and consumables away; cables to be managed for safety</p>	
Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.	Key Knowledge <ul style="list-style-type: none"> Recognise that some digital devices can capture images Recognise that images can be saved and viewed Know about what a ‘good’ image is Know how an image can be improved Know the effect of light on a photograph Know that images can be changed after taking Recognise that some images are not accurate 	Skills/Learning Checkpoints <ul style="list-style-type: none"> Use a device to take a photo Take photos in landscape and portrait Improve taking of a photograph Experiment with different light sources Use tools to change an image Identify real and distorted/changed images 	Vocabulary <p>music, quiet, loud, feelings, emotions, pattern, rhythm, pulse, pitch, tempo, rhythm, notes, create, emotion, beat, instrument, open, edit.</p>	Ideas for Great Teaching <p>NCCE Creating media – digital photography https://teachcomputing.org/curriculum/key-stage-1/creating-media-digital-photography</p> <p>Hardware/software requirements: Suitable photo app (eg: Windows photo or inbuilt iPad camera and tools)</p>
Computational Thinking: <ul style="list-style-type: none"> Understanding and using algorithms De-bugging Abstraction Decomposition Pattern recognition 	Key Knowledge <p>Knowing and exploring algorithms Algorithms: <ul style="list-style-type: none"> An algorithm is a series of set of instructions Know the sequence of an algorithm is important Be aware algorithms can be represented in simple formats [flow chart, storyboards, and narrative text] De-bugging <ul style="list-style-type: none"> Debugging is identifying and removing errors from an algorithm. Abstraction: <ul style="list-style-type: none"> Abstraction is taking out instructions you do not need in an algorithm Decomposition: <ul style="list-style-type: none"> Decomposition is solving a problem in smaller parts to make it easier Pattern & Recognition: <ul style="list-style-type: none"> Know which parts of instructions/algorithms are similar and different </p>	Skills/Learning Checkpoints <p>I can program a computer algorithm to perform a specific task</p> <p>I can check, debug and update my program</p> <p>I can improve and add to a program (existing and ones I have created)</p> <p>I can create sequenced / ordered parts of a program</p> <p>I can make changes to programs (existing and ones I have created)</p>	Vocabulary <ul style="list-style-type: none"> Sequence – a series of instructions/commands Algorithm - A precise set of ordered steps that can be followed by a human and implemented on a computer to achieve a task Program - A set of ordered commands that can be run by a computer to complete a task Debugging - The process of finding and correcting errors in a program Decomposition - To break down a task into smaller, more achievable steps Command - A single instruction that can be used in a program to control a computer Run - To action the commands in a program Sprite - a 2-D image in a programme Code - The commands that a computer can run 	Ideas for Great Teaching <p>NCCE Programming A – Robot algorithms https://teachcomputing.org/curriculum/key-stage-1/programming-a-robot-algorithms</p> <p>NCCE Programming B -Programming quizzes https://teachcomputing.org/curriculum/key-stage-1/programming-b-an-introduction-to-quizzes</p> <p>Hardware/software requirements: Bee-bot robots Scratch Jr</p>
Technology should be used safely, respectfully, and responsibly.	Electronic safety (e-safety) incorporates: <ul style="list-style-type: none"> Safe, respectful, responsible use of technology Identifying dangers Know where or who to contact if they have concerns 			

Key Concepts, Knowledge, Vocabulary and Skills – Computer scientists: Year 3

Key Concept	Key Knowledge and Vocabulary		Skills: Techniques & Application	
Computer components and resultant hardware	Global computers <ul style="list-style-type: none"> Recognise the components within digital devices [processor, memory storage, motherboard, display, speakers, microphone – other input devices webcam, finger-print scanner] Recognise what computers do when they receive instructions that can not be executed [error messages] Recognise one or more computing language [binary, html (hyper-text markup language)] Understand how computers can be connected to form a network either wirelessly or via LAN The internet is a global network of computers, including online devices and servers The World Wide Web is the collection of public webpages stored on the internet [understand URLs and domains] URL means Universal Resource Locator and is an address to locate data A domain is the name of a website e.g. google or BBC bitesize Understand that copyright exists and that they are whole libraries of content available for public use 		Global computers <ul style="list-style-type: none"> Troubleshoot the computer/device to identify possible visual errors such as connections to devices, e.g. microphone, speakers, camera, printers Locate general device information e.g. wifi network, storage capacity and usage, which programmes are installed etc Recognise when a problem can be solved individually without 'technical support' Use the language URL and domain to describe the location of information and identify if there are errors with the domain or url Relate domains and urls to wider world context e.g. home address, room locations in a building 	
Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.	Key Knowledge <ul style="list-style-type: none"> To explain that an animation is made up of a sequence of images. Identify that a capturing device needs to be in a fixed position Recognise that smaller movements create smoother animations Addition of other media to animations Recognise how a project is exported then shared 	Skills/Learning Checkpoints <ul style="list-style-type: none"> Draw and create an animation flip book Predict and create movement Create a storyboard Use onion skinning to review and evaluate work Improve an animation Add other media to an animation 	Vocabulary animation, flip book, stop frame, frame, sequence, image, photograph, setting, character, events, onion skinning, consistency, evaluation, delete, media, import, transition. text, images, advantages, disadvantages, communicate, font, style, landscape, portrait, orientation, placeholder, template, layout, content, desktop publishing, copy, paste, purpose, benefits	Ideas for Great Teaching NCCE Creating media – stop-frame animation https://teachcomputing.org/curriculum/key-stage-2/creating-media-animation Hardware/software requirements: iMotion or Stop Motion Studio
	<ul style="list-style-type: none"> Recognise how text and images are used together to convey information Define landscape and portrait orientation Recognise different layout options for different purposes Recognise that DTP pages can be structured with placeholders Recognise font styles are used for effect Consider benefits of DTP applications 	<ul style="list-style-type: none"> Explain difference between text and images to communicate information Edit text and layout Choose appropriate page settings Add content to publications Choose a suitable layout Compare work on a desktop to that completed by hand 		NCCE Creating media – desktop publishing https://teachcomputing.org/curriculum/key-stage-2/creating-media-desktop-publishing Hardware/software requirements: Canva
Computational Thinking: <ul style="list-style-type: none"> Understanding and using algorithms De-bugging Abstraction Decomposition Pattern recognition 	Key Knowledge Programming Algorithms: <ul style="list-style-type: none"> Different algorithms work for different purposes; there are patterns Programs are executed by following precise and unambiguous instructions It is possible to predict, and test completed algorithms De-bugging <ul style="list-style-type: none"> Debugging is necessary for all algorithms for a variety of situations A systematic approach to debugging means describing, explaining and correcting errors Abstraction: <ul style="list-style-type: none"> A sequence of events will have more important information Information is not always needed for the sequence of events and so some instructions can be removed. Decomposition: <ul style="list-style-type: none"> Decomposition can be explored without computers (unplugged activity): Visual representation, including mathematics can be used to solve problems. Pattern & Recognition: Patterns can be recognised and included within a program	I can recognise (and use) algorithms being represented in different ways (e.g.: as blocks) I can recognise, use and build a sequence of commands. I can decide actions for each individual sprite in a program. I can add various additional features to improve a program.	Vocabulary <ul style="list-style-type: none"> Algorithm - A precise set of ordered steps that can be followed by a human and implemented on a computer to achieve a task Bug – an unexpected problem in the code Code - The commands that a computer can run Commands - A single instruction that can be used in a program to control a computer Debugging - The process of finding and correcting errors in a program Programming - the process of writing commands (coding) Run - To action the commands in a program Sequence – a series of instructions/commands Sprite - a 2-D image in a programme 	Ideas for Great Teaching NCCE Programming A – Sequencing sounds https://teachcomputing.org/curriculum/key-stage-2/programming-a-sequence-in-music NCCE Programming B – Events and actions in programs https://teachcomputing.org/curriculum/key-stage-2/programming-b-events-and-actions Hardware/software requirements: Scratch
	Technology should be used safely, respectfully, and responsibly. Electronic safety (e-safety) incorporates: <ul style="list-style-type: none"> Safe, respectful, responsible use of technology Identifying dangers Know where or who to contact if they have concerns 			

Key Concepts, Knowledge, Vocabulary and Skills – Computer scientists: Year 4

Key Concept	Key Knowledge and Vocabulary		Skills: Techniques & Application	
Computer components and resultant hardware	Networks and data <ul style="list-style-type: none"> There are a range of connected networks e.g. home network, school, organisation, national and intranets Computer networks provide multiple services, such as the world wide web and opportunities for communication and collaboration There are types of different network hardware where computers are linked e.g. <i>desktop computers, mobile devices, servers, routers</i> A server is a computer or system that stores all the data of a website to allow access from another device (clients) and its user e.g. cloud storage An IP address (internet protocol address) is a unique address that identifies a device on the internet or a local network, using numbers Digital computers use binary to represent all data Packets of data are units of <i>data</i> made into a single package that travels along a given network path		Networks and data <ul style="list-style-type: none"> Identify the range of connected networks e.g. <i>home network, school, organisation, national and intranets</i> Describe and design/map(draw) real and fictional networked devices and how they connect through websites, web pages, apps and IPs Locate information about storage capacity on devices e.g. <i>file explorer</i> Store and share data and understand the role of packets Identify suitable storage locations for own information: e.g. <i>saving files onto personal or school networks</i> 	
Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.	Key Knowledge	Skills/Learning Checkpoints	Vocabulary	Ideas for Great Teaching
	<ul style="list-style-type: none"> To identify that sound can be recorded To identify that an input device is needed to record sound To identify that an output device is needed to play audio Recognise that recorded audio can be stored on a computer Recognise that audio can be edited Recognise that sound can be represented visually Recognise that audio can be layered 	<ul style="list-style-type: none"> Use a computer to record audio Re-record and improve audio Save a project and play appropriate content Review and improve audio independently Combine audio to enhance recordings Evaluate effects of audio 	audio, microphone, speaker, headphones, input device, output device, sound, podcast, edit, trim, align, layer, import, record, playback, selection, load, save, export, MP3, evaluate, feedback	NCCE Creating media – audio production https://teachcomputing.org/curriculum/key-stage-2/creating-media-audio-editing Hardware/software requirements: Audacity or Garage Band (iPads), Twisted Wave or Audio Mass (Chromebooks)
	<ul style="list-style-type: none"> Know how images can be edited, including: rotating, flipping, cropping, adjusted and filtered Know how images can be cloned, copied, retouched and pasted Know how different compositions (eg: text) can be added 	<ul style="list-style-type: none"> Rotate and crop images Change colours in an image Use cloning in an image Copy between images Choose and combine suitable images Review images; combine image and text 	image, edit, digital, crop, rotate, undo, save, adjustments, effects, colours, hue, saturation, sepia, vignette, image, retouch, clone, select, combine, made up, real, composite, cut, copy, paste, alter, background, foreground, zoom, undo, font.	NCCE Creating media – photo editing https://teachcomputing.org/curriculum/key-stage-2/creating-media-photo-editing Hardware/software requirements: iPaint/Pixlr/PhotoPea
Computational Thinking: <ul style="list-style-type: none"> Understanding and using algorithms De-bugging Abstraction Decomposition Pattern recognition 	Key Knowledge	Skills/Learning Checkpoints	Vocabulary	Ideas for Great Teaching
	Being a Programme Engineer A programme engineer will design using programming language for an intended purpose. Algorithms: <ul style="list-style-type: none"> Understand that a program is a sequence of statements written in programming language Algorithms for use when programming can become increasingly more precise Remix existing code De-bugging <ul style="list-style-type: none"> Being systematic and applying logical reasoning can assist in the detection and correction of errors in programs Abstraction: <ul style="list-style-type: none"> Abstraction (removing of instructions) can help to focus on what is important in a design Abstraction (removing of instructions) and pattern recognition can help to modify code Decomposition: <ul style="list-style-type: none"> Decomposition can be explored without computers (unplugged activity): Visual representation, including mathematics can be used to solve problems Pattern & Recognition: <ul style="list-style-type: none"> Procedures, pattern matching, and simple selection can be used in algorithms Know that a computer can repeatedly call a procedure A procedure is set of coded instructions within the program Patterns are similarities or characteristics that some of the problems share Pattern recognition involves finding the similarities or patterns among small, decomposed problems that can help us solve more complex problems more efficiently Simple selection is a decision or question. In an algorithm, there may need to be a question because the algorithm has reached a step where one or more options are available. Depending on the answer given, the algorithm will follow certain steps and ignore others	I can use block and text-based programming (eg: Scratch, Logo). I can identify and use loops (patterns) within programming. I can choose which values to change within a loop. I can predict outcomes in a program and develop it by debugging. I can reuse code so more than one loop/sequence runs at the same time.	<ul style="list-style-type: none"> Algorithm - a precise set of ordered steps that can be followed by a human and implemented Animate – to create movement using a series of still images Code – the commands that a computer can run Code snippet – a section of code on a computer to achieve a task Commands - a single instruction that can be used in a program to control a computer Count-controlled loop – a set of instructions that repeats a set amount of times Infinite loop – a set on instructions that repeats endlessly Debugging - the process of finding and correcting errors in a program Duplicate – to copy Programming - the process of writing commands (coding) Sprite – a 2-D image in a programme Variable – a value that can change depending on instructions. 	NCCE Programming A – Repetition in shapes https://teachcomputing.org/curriculum/key-stage-2/programming-a-repetition-in-shapes NCCE Programming B – Repetition in games https://teachcomputing.org/curriculum/key-stage-2/programming-b-repetition-in-games Hardware/software requirements: FMS Logo (free download) Or Turtle Academy (can be used on tablets/iPads) Scratch
Technology should be used safely, respectfully, and responsibly.	Electronic safety (e-safety) incorporates: <ul style="list-style-type: none"> Safe, respectful, responsible use of technology Identifying dangers Know where or who to contact if they have concerns 			

Key Concepts, Knowledge, Vocabulary and Skills – Computer scientists: Year 5

Key Concept	Key Knowledge and Vocabulary		Skills: Techniques & Application	
Computer components and resultant hardware	Computing & Security <ul style="list-style-type: none"> There are a range of computing technology that is used as part everyday life e.g. <i>contactless payments, mobile apps, entry systems, security surveillance systems</i> The security of data relates to passwords, access privileges/rights and data encryption Data encryption means the process of converting data into an unusable form (but knowing this does not itself stop hacking or data theft) Malware (short for malicious software) means software which is designed to harm or exploit any programmable device, service or network e.g. <i>viruses, ransomware and spyware</i> A computer virus is a type of malicious code or program written to alter the way a computer operates and that is designed to spread from one computer to another Antivirus is a kind of software used to prevent, scan, detect and delete viruses from a computer 		Computing & Security <ul style="list-style-type: none"> Check network connection and availability for owned devices, including 'levels of access' and the source of connection Identify network speed and how to improve connection Explain the impact a server may have on a connection speed and suggest improvements Transfer data safely and securely across the internet with an understanding of data protection rules, including encryption of messages Describe and map(draw) real and fictional networked devices and the intended security and access privileges/rights, including home and organisational access, as well as software program installation rights Identify unsafe networks or websites, including signs of malware or viruses	
Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.	Key Knowledge <ul style="list-style-type: none"> Know features of video as a visual media format Recognise which devices can and can't record video Explain purpose of a storyboard Identify how a video can be edited, explaining its limitations Recognise the need to regularly review and edit a project Recognise projects need to be exported to be shared 	Skills/Learning Checkpoints <ul style="list-style-type: none"> Identify and compare features of videos Use a digital device to record video Use a range of filming techniques Outline scenes of a video in a storyboard Improve, store, retrieve and export video Edit and evaluate a created video 	Vocabulary <p>drawing tools, object, toolbar, vector drawing, move, resize, colour, rotate, duplicate/copy, zoom, select, align, modify, layers, order, copy, paste, group, ungroup, reuse, reflection</p>	Ideas for Great Teaching <p>NCCE Creating media – video production https://teachcomputing.org/curriculum/key-stage-2/creating-media-video-editing</p> <p>Hardware/software requirements: iPads/tablets for recording; iMovie or Canva video editor for editing of videos</p>
	<ul style="list-style-type: none"> Know that a vector drawing comprises different objects Recognise the different layers of each object Recognise that drawings can be scaled without an impact on quality Recognise that objects can be modified in groups Know that alignment and size guides can help consistency 	<ul style="list-style-type: none"> Experiment with shape and line tools Identify, explain and move/resize and rotate objects Use different tools, including zoom and resize grids Change order of layers in a drawing Duplicate, group, ungroup and reuse objects Reflect and compare vector to freehand paint drawings. 	video, audio, camera, talking head, panning, close up, video camera, microphone, lens, mid-range, long shot, moving subject, side by side, angle (high, low, normal), static, zoom, pan, tilt, storyboard, filming, review, import, split, trim, clip, edit, reshoot, delete, reorder, export, evaluate, share.	<p>NCCE Creating media – vector graphics https://teachcomputing.org/curriculum/key-stage-2/creating-media-introduction-to-vector-graphics</p> <p>Hardware/software requirements: Google Drawings (Powerpoint can be used for adapted vector-style drawings)</p>
Computational Thinking: <ul style="list-style-type: none"> Understanding and using algorithms De-bugging Abstraction Decomposition Pattern recognition 	Key Knowledge <p>Complex Programming with conditions and patterns</p> <p>Algorithms:</p> <ul style="list-style-type: none"> Write complex algorithms for a purpose, evaluate work, and identify errors Include a sequence, selection, and repetition commands, combined with variables as required, to implement a design The flow of a program refers to the sequence of the instructions in a program Understand that websites can be altered by exploring the code beneath the site <p>De-bugging</p> <ul style="list-style-type: none"> Being systematic and applying logical reasoning can assist in the detection and correction of errors in programs <p>Abstraction:</p> <ul style="list-style-type: none"> Understanding abstraction (removing of instructions) can help to identify errors in code i.e. what has been removed accidentally or deliberately <p>Decomposition:</p> <ul style="list-style-type: none"> Understand decomposition is and how it facilitates problem solving of increasing complex problems <p>Pattern & Recognition:</p> <ul style="list-style-type: none"> Include patterns and selection to create specific outputs in algorithms. Compare various programmes to identify commonalities / similarities. A count-controlled loop repeats a series of one or more instructions until a determined number of repetitions of the loop/sequence has been completed An infinite loop is a sequence of instructions in a computer program which loops endlessly, either due to the loop having no terminating condition A conditional loop is a sequence of instructions in a computer program which repeats set of commands depending on the conditions set 	Skills/Learning Checkpoints <p>I can use previous learning to explore physical computing (eg: micro:bit).</p> <p>I can connect and program components.</p> <p>I can create, test and debug a project.</p> <p>I can create, identify and modify conditions in a program.</p> <p>I can use condition and selection to test and share my program with others.</p>	Vocabulary <ul style="list-style-type: none"> Algorithm - a precise set of ordered steps that can be followed by a human and implemented Microcontroller – a self-contained microcomputer. Used to perform one specific job. Components – Hardware Condition – a state of something in the programme that's true or false Conditional statement - used to make decisions in code based on whether a certain condition is true or false Count-controlled loop - a set of instructions that repeats a set amount of times Infinite loop - a set on instructions that repeats endlessly Input – information provided to a computer (mouse clicks, keys on a keyboard) Output – the result of an input Debugging - the process of finding and correcting errors in a program Programming - the process of writing commands (coding) Operator - An operator is a character, or characters, that determine what action is to be performed or considered (= > < etc.) Selection - Part of a program where if a condition is met, then a set of commands is run Variable - a value that can change depending on instructions. 	Ideas for Great Teaching <p>NCCE Programming A – Selection in physical computing https://teachcomputing.org/curriculum/key-stage-2/programming-a-selection-in-physical-computing</p> <p>NCCE Programming B -Selection in quizzes https://teachcomputing.org/curriculum/key-stage-2/programming-b-selection-in-quizzes</p> <p>Hardware/software requirements: Micro:bit/Crumble devices + Microsoft makecode</p> <p>Scratch</p>
Technology should be used safely, respectfully, and responsibly.	Electronic safety (e-safety) incorporates: <ul style="list-style-type: none"> Safe, respectful, responsible use of technology Identifying dangers Know where or who to contact if they have concerns 			

Key Concepts, Knowledge, Vocabulary and Skills – Computer scientists: Year 6

Key Concept	Key Knowledge and Vocabulary		Skills: Techniques & Application	
Computer components and resultant hardware	Compatibility and design <ul style="list-style-type: none"> There are links between internal parts within a computer, to ensure compatibility. Compatibility is the ability to live and work together without any discrepancy Not all hardware/components are compatible with each other (Window, Apple, Linux, Chrome) app compatibility Computer software develops solve problems of compatibility, i.e. <i>compatible with Windows</i> Updates refer to software improvements that are made and support the functionality of the software and hardware effectiveness Servers <ul style="list-style-type: none"> A private server is any computer used as a server that is privately administrated. A public server is data can be accessed by others who are also connected to it Data transmission between digital computers over networks, including the internet i.e. IP addresses and packet switching An IP address is a unique address that identifies a device on the internet or a local network. IP stands for "Internet Protocol," which is the set of rules governing the format of data sent via the internet or local network <p>Packet switching is the transfer of small pieces of data across various networks. These data chunks or "packets" allow for faster, more efficient data transfer, i.e. <i>When a user sends a file across a network, it gets transferred in smaller data packets, not in one pieces</i></p>		Compatibility and design <ul style="list-style-type: none"> Link different parts of computer hardware to achieve an outcome e.g.: <i>raspberrypi, maKey-maKey</i> Recognise where hardware components are compatible and not compatible Critically analyse different networks, viewing their purpose and suitability for specific outcomes Describe and design (draw) real and fictional networks, viewing their purpose and suitability for specific outcomes <p>Recognise the need, rights/permissions and need to run software updates, which in turn support hardware</p>	
Digital Literacy relates to having functional IT skills, which support the creation or presentation of content.	Key Knowledge <ul style="list-style-type: none"> Recognise relationship between HTML and visual display Recognise that web pages can contain different media types Recognise that web pages are written by people Recognise that hyperlinks make up a web page Recognise components of a web layout Know ownership (copyright) rights Recognise the need to preview and navigation paths of pages Recognise implications of linking content owned by others 	Skills/Learning Checkpoints <ul style="list-style-type: none"> Explore and discuss different websites Plan features of a webpage Find and use copyright-free images Add, preview and evaluate a webpage Create a webpage; describing and explaining navigation paths Create hyperlinks, recognising owned content 	Vocabulary <p>website, web page, browser, media, Hypertext Markup Language (HTML), logo, layout, header, media, purpose, copyright, fair use, home page, preview, evaluate, device, Google Sites, breadcrumb trail, navigation, hyperlink, subpage, evaluate, implication, external link, embed.</p>	Ideas for Great Teaching <p>NCCE Creating media – web page creation https://teachcomputing.org/curriculum/key-stage-2/creating-media-web-page-creation</p> <p>Hardware/software requirements: Google sites, Microsoft Sway or Powerpoint (non-web-based version)</p>
	<ul style="list-style-type: none"> Know that 3-D models can be created on a computer Recognise that they can be viewed from different perspectives Know that tools can be used to manipulate models Know how placeholders can be used to create holes in 3-D models 	<ul style="list-style-type: none"> Add, view and move shapes Resize, lift/lower and recolour objects Rotate, group and duplicate objects Size, combine and place hold objects Plan and create own 3-D model 	<p>TinkerCAD, 2D, 3D, shapes, select, move, perspective, view, handles, resize, lift, lower, recolour, rotate, duplicate, group, cylinder, cube, cuboid, sphere, cone, prism, pyramid, placeholder, hollow, choose, combine, construct, evaluate, modify.</p>	<p>NCCE Creating media – 3D modelling https://teachcomputing.org/curriculum/key-stage-2/creating-media-3d-modelling</p> <p>Hardware/software requirements: TinkerCAD platform</p>
Computational Thinking: <ul style="list-style-type: none"> Understanding and using algorithms De-bugging Abstraction Decomposition Pattern recognition 	Key Knowledge <p>Mastering Programming</p> <p>Algorithms:</p> <ul style="list-style-type: none"> Increasingly complex algorithms are written for purpose Generic code can be written for use across multiples projects A variable are storage locations with assigned names (categories) that hold data that you can change or use later Controllable device are ones which can be programmed and controlled <p>De-bugging</p> <ul style="list-style-type: none"> Editing code contributes to unsuccessful and successful outputs and applications A software bug is an error in a programme that causes it to produce an incorrect or unexpected result. The elimination of software errors is called bug fixing A bug fix is a change to a system or product designed to handle a programming bug/glitch <p>Abstraction:</p> <ul style="list-style-type: none"> Logical reasoning can help to explain how processes can be more efficient, reducing the number of potential steps needed in a program Knowing abstraction is part of the planning process before creating algorithms <p>Decomposition:</p> <ul style="list-style-type: none"> Use decomposition across a range of situations, to break down complex programs, making improvements where required <p>Pattern & Recognition:</p> <ul style="list-style-type: none"> Recognise, and make use, of patterns across programming projects, identifying variables needed, and explain their uses in selection and repetition A nested loop is a loop within a loop, an inner loop within the body of an outer one Iteration is the repetition of a process such as a loop 	<p>I can use, modify and create variables within a project.</p> <p>I can use different values in variables, predicting then testing these.</p> <p>I can write increasingly complex algorithms, using variables to extend projects.</p> <p>I can use abstraction/modifying to increase efficiency of a program.</p> <p>I will use a range of approaches to debug programs.</p>	Vocabulary <ul style="list-style-type: none"> Algorithm - a precise set of ordered steps that can be followed by a human and implemented Accelerometer – a device that detects acceleration Code - the commands that a computer can run Debugging - the process of writing commands (coding) Event – an action carried by software Programming - the process of writing commands (coding) Input - information provided to a computer (mouse clicks, keys on a keyboard) Process - A program, or part of a program, that is running on a computer Output - the result of an input Operator - An operator is a character, or characters, that determine what action is to be performed or considered (= > < etc.) Set – an unordered collection of objects Variable - a value that can change depending on instructions. 	Ideas for Great Teaching <p>NCCE Programming A – Variables in games https://teachcomputing.org/curriculum/key-stage-2/programming-a-variables-in-games</p> <p>NCCE Programming B - https://teachcomputing.org/curriculum/key-stage-2/programming-b-sensing</p> <p>Hardware/software requirements: Scratch Micro:bit + Microsoft makecode</p>
Technology should be used safely, respectfully, and responsibly.	<p>Electronic safety (e-safety) incorporates:</p> <ul style="list-style-type: none"> Safe, respectful, responsible use of technology Identifying dangers Know where or who to contact if they have concerns 			