Computer Science KS3 Overview (2023-24)

The computing Program of study for KS3 computing aims to develop students' interest in computing, and computer Science. We aim to embed a deep interest in the subject and equip students with the necessary skills and knowledge so they can be successful and allow them to fully participate in the digital world we live in. We also look to build students ability in computational thinking, so that they use it within their own programming, and more broadly within all their digital interactions.

Year 7						
	Autumn 1	**Autumn 2**	**Spring 1**	**Spring 2**	**Summer 1**	**Summer 2**
	Introduction to Computer Science	**Media – Vector Graphics**	**Computational Thinking**	**Programming intro (scratch)**	**Spreadsheets**	**Scratch (Part 2)**
Est Weeks	8	6	6	6	6	6
Knowledge	IT systems (files, folders, organisation), correct use of email, netiquette, web awareness, using MS office. Introduce school wellbeing and personal development platform – YouHQ.	A topic which focuses on Vector graphics versus raster graphics, using Inkscape / vectr.com. The intent is to ultimately create digital artefacts which incorporate layering, grouping, combining, subtracting objects from each other, and editing the paths of the objects to create images that meet a brief.	A first look at computational thinking. Introduction to the key parts of flowcharts, algorithms, abstraction, decomposition, and logical questioning.	Programming with Scratch with further promotion of computational thinking and problem solving. Looking at fundamental programming concepts including, sequencing, variables, selection, operators, and iteration,	An introduction to spreadsheets and how they can organise and display data. Use of formulas, conditional formatting, functions, and graphs to store, analyse and visualise data.	Investigating code (debugging) and development of problem-solving skills. An introduction in Scratch to different types of iteration, lists, data types, and methods for sorting data
Skills / software	Microsoft software File organisation Email use MS teams Online Safety YouHQ	Creating vector graphics using Inkscape Vectr.com Presentation skills	Using and creating flowcharts Flowol4 Develop algorithms	Using scratch to create simple programs, and development of computational thinking. Kodu Lab	MS Excel Office 365 MS teams Spreadsheet structures Basic formula using: *, -, +, / Functions - SUM, MIN, MAX, AVERAGE Graphs Conditional formatting	Scratch Sorting data Computational thinking
NC links [National Curriculum](#)	Understand a range of ways to use technology safely, respectfully, responsibly, and	Create, reuse, revise and repurpose digital artefacts for a given audience, with	Design, use, and evaluate computational abstractions that model the state and	Use two or more programming languages, one of which is textual, to	Undertake creative projects that involve selecting, using, and combining multiple	Use two or more programming languages, one of which is textual, to solve a variety of computational

Computing key stages 3 and 4	securely, including protecting their online identity and privacy; recognise inappropriate content, contact, and conduct, and know how to report concerns.	attention to trustworthiness, design, and usability. Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting, analysing data, and meeting the needs of known users.	behaviour of real-world problems and physical systems. Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming.	solve a variety of computational problems; design and develop modular programs that use procedures or functions.	applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users. Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design, and usability	problems; make appropriate use of data structures such as lists, tables, or arrays. Design and develop modular programs that use procedures or functions. 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
External links	The impact of ICT on society: https://www.bbc.co.uk/bitesize/guides/zbxbkqt/revision/1 E Safety: https://www.bbc.co.uk/bitesize/guides/zrtrd2p/revision/1 How ICT has changed communication and collaboration: https://www.bbc.co.uk/bitesize/guides/zg6g87h/revision/1 Search engines: https://www.bbc.co.uk/bitesize/guides/zpkhpv4/revision/1 Tech through time: https://www.bbc.co.uk/bitesize/guides/z4p4jxs/revision/1	Graphics software: https://www.bbc.co.uk/bitesize/guides/zv2v4wx/revision/1 Online vector graphics www.vectr.com Bias and reliability: https://www.bbc.co.uk/bitesize/guides/z2g2mp3/revision/1	Introduction to Computational thinking: https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1	Control and sequencing: https://www.bbc.co.uk/bitesize/guides/zfjsgk7/revision/1 Introduction to programming: https://www.bbc.co.uk/bitesize/guides/zts8d2p/revision/1 Scratch: https://scratch.mit.edu/ Programming basics: https://www.bbc.co.uk/bitesize/guides/zwmbgk7/revision/1	Spreadsheets: https://www.bbc.co.uk/bitesize/guides/zdydmp3/revision/1 Presenting information on computers: https://www.bbc.co.uk/bitesize/guides/zksk7ty/revision/1 Modelling simulation: https://www.bbc.co.uk/bitesize/guides/zyqfr82/revision/1	Scratch: https://scratch.mit.edu/ Arrays and Lists: https://www.bbc.co.uk/bitesize/guides/zy9thyc/revision/1 Selection in programming: https://www.bbc.co.uk/bitesize/guides/z2p9kqt/revision/1 Iteration: https://www.bbc.co.uk/bitesize/guides/z3khpv4/revision/1 Procedures and Functions: https://www.bbc.co.uk/bitesize/guides/zqh49j6/revision/1

Year 8						
Term	**Autumn 1**	**Autumn 2**	**Spring 1**	**Spring 2**	**Summer 1**	**Summer 2**
	Computer Systems	**Computer Networks**	**Web development**	**Binary**	**Python Programming**	**Physical and embedded computing**
Est Weeks	6	6	6	6	8	6
Knowledge	Exploration of the different parts of computing systems: from the OS to hardware. This will also include a look at Logic, AI, sharing code etc	A detailed look into computer networks including, network components, layouts and security, protocols for transferring data, LAN, WAN, the internet, and transferring data across networks. Also, this topic looks at the internet as a source of information and how to scrutinise webpages / resources retrieved.	An exploration of the building blocks of the World Wide Web, HTML, and CSS. A revisit to internet safety, searching the web, and a look at how search engines effectively retrieve information and catalogue webpages.	The use of AND/OR/NOT gates, truth tables, and building simple logic circuits.	First look at the textual programming language, Python, use an IDE to develop code, revisiting the idea of an algorithm, look at user Inputs into programs, different data Types & casting, identify different data structures (Lists/Arrays) and how repetition can affect a program.	A look at embedded computing including the programming of a Micro: bit. Identify the key parts of a Micro: bit program and apply programming knowledge to extend existing programs. Further develop event driven programming skills and strengthen skills of decomposition, algorithm design.
Skills / software	Identify computer components, Little Man Computer simulator. PPT and presentation skills	Network security. Data management. Computer programming. Computer networking and communication tools.	HTML debugging Adding Multimedia to a Website Adding CSS to a Website Adding Tables and Hyperlinks to a Website Multiple Web Pages	Logic circuit use in computer science. Truth Tables Computational Thinking	Problem Solving Sequencing Selection Iteration Reflection Programming Using an IDE	Micro: bit / Arduinos (Scratch)
NC links National Curriculum Computing key stages 3 and 4	Understand the hardware and software components that make up computer systems, and how they communicate with one another and with	Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other networks. 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	Create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design, and usability. Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting, analysing data, and meeting the needs of known users	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]	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. Use two or more programming languages, one of which is textual, to solve a variety of computational problems; make appropriate	To make appropriate use of data structures and develop modular programs that use procedures or functions. Undertake creative projects that involve challenging goals. Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems

		know how to report concerns.			use of data structures such as lists, tables, or arrays.	
External links	The CPU and the fetch-execute cycle: https://www.bbc.co.uk/bitesize/guides/zws8d2p/revision/1 Software: https://www.bbc.co.uk/bitesize/guides/zcxgr82/revision/1 Digital devices: https://www.bbc.co.uk/bitesize/guides/zxb72hv/revision/1	Networks: https://www.bbc.co.uk/bitesize/guides/zc6rcdm/revision/1	HTML: https://www.bbc.co.uk/bitesize/guides/z993tv4/revision/1 Internet and communication: https://www.bbc.co.uk/bitesize/guides/z8nk87h/revision/1	Logical Reasoning: https://www.bbc.co.uk/bitesize/guides/z8jfyrd/revision/1 Binary: https://www.bbc.co.uk/bitesize/guides/z26rcdm/revision/1 Boolean logic: https://www.bbc.co.uk/bitesize/guides/zqp9kqt/revision/1	Introduction to programming: https://www.bbc.co.uk/bitesize/guides/zts8d2p/revision/1	Micro:bit projects: https://microbit.org/projects/make-it-code-it/

	Year 9					
Term	**Autumn 1**	**Autumn 2**	**Spring 1**	**Spring 2**	**Summer 1**	**Summer 2**
	Cybersecurity and cybercrime	**Animations**	**Audiovisual representations**	**Data Science**	**Python with data**	**Physical and embedded computing**
Est Weeks	7	6	6	7	6	7
Knowledge	This topic focuses on online security and the risks to you and your data online. It looks at social engineering, hacking, bots, the Data Protection Act, the Computer Misuse Act, Hacking, Malware, protection methods such as anti-malware and authentication.	A look at animation using the program, Blender. Films, television, computer games, etc have been transformed by computer-based 3D modelling and animation. Using Blender to make animations, and to gain an appreciation of how this important creative field uses technology to model and animate.	Representing images and sounds digitally using audacity, paint.net and GIMP. Here the focus is on the making of digital media such as images and sounds, and the understanding of the underlying principles which underpin digital representations.	Introduction to data science with the empowerment it gives. Knowing how to use data to investigate problems and make changes to the world around them. Using both worldwide and local data sets and gain an appreciation of how visualising data can help with seeing relationships and trends.	Continued development of python programming skills with emphasis on data processing and presentation.	A look at embedded computing including the programming of a Micro: bit. Identify the key parts of a Micro: bit program and apply programming knowledge to extend existing programs. Further develop event driven programming skills and strengthen skills of decomposition, algorithm design.
Skills / Software	Critical thinking, self-awareness, and a better appreciation of societal issues surrounding technology and its use(s).	Blender – use of filters, creation of animations, using layers within animations, rendering	Image manipulation including altering colour balance, cropping, retouching, constructing images from layers, masking tools, flattening images etc	Data visualisation using MS Excel, online data representation tools, and	Mu IDE Visual Studio Python IDLE	Micro:bit projects:
NC links [National Curriculum Computing key stages 3 and 4](#)	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.	Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems. Create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design, and usability.	Understand how instructions are stored and executed within a computer system; understand how data of several types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits	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. Create, reuse, revise and	Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems. Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility	To make appropriate use of data structures and develop modular programs that use procedures or functions. Undertake creative projects that involve challenging goals. Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and

				repurpose digital artefacts for a given audience, with attention to trustworthiness, design, and usability	of alternative algorithms for the same problem. Use two or more programming languages, one of which is textual, to solve a variety of computational problems; make appropriate use of data structures such as lists, tables, or arrays.	physical systems
External Links	IBM cybersecurity game: https://www.ibm.com/security/digital-assets/cybersecurity-ops/terminal/ Graphics software: https://www.bbc.co.uk/bitesize/guides/zv2v4wx/revision/12 Collaboration: https://www.bbc.co.uk/bitesize/guides/zg6g87h/revision/1 Search engines: https://www.bbc.co.uk/bitesize/guides/zpkhpv4/revision/1	Graphics Software: https://www.bbc.co.uk/bitesize/guides/zv2v4wx/revision/1 Computer models: https://www.bbc.co.uk/bitesize/guides/zyqfr82/revision/1	Online image editor www.pixlr.com/e Representing test, images, and sound: https://www.bbc.co.uk/bitesize/guides/zpfdwmn/revision/1 Binary: https://www.bbc.co.uk/bitesize/guides/z26rcdm/revision/1 Bias and reliability: https://www.bbc.co.uk/bitesize/guides/z2g2mp3/revision/1	Pattern recognition: https://www.bbc.co.uk/bitesize/guides/zxxbgk7/revision/1	Sorting: https://www.bbc.co.uk/bitesize/guides/z2m3b9q/revision/1 Searching: https://www.bbc.co.uk/bitesize/guides/zgr2mp3/revision/1	Micro:bit projects: https://microbit.org/projects/make-it-code-it/