COMPUTING KNOWLEDGE AND SKILLS PROGRESSION OVERVIEW

Kingfisher Hall Academy

Ensuring That Our Computing is a Force For Positive Change

Intent

Computing enables children to find, explore, analyse, exchange and present information. We want children to know more, remember more and understand more in computing so that they leave primary school computer literate. Computing skills are a major factor in enabling children to be confident, creative and independent learners and it is our intention that children have every opportunity available to allow them to achieve this. Computing helps develop pupil's learning and results in the acquisition of knowledge of the world around them that ensures all pupils can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation - analysing problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems. A computing curriculum taught through effective lesson plans and DB Primary programmes, prepares pupils to live safely in an increasingly digital British society where pupils can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems and be a force for positive change.

Key Stage 1 National Curriculum Expectations

Pupils should be taught to:

- understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions;
- create and debug simple programs;
- use logical reasoning to predict the behaviour of simple programs;
- use technology purposefully to create, organise, store, manipulate and retrieve digital content;
- recognise common uses of information technology beyond school;
- use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies.

Key Stage 2 National Curriculum Expectations

Pupils should be taught to:

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts;
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output;
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs;
- understand computer networks including the internet; how they can provide multiple services, such as the world wide web, and the opportunities they offer for communication and collaboration;
- use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content;
- select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information;
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Computer Science Coding, programming, data handling	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	Program a (short set of) instructions on e.g. DB coding Identify and start to verbalise problems in a simple program (written by someone else). Start to give commands one at a time to control direction and movement. (DB coding) Begin to control the nature of events: repeat, loops, single events and add and delete features; (DB coding) Begin to improve/change their sequence of commands by debugging; (DB Primary) Understand what an algorithm is (a sequence of instructions or set of rules for performing a specific task) (DB Primary) Understand that algorithms need to be precise, simple, clear and limited. (DB Primary) Understand that an algorithm is implemented as program on a digital device. (DB Primary)	Understand what algorithms are, how they are implemented as programs on digital devices and that programs execute by following a sequence of instructions (DB Primary) Use logical reasoning to predict the behaviour of simple programs Follow and predict the outcome of a program write/input and test a simple a program/ code to achieve a desired outcome (DB Primary) Identify a bug in my code (where the algorithm has not achieved the desired outcome) debug a program (DB Primary) Identify and describe bugs in a simple program, and start to suggest corrections. Verbalise what will happen in a simple program before activating. (DB Primary)	Write programs that accomplish a simple purpose (e.g. a PowerPoint animation). Start breaking problems into smaller parts, e.g. the background (DB Primary) Debug a simple program independently, and start to identify bugs in their own work. (DB Primary) Explain how some simple algorithms work (DB Primary) Begin to use logical thinking to solve an open-ended problem by breaking it up into smaller parts; (DB Primary) write a program, putting commands into a sequence to achieve a specific outcome; (DB Primary) Start to use variables to create an effect, e.g. when, if, repetition, loop; (DB Primary) Start to talk about the different ways data can be organised;	Talk about the different ways data can be organised; sort and organise information to use in other ways; search a ready-made database to answer questions; Use logical thinking to solve an open-ended problem write a program, putting commands into a sequence to achieve a specific outcome; (DB Primary) Give a set of instructions to follow and predict what will happen; (DB Primary) Keep testing a program and recognise when it needs to be debugged; (DB Primary) Use variables to create an effect, e.g. repetition, if, loop (DB Primary) Start to design programs for a specific goal, i.e. planning before writing. Identify and fix bugs in their own programming, e.g. for a goal that's specified to them. Explain what logical reasoning means.	Design and write programs for a given purpose in more abstract contexts e.g. Excel formulas. Start using a range of inputs (e.g. sensors, music) to inform selection commands. Use precise language to explain how to debug a program. Begin to use external triggers and infinite loops to demonstrate control; (DB) Follow a sequence of instructions, e.g. in a flowchart and modify a flowchart using symbols; Start to use conditional statements and edit variables. (DB) Start to construct data on the most appropriate application. Begin to know how to interpret data, including spotting inaccurate data and comparing data. (DB) Use keyboard shortcuts and functions to input data on spreadsheets.	Solve problems they identify themselves, designing and writing programs to address this. Work confidently with sequence, selection, and repetition; work with variables and various forms of input and output. (DB) Alter and improve their own and others' programs, explaining why, and predicting and/or describing the effect. Know how to interpret data, including spotting inaccurate data and comparing data; Use keyboard shortcuts and functions to input data on spreadsheets and create formulas for spreadsheets; Add data to an existing database; Use conditional statements and edit variables; Decompose a problem into smaller parts to design an algorithm for a specific outcome and use this to write a program; (DB) Keep testing a program and recognise when it needs to be debugged;

Computer Literacy Multimedia – text and sound, IT, practical skills	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	Follow instructions to create content on simple editing programs like Word and Paint. Manipulate simple digital content Organise/store: Save files when the location is set for them. Retrieve: Do a simple search with support, e.g. for a postcode Find letters on a qwerty keyboard, e.g. type their name. Manipulate a mouse without looking (i.e. eyes on screen).	Go beyond teacher instruction to create content in simple editing programs. With support, create simple presentations e.g. a poster through Publisher or Word. Manipulate digital content e.g. highlight and delete text in Word, zoom in on a digital map. Follow instructions to save files to a specific location. Retrieve: Visit a known website and select some information (copying onto paper). Analyse/evaluate: Start to make selections, e.g. from or within sources of information. Navigate a qwerty keyboard, e.g. type a simple sentence and use cursor keys, back-space, etc. Use the double-click function.	Follow instructions to create content in a range of editing programs. Deliver a short presentation with digital content e.g. recount with photos on IWB. Copy and paste e.g. from website text into a Word document. Manipulate more digital content e.g. resize images, alter the font, take a screengrab. Save files appropriately without support. Retrieve: Perform a keyword search e.g. within Word or on a search engine. Start to select and order information according to relevance. Increased speed with a qwerty keyboard, e.g. can type several sentences in a lesson without struggling. Highlight, drag, right-click and double-click.	Use more than one finger to type letters, and both thumbs for the spacebar. Use a mouse to manipulate text, images and controls. Select between software (e.g. Publisher vs Word) and explain their reasoning. Deliver a short presentation with purpose-made digital backdrop e.g. PowerPoint. Use a range of selection, annotation and other tools, e.g. measuring distance -digimap. Take a screengrab and insert it into another program. Create and (re)name folders to collect digital content. Search and find files on a computer. Use a search engine and make decisions about which site to visit. Select and sort by relevance, start to analyse reliability and explain their reasons.	Select and use a range of editing software, and move simple content purposefully between programs. Create a presentation with text/images to support them in showcasing work Create and (re)name folders, moving files as appropriate. Find 'lost' files on a computer, identifying and recording the directory details. Use a search engine and explain the rationale/purpose behind which site they choose to visit. Select and sort by relevance and reliability, and explain their reasoning. Start to position hands correctly, moving fingers rather than arms to type. Confident use of a mouse.	Carefully select and move content within and between applications Use a range of supporting material to showcase their work, and take questions. Manipulate folders by creating, renaming and even deleting Find files, identify the directory details and move/resave elsewhere if appropriate. Use search technologies effectively, explaining how the algorithms select and rank results. Type at 2 letters per second, using different fingers and minimising arm/wrist movement. Confident use of a mouse.

Online Safety Research, e-safety	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	Recognise that their actions may have negative consequences. Identify what things count as personal information; Identify what is appropriate and inappropriate behaviour on the internet; Understand that there may be dangers online, and explain who they'll talk to if they're worried.	Understand who they can report things to if they're worried about anything digital. With support, set up a video conference call, e.g. Skype or FaceTime. Start to verbalise other people's needs and feelings. Agree and follow sensible online safety rules, e.g. taking pictures, sharing information, storing passwords; Seek help from an adult when they see something that is unexpected or worrying; Demonstrate how to safely open and close applications, log on and log off from websites;	Start to locate online safety procedures, e.g. the report abuse button, screengrabs to desktop. Able to list some forms of personal data (e.g. home address, date of birth). Write a short post on a safe site like ChatFOSS or Playkids Talk. Verbalise the possible consequences of their online behaviour. Be polite when challenging others' values and opinions. Show respect for online content, e.g. distinguishing between public and private material.	Understand and use a range of online safety procedures, e.g. saving a screengrab so it can be retrieved. Review privacy settings to protect personal data. Communications technology: Take simple notes from a conference call or phone call, e.g. directions to a shop. Write posts and respond respectfully on e.g. ChatFOSS, Playkids Talk, Kudos. Use their knowledge of consequences to manage and adapt their own behaviour. Show awareness of cultural and religious context, adjusting their style of challenge accordingly. Show respect for online content, e.g. by acknowledging a source.	Screengrab an online message and forward it by email. Verbalise what is meant by personal data, and explain how we might reveal it inadvertently. Take notes from a conference/phone call, asking clarifying questions and checking for accuracy. Write, reply-to and forward short emails. Using IT respectfully Challenge others' values and opinions sensitively (and when appropriately), beginning to cope with / compensate for other people's lack of respect. Show respect for online content, e.g. by investigating permissions.	Constantly show awareness of e-safety, e.g. checking settings as a priority within a new app. Automatically review their messages, texts, posts etc to check for personal data. Sensitively cope with and compensate for other people's lack of respect. Show awareness of, and explain, privacy, copyright and plagiarism.

Vocabulary	
KS1	❖ commands, add sound, tools, settings, undo, redo, text, image, size, poster, launch, application, software, window, minimise, restore, size, move, screen, close, click, drag, log on, log off, keyboards, keys, mouse, click, button, double click, drag, present, filter, Google, search engine, image, keyboard, email, internet, subject, address, communicate, sender, safe, secure, algorithm, instruction, order, debug, program, turn, left, right, clockwise, anticlockwise, blocks, sequence, project, repeat, repeat forever, invisible, grow, shrink, safe, meet, accept, reliable, tell, online, trusted, adult, information, safety, personal, key, question, tell, safe, share, stranger, danger, internet.
LKS2	❖ draw, object, shape, line, line colour, fill colour, group, ungroup, font, size, text box, format, image, wrap text, plan, link, image, object, link, hyperlink, minimise, restore, size, move, screen, split, create, organise, file, folder, close, exit, search, print, password, screenshot, snipping tool, shift, undo, redo, menu, dictionary, highlight, cursor, toolbar, spellcheck, audio, sound, video, movie, embed, link, file format, animate, animation, still image, thaumatrope, zoetrope, zoopraxiscope, stereoscope, flip book, frame, onion skinning, loop, frame rate, record, stop, play, stop motion, stop frame, Google Docs, insert, table, filter, Google, search engine, image, keyboard, email, subject, address, communicate, sender, safe, secure, internet, world wide web, social media, decompose, decomposing, logical sequence, flowchart, sprite, block, command, algorithm, answer, correct, errors, program, algorithm, instructions, commands, forward (fd), left (lt), right (rt), move, turn, clear screen (cs), variable, safe, meet, accept, reliable, tell, online, trusted, adult, information, safety, personal, internet, world wide web, communicate, message, social media, email, password, cyberbullying/bullying, plagiarism, profiles, account, private, public.
UKS2	❖ window, layout, text, font, colour, format, heading, hyperlink, 2D shape, 3D shape, orbit, pan, zoom, eraser, dimension, measurement, guide, audio, record, edit, play stop, skip, waveform, input, output, record, edit, play podcast, digital content, downloadable, backing track, voiceover, mute, gain, production, post-production, documentary, project, evaluation, screening, ceremony, upload, Google Docs, insert, table, spreadsheet, cell, row, column, formula/formulas, calculate, format, edit, insert, ascending, descending, world wide web, search, search engine, advanced search, results, Google, browser, terms of use, bias, authority, citation, plagiarism, source, website, secure, https, site, domain, website, browser, address bar, flowchart, algorithm, control, output, symbol, start, stop, delay, process, decision, loop, backdrop, script, block, repeat, commentary, sequence, consequence, debug, program, Kodu, world, object, tool palette, program environment, smooth, flatten, raise, spam, link, privacy, virus, scam, phishing, inbox, junk, sender, subject, secure, safe, account, online, private, social media, adverts, cyberbullying, reporting, anonymous, victim, fraud/fraudulent, policy, private/personal.