MANORFIELD

STONEY STANTON

COMPUTING CURRICULUM

1. INTENT, IMPLEMENTATION AND IMPACT

Intent

The aim of the Computing curriculum at Manorfield is to provide a high-quality computing education; which equips children to build and apply computational thinking and creativity to understand and change the world. The curriculum will teach children key knowledge in a safe and responsible manner.

We provide a knowledge rich, varied curriculum which fuses both key skills and concepts together whilst also challenging pupils to become greater computational thinkers. By using progressive objectives and revisiting strands, across year groups, using a range of themes, hardware and software, it allows children to securely embedded knowledge by applying key skills. Children develop the fundamental principles and concepts of computer science, analytical problem-solving skills and learn to evaluate and apply information technology skills.

A key part of learning is online safety. This is taught through Computing lessons as well as PSHCE lessons and assemblies. This enables children to become responsible, competent, confident and creative users of technology, both at school and at home.

We want children to leave Manorfield confident to use a range of hardware and software and to be able to produce high-quality purposeful products. At Manorfield, the curriculum we use gives pupils pieces of a toolkit which enables them to move forward to be an active participant in a vast growing digital world, becoming confident and respectful digital citizens, making the impossible possible and creating solutions to problems in everyday life.

Implementation

We have chosen to implement computing through the 'Teach Computing' scheme. This programme of work gives all teaching staff the confidence and the knowledge to best embed and cover every element of the computing curriculum.

A variety of hardware and software is used to create stimulating and engaging lessons that challenge the pupil's computational thinking. This can mean doing lessons away for the computer. These activities are physical in nature and provide kinaesthetic experiences which help pupils understand abstract concepts and deepen learning.

Every year group learns through units within the same four themes (computer systems and networks, creating media, programming and data and information). This means that each of the themes are revisited regularly and pupils revisit each theme through a new unit that consolidate and builds on prior learning within that theme. This approach allows us to use a spiral curriculum, progressing skills and concepts from one year group to the next.

Applications of online safety is implemented through Computing and PHSCE. Children have the opportunity to explore and respond to key issues such as digital communication, cyber-bullying, online safety, security, plagiarism and social media.

Impact

Our approach to the curriculum results in a fun, engaging and high-quality computing education. The quality of children's learning is evident in their confident and competency to navigate digital technology and through discussions and evaluation of their own work, as well as their peers.

Each class has a computing floor book, which highlights key learning within the classroom. Our floor books use a range of evidence, including pupil voice, to demonstrate children's learning. This is also used as a retrieval tool to activate prior knowledge and for children to reflect on their previous learning. Additionally, every child has their own Computing book which depicts individual elements of their learning journey.
Currently be trialled, floor books are being completed by children in Year 5, under supervision of the class teacher to ensure key elements of learning are being captured. This allows children to demonstrate and take ownership of their learning. After review, if successful, this will be rolled out to UPKS2 and potentially LKS2.

Knowledge and understanding is monitored by assessing the work according to the outcomes. This is used to feed into teachers' future planning, and it enables teachers to revisit misconceptions and knowledge gaps in future teaching. The outcomes of pupils will be monitored by the class teacher, subject lead and SLT through assessment and marking, tracking, book scrutiny and pupil interviews.

Much of the subject-specific knowledge developed in our Computing lessons equip pupils with experiences which will benefit them in secondary school, further education and future workplaces. From research methods, use of presentations and revisiting strands repeatedly through a range of themes during children's time in primary school will ensure the learning is embedded and skills are successfully developed. Computing at Manorfield gives children the building blocks that enable them to pursue a wide range of interests and vocations in the next stage of their lives and ensure they are competent and safe users of technology.

Assessment

Assessment is ongoing throughout each Computing topic. For each lesson a feedback sheet is completed, detailing misconceptions and next steps for learning, alongside a whole class Computing floor book. If a child has a misconception these are addressed in lesson or addressed in subsequent lessons. Summative assessment can be seen in a number of different ways in lessons. For example, a pupil may be asked to independently open, save or print a program.

Currently be trialled in KS1, is individual assessment sheets. Each assessment sheet contains the overview of the unit, with the key learning of each lesson. The class teacher can assess each child against each objective for the lesson. Alongside feedback sheets, this allows a detailed understanding of each individual and whole class understanding, feeding into next steps of learning. After review, if successful, will be rolled out across other phases.

What does greater depth look like in Computing?

- Children who approach problem solving situations with persistence, resilience and confidence.
- Children who have a firm grasp of Microsoft products (Word, PowerPoint, Excel etc.) and can use or combine these for a variety of purposes.
- Children who show a comprehensive understanding of coding and can work with various forms of input and output confidently.
- Children who are able to confidently evaluate the validity of a website and can state the source of the information found on the internet.
- Children who know how to navigate the internet safely and effectively and know what a problem looks like and how to report it immediately.
- Children who fully understand, explore and apply skills and ideas in different ways, in different situations and in different subjects.
- Children who can apply their knowledge from other subjects to help them solve technological problems.
- Children who are able to constantly review, analyse and evaluate their work and will make improvements without being asked

2. NATIONAL CURRICULUM COVERAGE

	Year 1			Year 2		
	AUT	SPR	SUM	AUT	SPR	SUM
Pupils Should be taught to:						
1. Recognise common uses of information technology beyond school.	Aut1		Sum1	Aut2		
2. Use technology purposefully to create, organise, store, manipulate and retrieve digital content.	Aut1 Aut2		Sum1 Sum2	Aut1	Spr2	Sum2
3. Use technology safely and respectfully, keeping personal information private.		Spr1	Sum1 Sum2			Sum1 Sum2
4. Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions.		Spr2	Sum1		Spr1	Sum2
5. Create and debug simple programs.		Spr2	Sum1		Spr1	Sum2
6. Use logical reasoning to predict the behaviour of simple programs.		Spr2	Sum1		Spr1	Sum2

	Year 3			Year 4		
	AUT	SPR	SUM	AUT	SPR	SUM
Pupils Should be taught to:						
1. Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.	Aut2	Spr2		Aut1 Aut2		
2. Use sequence, selection, and repetition in programs; work with variables and various forms of input and output.	Aut1 Aut2	Spr2		Aut1 Aut2		Sum2
3. Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.	Aut2	Spr2		Aut1 Aut2		
4. 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.	Aut1				Spr2	
5. Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.		Spr1	Sum2		Spr1	
6. 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.	Aut1 Aut2	Spr1 Spr2	Sum1 Sum2		Spr1	Sum1 Sum2
7. Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.			Sum1			Sum1

	Year 5			Year 6		
	AUT	SPR	SUM	AUT	SPR	SUM
Pupils Should be taught to:						
1. Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.	Aut1	Spr1	Sum2	Aut1		
2. Use sequence, selection, and repetition in programs; work with variables and various forms of input and output.		Spr1	Sum2			
3. Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.		Spr1	Sum2			
4. 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.	Aut1			Aut1		
5. Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.				Aut1		
6. 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.	Aut1 Aut2	Spr2	Sum1	Aut1 Aut2	Spr1 Spr2	Sum1 Sum2
7. Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.	Aut1			Aut1 Aut2	Spr1	Sum1 Sum2

3. LONG TERM PLAN

	Autumn		Spring		Summer	
Year 1	**Computing Systems and Networks** **Aspect:** Technology around us **Focus:** Recognising technology in school and using it responsibly.	**Creating Media A** **Aspect:** Digital painting **Focus:** Choosing appropriate tools in a program to create art and making comparisons with working non-digitally.	**Data Handling** **Aspect:** Grouping data **Focus:** Exploring object labels, then using them to sort and group objects by properties.	**Programming B** **Aspect:** An introduction to animation **Focus:** Design and programming the movements of a character on screen to tell stories.	**Programming A** **Aspect:** Moving a robot **Focus:** Writing short algorithms and programs for floor robots and predicting program outcomes.	**Creating Media B** **Aspect:** Digital writing **Focus:** Using a computer to create and format text, before comparing to writing non-digitally.
Year 2	**Creating Media B** **Aspect:** Making music **Focus:** Using a computer as a tool to explore rhythms and melodies, before creating a musical composition.	**Computing Systems and Networks** **Aspect:** IT around us **Focus:** Identifying IT and how it's responsible use improves our world in school and beyond.	**Programming B** **Aspect:** An introduction to quizzes **Focus:** Designing algorithms and programs that use events to trigger sequences of code to make an interactive quiz.	**Creating Media A** **Aspect:** Digital photography **Focus:** Capturing and changing digital photographs for different purposes	**Data Handling** **Aspect:** Pictograms **Focus:** Collecting data in tally charts and using attributes to organise and present data on a computer.	**Programming A** **Aspect:** Robot algorithms **Focus:** Creating and debugging programs and using logical reasoning to make predictions.
Year 3	**Computing Systems and Networks** **Aspect:** Connecting computers **Focus:** Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks.	**Programming A** **Aspect:** Sequence in music **Focus:** Creating sequences in a block-based programming language to make music.	**Creating Media B** **Aspect:** Desktop publishing **Focus:** Creating documents by modifying text, images, and page layouts for a specified purpose.	**Programming B** **Aspect:** Events and actions (maze) **Focus:** Writing algorithms and programs that use a range of events to trigger sequences of actions.	**Data Handling** **Aspect:** Branching databases **Focus:** Building and using branching databases to group objects using yes/no questions.	**Creating Media A** **Aspect:** Animation **Focus:** Capturing and editing digital still images to produce a stop-frame animation that tells a story

Year 4	**Programming A** **Aspect:** Repetition in shapes **Focus:** Using a text-based programming language to explore count-controlled loops when drawing shapes.	**Programming B** **Aspect:** Repetition in games **Focus:** Using a block-based programming language to explore count-controlled and infinite loops when creating a game.	**Creating Media B** **Aspect:** Photo editing **Focus:** Manipulating digital images, and reflecting on the impact of changes and whether the required purpose is fulfilled.	**Computing Systems and Networks** **Aspect:** The internet **Focus:** Recognising that internet as a network of networks including the WWW, and why we should evaluate online content.	**Creating Media A** **Aspect:** Audio editing **Focus:** Capturing and editing audio to produce a podcast, ensuring that copyright is considered.	**Data Handling** **Aspect:** Data logging **Focus:** Recognising how and why data is collected over time, before using data loggers to carry out an investigation
Year 5	**Computing Systems and Networks** **Aspect:** Systems and Searching **Focus:** Identifying how information is shared between digital systems. Understand how information is found on WWW, and how search engines work.	**Creating Media A** **Aspect:** Vector drawing **Focus:** Creating images in a drawing program by using layers and grouping objects.	**Programming A** **Aspect:** Selection in physical computing **Focus:** Exploring conditions using a programmable microcontroller.	**Data Handling** **Aspect:** Flat-file databases **Focus:** Using a database to order data and create charts to answer questions.	**Creating Media B** **Aspect:** Video editing **Focus:** Planning, capturing, and editing video to produce a short film.	**Programming B** **Aspect:** Selection in quizzes **Focus:** Exploring selection in programming to design and code an interactive quiz.
Year 6	**Computing Systems and Networks** **Aspect:** Communication and collaboration **Focus:** Explore how data is transferred over the internet, focussing on addresses then structure of data packets.	**Creating Media A** **Aspect:** 3D modelling **Focus:** Planning, developing, and evaluating £D computer models pf physical objects.	**Programming A** **Aspect:** Variables in games **Focus:** Exploring variables when designing and coding a game.	**Data Handling** **Aspect:** Spreadsheets **Focus:** Answering questions by using spreadsheets to organise and calculate data.	**Creating Media B** **Aspect:** Webpage creation **Focus:** Designing and creating webpages, considering copyright, aesthetics and navigation.	**Programming B** **Aspect:** Sensing **Focus:** Designing and coding a project that captures inputs from a physical device.

4. PROGRESSION OF SKILLS

Computing Systems and Networks

EYFS	Year 1	Year 2
— Use different digital devices. — Recognise that you can access content on a digital device. — Use a mouse, touchscreen, or appropriate access device to target and select options on screen. — Recognise a selection of digital devices. — Recognise the basic parts of a computer e.g., mouse, screen, keyboard. — Select a digital device to fulfil a specific task, e.g., to take a photo	— Recognise a range of digital devices. — Select a digital device to fulfil a specific task, e.g., to take a photo. — Name a range of digital devices, e.g. laptop, phone, games console. — Log on to the school computer / unlock the school tablet with support. — Identify the basic parts of a computer, e.g., mouse, keyboard, screen. — Use a suitable access device (mouse, keyboard, touchscreen, switch) to access and control an activity on a computer. — Open key applications independently. — Save and open files with support. — Add an image to a document from a given folder/source with support.	— Recognise what a computer is (input > process > output). — Recognise that a range of digital devices contain computers, e.g. phone, games console, smart speaker. — Explain what the basic parts of a computer are used for. — Identify and use input devices, e.g. mouse, keyboard; and output devices, e.g., speakers, screen. — Open key applications independently. — Save and open files to/from a given folder. — Add an image to a document from a given folder/source. — Resize an image in a document. Highlight text and use arrow keys. — Capture media independently (e.g. take photos, record audio).

Year 3	Year 4	Year 5	Year 6
— Describe what a computer is (input > process > output). — Explain the difference between input and output devices on a computer. — Know where to save and open files (e.g., in shared folder). — Save files with appropriate names. — Use a keyboard effectively to type in text. — Use left-, right and double-click on the mouse. — Add an image to a document from the internet. Resize and move an image in a document.	— Recognise that you can organise files using folders. — Explain what a good file name would look like. — Delete and move files. — Use key parts of a keyboard effectively, e.g., shift, arrow keys, delete). — Know how to copy and paste text or images in a document. — Crop an image and apply simple filters. — Use a search engine to find specific information — Recognise that school computers are connected on a network.	— Type using fingers on both hands. — Use common keyboard shortcuts, e.g., ctrl C (copy), ctrl V (paste). — Explain what makes a strong password. — Use folders to organise files. — Know how to mute and unmute audio on a computer or tablet. — Recognise that there is more than one search engine, and they may produce different results. — Use a search engine effectively to find information and images. — Know how to search for an application on a computer/tablet.	— Type efficiently using both hands. — Use a range of keyboard shortcuts. — Recognise that different devices may have different operating systems. — Organise files effectively using folders and files names. — Use the advanced search tools when using a search engine to find specific informa tion and images. — Explain the basic function of an operating system. — Recognise common file types and extensions e.g. jpeg, png, doc, wav — Recognise a range of Internet services,

— Use a search engine to find simple information. — Recognise that school computers are connected.			e.g., email, VOIP (e.g., Skype, FaceTime), World Wide Web, and what they do.

Presenting Information and Creating Multimedia

EYFS	Year 1	Year 2
— Use technology to explore and access digital content. — Operate a digital device with support to fulfil a task. — Create simple digital content, e.g. digital art.	— Create digital content, e.g., digital art. — Choose media from a selection (e.g. images, video, sound) to present information on a topic. — Recognise that you can find out information from a website. — Recognise that you can edit digital content to change its appearance. — Select basic tools/options to change the appearance of digital content, e.g., filter on an image / font / size of paintbrush. — Combine media with support to present information, e.g., text and images.	— Create simple digital content for a purpose, e.g. digital art, poster, music. — Recognise that we can use technology to record and playback audio or take and view photographs. — Apply edits to digital content to achieve a particular effect, e.g., emphasise part of a text. — Present ideas and information by combining media, e.g., text and images. — Explain that you can search for information on the internet. — Plan out digital content, e.g., a simple sketch or storyboard. — Identify the common features of digital content, e.g. title, images. — Recognise that we can use different types of media to convey Information, e.g. text, image, audio, video.

Year 3	Year 4	Year 5	Year 6
— Present ideas and information by combining media independently, e.g. text and images. — Design and create simple digital content for a purpose/audience, e.g., poster. — Edit digital content to improve it, e.g., resize text. — Identify the features of a good piece of digital content. — Explain why we use technology to create digital content.	— Collect, organise and present information using a range of media. — Design and create digital content for a specific purpose, e.g. poster, animation. — Edit digital content to improve it according to feedback. — Identify the features of a good piece of digital content and apply these in own design. — Explain the benefits of using technology to present information. — Know where to find copyright-free content,	— Identify and use appropriate hardware and software to fulfil a specific task. — Remix and edit a range of existing and their own media to create content. — Consider the audience when designing and creating digital content. — Recognise the benefits of using technology to collaborate with others — Identify success criteria for creating digital content for a	— Select, combine and remix a range of media to create original content. — Consider all steps of the design process when creating content (e.g., identify problem, plan, create, evaluate, share.) — Identify the most effective tools to present information for a specific purpose. — Explain the benefits of using technology to collaborate with others.

— Recognise why we use different types of media to convey information, e.g. text, image, audio, video.	e.g., creative commons images. — Collaborate with peers using online tools, e.g., blogs, Google Drive, Office 365, if available.	given purpose and audience. — Evaluate their own content against success criteria and make improvements accordingly.	— Evaluate existing digital content in terms of effectiveness and design.

Data and Information

EYFS	Year 1	Year 2
— Access content in a range of formats, e.g., image, video, audio. — Answer basic questions about information displayed in images e.g..	— Recognise different forms of digital content, i.e., text, image, video and audio. — Collect simple data (e.g., likes/dislikes) on a topic. — Present simple data using images, e.g. number of animals. — Recognise tally charts and pictograms and why we use them.	— Recognise tally charts, charts, pictograms and branching databases, and why we use them. — Explain information shown in a simple chart or pictogram. — Identify the key features of a chart or pictogram. — Collect data on a topic (eye colour, pets etc.) and present in a pictogram or chart. — Modify simple charts/pictograms, e.g., add title, item or labels.

Year 3	Year 4	Year 5	Year 6
— Recognise charts, pictograms and databases, and why we use them. — Present information using a suitable chart — Explore a record card database to find out information. — Use filters in a database to find out specific information. — Name the key parts of a database, e.g. record, field, search. — Answer questions about information in a database. Create questions using yes or no. — Name some benefits of using a computer to create charts and databases. — Recognise that search engines store information in databases.	— Draw conclusions from information stored in a database, chart or table. — Design a questionnaire and collect a range of data on a theme. — Choose appropriate formats to present data to convey information. — Recognise that data can be collected on digital devices and sensors automatically. — Use a computer program to sort data by attributes. — Present the same date in a graph and in a chart. — Know that you use a web browser to access information stored on the internet. — Appreciate that you need to use specific software to work with	— Explain the difference between data and information. — Appreciate that different programs work with different types of data, e.g. text, number, video, paper database. — Explain the difference between the Internet and the World Wide Web. — Know the difference between a search engine and a web browser. — Explain the basics of how search engines work. — Perform searches for information using advanced settings in search engines. — Recognise the — benefits and risks of sharing data online.	— Recognise what a spreadsheet is and what it is used for. — Explain the difference between physical, mobile and wireless networks. — Use simple formulae in a spreadsheet to find out information from a set of data. — Collect data for a purpose and plan out a spreadsheet to present it effectively, using relevant formulae. — Produce graphs from data in a spreadsheet to answer a question. — Analyse and evaluate data and information in a spreadsheet, chart or database. — Recognise that poor quality data leads to unreliable results.

— Compare database and branching data to a picogram.	video, images, audio etc.	— Use, Create and compare visual databases.	

Programming and Algorithms

EYFS	Year 1	Year 2
— Explore technology — Repeat an action with technology to trigger a specific outcome. — Recognise the success or failure of an action. — Follow simple instructions to control a digital device. — Recognise that we control computers. — Input a short sequence of instructions to control a device.	— Recognise that computers don't have a brain. — Explain that we control computers by giving them instructions. — Create a simple program e.g., to control a floor robot. - Create a simple algorithm. — Predict the outcome of a simple algorithm or program. — Explain what an algorithm is – a sequence of instructions to make something happen. — Recognise that the order of instructions in an algorithm is important. — Debug an error in a simple algorithm or program e.g., for a floor robot.	— Explain that computers have no intelligence and we have to program them to do things. — Create a program with multiple steps e.g., to control a floor robot. — Predict the outcome of an algorithm or program with multiple steps. — Recognise that the instructions in an algorithm need to be clear and unambiguous. — Identify and correct errors in each algorithm or program and recognise the term debugging. — Explain what an algorithm is, and that when inputted on a computer it is called a program. — Plan out a program by creating an algorithm and evaluate its success.

Year 3	Year 4	Year 5	Year 6
— Predict the outcome of a block or text-based program (Scratch/discovery coding). — Modify an existing program, e.g., change background, number of times things happen — Identify repeated steps in a program or algorithm. — Create examples of algorithms containing count-controlled loops. — Use a count-controlled loop to make a program more efficient. — Recognise that we can create an algorithm to help plan out a program. — Recognise and use a forever loop in a program or algorithm. — Identify errors in a block or text-based	— Create a program using a range of events/inputs to control what happens. — Recognise that we can decompose a problem into smaller parts to help solve it. — Explain when to use forever loops and count-controlled — loops, and use them in programs. — Recognise selection in a program or algorithm. — Use selection in algorithms in — programs to alter what happens when a condition changes, e.g., if…then… — Design a program for a purpose. — Recognise common mistakes in programs and how to correct them.	— Name a range of sensors in physical systems. — Recognise that different solutions may exist for the same problem. — Predict what will happen in a program or algorithm when the input changes (e.g. sensor, data or event). — Use two-way selection in programs and algorithms, i.e. if…then…else… — Recognise variables in a program and what they do. — Create programs including repeat until loops. — Create and use simple variables, e.g., to keep score. — Evaluate a program and make improvements to the	— Design and program a physical computing system that uses sensors. — Recognise and use procedures (sub-routines) in programs. — Plan out a program in detail, including task, algorithm, code and execution level. — Explain common errors in programs and how to fix them. — Use nested selection statements in a program or algorithm effectively. — Combine a variable with relational operators (< = >) to determine when a program changes, e.g., if score > 5, say "well done". — Recognise key concepts (sequence, selection, repetition and variables) in a

program and correct them. — Recognise that different inputs can be used to control a program.		code or design accordingly. — Create an algorithm for a physical system containing a sensor.	range of languages and contexts.

5. SEND IN COMPUTING

We teach Computing to all children, whatever their ability. DT forms part of the school's curriculum policy to provide a broad and balanced education to all children. We provide learning opportunities matched to the needs of children with learning difficulties and we consider each child's abilities.
"*A core principle of the Teach Computing Curriculum is inclusive and ambitious. The Teach Computing Curriculum has been written to support all pupils. Each lesson is sequenced so that it builds on the learning from the previous lesson, and where appropriate, activities are scaffolded so that all pupils can succeed and thrive. Scaffolded activities provide pupils with extra resources, such as visual prompts, to reach the same learning goals as the rest of the class. Exploratory tasks foster a deeper understanding of a concept, encouraging pupils to apply their learning in different contexts and make connections with other learning experiences. As well as scaffolded activities, embedded within the lessons are a range of pedagogical strategies which support making computing topics more accessible."* Teach Computer – Teacher Guide – Core Principle.

TASKS	PROBLEM SOLVING	HIGH EXPECTATIONS
Incorporate learning materials that are accessible for learners of all abilities. For learners with special educational needs and disabilities, specific resources or approaches may be required to enable them to access the curriculum. Ensure you have considered what barriers learners may have within a lesson and embed support strategies to help them overcome these. Scaffold learning so that learners benefit from support during initial phases of learning. Adapt tasks to make the curriculum accessible to all.	In computer science, there can be multiple solutions to a problem. Focus your instruction and encouragement on solving problems and the problem-solving process, rather than finding a single right answer. Emphasize guided inquiry, designing learning opportunities where learners can ask questions, explore, try different approaches and challenge their own and each other's ideas. Encourage learners to take ownership over their learning. If a learner struggles with complex, multi-step problem solving, give them additional support in the beginning, then slowly remove the support once learners build their skills and confidence.	One of the largest subject barriers we face is learners' own belief systems about who can succeed in computer science. If a teacher holds lower expectations of a learner, it can have a negative impact on a learner's achievement in the subject. Encourage learners to reflect on their perspectives and potential biases and challenge yourself to do the same. Build relationships with learners to identify opportunities to connect learning to their personal experience. Look for stories and experiences about using computer science that will be meaningful and relatable to your learners.
Creating an Inclusive Environment		
Vocabulary	**Vision Impairment**	**Space**
Whilst you model the skills and understanding required to develop a rich vocabulary knowledge, consider your use of words within a lesson. Familiarise learners with Tier 2 words by embedding them into classroom displays and lesson activities. It's	At Key Stage 1 and 2, coding is primarily taught using block-based programming languages such as Scratch. Carefully consider what inclusive practices are appropriate. For example, embedding the use of braille, allowing learners to orient themselves to the classroom space, careful selection of colours within resources,	The learning environment is important in making learners feel included. Incorporate visuals that will appeal to a wide range of learner interests and backgrounds. Include examples of learners and professionals with disabilities, the representation of a diverse range of figures in computing can send a

important that you find ways for learners to encounter these terms, as this will empower them to access a higher level of language with which they can communicate and understand ideas across the curriculum.	installing a screen reader and magnifier aids. Together these approaches support learners in solving complex challenges.	powerful message to your learners. Arrange the learning space to promote collaboration and hands-on activities, whilst also being mindful of how learners will access their workstations. Arrange aisles and workstations so that learners with mobility aids can get to all the areas they need to access to participate fully. Consider what assistive technology devices could be embedded into practice to give opportunities for all learners to fully access lesson content.
Curriculum Considerations	**Key Stage 1**	**Key Stage 2**
Computing equips learners to use computational thinking and creativity to understand the digital world we live in. Computing has deep links with mathematics, science and design and technology, and ensures that learners become digitally literate, offering the opportunity to learn in different ways.	At this stage, learning should be focused on the concept of computational thinking and equipping learners with the skills to tackle challenging problems using logical reasoning. Practical activities that encourage them to get hands-on with problems can help them visualise solutions. Giving learners the opportunity to predict behaviour of simple programs can also develop their problem-solving skills. It's important to use and to teach learners the correct technical terminology within lessons, to ensure that misconceptions are not embedded early into their computing education.	At this stage, learners begin to apply and build upon the skills learnt at Key Stage 1 through designing and writing programs that accomplish specific goals. Learners should be able to detect and correct errors in algorithms. When teaching learners to solve various problems, encourage them to be resilient and think outside the box. Learners should also be shown how to use technology safely, respectfully and responsibly. Learners need to be able to identify unacceptable behaviour and know how to report concerns.

Strategies to Scaffold Learning			
How can I support learners who struggle to access lessons because of literacy difficulties?	How can I support learners who struggle to retain vocabulary?	How can I support learners who need additional time to develop conceptual understanding?	How can I support learners who struggle with attention?
Model the correct use of vocabulary. Show examples of common errors/misconceptions and work with learners to improve literacy within given text. For those with appropriate access arrangements, encourage the use of a	Embed opportunities to recall key terms within lessons. Memorisation techniques such as tracked retrieval practice can give learners the opportunity to revisit topics across the curriculum.	Model answers and get learners to look at and discuss completed examples. Assess and use learners' prior knowledge to create links between old and new content.	Learn what hobbies or topics the learners are interested in. Find ways to incorporate this into lessons and questions. Use learners' names in written questions to further engage them in text. Give clear instructions within the form of a checklist. This will break down the task into more manageable chunks.

reader to support learners in reading and interpreting large sections of text. Chunk key information and create clear, easy-to-follow checklists. This can help your learner focus on one section at a time and have a clear set of goals. During classroom discussions, listen to the answers given and when re-iterating points, rephrase sentences to include key vocabulary. Consider your classroom display and how you can promote the definitions and use of Tier 2 words. Provide learners with a glossary of key terms which they can refer to during the lesson	Provides learners with a glossary of key terms which they can refer to during the lesson. Use rephrasing techniques to strengthen learner answers with correct vocabulary. Introduce new terms slowly and rehearse news words. Get learners to interact with the key terms in various ways such as writing, speaking, mini games, questioning and more.	Walk through examples together, giving learners the opportunity to ask questions. Address misconceptions early.	Praise learners on their contributions and for targets met, encourage them to continue and to have a growth mindset. Consider the learning environment and potential distractions and make appropriate arrangements to remove these barriers. Ensure instructions are clear and signposted. Be concise in teacher-led delivery. Chunk material in larger topics so learners can complete a range of engaging activities. Check in with the learners throughout the activity, initially to check they have understood the task, to praise work completed and to challenge them further.

6. KEY KNOWLEDGE AND VOCABULARY

Computing Key Concept Knowledge						
EYFS	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Computing Systems and Networks						
	To explain that technology is something that can help us.	To recognise different types of computers used in school.	To recognise that a digital device is made up of several parts. To recognise that a network is made up of a number of components.	To recognise that the World Wide Web is part of the internet. To outline how information can be shared via the World Wide Web.	To explain that computers can be connected together to form IT systems. To explain how search results are selected.	To recognise that data is transferred across networks using agreed protocols (data packets). To recognise computers connected to the internet allow people in different places to work together.
Presenting Information and Creating Multimedia						
	To recognise computers can be used to create art. To recognise that a keyboard is used to enter text into a computer. To recognise that the appearance of text can be changed	To recognise that some digital devices can capture images using a camera. To recognise that photographs can be change after they have been taken. To identify that computers can be used to play sounds	To explain that an animation is made up of a sequence of images. To recognise how text and images can be used together to convey information To recognise how different font styles and effects are used	To identify that an input device is needed to record sound. To identify that output devices are needed to play audio. To recognise that audio can be edited.	To identify that a vector drawing comprises separate objects and can be modified separately or as groups. To recognise that filming techniques can be used to create different effects. To identify that	To explain that 3D models can be created on a computer. To recognise that web pages are written by people. To recognise that web pages can contain different media types. To recognise that a

		of different instruments.	for particular purposes.	To use an application to change a part or a whole digital image.	videos can be edited on a recording device or on a computer.	website is a set of hyperlinked web pages.
Data and Information						
	To recognise that information can be presented.	To explain that we can present information using a computer. To use a computer program to present information in different ways.	To explain that a branching database is an identification tool, structured using yes/no questions.	To recognise that a sensor can be used as an input device for data collection, over time.	To explain that a computer program can be used to organise data. To explain that computer programs can be used to compare data visually.	To explain what an item of data is in a spreadsheet. To explain that formulas can be used to produce calculated data.
Programming and Algorithms						
	To understand that a program is a set of commands that a computer can run. To combine commands in a program.	To describe that a series of instructions is a sequence. To recognise that you can predict the outcome of a program. To explain what happens when we change the order of instructions.	To explain that programs start because of an input. To explain that the order of commands can affect a program's output.	To explain that we can use a loop command in a program to repeat instructions. To explain that in programming there are indefinite loops and count-controlled loops.	To explain that selection can be used to branch the flow of a program. To explain that a loop can be used to repeatedly check whether a condition has been met.	To define a 'variable' as something that is changeable. To explain that a variable can be used in a program, eg 'score'.

Computing Vocabulary Progression

	Computing Systems and Networks	Creating Media A	Programming A	Data and Information	Creating Media B	Programming B
Year 1	**Technology around us** technology, computer, laptop, desktop, keyboard, screen, click, drag, mouse, type, save, file, cursor, delete, text, Log in, username, password, log out	**Digital painting** Paint program, tool, paintbrush, erase, fill, undo, Piet Mondrian, primary colours, shape tools, line tool, fill tool, undo tool, Henri Matisse, Wassily Kandinsky, feelings, colour, brush style, George Seurat, Pointillism, prefer, dislike, like	**Moving a robot** Forwards, backwards, turn, clear, go, commands, instructions, directions, left, right, plan, algorithm, route, program	**Grouping data** Object, label, group, search, image, colour, shape, property, value, data set, less, most, fewest, the same	**Digital writing** Word processor, keyboard, keys, letters, Microsoft Word, letters, numbers, space, backspace, text cursor, toolbar, bold, italic, underline, undo, font, toolbar	**An introduction to animation** ScratchJr, command, sprite, compare, programming, programming area, block, joining, start, program, background, delete, reset, algorithm, predict, effect, change, value, block, instructions,
Year 2	**IT around us** Information technology (IT), computer, barcode, scanner/scan	**Digital photography** Device, camera, photograph, capture, image, digital, landscape, portrait, horizontal, vertical, field of view, narrow, wide, format, framing, focal point, subject, matter, flash, focus, background, foreground, editing, filter, changed, real	**Robot algorithms** Instruction, sequence, clear, unambiguous, algorithm, program, order, commands, prediction, artwork, design, route, mat, debugging	**Pictograms** More than, less than, most, least, organise, data, object, tally chart, votes, total, pictogram, enter, data, tally chart, compare, count, explain, attribute, group, same, different, most popular, least popular	**Making music** sound effects, pattern rhythm, pulse, pitch, tempo, note, instrument, create, open edit.	**An introduction to quizzes** Sequence, command, program, run, program, start, predict, blocks, actions, sprite, modify, match, debug, features, evaluate
Year 3	**Connecting computers** Digital device, input, output, process, program, connection, network, network switch, server, wireless access point (WAP)	**Animation** Animation, flip book, stop frame, animation, frame, sequence, image, photograph, setting, character, events, onion skinning, consistency, delete, frame, media, import, transition	**Sequence in music** Scratch, programming, blocks, commands, code, sprite, costume, stage, backdrop, motion, turn, point in direction, go to, glide, event, task, design, code, run the code, order, note, chord, algorithm, bug, debug	**Branching databases** Attribute, value, questions, table, objects, branching databases, objects, equal, even, separate, order, organise, selecting, pictogram, information, decision tree, questions	**Desktop publishing** Text, images, advantages, disadvantages, communicate, font, style, template, desktop publishing, copy, paste, layout, purpose, benefits	**Events and actions (maze)** Motion, event, sprite, algorithm, logic, move, resize, algorithm, extension block, pen up, set up, design, action, debugging, errors, setup, test
Year 4	**The internet** Internet, network, router, network security, network switch, wireless access point (WAP), router, website, web page, web address, router, routing,	**Audio editing** Audio, record, playback, microphone, speaker, headphones, input, output, start, stop, podcast, save, file, selection, edit, mixing,	**Repetition in shapes** Program, turtle, commands, code, snippet, algorithm, design, debug, logo commands, pattern, repeat, repetition, count-controlled	**Data logging** Data, table (layout), input device, sensor, data logger, logging, data point, interval, analyse, import, export, logged, collection, analyse, review, conclusion	**Photo editing** Image, edit, arrange, select, digital, crop, undo, save, search, copyright, composition, save, pixels, rotate, flip, adjustments, effects, colours,	**Repetition in games** Scratch, programming, sprite, blocks, code, loop, repeat, value, forever, infinite loop, count-controlled loop, animate, costume, event block,

	route tracing, browser, World Wide Web, content, links, files, use, download, sharing, ownership, permission, accurate, honest, adverts	time shift, export, MP3, evaluate, feedback	loop, value, decompose, procedure		hue/saturation, sepia, version, illustrator, clone, recolour, magic wand, sharpen, brighten, fake, real, composite, background, foreground, retouch, paste, alter, publication, elements, original, font style, border, layer	duplicate, modify, debug, refine, evaluate, algorithm
Year 5	**Sharing information** System, connection, digital, input, process, output, search, search engine, refine, index, crawler, bot, ordering, ranking, links, algorithm, search engine optimisation (SEO), web crawler, content creator, selection	**Vector drawing** Vector, drawing tools, shapes, object, icons, toolbar, move, resize, colour, rotate, duplicate/copy, zoom, select, alignment grid, handles, consistency, modify, layers, front, back, copy, paste, group, ungroup, reuse, improvement, evaluate, alternatives	**Selection in physical computing** Microcontroller, crumble controller, components, LED, Sparkle, crocodile clips, connect, battery box, program, repetition, infinite loop, count-controlled loop, condition, true, false, input, action, selection, motor, switch, algorithm, debug, evaluate	**Flat-file databases** Database, data, information, record, field, sort, order, group, search, criteria, value, graph, chart, axis, compare, filter, presentation	**Video editing** Video, audio, recording, storyboard, script, soundtrack, dialogue, capture, zoom, storage, digital, tape, AV (audiovisual), videographer, video techniques, zoom, pan, tilt, angle, content, camera, colour, export, trim/clip, titles, end credits, timeline, transitions, soundtrack, retake/reshoot, special effects, constructive feedback	**Selection in quizzes** Selection, condition, true, false, count-controlled loop, outcomes, conditional statement – the linking together of a condition and outcomes, algorithm, program, debug, implement, question, answer, task, input, outcomes, test, run, setup, share, evaluate, constructive
Year 6	**Communication and collaboration** Communication, protocol, data, address, Internet Protocol (IP) address, Domain Name Server (DNS) Packet, header, data payload, chat, explore, slide deck, reuse, remix, collaboration, internet, public, private, one-way, two-way, one-to-one, one-to-many	**3D modelling** 2D, 3D, 3D object, 3D space, view, resize, colour, lift, rotate, position, select, duplicate, dimensions, placeholder, hole, group, ungroup, modify, evaluate, improve	**Variables in games** algorithm, code, task, artwork, program, project, code, test, debug, improve, evaluate, share	**Spreadsheets** spreadsheet, data, data heading, data set, cells, columns and rows, data item, format, common attribute, formula, calculation, call reference, sigma, graph, evaluate, results, comparisons, questions, software, tools, data, propose	**Webpage creation** Website, web page, browser, media, Hypertext Markup Language (HTML), layout, header, media, purpose, copyright, fair use, evaluate, preview, device, breadcrumb, trail, navigation, hyperlink, subpage, implication, external link, embed	**Sensing** Micro-bit, input, process, output, flashing, USB, selection, condition, if… then… else, variable, random, navigation, design, task, step counter, plan, create, code, test, debug

7. RESOURCES AND WEBSITES

Ofsted Research review series: computing - GOV.UK (www.gov.uk)

Teach Computing Curriculum https://teachcomputing.org/curriculum