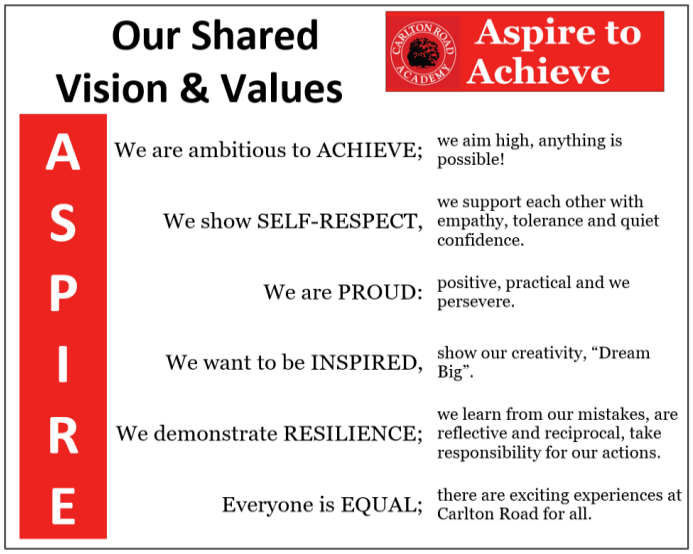
Computing Policy

The Context of our School and its Curriculum



Carlton Road Academy is a two-form entry school with our own attached Nursery based in Boston, Lincolnshire. With a cohort drawn from the immediate area, the school serves a diverse community with a greater-than-average number of EAL and Pupil Premium students; mobility is high. As a result is key that our approach to teaching and learning is accessible to all children, regardless of their background. Our ethos "Aspire to Achieve" is embedded throughout the school, it's curriculum and our knowledge expectations – we expect our children to "Aim High and "Dream Big".

Our "Aspire" curriculum brings to life the school's ethos and values. It embraces the whole child and their success in education – both academic ambition, practical skills and social achievements. At the same time, we expect the children to be proudly responsible of and for their own efforts, to persevere when the going is not always easy thereby making their individual contribution to the shared, equalitarian and democratic learning experience at Carlton Road.

Curriculum Intent

Computing

At Carlton Road Academy the primary aim of Computing is to equip our learners with the knowledge and skill development that will prepare them for a rapidly increasing digital world. This knowledge and understanding of Computing is becoming exponentially more important for children of this generation in terms of future employment opportunities.

Our Computing Curriculum has specific focuses to allow the children to be diverse in their understanding of technology, these are: Computing Systems and Networks, Programming, Data and Information and Creating Media. Online Safety and Security are interwoven within each block as is the effective use of tools and impact of technology in society, each ensuring that our children are becoming confident and competent in using a range of technology in a safe and effective way.

The intent of the computing curriculum at Carlton Road Academy is to deliver it in an accessible way, ensuring that all children develop a deeper understanding of just how important information technology is and will be to them as they grow up – being aware of how it not only effects their own daily lives but also in understanding the mechanics behind software, coding and technology as they know it.

Our curriculum promotes	A	Achievement,	all learners are academically ambitious, aim high, maximising their own potential from their considerably different starting points. Everyone reads with enjoyment and enthusiasm. Anything is possible!
Our curriculum instils	S	Self-respect,	creating confidence, tolerance and mutual respect within the whole school community, children support and encourage each other upholding British Values.
Our curriculum promotes pupil's	P	Pride,	they feel part of something bigger, maintain positivity, take pleasure in their work, using both practical skills and their own mindset to help them persevere.
Our curriculum enables students to be	I	Inspired;	its varied content and wide-reaching subject matter offers a wealth of learning opportunities that encourage creativity, from close to home and further afield; we "Dream Big" at Carlton Road.
Our curriculum develops learners who are	R	Resilient	to be honest in their understanding and learn from their mistakes, overcome setbacks and challenges, to take responsibility for their own learning, to be reciprocal and reflective in lessons.
Our curriculum embodies	E	Equality;	everyone is entitled to exciting experiences, cultural, sporting and enriching and to have their views and voices heard.

What out intent looks like in Computing

For all our pupils to develop through Computing, it is our responsibility to ensure that we prepare our children for a technological future. We want our children to become 'masters of technology' by the time they leave our academy.

A	**Achievement**	Children are given the tools to be successful and achieve in their own unique way. Achievement in Computing may differ from child to child but being able to present their ideas in a range of different formats will allow them to demonstrate their broad knowledge and understanding. Children will have been exposed varying levels of computing and technology outside of school, so it is important that they are aware that all achievements are viewed and celebrated in the same way, regardless of their complexity.
S	**Self-respect**	Children are made aware that it is easy to become comparable with their peers, especially over social media as well as in school. We encourage a culture of ensuring the children respect themselves through the use of technology and not minimize their achievements as a result of social comparison with others. This also applied to the work and content they create in lessons.
P	**Pride**	Throughout Carlton Road Academy, we want our children to feel a sense of pride about what they have produced, created or designed. We want them to be able to look back at the journey they have been on and be proud of how far they have come – from logging onto devices to being able to program and code their own characters and games.
I	**Inspired**	From a young age, children are emersed in the world of technology. We want them to realise and be inspired by it by demonstrating how it can allow them to become a ground-breaking or important ambassador for society in the future such as becoming a: programmer, teacher, digital marketer, web designer, app developer or scientist. It is important that the children know and understand the positive impact that a computing and technological world can have on their lives.
R	**Resilient**	Children are taught and encouraged how to be resilient in all aspects of their school journey and Computing is no different. In fact, solving errors and re-programming tasks is a fundamental part of the Computing curriculum. Children should be reassured that technology can present many difficulties and challenges, but it is important to remain resilient in order to maintain engagement with the subject.
E	**Equality**	Equal opportunities and equality is entwined within the teaching of Computing. We ensure every pupil has the opportunity to participate in all areas of learning through the use of various technological platforms. We ensure our learners experience a wide variety of digital media: laptops, i-pads, recording equipment, scientific equipment, audio, video, digital imagery and animation.

Curriculum Implementation

How we deliver our curriculum:

Teaching and learning turns 'thinking' (the task of the working memory) into 'knowledge' (our long-term memories) so that it can be recalled and used again and again.

The table below demonstrates the types of knowledge the children acquire and what that looks like in Computing.

	Forms of Knowledge	What that knowledge looks like in school	What that knowledge looks like in Computing…
A	**Academic Answers**	Children encounter facts and learn knowledge that is 'known' for example: number bonds, spellings, capital cities, the wives of Henry VIII, colours. Facts that can be straightforwardly shared, memorised and recalled.	The children will be exposed to and know the appropriate computing vocabulary in the context of what is being taught e.g., 'algorithm', 'network', 'cyberbullying', 'coding' or 'sequence'. Retrieval of this knowledge will be ongoing to ensure it becomes memorable and is embedded. Such activities will form part of an assessment of overall understanding for teachers, helping to inform future planning and provide additional intervention where required.
S	**Situational and Symbolic**	Children interpret knowledge in the context of what they comprehend from the cultures they know (context, community heritage). This includes their understanding of symbols such as: written, gestures, body language, pictorial, coded (such a computers or road signs).	Computing and technology is an immersive part of the children's lives, both at home and at school and we have a responsibility to ensure that connections between the two are made. During their time at Carlton Road Academy, the children will know that almost all electronic devices in their home have been coded and programmed; they will learn how this is done at school. They will make connections with computer symbols relating to, for example, on/off, contrast, mathematical operations, sound, letters and coding.
P	**Practical – the 'How to?'**	Children learn practical knowledge when they need to know the 'How to…'e.g., ride a bike, read a map. The knowledge may come in steps or stages. It could be written down to follow like a recipe or automatically retrieved, once learnt, such as how to swim.	It is essential that during Computing the children participate in activities they can independently carry out such as turning devices on/off and logging into apps and programs. This is one of many key aspects to their sequence of learning and once understood will allow access to the further learning of 'how to' tasks. For example, children must know what an 'algorithm' is to manipulate, change, fix, de-bug and create others. Children are given a range of digital platforms to try and display their learnt skills and understanding across the various strands within the Computing curriculum.
I	**Implicit and Incidental**	Implicit knowledge often is unconsciously obtained, and we may not recall learning it or be able to verbalise it: such as how to walk or talk. It builds on past experiences. Incidental knowledge is similar in that we acquire it from experiences, but these are unplanned or unintended.	The Computing curriculum is progressive. Over time, children will become more familiar with a range of devices, each time being able to build upon prior experience. For example, the use of a keyboard will not change drastically, and learners can continue to use knowledge acquired to move forward in Computing – this will quickly become implicit as it is something that will be a constant. Being able to do this will assist pupils when they type on smaller devices or on keyboards that are touch screen or have different layout. The same applies to coding a Beebot, this knowledge is transferable to more complex coding through changing algorithms and block codes.
R	**Relationships and Real Life**	This is knowledge that supports children build relationships and understand how social interactions work; the knowledge behind 'real life' skills such as empathy, friendship, honesty. For some it comes naturally, most children need a level of coaching to acquire it.	Real-life links are made throughout the Computing curriculum through demonstrating the 'how'. Children are provided with opportunities to develop an understanding of how technology (both in the home and workplace), the Internet and a range of computing systems work, giving the subject purpose and providing an insight into the world of work. Relationships and how we behave, is fundamental to the teaching of Online Safety. Here, children will be shown the importance of demonstrating kindness, tolerance, self-respect and friendship on digital platforms such as social media. They will know and understand the consequences and impact of negative behaviour as well as how to stay safe. Opportunities will be provided for children to communicate their opinions and ideas in a supportive environment. Communication through computing can be a powerful tool, be it video, audio or via student email.
E	**Experiences and Experts**	This is knowledge built up from a range of experiences both undertaken or encountered personally such as a visit to a place of historical interest, it may be explicitly taught, or delivered by an "expert" such as a professor, or sensorily observed such as an experiment.	As part of Computing curriculum, it is important that we provide our pupils with opportunities to see how the knowledge and skills they are learning can be applied to real-life. If children are given these 'tangible' experiences, it will enhance their passion for the subject. Talking to experts is one such way that they can do this.

In Computing we teach to secure that knowledge in the following ways:

	How the children will acquire their knowledge:	What that practice looks like in Computing
A	*Active* construction of knowledge, the acquisition of *vocabulary,* teacher *articulation* of learning processes and the *asking and answering* of questions.	When planning Computing lessons, our teachers support pupils in the acquisition of knowledge, through the use of key concepts, terms and vocabulary, providing opportunities to build a shared and consistent understanding. Glossaries, concept maps, and displays, along with regular recall and revision, support this approach. Questioning is also a fundamental part of the learning process with children demonstrating their understanding through asking and answering a range of questions.
S	*Staged* development enables children to join up intertwined groups of meaningful knowledge into *schemas*. This comes *semantically*, through the *senses,* through *skills* and *socially.*	During the journey a child goes on in our curriculum, they will make connections. Vocabulary and meaning (semantics) will coincide with events the learners have experienced. In computing, children may have their own experiences of coding as part of games they play digitally and online – they will be fully aware that games can have errors that need fixing. The schema of programming or coding something, in addition to their own personal understanding, will help them place the knowledge they are learning. They will use physical computing and activities that offer tactile and sensory experiences to enhance their learning. Combining electronics and programming with arts and crafts (especially through exploratory projects) providing them with a creative, engaging context to explore and apply computing concepts.
P	*Practically:* children access a wide range of memorable learning through *play*, the *power of stories pictures and print* and through *problem-solving* activities.	Computing is an ideal subject for developing problem-solving skills. The computer science sections of our curriculum expose children to errors, misconceptions, de-bugging and opportunities to edit and make improvements. Many aspects of computing will involve trial and error, ensuring learners are given the chance to explore and examine solutions. This gives teachers a perfect opportunity to encourage resilience in our children, one of the school's key values. Our cross-curricular projects will enable children to access memorable learning through practical video, audio, digital imagery and animation. They will also carry out project-based learning activities which provide them with the opportunity to apply and consolidate their knowledge and understanding.
I	*Internalisation* of learning through *interaction, instruction, imitation* and *integration* aids the movement of thoughts to long term memory.	Teachers will model processes or practices in Computing — everything from debugging code to binary number conversions — using techniques such as worked examples and live coding. Modelling is particularly beneficial to novices, providing scaffolding that can be gradually taken away, with children eventually becoming more confident at finding their own solutions. The curriculum encourages collaboration, specifically using paired programming and peer instruction, and also structured group tasks. Working together stimulates classroom dialogue, articulation of concepts, and development of shared understanding. Computing lessons provide activities with different levels of direction, scaffolding, and support that promote learning, ranging from highly structured to more exploratory tasks. Adapting your instruction to suit different objectives helps keep all pupils engaged and encourage greater independence. Such hands-on approaches and interaction aid the transfer of information into the long-term memory.
R	*Retrieval* of knowledge *through repetition, revision, recycling* and *routine* prompts memory "muscle" to work, making knowledge "stick".	Computing is taught across the year so it is important that knowledge is revisited, through retrieval tasks and through other curriculum subjects such as data handling in Maths or Science, animation in English, digital images in Humanities and Art. Learning is also sequential and progressive, so each year the children build on their knowledge and skills from the previous years – these links to prior learning are made explicit to the children. Lessons across each sequence follow a given format, teaching concepts by first unpacking complex terms and ideas, exploring these ideas in unplugged and familiar contexts, then repacking this new understanding into the original concept. This approach can help pupils develop a secure understanding of complex concepts. Modelling, collaboration and project work are also a part of this learning process. Such familiarity with this way of learning means that children can concentrate on developing their understanding of the concept, rather than focussing on the routines, which become more automatic.
E	Special *experiences* linked to learning objectives and opportunities in specific *environments* can enhance the probability of long-term memory retaining key messages.	We aim to present children with exciting and innovative experiences through our Computing curriculum, making it purposeful and as relevant to real-life as possible, thus aiding the transfer to the long-term memory. Children have access to a wide-range of technology and programmes to support their knowledge and skill development both within the subject and across the curriculum. They are provided with the opportunity to apply their learning in different ways.

Teaching Computing

As an academy we follow the Teach Computing curriculum, it is structured across the same four primary themes in every year group: Computing Systems and Networks, Programming, Data and Information and Creating Media. These are divided into ten units (or strands) as shown below. In it is line with the aims of the National Curriculum.

The Teach Computing Curriculum uses the National Centre for Computing Education's computing taxonomy to ensure comprehensive coverage of the subject. All learning outcomes can be described through a high-level taxonomy of ten strands, ordered alphabetically as follows:

- **Algorithms** — Be able to comprehend, design, create, and evaluate algorithms
- **Computer networks** — Understand how networks can be used to retrieve and share information, and how they come with associated risks
- **Computer systems** — Understand what a computer is, and how its constituent parts function together as a whole
- **Creating media** — Select and create a range of media including text, images, sounds, and video
- **Data and information** — Understand how data is stored, organised, and used to represent real-world artefacts and scenarios
- **Design and development** — Understand the activities involved in planning, creating, and evaluating computing artefacts
- **Effective use of tools** — Use software tools to support computing work
- **Impact of technology** — Understand how individuals, systems, and society as a whole interact with computer systems
- **Programming** — Create software to allow computers to solve problems
- **Safety and security** — Understand risks when using technology, and how to protect individuals and systems.

The taxonomy provides categories and an organised view of content to encapsulate the discipline of computing. Whilst all strands are present at all phases, they are not always taught explicitly

For these to be coherent, the lessons within a unit are taught in order. However, across a year group, the units themselves do not need to be taught in order, with the exception of 'Programming' units, where concepts and skills rely on prior learning and experiences.

The units for key stages 1 and 2 are based on a spiral curriculum. This means that each of the themes is revisited regularly (at least once in each year group), and pupils revisit each theme through a new unit that consolidates and builds on prior learning within that theme. This style of curriculum design reduces the amount of knowledge lost through forgetting, as topics are revisited yearly. It also ensures that connections are made even if different teachers are teaching the units within a theme in consecutive years.

In KS1 these are:

	Computing systems and networks	Creating media	Programming A	Data and Information	Creating Media	Programming B
Year 1	**Technology around us** Recognising technology in school and using it responsibly	**Digital painting** Choosing appropriate tools in a program to create art, and making comparisons with working non-digitally	**Moving a robot** Writing short algorithms and programs for floor robots, and predicting program outcomes.	**Grouping data** Exploring object labels, then using them to sort and group objects by properties.	**Digital writing** Using a computer to create and format text, before comparing to writing non-digitally	**Programming animations** Designing and programming the movement of a character on screen to tell stories.
	Effective use of tools					
	Impact of technology					
	Safety and security					

	Computing systems and networks	Creating media	Programming A	Data and Information	Creating Media	Programming B
Year 2	**Information technology around us** Identifying IT and how its responsible use improves our world in school and beyond.	**Digital photography** Capturing and changing digital photographs for different purposes.	**Robot algorithms** Creating and debugging programs, and using logical reasoning to make predictions.	**Pictograms** Collecting data in tally charts and using attributes to organise and present data on a computer.	**Digital music** Using a computer as a tool to explore rhythms and melodies, before creating a musical composition.	Programming quizzes Designing algorithms and programs that use events to trigger sequences of code to make an interactive quiz.
	Effective use of tools					
	Impact of technology					
	Safety and security					

In Ks2 these are:

	Computing systems and networks	Creating media	Programming A	Data and Information	Creating Media	Programming B
Year 3	**Connecting computers** Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks.	**Stop-frame animation** Capturing and editing digital still images to produce a stop-frame animation that tells a story.	**Sequencing sounds** Creating sequences in a block-based programming language to make music	**Branching databases** Building and using branching databases to group objects using yes/no questions.	**Desktop publishing** Creating documents by modifying text, images, and page layouts for a specified purpose.	**Events and actions in programs** Writing algorithms and programs that use a range of events to trigger sequences of actions.
	Effective use of tools					
	Impact of technology					
	Safety and security					

	Computing systems and networks	Creating media	Programming A	Data and Information	Creating Media	Programming B
Year 4	**The internet** Recognising the internet as a network of networks including the WWW, and why we should evaluate online content.	**Audio production** Capturing and editing audio to produce a podcast, ensuring that copyright is considered.	**Repetition in shapes** Using a text-based programming language to explore count-controlled loops when drawing shapes.	**Data logging** Recognising how and why data is collected over time, before using data loggers to carry out an investigation.	**Photo editing** Manipulating digital images, and reflecting on the impact of changes and whether the required purpose is fulfilled.	**Repetition in games** Using a block-based programming language to explore count-controlled and infinite loops when creating a game
	Effective use of tools					
	Impact of technology					
	Safety and security					

	Computing systems and networks	Creating media	Programming A	Data and Information	Creating Media	Programming B
Year 5	**Systems and searching** Recognising IT systems in the world and how some can enable searching on the internet.	**Video production** Planning, capturing, and editing video to produce a short film.	**Selection in physical computing** Exploring conditions and selection using a programmable microcontroller.	**Flat-file databases** Using a database to order data and create charts to answer questions.	**Introduction to vector graphics** Creating images in a drawing program by using layers and groups of objects.	**Selection in quizzes** Exploring selection in programming to design and code an interactive quiz.
	Effective use of tools					
	Impact of technology					
	Safety and security					

	Computing systems and networks	Creating media	Programming A	Data and Information	Creating Media	Programming B
Year 6	**Communication and collaboration** Exploring how data is transferred by working collaboratively online	**Webpage creation** Designing and creating webpages, giving consideration to copyright, aesthetics, and navigation.	**Variables in games** Exploring variables when designing and coding a game.	**Introduction to spreadsheets** Answering questions by using spreadsheets to organise and calculate data.	**3D modelling** Planning, developing, and evaluating 3D computer models of physical objects.	**Sensing movement** Designing and coding a project that captures inputs from a physical device
	Effective use of tools					
	Impact of technology					
	Safety and security					

Online Safety
Overviews for each unit show the links between the content of the lessons and the national curriculum and Education for a Connected World framework. These references have been provided to show where aspects relating to online safety, or digital citizenship, are covered within the Teach Computing Curriculum. Not all of the objectives in the Education for a Connected World framework are covered in the Teach Computing Curriculum, so it is taught through our PSHE (Jigsaw) curriculum and through bespoke online safety weeks in school.

Evidence of Key Knowledge

Each child from Y1-Y6 has both an electronic and paper folder to store their work. This moves up with the children as they move through the school. They use each form of evidence to:

- Regularly add to their knowledge harvest – to show the growth of the skills and knowledge.
- Store 'unplugged' paper activities that are completely away from technology.
- Take photos or videos of the work that has been produced, this will be annotated against the learning question.

Collecting such evidence, provides the children with the opportunity to refer back to their learning during each block and to see how they are progressing between blocks and through the year groups.

Impact

Assessment is used to monitor progress and to identify any child needing additional support as soon as they need it.

Formative assessment

Every lesson includes formative assessment opportunities. These opportunities are identified on planning to ensure that misconceptions are recognised and addressed if they occur. They vary from teacher observation or questioning, to marked activities. These assessments are vital to ensure that teachers are adapting their teaching to suit the needs of the pupils that they are working with, and you are encouraged to change parts of the lesson, such as how much time you spend on a specific activity, in response to these assessments. The learning objective and success criteria are introduced in the slides at the beginning of every lesson. At the end of every lesson, pupils are invited to assess how well they feel they have met the learning objective using thumbs up, thumbs sideways, or thumbs down. This gives pupils a reminder of the content that has been covered, as well as a chance to reflect. It is also a chance for teachers to see how confident the class is feeling so that they can make changes to subsequent lessons accordingly.

Summative assessment

Every unit includes an optional summative assessment framework in the form of either a multiple choice quiz (MCQ) or a rubric. All units are designed to cover both skills and concepts from across the computing national curriculum. Units that focus more on conceptual development include an MCQ. Units that focus more on skills development end with a project and include a rubric. However, within the 'Programming' units, the assessment framework (MCQ or rubric) has been selected on a best-fit basis

This policy was most recently updated in:

October 2022