Dream Reach Achieve

Whittington Green School – Computing

Dream Reach Achieve

CURRICULUM INTENT: A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with mathematics, science, and design and technology, and provides insights into both natural and artificial systems. The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of content. Computing also ensures that pupils become digitally literate – able to use, and express themselves and develop their ideas through, information and communication technology – at a level suitable for the future workplace and as active participants in a digital world.

Aims

The national curriculum for computing aims to ensure that all pupils:

- can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- are responsible, competent, confident and creative users of information and communication technology.

Attainment targets

By the end of each key stage, pupils are expected to know, apply and understand the matters, skills and processes specified in the relevant programme of study.
Schools are not required by law to teach the example content in [square brackets].

Subject content Key stage 3

Pupils should be taught to:

- design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems
- understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem
- use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
- understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]

- understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems
- understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits
- undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
- understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.

Cultural capital	- Online bullying - Online relationships - Privacy and security - Internet Safety Day - Hour of Code
Adaptations for SEND	A broad and relevant curriculum will be taught to all students in Computing, this will be appropriately scaffolded by LSA's and teachers to ensure every student can access the curriculum.

Year 7

Pupils should be taught to:

- design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems Y
- understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem
- use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
- understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]
- understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems
- understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits
- undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
- understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.

Term/Unit	Topic Overview / Content	Learning Expectation / Concepts and Skills	Recall and review of past learning	National Curriculum target	Key Vocab/Concepts
Term 1	**Clear Messages in Digital Media** This unit is designed to build upon learners' experience in key stage 2. It requires learners to use a range of different skills across several	• Choose search terms relating to a particular issue – online safety • Use tools to copy an image into another application • Identify key features of a good poster focusing on online safety • Plan a poster to clearly convey a message • Choose and download a suitable image following the theme • Create a poster using a desktop publishing application	KS2 Y3 Creating Media – Desktop Publishing KS2 Y4 Computing Systems & Networks – The Internet KS2 Y6 Computing Systems & Networks –	• undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users	• Search term • Screenshot • Annotate • Landscape • Portrait • Download • Heading • Subheading • Body • Text • Brand • Logo • Slide • Comment

	pieces of software. Learners will work between different applications to create a poster and slides on a given theme. The central theme focuses on embedding online safety and secure ways of working.	• Modify a logo using a graphic editing program • Choose how to combine text and graphics in a slide • Use digital tools to provide feedback on design choices • Plan a consistent layout for a set of slides • Modify a logo so that it fits in with the planned slide styles • Create a styled set of slides based on a plan • Search for suitable text for slides • Search for and add a suitable image • Evaluate content against a rubric • Plan how to deliver a presentation • Explain work to others through a presentation	Communication and Collaboration	• create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability • understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.	• Content • Licence • Present • Design choices
Term 2/3	**Networks – From Semaphores to the Internet** This unit begins by defining a network and addressing the benefits of networking, before covering how data is transmitted across networks using protocols. The types of hardware required are	• Define what a computer network is and explain how data is transmitted between computers across networks • Define 'protocol' and provide examples of non-networking protocols • List examples of the hardware necessary for connecting devices to networks • Compare wired to wireless connections and list examples of specific technologies currently used to implement such connections • Define 'bandwidth', using the appropriate units for measuring the rate at which data is transmitted,	KS2 Y3 Computing Systems & Networks – Connecting computers KS2 Y4 Computing Systems & Networks – The Internet KS2 Y6 Computing Systems & Networks – Systems & Searching	• understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems	• Network • Protocol • Mainframe • Personal computer • Stand-alone • HTTP • Network cable • Hub • Server • Router • ISP • Wired • Wireless • 3G, 4G, 5G • WiFi • Bandwidth • Bit

	explained, as is wired and wireless data transmission. Learners will develop an understanding of the term's 'internet' and 'World Wide Web', and of the key services and protocols used	and discuss familiar examples where bandwidth is important • Define what the internet is • Explain how data travels between computers across the internet • Describe key words such as 'protocols', 'packets', and 'addressing' • Explain the difference between the internet, its services, and the World Wide Web • Describe how services are provided over the internet • List some of these services and the context in which they are used • Explain the term 'connectivity' as the capacity for connected devices ('Internet of Things') to collect and share information about me with or without my knowledge (including microphones, cameras, and geolocation) • Describe how internet-connected devices can affect me • Describe components (servers, browsers, pages, HTTP and HTTPS protocols, etc.) and how they work together			• Megabit • Gigabit • Broadband • Buffering • Internet • Packet • IP address • Packet header • Packet payload • Transmission Control • Internet Protocol • World Wide Web • WWW • Internet services • Email • Voice over Internet Protocol (VoIP) • Internet of Things (IoT) • Spam • Privacy • Security • Web browser • Web server • Web page • Search engine • HTTPS • URL • Domain name • Domain name system
Term 3/4	**Using Media – Gaining Support for a Cause** During this unit, learners	• Select the most appropriate software to use to complete a task • Identify the key features of a word processor • Apply the key features of a word processor to format a document	KS2 Y3 Creating Media – Desktop Publishing	• undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a	• Application software • Word processor • Formatting • Fonts • Icons

	develop their understanding of information technology and digital literacy skills. They will use the skills learnt across the unit to create a blog post about a real-world cause that they would like to gain support for. Learners will develop software formatting skills and explore concerns surrounding the use of other people's work, including licensing and legal issues.	• Evaluate formatting techniques to understand why we format documents • Select appropriate images for a given context • Apply appropriate formatting techniques • Demonstrate an understanding of licensing issues involving online content by applying appropriate Creative Commons licences • Demonstrate the ability to credit the original source of an image • Critique digital content for credibility • Apply techniques to identify whether or not a source is credible • Apply referencing techniques and recognise the concept of plagiarism • Evaluate online sources for use in own work • Construct a blog using appropriate software • Create content for a blog based on credible sources • Apply referencing techniques that credit authors appropriately • Design the layout of the content to make it suitable for the audience • Construct a blog using appropriate software • Create content for a blog based on credible sources • Apply referencing techniques that credit authors appropriately • Design the layout of the content to make it suitable for the audience	KS2 Y4 Creating Media – Audio Production KS2 Y4 Creating Media – Image Editing KS2 Y6 Creating Media – Webpage Creation	range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users • create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	• Appropriate • Copyright • Licensing • Creative Commons • Text wrapping • Cropping • Recolouring • Credibility • Source • Audience • Plagiarism • Referencing • Citation • Paraphrase • Blog
Term 5	**Scratch Programming**	• Compare how humans and computers understand instructions (understand and carry out)	KS2 Y3 Programming A –	• understand several key algorithms that reflect computational	• Sequencing • Subroutines • Instructions

	The aim of this unit is to build learners' confidence and knowledge of the key programming constructs. The main programming concepts covered in this unit are sequencing, variables, selection, and count-controlled iteration.	• Define a sequence as instructions performed in order, with each executed in turn • Predict the outcome of a simple sequence • Modify a sequence • Define a variable as a name that refers to data being stored by the computer • Recognise that computers follow the control flow of input/process/output • Predict the outcome of a simple sequence that includes variables • Trace the values of variables within a sequence • Make a sequence that includes a variable • Define a condition as an expression that will be evaluated as either true or false • Identify that selection uses conditions to control the flow of a sequence • Identify where selection statements can be used in a program • Modify a program to include selection • Create conditions that use comparison operators (>,<,=) • Create conditions that use logic operators (and/or/not) • Identify where selection statements can be used in a program that include comparison and logical operators • Define iteration as a group of instructions that are repeatedly executed • Describe the need for iteration	Sequencing Sounds KS2 Y3 Programming B – Events and Actions in Programs KS2 Y4 Programming A – Repetition in Shapes KS2 Y4 Programming B – Repetition in Games KS2 Y5 Programming A – Selection in Physical Computing KS2 Y5 Programming B – Selection in Quizzes KS2 Y6 Programming A – Variables in Games KS2 Y6 Programming B – Sensing Movement	thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem • use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions • understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming • create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	• Execute • Variables • Commands • Input • Process • Output • Storage • Tracing • Expressions • Evaluate • Conditions • Selection • If statements • Operators • Logic • Comparison • Iteration • Count-controlled • Condition-controlled • Debugging

		• Identify where count-controlled iteration can be used in a program • Implement count-controlled iteration in a program • Detect and correct errors in a program (debugging) • Independently design and apply programming constructs to solve a problem (subroutine, selection, count-controlled iteration, operators, and variables)			
Term 6	**Vector Graphics** This unit offers learners the opportunity to design graphics using vector graphic editing software. By the end of the unit learners will have produced an illustration, a logo, or some icons using vector graphics.	• Use tools to draw and modify shapes • Change the position and rotation shapes • Explain how z-order determines what is visible • Use tools to align and distribute objects to create uniformity • Explain how grouping can be used to work with several objects at once • Combine two shapes using union, intersection, and difference • Explain that vector graphics are made up of paths • Create and modify straight and curved paths • Change shapes to paths and edit them • Choose a project and plan a design • Combine tools and techniques to create a vector image • Evaluate the project against its given purpose • Explain how markup defines what a vector graphic looks like • Change an object by modifying its markup • Plan improvements and implement them to develop a project	KS2 Y5 Creating Media – Introduction to Vector Graphics	• undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users • create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	• Vector • Fill • Stroke • Rotate • Reposition • Z-order • Layer • Handle • Object • Select • Group • Ungroup • Align • Distribute • Combine • Union • Difference • Intersection • Path • Node • Freehand • Object • Logo • Illustration • Icon • Markup • Scalable

		• Explain key differences between vector and bitmap images • Outline which image type best suits which uses • Evaluate their image against a rubric •			• SVG (scalable vector graphic) • Bitmap