

Baydon St Nicholas CE VA Primary School Curriculum Overview Areas of Study 2026-2027

Robins 2026-2027 (YEAR 1)

COMPUTING Robins YEAR 1 UNIITS	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Area of Learning	Computing Systems and Networks Improving mouse skills	Word processing Exploring word-processing software by using keyboard shortcuts, importing images and applying simple editing tools such as bold, italics, underlining and font colour.	Programming 1: Commands unplugged	Skills Showcase: Rocket to the moon	Programming 1: Algorithms and debugging	Online Safety
Unit Outcomes	- Use computers more purposefully - Log in and navigate around a computer - Drag, drop, click and control a cursor using a mouse - Use software tools to create art on the computer	- Identify the main keys on a keyboard and use them for typing. - Use the ‘spacebar’ correctly when typing short words or sentences. - Type capital letters using the ‘Shift’ key. - Explain that the keyboard is an input device used to enter information. - Type sentences using a word processor and make simple text changes. - Select text and use on-screen buttons to alter its appearance. - Create documents with images and text, using keyboard shortcuts. - Use copy and paste, different text styles and credit source materials.	- Describe what a command is and what it does. - Recall the key rules a command must follow. - Give examples of commands from their life. - Follow commands and instructions given by others - Describe that instructions are made of commands that must be in the correct order. - Create correct commands for others to follow. - Order commands to create clear instructions. - Recognise and fix errors in commands so instructions work correctly. - Check and share instructions so others can follow them.	- Use a computer to make a list. - Explain the benefits of making a list on the computer. - Use a basic range of tools on art software to design a rocket. - Sequence instructions. - Follow instructions to build their model rocket. - Evaluate and edit their rocket design.	- Recall that a monarch is a king or queen. - Identify some of the monarch’s roles. - Explain that a king or queen is crowned in a special ceremony called a coronation. - Name some of the main steps in the coronation ceremony. - Use sources to explain how William the Conqueror became King of England. - Explain how William the Conqueror kept order and conquered England. - Explain how castles have changed over time. - Identify that the power of monarchs has changed over time. - Make comparisons between past and present monarchies.	- Discuss what the internet is and how it can be used. - Recognise that the internet may affect mood or emotions. - Recognise how internet use can affect and upset others. - Identify which information is appropriate to share and post online and which is not.

COMPUTING Robins YEAR 1 UNIITS	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 1 Key Skills	• Learning how to explore and tinker with hardware to find out how it works. • Learning where keys are located on the keyboard. • Using a basic range of tools within graphic editing software. • Developing control of the mouse through dragging, clicking and resizing images to create different effects. • Developing an understanding of different software tools. • Recognising devices that are connected to the internet. • Logging in and out and saving work on their own account.	• Developing confidence with the keyboard and the basics of touch typing. • Developing word processing skills, including altering text, copying and pasting and using keyboard shortcuts. • Using word processing software to type and reformat text. • Searching for appropriate images to use in a document. • Understanding what online information is. • Identifying whether information is safe or unsafe to be shared online. •	• Explaining what they are trying to achieve with their instructions. • Writing clear, sequenced instructions for familiar tasks. • Using terms like 'start,' 'end' and 'next' to describe the steps in instructions. • Changing their instructions or algorithms into code that the robot understands.	Learning where keys are located on the keyboard. Learning how to operate a camera to take photos and videos. Using logical reasoning to predict the behaviour of simple programs. Developing the skills associated with sequencing in unplugged activities. Following a basic set of instructions. Assembling instructions into a simple algorithm. Learning to debug instructions when things go wrong. Learning to debug an algorithm in an unplugged scenario. Using a basic range of tools within graphic editing software. Taking and editing photographs. Developing control of the mouse through dragging, clicking and resizing of images to create different effects. Developing an understanding of different software tools. Recognising devices that are connected to the internet. Understanding that technology can be used to represent data in different ways, such as pictograms, tables, pie charts, bar charts, and block graphs. Logging in and out and saving work on their own account.	• Sequencing up to six photographs, focusing on the intervals between events. • Knowing where people/events studied fit into a chronological framework. • Recognising some things which have changed/stayed the same as the past. • Identifying simple reasons for changes. • Identifying similarities and difference between ways of life at different times • Finding out about people, events and beliefs in society. • Making comparisons with their own lives. • Using artefacts, photographs and visits to museums to ask and answer questions about the past. • Making simple observations about a source or artefact. • Using sources to show an understanding of historical concepts (see above). • Recognising different ways in which the past is represented (including eye-witness accounts). • Comparing pictures or photographs of people or events in the past. • Asking a range of questions about stories, events and people. • Understanding the importance of historically-valid questions. • Understanding how we use books and sources to find out about the past. Using a source to answer questions about the past. • Evaluating the usefulness of sources to a historical enquiry. • Selecting information from a source to answer a question. • Making links and connections across a unit of study. • Making simple conclusions about a question using evidence to support. • Communicating answers to questions in a variety of ways, including discussion, drama and writing (labelling, simple recount). • Using relevant vocabulary in answers. • Describing past events and people by drawing or writing. • Expressing a personal response to a historical story or event through discussion, drawing our writing.	• Recognising devices that are connected to the internet. • Understanding that we are connected to others when using the internet. • Understanding some of the ways we can use the internet. • When using the internet to search for images, learning what to do if they come across something online that worries them or makes them feel uncomfortable. • Understanding how to interact safely with others online. • Recognising how actions on the internet can affect others. • Recognising what a digital footprint is and how to be careful about posting online. • Discussing ways to balance time spent online and offline.

COMPUTING Robins YEAR 1 UNIITS	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 1 Key Knowledge	- Log in and log out means to begin and end a connection with a computer - A computer and mouse can be used to click, drag, fill, select, add backgrounds, text, layers, shapes and clipart. - Passwords are important for security and to keep us safe.	- Touch typing is the fastest way to type. - I can make text a different style, size and colour. - Copy and paste is a quick way of duplicating text.	- There may be an error if a set of instructions does not give the expected result. - Errors could result from sequencing, unclear instructions or missing steps.	When we create something on a computer it can be more easily saved and shared than a paper version. Some of the simple graphic design features of a piece of online software. A spreadsheet is an electronic table for sorting data.	- To know that beyond living memory is more than 100 years ago. - To know that events in history may last different amounts of time. - To know that some events are more significant than others. - To know the impact of a historical event on society. - To know that ‘historically significant’ people are those who changed many people’s lives. - To know that historians use evidence from sources to find out more about the past. - To know that the past is represented in different ways. - To know that a monarch is a king or queen. - To begin to understand that power is exercised in different ways in different culture, times and groups e.g. monarchy. - To know that in the past monarchs had absolute power. - To be aware of the achievements of significant individuals.	- To know that the internet is many devices connected to one another. - To know what to do if you feel unsafe or worried online – tell a trusted adult. - To know that people you do not know on the internet (online) are strangers and are not always who they say they are. - To know that to stay safe online it is important to keep personal information safe. - To know that ‘sharing’ online means giving something specific to someone else via the internet and ‘posting’ online means placing information on the internet.
Year 1 Key Vocabulary	- account - click - clipart - computer - drag - drag and drop - layers - log off - log on - mouse - password - predict - resize - screen (monitor) - software - tool - username	- backspace - bold - copy - cut - delete - highlight - italics - keyboard shortcut - paste - underline - undo - word processor	- command - error - instructions - order	- computer - distance - equipment - file - list - materials - open - order - program - save - select - share - tool	- king - monarchy - power - queen - ruler	- app - appropriate - device - digital footprint - feelings - going online - in-person interactions - internet - kindness - offline activity - online activity - online experience - online interactions - online safety - personal information - pop-up - posting online - report - responsible digital citizen - screen time - sharing online - stranger - technology - trusted adult - unkind - website

Swifts 2026-2027 (YEAR 2 and 3)

Computing SWIFTS YEAR 2 & 3	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Area of Learning	Programming’: Scratch	Video trailers – Option 2: Using iPads	Microsoft Office 365: Creating media: Website design	Programming 1: Further programming with Scratch	Computational Thinking	Online Safety
Unit Outcomes	- Identify Scratch as a coding application and explore its different code blocks. - Make predictions about what code blocks will do and test these ideas. - Create a simple animation by combining motion, speech and wait blocks. - Plan how to remix an existing animation by choosing which parts to change. - Alter and remix code to create a new version of an animation. - Test and debug animations to fix problems and improve the code. - Reflect on the changes made, explaining what worked well and what could be improved.	- Describe the purpose of a trailer. - Create a storyboard for a book trailer. - Consider camera angles when taking photos or videos. - Import videos and photos into film editing software. - Add text to a video. - Incorporate transitions between images. - Evaluate their own and others’ trailers.	- Create a Sway with a title, image and a completed first header section. - Create a clear plan for their web page and beginning to create it. - Create a professional-looking web page with useful information and a clear style, which is easy for the user to read and find information from. - Create a clear plan by referring back to their checklist to include a range of features. - Create a web page with clear sections and with a range of features in.	- Identify how variables and if statements are used in Scratch games. - Explain what a variable is, tracking and how a condition changes what happens. - Create variables to keep and display scores. - Use sensing blocks and if statements to control game actions. - Combine variables, sensors and if/else blocks to build a multiplication game. - Debug code by finding and fixing errors. - Evaluate the game by explaining what worked well and what could be improved.	- Understand that problems can be solved more easily using computational thinking. - Recognise decomposition, abstraction, algorithm design and pattern recognition as key computational thinking skills. - Explain how decomposition and abstraction simplify problem-solving. - Identify patterns in problems and use them to solve problems. - Design clear algorithms and justify their choices. - Create logical sequences of steps to complete a task or project. - Use computational thinking skills to code, refine and evaluate their work.	- Differentiate between fact, opinion and belief online. - Explain how to deal with upsetting online content. - Recognise that digital devices communicate with each other to share personal information. - Explain what social media platforms are used for. - Recognise why social media platforms are age-restricted
Year 2 Key Skills	Understanding programming - Recognising that robots are programmed by humans. - Recognising examples of computers being programmed by humans to perform simple tasks (computer games, animations, computer-generated music) Plan - Explaining what they are trying to achieve with their instructions (algorithms). - Writing clear, sequenced instructions (algorithms). Create - Changing their instructions or algorithms into code that the robot understands.	Plan - Making decisions about the digital content they make. - Ordering ideas so that content makes sense. - Identifying the audience and purpose of digital content. - Planning how to use the space on a digital page. - Importing images onto a digital page.	Create - Creating digital content - Using a basic drawing or writing tools to create a simple piece of digital content (e.g., drawing a picture, typing words). - Selecting and using simple editing tools (e.g., formatting, resizing). - Typing simple words and short phrases. - Importing images onto a digital page. - Positioning images and text on a digital page. - Typing familiar words and short sentences. - Cutting, copying and pasting. Create - Design principles - Choosing and arranging simple elements (e.g., text, images, colours) to make digital content clear and appealing. - Using basic tools to change the appearance of digital content (e.g., selecting different brush styles, filling shapes with colour, formatting text). - Arranging text and images to create an organised, clear layout. Evaluate - Explaining their decisions about the digital content they created and the tools they used to make it.	Understanding programming - Recognising that robots are programmed by humans. - Recognising examples of computers being programmed by humans to perform simple tasks (computer games, animations, computer-generated music). Plan - Explaining what they are trying to achieve with their instructions (algorithms). - Writing clear, sequenced instructions (algorithms) for familiar tasks. - Using terms like ‘start,’ ‘end’ and ‘next’ to describe the steps in instructions (algorithms). - Explaining what they are trying to achieve with their program. - Verbally planning what they aim to achieve with a program. Create - Changing their instructions or algorithms into code that the robot understands. - Using basic block code to implement an algorithm. - Looking at different forms of code (Bee-bot and Scratch) and predicting how a program will execute. - Tinkering with a basic existing visual code to see how it affects a program. Evaluate - Beginning to identify errors in instructions (algorithms). - Making suggestions for how to fix errors in instructions (algorithms). - Identifying errors in code within a simple program. - Correcting errors or bugs in simple programs they have created.	Understanding programming - Recognising that robots are programmed by humans. - Recognising examples of computers being programmed by humans to perform simple tasks (computer games, animations, computer-generated music). Plan - Explaining what they are trying to achieve with their instructions (algorithms). - Writing clear, sequenced instructions (algorithms) for familiar tasks. - Using terms like ‘start,’ ‘end’ and ‘next’ to describe the steps in instructions (algorithms). - Explaining what they are trying to achieve with their program. - Verbally planning what they aim to achieve with a program. Create - Changing their instructions or algorithms into code that the robot understands. - Using basic block code to implement an algorithm. - Looking at different forms of code (Bee-bot and Scratch) and predicting how a program will execute. - Tinkering with a basic existing visual code to see how it affects a program. Evaluate - Beginning to identify errors in instructions (algorithms). - Making suggestions for how to fix errors in instructions (algorithms). - Identifying errors in code within a simple program. - Correcting errors or bugs in simple programs they have created.	Being online Recognising when they are online. Identifying programs (e.g. apps and websites) they use online. Recognising indicators of being logged into an account (e.g. avatar, username, initials). Online relationships Recognising that being kind, polite and safe is just as important online as in-person. Risks and harms Describing how someone might change their account if they have their password. Responsible and safe use Checking with an adult before they use new apps or websites. Beginning to remember their username and password for an online account. Treating devices carefully. Reporting online Recognising when something online scares or worries them. Describing how to respond safely to an online message.

Computing SWIFTS YEAR 2 & 3	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 3 Key Skills	- Working towards a given goal that a program needs to accomplish. - Breaking down what they want to achieve into smaller, manageable parts. - Tinkering with an existing text-based code to see how it affects a program. - Remixing code to alter and add to an existing program. - Recognising the relationship between what is happening in a program and the written (block) code. - Working backwards, beginning to identify the code they think a program uses. - Running small chunks of code at a time to find the error or bug.	Using logical thinking to explore more complex software; predicting, testing and explaining what it does. Taking photographs and recording video to tell a story. Using software to edit and enhance their video adding music and text on screen with transitions.	- Building a web page and creating content for it. - Designing and creating a webpage for a given purpose. - Using software to work collaboratively with others.	Working towards a given goal that a program needs to accomplish. Breaking down what they want to achieve into smaller, manageable parts. Tinkering with an existing text-based code to see how it affects a program. Remixing code to alter and add to an existing program. Creating loops to make code more efficient in block-based programs. Beginning to use variables in block-based programming languages to make programs more interactive. Including a conditional statement in block-based programming languages. Recognising the relationship between what is happening in a program and the written (block) code. Working backwards, beginning to identify the code they think a program uses. Running small chunks of code at a time to find the error or 'bug.'	Breaking down what they want to achieve into smaller, manageable parts. Using logic, pattern recognition and decomposition to solve simple problems. Remixing code to alter and add to an existing program. Recognising repeating patterns in a program or code. Creating loops to make code more efficient in block-based programs. Beginning to use variables in block-based programming languages to make programs more interactive. Including a conditional statement in block-based programming languages. Recognising the relationship between what is happening in a program and the written (block) code.	- Recognising how social media platforms are used to interact. - Recognising that different information is shared online, including facts, beliefs and opinions. - Learning how to identify reliable information when searching online. - Learning how to stay safe on social media. - Considering the impact technology can have on mood.
Year 2 Key Knowledge	To know that humans need to give robots instructions to follow and that they will carry out these instructions exactly, even if they are wrong. To know that humans need to give instructions in the correct language for the robot to understand To know that there may be an error if a set of instructions (an algorithm) does not give the expected result. To know that errors could result from sequencing, unclear instructions or missing steps. To know that th	Create - Design principles To know that colours, shapes and space can be used to make content clearer and more interesting. To know that size, position and space can be used to organise digital content. To know that different fonts and colours can make text easier to read. To know that images and text work together to communicate.	To know that ordering ideas is important to make content clear. To know that digital tools can be used to make content that is presented clearly. To know that digital content should be organised to suit its purpose and audience.	To know that instructions (algorithms) must give every step of a task. To know that an algorithm must give clear, sequenced instructions. To know that an algorithm is a set of instructions used to carry out a task. To know that an algorithm becomes a program when it is coded. To know that programs execute the exact instructions they are given, even if they are incorrect. To know that a program is a series of instructions (algorithms) that are written for a computer to follow. To know that a person can program a device by giving it an algorithm/algorithms to follow.	To know that there may be an error if a set of instructions (an algorithm) does not give the expected result. To know that errors could result from sequencing, unclear instructions or missing steps. To know that there must be an error if a program does not execute as expected. To know that an error in a computer program is known as a 'bug' and fixing errors is known as 'debugging'.	Being online "To know that 'online' means being connected to other devices or the internet. ("Statement also found in Computing systems.) To know that an 'account' is a personal space within an online program. Online relationships To know they should not speak to or message unfamiliar people online. To know that they can still hurt someone's feelings online. Risks and harms To know that passwords help keep online accounts and programs safe. To know that logging in to their account helps keeps them safe online. Responsible and safe use To know that using someone else's login is not safe or respectful. Reporting online To know who to report concerns to about online content or if they receive messages from someone they don't know. To know they should tell an adult if something in their account has changed.
Year 3 Key Knowledge	Decomposition is the process of breaking down a task or problem into smaller parts. Breaking down a problem into smaller parts makes it easier to solve.	- To know that different types of camera shots can make my photos or videos look more effective. - To know that I can edit photos and videos using film editing software. - To understand that I can add transitions and text to my video.	- To know that a website is a collection of pages that are all connected. - To know that websites usually have a homepage and subpages as well as clickable links to new pages, called hyperlinks. - To know that websites should be informative and interactive.	Breaking down a problem into smaller parts makes it easier to solve the problem. Loops are used to save time when writing code by reducing repetition. A variable is a container or holder for storing information that can change, e.g. numbers or text. Conditional statements tell the computer what to do next based on a user's input. It is important to identify where the mistake is in the programming as part of the debugging process. Errors in a program could result from sequencing errors, coding errors or missing code.	Decomposition is the process of breaking down a task or problem into smaller parts. Breaking down a problem into smaller parts makes it easier to solve the problem. Abstraction is identifying the important detail and ignoring irrelevant information. Loops are used to save time when writing code by reducing repetition. A variable is a container or holder for storing information that can change, e.g. numbers or text. Conditional statements tell the computer what to do next based on a user's input. It is important to identify where the mistake is in the programming as part of the debugging process. Errors in a program could result from sequencing errors, coding errors or missing code.	- That not everything on the internet is true: people share facts, beliefs and opinions online. - The internet can affect people's moods and feelings. - Privacy settings limit who can access important personal information, such as names, ages, gender etc. - What social media is and that age restrictions apply.

Computing SWIFTS YEAR 2 & 3	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 2 Key Vocabulary	- Programming - program - command - error - instructions - algorithm - sequence - code - debug - evaluate - bug	Creating media edit text tool cursor keys digital content delete format cut copy paste image	Creating media edit text tool cursor keys digital content delete format cut copy paste image	Programming program command error instructions algorithm sequence code debug evaluate bug	Software open close toolbar program application (app) software icon menu settings	Online safety online personal information trusted adult pop up Skills drag and drop scroll click/tap select double click log in/out right click/press and hold left click/press
Year 3 Key Vocabulary	Loop	- application - camera angle - clip - cross dissolve - edit - fade to black - fade to white - film - film editing software - graphics - import - key events - music - photo - plan - recording - slide - sound effects - storyboard - time code - trailer - transition - video - voiceover - wipe	- assessment - audience - checklist - collaboration - content - contribution - create - design - embed - evaluate - features - Google Sites - hobby - homepage - hyperlinks - images - insert - online - plan - progress - published - record - review - style - subpage - tab - theme - web page - website - world wide web	- condition - if statement - variable	- abstraction - algorithm - algorithm design - computational thinking - debug - decomposition - effectiveness - evaluate - instructions - logical - pattern recognition - relevant - remixing - sequence - unplugged	- accurate - age restrictions - autocomplete - belief - charity - content - digital device - fact - fake news - hoax - internet - internet of things - opinion - online emotions - organisation - permission - privacy settings - reliable - search - search engine - share - smart devices - social media platforms

Firecrests 2026-2027 (YEAR 4 and 5)

Computing FIRECRESTS YEAR 4 & 5	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Area of Learning	Programming: Micro Bit	Computer Systems and Networks: Exploring AI	Mars Rover 1	Skills Showcase: Mars Rover 2	Bletchley Park and the history of computers	Skills showcase: Inventing a product Designing an electronic product through coding, debugging and computer-aided design to understand how digital technologies support product development

Computing FIRECRESTS YEAR 4 & 5	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Unit Outcomes	- Explore and test simple programs on the micro:bit. - Predict and describe how code will work before running it. - Plan and create animations using LED displays. - Use inputs, variables and conditions to build interactive programs. - Create programs that react to sensor data, such as pedometers and weather checkers. - Create a program that responds to real-time sensor readings. - Test, debug and evaluate programs by identifying and fixing problems.	- Explain what AI is and its basic functions. - Identify real-life applications of AI that are commonly used in everyday life. - Identify how AI understands and processes text and image prompts. - Generate and refine prompts to achieve the best possible response from AI. - Identify how AI generates code and how it can be useful in web design. - Identify how AI can be a useful starting point for a project. - Explain the key ethical considerations of AI. - Debate the potential of AI replacing human roles, presenting well-structured arguments.	- Identify some types of data the Mars Rover could collect (for example, photos). - Explain how the Mars Rover transmits the data back to Earth and the challenges involved. - Read any number in binary, up to eight bits. - Identify input, processing and output on the Mars Rovers. - Use a column method to add two 8-bit binary numbers. - Relate binary signals (Boolean) to a simple character-based language, ASCII.	Create a pixel picture, explaining that a pixel is the smallest element of a digital image and that binary is used to code and transfer this data. Save a JPEG as a bitmap and recognise the difference in file size as well as explaining how pixels are used to transfer image data. Explain the ‘fetch, decode, execute’ cycle in relation to real-world situations. Create a profile with a safe and suitable username and password and begin to use 3D design tools. Independently take tutorial lessons, applying what they have learnt to their design and understand the importance of using an online community responsibly.	- Explain that codes can be used for a number of different reasons and decode messages. - Explain how to ensure a password is secure and how this works. - Explain the importance of historical figures and their contribution towards computer science. - Present information about their historical figures in an interesting and engaging manner. - Develop an idea for a computer of the future and create a simple design. - Produce a simple audio advert with simple edits, which demonstrate an understanding of how to use the software.	- Evaluate code, understand what it does and adapt existing code for a specific purpose. - Debug programs and make them more efficient using sequence, selection, repetition or variables. - Design appropriate housing for their product using CAD software, including any input or output devices needed to make it work. - Create an appealing website for their product aimed at their target audience, which explains what their product is and what it does using persuasive language. - Create an edited video of their project, articulating the key benefits. - Describe and show how to search for information online and be aware of the accuracy of the results presented.
Year 4 Key Skills	- Making links between different programming interfaces that they are faced with. - Recognising examples of programming elements in real-life applications (e.g. live loops in music or variables in pedometers). - Planning a program to be downloaded to a physical system/device to solve a particular problem. - Decomposing a program independently when given a specific outcome or task to achieve. - Programming a physical system that has the capability for sensory input (e.g. a micro:bit). - Independently using loops to make code more efficient in text-based or block-based programs. - Using nested loops to make code more efficient. - Explaining what a program does and how it works, referring to the inputs and outputs. - Becoming more efficient and effective at debugging their programs. - Systematically identifying mistakes, problems or ‘bugs’ in a program.	Identify different types of AI and their applications in everyday life. Exploring text-based and image-based AI tools to understand how they generate content. Applying coding skills like decomposition and pattern recognition to interact with AI applications.	- Learning that a separate computer can program external devices. - Recognising how the size of RAM affects the processing of data. - Recognising that computers transfer data in binary and understanding simple binary addition. - Learning that messages can be sent by binary code, reading binary up to eight characters and carrying out binary calculations. - Understanding how data is collected in remote or dangerous places. - Understanding how data might be used to tell us about a location. - Learn about different forms of communication that have developed with the use of technology.	Learning the difference between ROM and RAM. Recognising how the size of RAM affects the processing of data. Understanding the fetch, decode, execute cycle. Learning how the data for digital images can be compressed. Recognising that computers transfer data in binary and understanding simple binary addition. Understanding how bit patterns represent images as pixels. Using logical thinking to explore software more independently, making predictions based on their previous experience. Independently learning how to use 3D design software package TinkerCAD. Learn about different forms of communication that have developed with the use of technology.	Learning about the history of computers and how they have evolved over time. Using past experiences to help solve new problems. Writing increasingly complex algorithms for a purpose. Debugging quickly and effectively to make a program more efficient. Remixing existing code to explore a problem. Changing a program to personalise it. Evaluating code to understand its purpose. Predicting code and adapting it to a chosen purpose.	Identifying what the internet can be used for. Recognising visual and text-based programming languages. Recognising further examples of computers being programmed by humans to perform simple tasks. Capturing digital images (e.g. photos and video). Adding layers to create digital content both photos and videos. Using the features of a word processor (e.g. tabs and hyperlinks) to structure digital content. Editing and refining images (e.g., cropping, adjusting brightness/contrast. Editing and refining video content (e.g., trimming, adjusting brightness/contrast). Using content blocks to insert text and images into a webpage.

Computing FIRECRESTS YEAR 4 & 5	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 5 Key Skills	• Making links between different programming interfaces that they are faced with. • Recognising examples of programming elements in real-life applications (e.g. live loops in music or variables in pedometers). • Planning a program to be downloaded to a physical system/device to solve a particular problem. • Decomposing a program independently when given a specific outcome or task to achieve. • Programming a physical system that has the capability for sensory input (e.g. a micro:bit). • Independently using loops to make code more efficient in text-based or block-based programs. • Using nested loops to make code more efficient. • Explaining what a program does and how it works, referring to the inputs and outputs. • Becoming more efficient and effective at debugging their programs. • Systematically identifying mistakes, problems or ‘bugs’ in a program.	Identify different types of AI and their applications in everyday life. Exploring text-based and image-based AI tools to understand how they generate content. Applying coding skills like decomposition and pattern recognition to interact with AI applications. Analysing the effectiveness of prompts and refining them for improved AI outputs. Exploring ethical considerations around AI use and its impact on society.	• Learning that a separate computer can program external devices. • Recognising how the size of RAM affects the processing of data. • Learning the vocabulary associated with data: data and transmit. • Recognising that computers transfer data in binary and understanding simple binary addition. • Relating binary signals (Boolean) to the simple character-based language, ASCII. • Learning that messages can be sent by binary code, reading binary up to eight characters and carrying out binary calculations. • Understanding how data is collected in remote or dangerous places. • Understanding how data might be used to tell us about a location. • Learn about different forms of communication that have developed with the use of technology.	• Learning the difference between ROM and RAM. • Recognising how the size of RAM affects the processing of data. • Understanding the fetch, decode, execute cycle. • Learning how the data for digital images can be compressed. • Recognising that computers transfer data in binary and understanding simple binary addition. • Understanding how bit patterns represent images as pixels. • Using logical thinking to explore software more independently, making predictions based on their previous experience. • Independently learning how to use 3D design software package TinkerCAD. • Learn about different forms of communication that have developed with the use of technology.	• Learning about the history of computers and how they have evolved over time. • Using past experiences to help solve new problems. • Writing increasingly complex algorithms for a purpose. • Debugging quickly and effectively to make a program more efficient. • Remixing existing code to explore a problem. • Changing a program to personalise it. • Evaluating code to understand its purpose. • Predicting code and adapting it to a chosen purpose. • Using search and word processing skills to create a presentation. • Understanding how search engines work. • Using search engines safely and effectively. • Understanding the importance of secure passwords and how to create them. • Using the understanding of historic computers to design a computer of the future. • Planning, recording and editing an audio recording. • Creating and editing sound recordings for a specific purpose.	• Using past experiences to help solve new problems. • Writing increasingly complex algorithms for a purpose. • Debugging quickly and effectively to make a program more efficient. • Remixing existing code to explore a problem. • Changing a program to personalise it. • Evaluating code to understand its purpose. • Predicting code and adapting it to a chosen purpose. • Using logical thinking to explore software independently, iterating ideas and testing continuously. • Creating and editing videos, adding multiple elements: music, voiceover, sound, text and transitions. • Using design software Tinkercad to design a product. • Creating a website with embedded links and multiple pages. • Understanding how search engines work. • Using search engines safely and effectively.
Year 4 Key Knowledge	Code may sometimes need to be downloaded onto a physical system/device. Devices with sensors (such as pedometers, security systems, thermostats and light sensors) are often programmed to perform specific tasks in reaction to the input from the sensors. Running a program to identify errors should be done before checking the code. Errors in a program could be a result of forgetting to ‘end’ a loop. Typing and spacing are very important in text-based languages and can cause errors in code if used incorrectly.	AI is artificial intelligence and is used in everyday life. AI is trained on data to recognise patterns and generate outputs. AI can be used to generate written content. AI can be used to create visual content like pictures. AI can help generate basic HTML code to create the structure and layout of a website. There are ethical issues surrounding AI, including data privacy, bias and responsible use.	Mars Rover is a motor vehicle that collects data from space by taking photos and examining rock samples. What numbers using binary code look like and be able to identify how messages can be sent in this format. RAM is Random Access Memory and acts as the computer’s working memory. What simple operations can be used to calculate bit patterns	• To understand that bit patterns represent images as pixels. • To understand that the data for digital images can be compressed. • To know the difference between ROM and RAM. • To understand various techniques that will improve the design of a 3D object (using CAD software).	• The importance of having a secure password and what brute force hacking is. • The first computers were created at Bletchley Park to crack the Enigma code to help the war effort in World War 2. • About some of the historical figures that contributed to technological advances in computing. • What techniques are required to create a presentation using appropriate software. • Sound clips can be recorded using sound recording software. • Sound clips can be edited and trimmed.	What designing an electronic product involves. Which programming software/ language is best to achieve a purpose.

Computing FIRECRESTS YEAR 4 & 5	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 5 Key Knowledge	- Code may sometimes need to be downloaded onto a physical system/ device. - Devices with sensors (such as pedometers, security systems, thermostats and light sensors) are often programmed to perform specific tasks in reaction to the input from the sensors. - Running a program to identify errors should be done before checking the code. - Errors in a program could be a result of forgetting to 'end' a loop. - Typing and spacing are very important in text-based languages and can cause errors in code if used incorrectly.	AI is artificial intelligence and is used in everyday life. AI is trained on data to recognise patterns and generate outputs. AI can be used to generate written content. AI can be used to create visual content like pictures. AI can help generate basic HTML code to create the structure and layout of a website. There are ethical issues surrounding AI, including data privacy, bias and responsible use.	- Mars Rover is a motor vehicle that collects data from space by taking photos and examining rock samples. - What numbers using binary code look like and be able to identify how messages can be sent in this format. - RAM is Random Access Memory and acts as the computer's working memory. - What simple operations can be used to calculate bit patterns	- To understand that bit patterns represent images as pixels. - To understand that the data for digital images can be compressed. - To know the difference between ROM and RAM. - To understand various techniques that will improve the design of a 3D object (using CAD software).	- The importance of having a secure password and what brute force hacking is. - The first computers were created at Bletchley Park to crack the Enigma code to help the war effort in World War 2. - About some of the historical figures that contributed to technological advances in computing. - What techniques are required to create a presentation using appropriate software. - Sound clips can be recorded using sound recording software. - Sound clips can be edited and trimmed.	- What designing an electronic product involves. - Which programming software/ language is best to achieve a purpose. - The building blocks of computational thinking, for example, sequence, selection, repetition, variables and inputs and outputs.
Year 4 Key Vocabulary	- Programming - loop - variable - condition - if statement	- Data transfer and networks - Cloud - Network - World Wide Web (web) - web page	Computational thinking and problem solving - decomposition - abstraction - pattern recognition - algorithm design	Computational thinking and problem solving - decomposition - abstraction - pattern recognition - algorithm design	- acrostic code - audio advert - brute force hacking - Caesar cipher - chip and PIN system - cipher - combination - date shift cipher - discovery - invention - Nth letter cipher - password	abstraction adapt advert algorithm bug code coding debug design edit electronic evaluate image rights images information input loop photos video website

Computing FIRECRESTS YEAR 4 & 5	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 5 Key Vocabulary	- conditional statement - condition (LKS2) - variable (LKS2)	- AI - AI-generated image - AI-generated text - algorithm - applications - authenticity - code - considerations - debate - ethical - fake - generate - HTML - implications - instructions - modify - output - prompt - refine - response - trained	8-bit binary - ASCII - binary - Boolean - byte - CPU - data - data size - distance - input - Mars Rover - output - planet - processing - RAM - scientist - signal - store - transmit - UHF	3D - algorithm - binary - bitmap - CAD - compression - CPU - data - drag and drop - fetch-decode-execute - input - JPEG - memory - operating system - output - pixels - RAM - RGB - ROM - transmission - tinkering	- acrostic code - audio advert - brute force hacking - Caesar cipher - chip and PIN system - cipher - combination - date shift cipher - discovery - invention - Nth letter cipher - password - pigpen cipher - scrambled - script - secret - secure - technological advancement - trial and error	- abstraction - adapt - advert - algorithm - bug - code - coding - debug - design - edit - electronic - evaluate - image rights - images - information - input - loop - photos - product - program - repetition - selection - sequence - software - structure - variable - video - website

Owls 2026-2027 (YEAR 5 and 6)

Computing OWLS YEAR 5 & 6	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Area of Learning	Programming: Micro Bit	Computer Systems and Networks: Exploring AI	Mars Rover 1	Skills Showcase: Mars Rover 2	Bletchley Park and the history of computers	Skills showcase: Inventing a product Designing an electronic product through coding, debugging and computer-aided design to understand how digital technologies support product development

Computing OWLS YEAR 5 & 6	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Unit Outcomes	- Explore and test simple programs on the micro:bit. - Predict and describe how code will work before running it. - Plan and create animations using LED displays. - Use inputs, variables and conditions to build interactive programs. - Create programs that react to sensor data, such as pedometers and weather checkers. - Create a program that responds to real-time sensor readings. - Test, debug and evaluate programs by identifying and fixing problems.	- Explain what AI is and its basic functions. - Identify real-life applications of AI that are commonly used in everyday life. - Identify how AI understands and processes text and image prompts. - Generate and refine prompts to achieve the best possible response from AI. - Identify how AI generates code and how it can be useful in web design. - Identify how AI can be a useful starting point for a project. - Explain the key ethical considerations of AI. - Debate the potential of AI replacing human roles, presenting well-structured arguments.	- Identify some types of data the Mars Rover could collect (for example, photos). - Explain how the Mars Rover transmits the data back to Earth and the challenges involved. - Read any number in binary, up to eight bits. - Identify input, processing and output on the Mars Rovers. - Use a column method to add two 8-bit binary numbers. - Relate binary signals (Boolean) to a simple character-based language, ASCII.	- Create a pixel picture, explaining that a pixel is the smallest element of a digital image and that binary is used to code and transfer this data. - Save a JPEG as a bitmap and recognise the difference in file size as well as explaining how pixels are used to transfer image data. - Explain the ‘fetch, decode, execute’ cycle in relation to real-world situations. - Create a profile with a safe and suitable username and password and begin to use 3D design tools. - Independently take tutorial lessons, applying what they have learnt to their design and understand the importance of using an online community responsibly.	- Explain that codes can be used for a number of different reasons and decode messages. - Explain how to ensure a password is secure and how this works. - Explain the importance of historical figures and their contribution towards computer science. - Present information about their historical figures in an interesting and engaging manner. - Develop an idea for a computer of the future and create a simple design. - Produce a simple audio advert with simple edits, which demonstrate an understanding of how to use the software.	- Evaluate code, understand what it does and adapt existing code for a specific purpose. - Debug programs and make them more efficient using sequence, selection, repetition or variables. - Design appropriate housing for their product using CAD software, including any input or output devices needed to make it work. - Create an appealing website for their product aimed at their target audience, which explains what their product is and what it does using persuasive language. - Create an edited video of their project, articulating the key benefits. - Describe and show how to search for information online and be aware of the accuracy of the results presented.
Year 5 & 6 Key Skills	- Making links between different programming interfaces that they are faced with. - Recognising examples of programming elements in real-life applications (e.g. live loops in music or variables in pedometers). - Planning a program to be downloaded to a physical system/device to solve a particular problem. - Decomposing a program independently when given a specific outcome or task to achieve. - Programming a physical system that has the capability for sensory input (e.g. a micro:bit). - Independently using loops to make code more efficient in text-based or block-based programs. - Using nested loops to make code more efficient. - Explaining what a program does and how it works, referring to the inputs and outputs. - Becoming more efficient and effective at debugging their programs. - Systematically identifying mistakes, problems or ‘bugs’ in a program.	- Identify different types of AI and their applications in everyday life. - Exploring text-based and image-based AI tools to understand how they generate content. - Applying coding skills like decomposition and pattern recognition to interact with AI applications. - Analysing the effectiveness of prompts and refining them for improved AI outputs. - Exploring ethical considerations around AI use and its impact on society.	- Learning that a separate computer can program external devices. - Recognising how the size of RAM affects the processing of data. - Learning the vocabulary associated with data: data and transmit. - Recognising that computers transfer data in binary and understanding simple binary addition. - Relating binary signals (Boolean) to the simple character-based language, ASCII. - Learning that messages can be sent by binary code, reading binary up to eight characters and carrying out binary calculations. - Understanding how data is collected in remote or dangerous places. - Understanding how data might be used to tell us about a location. - Learn about different forms of communication that have developed with the use of technology.	- Learning the difference between ROM and RAM. - Recognising how the size of RAM affects the processing of data. - Understanding the fetch, decode, execute cycle. - Learning how the data for digital images can be compressed. - Recognising that computers transfer data in binary and understanding simple binary addition. - Understanding how bit patterns represent images as pixels. - Using logical thinking to explore software more independently, making predictions based on their previous experience. - Independently learning how to use 3D design software package TinkerCAD. - Learn about different forms of communication that have developed with the use of technology.	- Learning about the history of computers and how they have evolved over time. - Using past experiences to help solve new problems. - Writing increasingly complex algorithms for a purpose. - Debugging quickly and effectively to make a program more efficient. - Remixing existing code to explore a problem. - Changing a program to personalise it. - Evaluating code to understand its purpose. - Predicting code and adapting it to a chosen purpose. - Using logical thinking to explore software independently, iterating ideas and testing continuously. - Creating and editing videos, adding multiple elements: music, voiceover, sound, text and transitions. - Using design software Tinkercad to design a product. - Creating a website with embedded links and multiple pages. - Understanding how search engines work. - Using search engines safely and effectively.	- Using past experiences to help solve new problems. - Writing increasingly complex algorithms for a purpose. - Debugging quickly and effectively to make a program more efficient. - Remixing existing code to explore a problem. - Changing a program to personalise it. - Evaluating code to understand its purpose. - Predicting code and adapting it to a chosen purpose. - Using logical thinking to explore software independently, iterating ideas and testing continuously. - Creating and editing videos, adding multiple elements: music, voiceover, sound, text and transitions. - Using design software Tinkercad to design a product. - Creating a website with embedded links and multiple pages. - Understanding how search engines work. - Using search engines safely and effectively.

Computing OWLS YEAR 5 & 6	AUTUMN	AUTUMN 2	SPRING	SPRING 2	SUMMER	SUMMER 2
Year 5 & 6 Key Knowledge	- Code may sometimes need to be downloaded onto a physical system/ device. - Devices with sensors (such as pedometers, security systems, thermostats and light sensors) are often programmed to perform specific tasks in reaction to the input from the sensors. - Running a program to identify errors should be done before checking the code. - Errors in a program could be a result of forgetting to 'end' a loop. - Typing and spacing are very important in text-based languages and can cause errors in code if used incorrectly.	AI is artificial intelligence and is used in everyday life. AI is trained on data to recognise patterns and generate outputs. AI can be used to generate written content. AI can be used to create visual content like pictures. AI can help generate basic HTML code to create the structure and layout of a website. There are ethical issues surrounding AI, including data privacy, bias and responsible use.	- Mars Rover is a motor vehicle that collects data from space by taking photos and examining rock samples. - What numbers using binary code look like and be able to identify how messages can be sent in this format. - RAM is Random Access Memory and acts as the computer's working memory. - What simple operations can be used to calculate bit patterns	- To understand that bit patterns represent images as pixels. - To understand that the data for digital images can be compressed. - To know the difference between ROM and RAM. - To understand various techniques that will improve the design of a 3D object (using CAD software).	- The importance of having a secure password and what brute force hacking is. - The first computers were created at Bletchley Park to crack the Enigma code to help the war effort in World War 2. - About some of the historical figures that contributed to technological advances in computing. - What techniques are required to create a presentation using appropriate software. - Sound clips can be recorded using sound recording software. - Sound clips can be edited and trimmed.	- What designing an electronic product involves. - Which programming software/ language is best to achieve a purpose. - The building blocks of computational thinking, for example, sequence, selection, repetition, variables and inputs and outputs.
Year 5 & 6 Key Vocabulary	- conditional statement - condition (LKS2) - variable (LKS2)	- AI - AI-generated image - AI-generated text - algorithm - applications - authenticity - code - considerations - debate - ethical - fake - generate - HTML - implications - instructions - modify - output - prompt - refine - response - trained	8-bit binary - ASCII - binary - Boolean - byte - CPU - data - data size - distance - input - Mars Rover - output - planet - processing - RAM - scientist - signal - store - transmit - UHF	3D - algorithm - binary - bitmap - CAD - compression - CPU - data - drag and drop - fetch-decode-execute - input - JPEG - memory - operating system - output - pixels - RAM - RGB - ROM - transmission - tinkering	- acrostic code - audio advert - brute force hacking - Caesar cipher - chip and PIN system - cipher - combination - date shift cipher - discovery - invention - Nth letter cipher - password - pigpen cipher - scrambled - script - secret - secure - technological advancement - trial and error	- abstraction - adapt - advert - algorithm - bug - code - coding - debug - design - edit - electronic - evaluate - image rights - images - information - input - loop - photos - product - program - repetition - selection - sequence - software - structure - variable - video - website