



# THE BOWLAND FEDERATION OF SCHOOLS

## COMPUTING CURRICULUM NARRATIVE LKS2



### THE NATIONAL CURRICULUM

#### Purpose of study

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.

#### Key Stage 2 Pupils should be taught to:

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

### TEN COMPUTING STRANDS

ALGORITHMS	COMPUTER NETWORKS	COMPUTER SYSTEMS	CREATING MEDIA	DATA AND INFORMATION
Be able to comprehend, design, create, and evaluate algorithms	Understand how networks can be used to retrieve and share information, and how they come with associated risks	Understand what a computer is, and how its constituent parts function together as a whole	Select and create a range of media including text, images, sounds, and video	Understand how data is stored, organised, and used to represent real-world artefacts and scenarios
DESIGN AND DEVELOPMENT	EFFECTIVE USE OF TOOLS	IMPACT OF TECHNOLOGY	PROGRAMMING	SAFETY AND SECURITY
Understand the activities involved in planning, creating, and evaluating computing artefacts	Use software tools to support computing work	Understand how individuals, systems, and society as a whole interact with computer systems	Create software to allow computers to solve problems	Understand risks when using technology, and how to protect individuals and systems

LOWER KEY STAGE 2 TOPICS			
TITLE	COMPUTING SYSTEMS AND NETWORKS	CREATING MEDIA	PROGRAMMING A
CONCEPT	<p>Connecting computers - Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks.</p> <p>The internet - Recognising the internet as a network of networks including the WWW, and why we should evaluate online content.</p>	<p>Stop-frame animation - Capturing and editing digital still images to produce a stop-frame animation that tells a story.</p> <p>Audio production - Capturing and editing audio to produce a podcast, ensuring that copyright is considered.</p>	<p>Sequencing sounds - Creating sequences in a block-based programming language to make music.</p> <p>Repetition in shapes - Using a text-based programming language to explore count-controlled loops when drawing shapes.</p>
BIG IDEAS/KEY QUESTIONS/LEARNING FOCUS	<p>To explain how digital devices function (accepting inputs and producing outputs). To identify input and output devices. Can I design a digital device? To recognise how digital devices can change the way that we work. To explain how a computer network can be used to share information. Can I explain how messages are passed through multiple connections? Can I discuss why we need a network switch? To explore how digital devices can be connected. Can I explain the role of a switch, server, and wireless access point in a network?  To describe how networks physically connect to other networks. To recognise how networked devices make up the internet. Can I explain that the internet is used to provide many services? Can I recognise that the World Wide Web contains websites and web pages? To outline how websites can be shared and stored via the World Wide Web (WWW). To describe how content can be created, added and accessed on the World Wide Web (WWW). Can I suggest who owns the content on websites? Can I explain that there are rules to protect content? To evaluate the consequences of unreliable content.</p>	<p>To explain that animation is a sequence of drawings or photographs. Can I create an effective flip book—style animation? Can I predict what an animation will look like? To plan an animation. Can I use onion skinning to help me make small changes between frames? Can I evaluate the quality of my animation? To review and improve an animation. To evaluate the impact of adding other media to an animation.  Can I identify the input and output devices used to record and play sound? To explain that audio recordings can be edited and how to do it. To recognise the different parts of creating a podcast project. To apply audio editing skills independently. To combine audio to enhance my podcast project Can I evaluate the effective use of audio?</p>	<p>To explore a new programming environment – Sprite (objects, attributes, blocks). To identify that commands have an outcome. Can I create a program following a design? To explain that a program has a start. Can I explain that the objects in my project will respond exactly to the code? To recognise that a sequence of commands can have an order. To change the appearance of my project. To create a project from a task description. Can I implement my algorithm as code?  To identify that accuracy in programming is important. To create a program in a text-based language Can I write an algorithm to produce a given outcome? To explain what 'repeat' means. Can I use a count-controlled loop to produce a given outcome? To modify a count-controlled loop to produce a given outcome. To decompose a task into small steps. Can I explain that a computer can repeatedly call a procedure? Can I design a program that includes count-controlled loops? Can I develop my program by debugging it?</p>

VOCABULARY	<p><b>High Frequency VOCABULARY</b></p> <p>Use – Download – Sharing – Ownership - Permission  Information – Accurate – Honest – Content - Adverts</p>	<p><b>Subject Specific VOCABULARY</b></p> <p>Digital device – Input - Process - Output  Program – Digital - Non-digital  Connection – Network - Network Switch  Server - Wireless Access Point  Network Cables - Network Sockets</p> <p>Internet – Router - Network Security  Website - Web Page - Web Address – Routing - Web Browser  World Wide Web, - Content – Links – Files</p>	<p><b>High Frequency VOCABULARY</b></p> <p>Animation - Flip Book Sequence  Image - Photograph Setting – Character - Events  Consistency Evaluation - Delete</p> <p>Sound – Podcast Edit – Trim – Align – Layer - Import  Record – Playback – Edit – Selection – Load – Save - Export  Evaluate - Feedback</p>	<p><b>Subject Specific VOCABULARY</b></p> <p>Stop-Frame Animation – Frame - Onion Skinning  Media – Import - Transition</p> <p>Audio – Microphone – Speaker – Headphones  Input Device - Output Device  MP3</p>	<p><b>High Frequency VOCABULARY</b></p> <p>Costume – Stage - Backdrop  Motion – Turn - Point in Direction - Go To - Glide  Sequence – Event – Task - Design  Order – Note – Chord</p> <p>Pattern – Repeat – Repetition - Value  Trace - Decompose Procedure</p>	<p><b>Subject VOCABULARY</b></p> <p>Scratch – Programming Blocks - Sprite, Code - Commands, - Run The Code  Algorithm – Bug - Debug</p> <p>Turtle  Code snippet  Count-controlled Loop</p>

TITLE	DATA AND INFORMATION	CREATING MEDIA	PROGRAMMING B
CONCEPT	<p>Branching databases - Building and using branching databases to group objects using yes/no questions.</p> <p>Data logging - Recognising how and why data is collected over time, before using data loggers to carry out an investigation.</p>	<p>Desktop publishing - Creating documents by modifying text, images, and page layouts for a specified purpose.</p> <p>Photo editing - Manipulating digital images, and reflecting on the impact of changes and whether the required purpose is fulfilled.</p>	<p>Events and actions in programs - Writing algorithms and programs that use a range of events to trigger sequences of actions.</p> <p>Repetition in games - Using a block-based programming language to explore count-controlled and infinite loops when creating a game.</p>

BIG IDEAS/KEY QUESTIONS/LEARNING FOCUS	<p>To create questions with yes/no answers. To identify the attributes needed to collect data about an object. Can I arrange objects into a tree structure? To create a branching database and test to see if it works. To explain why it is helpful for a database to be well structured. To plan the structure of a branching database. To independently create an identification tool. Can I suggest real-world uses for branching databases?</p> <p>To explain that data gathered over time can be used to answer questions. To use a digital device to collect data automatically. To explain that a data logger collects 'data points' from sensors over time. Can I talk about the data that I have captured? To recognise how a computer can help us analyse data. To identify the data needed to answer questions and use a data logger to collect data. To use data from sensors to answer questions and draw conclusions from the data that I have collected. Can I explain the benefits of using a data logger?</p>		<p>To recognise how text and images convey information. To recognise that text and layout can be edited (font style, size, and colours). To choose appropriate page settings. Can I create a template for a particular purpose? To add content to a desktop publishing publication. To consider how different layouts can suit different purposes. To consider the benefits of desktop publishing Can I compare work made on desktop publishing to work created by hand?</p> <p>To explain that the composition of digital images can be changed (rotating and cropping). To explain that colours can be changed in digital images. To explain how cloning can be used in photo editing. To explain that images can be combined Can I use a range of tools to copy between images? To combine images for a purpose. To evaluate how changes can improve an image.</p>		<p>To explain how a sprite moves in an existing project. Can I identify a way to improve a program? To create a program to move a sprite in four directions. To adapt a program to a new context. To develop my program by adding features. Can I build more sequences of commands to make my design work? To identify and fix bugs in a program. To design and create a maze-based challenge. Can I evaluate my project?</p> <p>To develop the use of count-controlled loops in a different programming environment. Can I modify a snippet of code to create a given outcome? To explain that in programming there are infinite loops and count-controlled loops Can I recognise that some programming languages enable more than one process to be run at once? To develop a design that includes two or more loops which run at the same time. To modify an infinite loop in a given program. To design a project that includes repetition. To create a project that includes repetition. Can I evaluate the steps I followed when building my project?</p>	
	VOCABULARY	<p><b>High Frequency VOCABULARY</b></p> <p>Questions Equal – Even - Separate Structure – Compare – Order – Organise - Selecting</p> <p>Analyse – Review - Conclusion</p>	<p><b>Subject Specific VOCABULARY</b></p> <p>Attribute – Value – Table - Objects Branching Database – Database – Information - Decision Tree</p> <p>Data – Table - Layout Input device – Sensor - Data Logger – Logging Data Point - Interval Data Set Import - Export Logged - Collection</p>	<p><b>High Frequency VOCABULARY</b></p> <p>Advantages – Disadvantages - Communicate Landscape – Portrait – Orientation - Placeholder Copy - Paste Purpose - Benefits</p> <p>Adjustments – Effects – Colours – Hue – Saturation – Sepia - Vignette Made Up – Real Background - Foreground</p>	<p><b>Subject Specific VOCABULARY</b></p> <p>Text, - Images Font - Font Style Template - Layout, Content Desktop Publishing</p> <p>Image – Edit – Digital – Crop – Rotate – Undo - Save Retouch – Clone – Select - Combine Composite – Cut – Alter - Zoom - Font</p>	<p><b>High Frequency VOCABULARY</b></p> <p>Move - Resize Pen – Design - Event - Action</p> <p>Costume Repetition - Duplicate Modify – Refine - Evaluate</p>

