

Barwick and Stoford Key Stage 2 Curriculum Progression: COMPUTING

National Curriculum Skills: Key Stage 2

- Pupils should design, write, and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.
- They should use sequence, selection, and repetition in programs, work with variables and various forms of input and output.
- They should use logical reasoning to explain how simple algorithms work and to detect and correct errors in algorithms and programs.
- They should 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.
- They should use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.
- They should 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.
- They should use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Key Stage 2: Cycle A (2024 – 2025)

SYCAMORE CLASS (Upper Key Stage 2)					
NCCE Coverage– Computing systems and networks / Creating media A & B/ Programming A & B/ Data and information			Autumn	Spring	Summer
Overview: Systems and searching; Video production; Selection in physical computing; Flat-file databases; Introduction to vector graphics; Selection in quizzes.					
NCCE Projects	National Curriculum Skill	Specific Knowledge			
Systems and searching	They should use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.	Learners develop their understanding of computer systems and how information is transferred between systems and devices. Learners consider small-scale systems as well as large-scale systems. They explain the input, output, and process aspects of a variety of different real-world systems. Learners discover how information is found on the World Wide Web, through learning how search engines work (including how they select and rank results) and what influences searching, and through comparing different search engines.			
Video Production	They should 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.	This unit gives learners the opportunity to learn how to create short videos in groups. As they progress through this unit, they will be exposed to topic-based language and develop the skills of capturing, editing, and manipulating video. Active learning is encouraged through guided questions and by working in small groups to investigate the use of devices and software. Learners are guided with step-by-step support to take their idea from conception to completion. At the teacher's discretion, the use of green screen can be incorporated into this unit. At the conclusion of the unit, learners have the opportunity to reflect on and assess their progress in creating a video.			
Selection in Physical Computing	Pupils should design, write, and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts. They should use sequence, selection, and repetition in programs, work with variables and various forms of input and output.	In this unit, learners will use physical computing to explore the concept of selection in programming through the use of the Crumble programming environment. Learners will be introduced to a microcontroller (Crumble controller) and learn how to connect and program components (including output devices- LEDs and motors) through the application of their existing programming knowledge. Learners are introduced to conditions as a means of controlling the flow of actions and make use of their knowledge of repetition and conditions when introduced to the concept of selection (through the if, then structure).			
Flat-file computing	They should 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.	This unit looks at how a flat-file database can be used to organise data in records. Pupils use tools within a database to order and answer questions about data. They create graphs and charts from their data to help solve problems. They use a real-life database to answer a question and present their work to others.			
Introduction to vector graphics	They should 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.	In this unit, learners start to create vector drawings. They learn how to use different drawing tools to help them create images. Learners recognise that images in vector drawings are created using shapes and lines, and each individual element in the drawing is called an object. Learners layer their objects and begin grouping and duplicating them to support the creation of more complex pieces of work. This unit is planned using the Google Drawings app, other alternative pieces of software are available.			
Selection in quizzes	Pupils should design, write, and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts. They should use sequence, selection, and repetition in programs, work with variables and various forms of input and output. They should use logical reasoning to explain how simple algorithms work and to detect and correct errors in algorithms and programs.	In this unit, pupils develop their knowledge of selection by revisiting how conditions can be used in programs and then learning how the If... Then... Else structure can be used to select different outcomes depending on whether a condition is true or false. They represent this understanding in algorithms and then by constructing programs using the Scratch programming environment. They use their knowledge of writing programs and using selection to control outcomes to design a quiz in response to a given task and implement it as a program.			

Key Stage 2: Cycle B (2023 – 2024)

SYCAMORE CLASS (Upper Key Stage 2)					
NCCE Coverage– Computing systems and networks / Creating media A & B/ Programming A & B/ Data and information			Autumn	Spring	Summer
Overview: Communication and collaboration; Web page creation; Variables in games; Introduction to spreadsheets; 3D Modelling; Sensing movement.					
NCCE Projects	National Curriculum Skill	Specific Knowledge			
Communication and collaboration	They should 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.	In this unit learners explore how data is transferred over the internet. Learners initially focus on addressing, before they move on to the makeup and structure of data packets. Learners then look at how the internet facilitates online communication and collaboration; they complete shared projects online and evaluate different methods of communication. Finally, they learn how to communicate responsibly by considering what should and should not be shared on the internet.			
Web page creation	They should 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.	This unit introduces learners to the creation of websites for a chosen purpose. Learners identify what makes a good web page and use this information to design and evaluate their own website using Google Sites. Throughout the process learners pay specific attention to copyright and fair use of media, the aesthetics of the site, and navigation paths.			
Variables in games	They should use sequence, selection, and repetition in programs, work with variables and various forms of input and output. They should use logical reasoning to explain how simple algorithms work and to detect and correct errors in algorithms and programs.	This unit explores the concept of variables in programming through games in Scratch. First, learners find out what variables are and relate them to real-world examples of values that can be set and changed. Then they use variables to create a simulation of a scoreboard. In Lessons 2, 3, and 5, which follow the Use-Modify-Create model, learners experiment with variables in an existing project, then modify them, before they create their own project.			
Introduction to spreadsheets	They should 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.	This unit introduces the learners to spreadsheets. They will be supported in organising data into columns and rows to create their own data set. Learners will be taught the importance of formatting data to support calculations, while also being introduced to formulas and will begin to understand how they can be used to produce calculated data. Learners will be taught how to apply formulas that include a range of cells and apply formulas to multiple cells by duplicating them. Learners will use spreadsheets to plan an event and answer questions. Finally, learners will create charts, and evaluate their results in comparison to questions asked.			
3D Modelling	They should 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.	Learners will develop their knowledge and understanding of using a computer to produce 3D models. Learners will initially familiarise themselves with working in a 3D space, moving, resizing, and duplicating objects. They will then create hollow objects using placeholders and combine multiple objects to create a model of a desk tidy. Finally, learners will examine the benefits of grouping and ungrouping 3D objects, then go on to plan, develop, and evaluate their own 3D model of a building.			
Sensing movement	They should use sequence, selection, and repetition in programs, work with variables and various forms of input and output. They should use logical reasoning to explain how simple algorithms work and to detect and correct errors in algorithms and programs. They should 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.	This unit is the final KS2 programming unit and brings together elements of all the four programming constructs: sequence from Year 3, repetition from Year 4, selection from Year 5, and variables (introduced in Year 6 – 'Programming A'). It offers pupils the opportunity to use all of these constructs in a different, but still familiar environment, while also utilising a physical device – the micro: bit. The unit begins with a simple program for pupils to build in and test within the new programming environment, before transferring it to their micro: bit.			