

Subject Vision

- Learning allows pupils to develop a range of skills to use in the real world
- The teaching of Computer Science and ICT will develop independent pupils who think logically and can create creative solutions to solve problems.
- Teaching Computer Science and ICT allows us to create future leaders who can solve problems and think critically about information. Pupils will have the opportunity to investigate different problems from the real world, come up with solutions and critique the solutions from others.
- All pupils have the right to study Computer Science and ICT as it will allow them to break down problems into sub problems, understand that every problem has many solutions and to think critically about the different solutions.
- Pupils will have opportunities to work alongside businesses to see how these skills can be used in the workplace. In a world suffused by computation, every school-leaver should have an understanding of computing and ICT.

National Curriculum KS3 Computing

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 2 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, reuse, revise and repurpose 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

[National curriculum in England: computing programmes of study - GOV.UK \(www.gov.uk\)](https://www.gov.uk/guidance/national-curriculum-in-england-computing-programmes-of-study)

End Points

EP1. Exploring User Interface Design Principles and Project Planning Techniques (ICT)

EP2. Collecting, Presenting and Interpreting Data (ICT)

EP3. Effective Digital Working Practices (ICT)

EP4. Computer systems (CS)

EP5. Computational thinking, algorithms and programming (CS)

EP6. In house systems and digital safety (CS) (ICT)

Subject Domains of Knowledge

D1. Systems architecture

D2. Memory and storage

D3. Computer networks, connections, protocols and security

D4. Systems software

D5. Algorithms

D6. Programming (fundamentals, languages, environments and producing)

D7. Boolean logic

D8. Interfaces

Subject Key Concepts

C1. Audience needs

C2. Design principles

C3. Project planning techniques

C4. Create a project plan

C5. Create an initial design

C6. Ethical, legal, cultural and environmental impacts of digital technology

C7. Data representation



Computing & ICT Year 7

Medium Term Curriculum Plan

Year 7: Computing

Units	Unit 1: Respect Online	Unit 2: Scratch
Unit Overview	In this unit students will learn how to use our school network and how to use it appropriately with an awareness of digital safety (respecting others online, spotting strangers and the effects of cyberbullying). Students will be introduced to a range of presentation software including PowerPoint .	In this unit students begin by reverse-engineering some existing games leading on to developing their own games, they will plan, program and review their work. Moving on to test and debug programs.
Lesson Sequence		
Key Domains and Concepts taught in this Unit / Term	D8. Interfaces C1. Audience needs C5. Create an initial design C6. Ethical, legal, cultural and environmental impacts of digital technology C7. Data representation	D5. Algorithms D6. Programming (fundamentals, languages, environments and producing) D7. Boolean logic D8. Interfaces C1. Audience needs C2. Design principles C3. Project planning techniques C4. Create a project plan C5. Create an initial design
KS4 End Points	EP2. Collecting, Presenting and Interpreting Data (ICT) EP6. In house systems and digital safety (CS) (iCT)	EP1. Exploring User Interface Design Principles and Project Planning Techniques (ICT) EP3. Effective Digital Working Practices (ICT) EP5. Computational thinking, algorithms and programming (CS) EP6. In house systems and digital safety (CS) (ICT)

<p>Declarative Knowledge (Students should know)</p>	<p>How to use the school network appropriately and safely. They will learn about monitoring software and the need to be respectful in everything they do</p>	<p>Students will be able to create a fully working game with lives, scoring and some randomisation of objects. They will learn to test and debug their programs.</p>
<p>Procedural Knowledge (Students should be able to do)</p>	<p>Whilst completing this unit, learners will also learn how to</p> <ul style="list-style-type: none"> • Logging in • Navigating the SharePoint • Using school email system • Using presentation software effectively. • Spot strangers, and the effects of cyberbullying. 	<p>They will be able to develop their programs</p> <ul style="list-style-type: none"> • planning and developing their own games, learning to <ul style="list-style-type: none"> ○ incorporate variables, ○ procedures (using the Broadcast function), ○ lists and operators.
<p>Developing T3 Literacy and Numeracy</p>	<p>Computing, password, secure, hazards, Email, recipient, network, Online, comments, community, Cyberbullying, presentation software, Catfishing</p>	<p>Language or words associated with the animation package (Scratch), for example: sprite, interface, cursor, presentation mode, block, script, broadcast, timer</p> <p>Vocabulary associated with programming a computer, for example: default, operator, boolean, program, loop, variable, troubleshoot, debug, pseudocode</p> <p>Learners will need to use structures that describe programming commands, such as: Repeat ... until, If ... else, When ..., Forever.</p>
<p>Assessment (Summative and Formative)</p>	<p>Ongoing review of class notebooks verbal feedback given Test on MS forms to generate % (recorded in line with data capture) used formatively to identify any areas of misconception and addressed in tick time activities.</p>	<p>Ongoing review of class notebooks verbal feedback given Test on MS forms to generate % (recorded in line with data capture) used formatively to identify any areas of misconception and addressed in tick time activities.</p>

<p>Links to Prior Learning</p>	<p>At Key stage 2 students will have learnt about the dangers of being online, as such this unit revisits and fills any gaps in knowledge that may have developed</p>	<p>No previous learning is necessary with this unit but most pupils will have had experience of programming or sequencing instructions from units studied in previous years using software such as Logo, RoboMind, Kodu or GameMaker. They should be familiar with using screen objects.</p>
<p>Next steps in learning</p>	<p>Throughout the year staying safe online will be revisited both here and through the ethics curriculum</p>	<p>Through KS3 block based programming will be revisited in various guises (eg microbits) and will be developed to HTML and Python</p>
<p>Common Barriers to learning in this unit</p>	<p>Builds on KS2 SOW – students arrive from a variety of schools with varying knowledge and experiences. Also, to be factored in is the loss of learning from the covid pandemic and its ongoing effects.</p>	<p>Builds on KS2 SOW – students arrive from a variety of schools with varying knowledge and experiences. Also, to be factored in is the loss of learning from the covid pandemic and its ongoing effects. Prior to this some students arrived with strong scratch skills others had never seen it.</p>

Units	Unit 3: Microbits	Unit 4: Spreadsheets 1
Unit Overview	The BBC micro:bit is a pocket-sized computer that introduces students how software and hardware work together. In this unit they learn about emulators and get to compare the difference in using an emulator and the micro:bit.	Spreadsheets impact on our daily lives but many have little understanding how they work or how they can be used to manipulate data to get meaningful results. This unit is the first step in redressing this introduction functions such as COUNTIF statements, building up from simple formulae.
Lesson Sequence		
Key Domains and Concepts taught in this Unit / Term	D1. Systems architecture D2. Memory and storage D4. Systems software D5. Algorithms D6. Programming (fundamentals, languages, environments and producing) D7. Boolean logic D8. Interfaces C1. Audience needs	D5. Algorithms D6. Programming (fundamentals, languages, environments and producing) D7. Boolean logic D8. Interfaces C1. Audience needs C2. Design principles C3. Project planning techniques C7. Data representation
KS4 End Points	EP3. Effective Digital Working Practices (ICT) EP4. Computer systems (CS) EP5. Computational thinking, algorithms and programming (CS) EP6. In house systems and digital safety (CS) (ICT)	EP1. Exploring User Interface Design Principles and Project Planning Techniques (ICT) EP2. Collecting, Presenting and Interpreting Data (ICT) EP3. Effective Digital Working Practices (ICT) EP6. In house systems and digital safety (CS) (ICT)
Declarative Knowledge (Students should know)	Students will use micro:bits to develop their understanding of programming, problem solving, giving and receiving formative feedback.	The spreadsheet unit for Year 7 takes students from having very little knowledge of spreadsheets to being able to confidently model simple data with a spreadsheet.

<p>Procedural Knowledge (Students should be able to do)</p>	<ul style="list-style-type: none"> to create simple games of increasing complexity using Microsoft MakeCode block-based programming moving on to Python by the end of the unit. 	<ul style="list-style-type: none"> Use basic formulas to writing their own COUNTIF statements. <p>This unit will give students a good set of skills that they can use in computing lessons and in other subject areas.</p>
<p>Developing T3 Literacy and Numeracy</p>	<p>Key Words – Variable, Abstraction, Decomposition, Event Handler, Java Script Block Editor, Scripting Language, Pseudocode</p>	<p>Key Words and Definitions– Data, Cell, Cell Reference, Active Cell, Row, Column, Range, Select, Drag Handle, Autofill, formula, Information Source, Primary Source, Secondary Source, Chart, Pie Chart, Bar Chart, Series, Axis /Axes, Labels, Headers, Function, Maximum, Minimum, Header, Filter, Average, Criterion / Criteria, Condition, Conditional Formatting, Worksheet, Value, Label</p>
<p>Assessment (Summative and Formative)</p>	<p>Ongoing review of class notebooks verbal feedback given</p> <p>Test on MS forms to generate % (recorded in line with data capture) used formatively to identify any areas of misconception and addressed in tick time activities.</p>	<p>Ongoing review of class notebooks verbal feedback given</p> <p>Test on MS forms to generate % (recorded in line with data capture) used formatively to identify any areas of misconception and addressed in tick time activities.</p>
<p>Links to Prior Learning</p>	<p>Students may have studied in previous years software such as Logo, RoboMind, Kodu or GameMaker. The have had their first experience of scratch in KS3 also</p>	<p>At Key Stage 2 students may have learnt about spreadsheets, as such this unit revisits but experience indicates that the vast majority have very little prior learning in this area .</p>
<p>Next steps in learning</p>	<p>They will get their first taste of Python programming this will continue to be developed throughout the KS ready for GCSE computer science.</p>	<p>Spreadsheets will be revisited in Years 8 and 9 in order to build the students skills set ready for the requirements of the Key Stage 4 courses offered</p>
<p>Common Barriers to learning in this unit</p>	<p>Mind blocks and self belief are the two common barriers, we have built to this stage already though and will use familiarity and success elsewhere to break these down.</p>	<p>Mind blocks and self belief are the two common barriers. Students will build on their Maths knowledge and will use familiarity and success elsewhere to break these issues down.</p>



Computing & ICT Year 7

Units	Unit 5: Code.org Coding I	There are only 5 planned units in Year 7 due to the CAT tests and settling in period of term 1
Unit Overview	Students are challenged to further develop their coding skills by learning to create computer programs using another block-based language, develop problem-solving skills, and work through fun challenges! They make games and creative projects to share with others.	
Lesson Sequence		
Key Domains and Concepts taught in this Unit / Term	D4. Systems software D5. Algorithms D6. Programming (fundamentals, languages, environments and producing) D7. Boolean logic D8. Interfaces C1. Audience needs	
KS4 End Points	EP1. Exploring User Interface Design Principles and Project Planning Techniques (ICT) EP2. Collecting, Presenting and Interpreting Data (ICT) EP3. Effective Digital Working Practices (ICT) EP4. Computer systems (CS) EP5. Computational thinking, algorithms and programming (CS)	
Declarative Knowledge (Students should know)	Students will learn the importance of computational thought and sequencing. They will also develop their debugging skills by modifying and improving programs.	

Procedural Knowledge (Students should be able to do)	Write and debug programs, using critical thinking skills and improving trial and error testing	
Developing T3 Literacy and Numeracy	Code, Event, Program, Algorithm, Bug, Debugging, Sequencing, Programming	
Assessment (Summative and Formative)	Ongoing review of class notebooks verbal feedback given Test on MS forms to generate % (recorded in line with data capture) used formatively to identify any areas of misconception and addressed in tick time activities.	
Links to Prior Learning	Direct links to what is learnt in the scratch and micro:bits units and aims to cement the prior learning of these units	
Next steps in learning	Part II is in Year 8 as are additional programming opportunities	
Common Barriers to learning in this unit	The way these materials are set up they are low stakes and students can work at their own pace through the materials.	