Curriculum Plan KS3 – 2021/2022

Topic	Using Computers Safely Effectively & Responsibly	
Learning Objectives	**At the end of this Unit all pupils should be able to:** • use basic file management techniques to create folders, save, copy, move, rename and delete files and folders and make backup copies of files • recognise extensions for common file types such as .doc or .docx, .ppt, .jpg etc • keep their files in well organised and appropriately named folders • explain what constitutes a "strong" password for an online account • describe a code of conduct • list some of the dangers and drawbacks of social networking sites • list some possible responses to cyberbullying • send and reply to emails, send attachments • use a search engine to find information •	**Most pupils will be able to:** • describe guidelines for keeping their identity secure on the Internet • describe what is meant by identity theft and how to minimize the risks of this • identify a probable phishing email and deal with it appropriately • describe how to minimize the danger of having their computer infected by a virus • resize images before attaching to emails • explain the advantages and disadvantages of email as a method of communication **Some pupils will be able to:** • manage a Contacts list efficiently for email • use an email signature • use the advanced features of a search engine describe why the information they find may not be accurate
Skills to be Gained	This is a theoretical unit covering the necessary basic knowledge to use computers safely, effectively and responsibly. Pupils begin by looking at file management and security. The unit then moves on to e-safety (cyber-bullying, phishing etc.), and online profiles to give pupils a better understanding and awareness of using social media. The functionality and operation of email and search engines and how to use them effectively are covered	
Assessment	Pupils will sit a multiple choice test as their final assessment.	
Links with Prior/Subsequent Learning	All pupils will have a basic knowledge of using a computer, creating and managing files. They may already have email accounts and be able to send and receive emails, and will have used a search engine such as Google.	
Numeracy/Literacy Skills	File extension, Camel caps, folder, subfolder, root folder/directory, Recycle bin, backup, shortcut key combination, backup, zip, social networking, cyberbullying, online profile, privacy settings, phishing, hacking, biometrics, encryption, virus, email provider, salutation, email signature, Carbon Copy (CC) Blind Carbon copy (BCC), attachment, search engine, server	
NC Links	• 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	

Curriculum Plan KS3 – 2021/2022

<table>
<tr><th>Topic</th><th colspan="2">Understanding Computers</th></tr>
<tr><td>Learning Objectives</td><td>At the end of this Unit all pupils should be able to:
Distinguish between hardware and software
Give examples of computer hardware and software
Draw a block diagram showing CPU, input, output and storage devices
Name different types of permanent storage device
Suggest appropriate input and output devices for a simple scenario
Explain what RAM and ROM are used for
Show how numbers and text can be represented in binary
Explain the impact of future technologies</td><td>Most pupils will be able to:
Perform simple binary arithmetic
State strengths and weaknesses of different storage devices
Describe briefly how data is stored on a CD
Some pupils will be able to:
Identify input and output devices for more complex scenarios
Explain how characters are encoded using the ASCII system
Use an ASCII reference chart to convert a character into binary and its decimal equivalent</td></tr>
<tr><td>Skills to be Gained</td><td colspan="2">It is a theoretical unit covering the basic principles of computer architecture and use of binary. Pupils will revise some of the theory on input and output covered in previous learning and continue to look at the Input-Process-Output sequence and the Fetch-Decode-Execute cycle through practical activities. Pupils will then look at some simple binary to decimal conversion and vice versa, and learn how text characters are represented using the ASCII code. This will be followed by some simple binary addition. Pupils will learn more in depth how storage devices represent data using binary patterns and physically save these patterns. Finally, they will look at a brief history of communication devices, how new technologies and applications are emerging and the pace of change.</td></tr>
<tr><td>Assessment</td><td colspan="2">Pupils will sit an end-of-unit test.</td></tr>
<tr><td>Links with Prior/Subsequent Learning</td><td colspan="2">No previous learning is necessary with this unit. Many pupils may have a basic understanding of binary and its use to represent text and images from previous years. They may also have an understanding of input and output devices and their role in the Input – Process – Output sequence.</td></tr>
<tr><td>Numeracy/Literacy Skills</td><td colspan="2">Vocabulary associated with this Unit, such as:
Input, process, output, device, hardware, software, fetch, decode, execute, binary, conversion, memory, RAM, ROM, denary, ASCII, code, pits, lands, burn, read, write,
data, track</td></tr>
<tr><td>NC Links</td><td colspan="2">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; be able to convert between binary and decimal, and perform simple binary arithmetic</td></tr>
</table>

Curriculum Plan KS3 – 2021/2022

Topic	Spreadsheet modelling	
Learning Objectives	At the end of this Unit all pupils should be able to: • Give examples of how computer models are used in the real world • Format a simple spreadsheet model • Use simple formulae and functions • Name cells in a spreadsheet model • Use a simple spreadsheet model to explore different "what if" scenarios • Create a basic pie chart to display results Most pupils will be able to: • Explain what is meant by a financial model • Explain the advantages of naming cells in a spreadsheet model	• Format, construct and manipulate a simple spreadsheet model using formulae • Use conditional functions in calculations Use conditional formatting • Use a spreadsheet model to predict and test the outcomes for different scenarios Some pupils will be able to: • Justify the formatting they have used in a spreadsheet model • Present information from a spreadsheet model in a variety of formats • Create a macro and assign it to a button on the spreadsheet • Customise a chart to present information effectively Evaluate the effectiveness of a computer model
Skills to be Gained	It is a practical, skills-based unit covering the principles of creating and formatting basic spreadsheets to produce and use simple computer models. It is suitable for pupils who have a basic knowledge of spreadsheets including cell references, simple formulae and formatting, although these topics are revised in the first lesson, making it also suitable for pupils new to spreadsheets. The unit is centred around creating a financial model for a TV show. Pupils start by looking at different types of model and then use basic spreadsheet techniques to create and format a simple financial model to calculate the expected income from viewers' voting. The model is then extended to include sales from merchandising, with the introduction of "what if" scenarios. Finally the pupils create a seating plan, book seats and calculate income from seat sales. Spreadsheet features covered include SUM, MAX, IF and COUNTIF functions, cell naming for absolute referencing, conditional formatting, validation, charting and simple macros	
Assessment	Pupils will create an Assessment Portfolio showing their final spreadsheet. They will also answer questions on spreadsheet modelling and complete a self-assessment.	
Links with Prior / Subsequent Learning	The tasks in this unit assume that most pupils will have some experience of creating basic spreadsheets from Key Stage 2. Pupils' knowledge and experience is assessed during the first lesson of this unit so that teachers can adapt subsequent lessons accordingly.	
Numeracy/Literacy Skills	Model, simulation, cell, row, column, format, decimal, integer, currency, formula, relative reference, absolute reference, validation, macro, pie chart.	
NC Links	• Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems.	

Curriculum Plan KS3 – 2021/2022

Topic	INRODUCTION TO PYTHON	
Learning Objectives	**At the end of this Unit all pupils should be able to:** • Run simple Python programs in Interactive and Script mode • Write pseudocode to outline the steps in an algorithm prior to coding • Write programs using different types of data (e.g. strings and integers) • Correctly use different variable types (e.g. integer and floating point), assignment statements, arithmetic operators • Distinguish between syntax and logic errors and be able to find and correct both types of error • Use relational operators to control the order in which program statements are executed and in what order (if and while statements)	**Most pupils will be able to:** • Write an error-free, well-documented program involving selection and iteration • Describe how a binary search is carried out • Explain the advantages of a binary search over a linear search for an ordered list **Some pupils will be able to:** • Devise their own algorithms to solve reasonably complex problems, e.g. a binary search • Test and debug their programs, and correct both syntax and logic errors • Make allowances in their programs for user input errors, ensuring that the program still runs to a successful conclusion
Skills to be Gained	The unit is an introduction to Python, a powerful but easy-to-use high-level programming language. Although Python is an object-oriented language, at this level the object-oriented features of the language are barely in evidence and do not need to be discussed. The focus is on getting pupils to understand the process of developing programs, the importance of writing correct syntax, being able to formulate algorithms for simple programs and debugging their programs.	
Assessment	Pupils will write and run a program and submit the code and screenshots of the program running in a learning Portfolio.	
Links with Prior / Subsequent Learning	No previous learning is necessary with this unit. Pupils may have had some experience of using variables and with a variety of relational operators such as **If** and **Repeat** in graphical block-based languages such as Scratch. Applying this knowledge will help their understanding of a text-based language such as Python.	
Numeracy/Literacy Skills	Vocabulary associated with programming and particularly Python, such as: Integrated development, IDLE, interactive mode, Script mode, variable, string, syntax, assignment statement, augmented assignment operator, data type, integer, float, round, BIDMAS, selection, iteration, syntax error, logic error, debug, binary search	
NC Links	• Use two or more programming languages, one of which is textual, to solve a variety of computational problems; make appropriate use of data structures; design and develop modular programs that use procedures and functions • 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.	

Curriculum Plan KS3 – 2021/2022

<table>
<tr><th>Topic</th><th colspan="2">Computational Thinking & Logic</th></tr>
<tr><td>Learning Objectives</td><td>
At the end of this Unit all students should be able to:
• Be able to ask logical questions to solve problems
• Know the common Boolean operators:
– AND
– OR
– NOT
• Know different logic gates including:
– AND gates
– OR gates
– NOT gates
• Understand what an algorithm is
• Create a sequence of instructions to achieve a goal

Most students will be able to:
• Understand how Boolean operators can be represented in written expressions and
Venn diagrams
• Understand how logic is used in different situations
• Be able to complete truth tables for logic gates and circuits with up to three inputs
• Understand how loops can be used to reduce the amount of code required for a solution
• Be able to refine algorithms to reduce the number of instructions required
• Understand the difference between lossy and lossless compression
• Be able to use an algorithm to communicate data
– Understand how the algorithm can be improved
– Use a binary tree to further improve the algorithm
• Understand why compression is needed for video transmission and photo storage
• Understand how abstractions are used in everyday life
• Be able to create abstractions for different purposes
• Understand how networks are used to make an abstraction of a maze
• Understand how decomposition can be used to break down problems into more manageable components
• Be able to break down a large Computing problem into its parts and understand:
– how images are converted to binary using pixels
– how text is converted to binary using ASCII

Some students will be able to:
• Understand how nested loops can be used to improve solutions further
• Be able to use an algorithm to communicate data
– Use a binary tree to further improve the algorithm
• Understand network (graph) theory terms including:
– Nodes
– Edges
• Be able to break down a large Computing problem into its parts and understand:
– how data is broken up into packets
– how data is sent through a network
</td></tr>
</table>

Curriculum Plan KS3 – 2021/2022

Skills to be Gained	This unit introduces students to the world of computational thinking and logic. With the help of many unplugged activities, students get to understand the power of problem solving and the different methods that Computer Scientists use to tackle problems. All activities that can be carried out by computer have a paper alternative. This unit includes many novel activities to introduce key topics. For example, logical deductions and logical puzzles are used to show logical thinking, water pipes are used to introduce logic gates, network topology is used to show how mazes can be solved and phone messaging is used to demonstrate decomposition.
Assessment	Homework is given for each lesson. These consist of a mixture of short, factual questions and longer problems that need to be solved. A final multi-choice test of 40 questions is also given
Links with Prior/Subsequent Learning	Students should have studied algorithms and computational thinking topics either at Key Stage 2 or Key Stage 3. Some knowledge or experience with Scratch will be advantageous but is not essential. This unit has been written to be accessible for years 7-9. The unit assumes no specific prior knowledge, however some of the tasks can be challenging. It will depend on the experience and ability of students as to which year group is most appropriate for teaching this unit.
Numeracy/Literacy Skills	Vocabulary associated with this unit, such as: Logical thinking, logic, Boolean operators, AND, OR, NOT, logic gates, AND gate, OR gate, NOT gate, algorithm, sequence, Venn diagram, truth table, circuit, loop, nested loop, instructions, binary tree, abstraction, network, decomposition, pixels, ASCII, nodes, edges, packets, source, destination.
NC Links	• Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems • Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem • Use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions • Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]