

Curriculum Map							
Year 8							
Half term	Unit title with hyperlink to scheme of work	Unit summary	Skills & content covered	Skills & content revisited	Links to GCSE skills and content	Summary of formative marking, feedback and student response	Summative assessment schedule, including assessment criteria
Autumn 1	Computer Systems	This unit provides students with an overview of how the computing system operates. Students will learn a wide range of layers to a computer system from software to the physical components that store and execute programs, to the important factors of building a network and how it operates.	Understand the difference between computer hardware and software. Explain when a particular input/output/storage device should be used. Define what a network is. Discuss the uses of components required to build a network.	Students will use prior knowledge of devices they have used in real life settings to help support them to understand the topic.	Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems	Homework, Teams activities and oral feedback	End of unit assessment
Autumn 2	Data Representation	This unit provides students with the essential knowledge relating to binary representations. The activities gradually introduce learners to binary digits and how they can be used to represent text and numbers. The concepts are linked to practical applications and problems that the learners are familiar with.	Understand that data is represented using binary. Be able to carry out binary/denary conversions. Understand why humans use hexadecimal and carry out all hexadecimal conversions. Students carry out binary arithmetic using binary shifts. Understand the importance and limitation of ASCII.	<i>Students will use knowledge learnt in previous topic (Computer Systems) to understand how computers store data and how logic gates work.</i>	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]	Homework, Teams activities and oral feedback	End of unit assessment
Spring 1	HTML	Learners will explore the technologies that make up the internet and World Wide Web. Starting with an exploration of the building blocks of the World Wide Web, HTML, and CSS, learners will investigate how websites are catalogued and organised for effective retrieval using search engines. By the end of the unit, learners will have a functioning website.	Describe what HTML is. Modify HTML tags using inline styling to improve the appearance of web pages. Use CSS to style static web pages. Assess the benefits of using CSS to style pages instead of in-line formatting. Explain how search engines 'crawl' through the World Wide Web and how they select and rank result.	Pupils have previously learnt how to identify target audience and capture the attention. Students will apply this knowledge when designing and creating their websites.	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	Homework, building a website and oral feedback	End of unit assessment
Spring 2	Python	This unit introduces learners to text-based programming with Python. This is the programming language that they will be learning throughout the GCSE course. The lessons form a journey that starts with simple programs involving input and output, and gradually moves on through arithmetic operations, comparison operators, selection, and iteration.	Describe what algorithms and programs are and how they differ. Write programs using print(), input() and variables. Use multi-branch selection to control the flow of program execution with the use of arithmetic/comparison operators. Use iteration to control the flow of program execution.	Students should have some experience in writing programs. Whilst much of their experience of this is through drag and drop programming they have studied one unit introducing them writing code in HTML and CSS.	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	Homework, creating programs when provided with a given scenarios and oral feedback	End of unit assessment
Summer 1	Cryptography	Students will explore different forms of cryptography and the benefits of using encryption.	Students will be able to use Caesar Cipher to encrypt and decrypt messages. Learn the importance of the enigma machine to modern day computer science.	Students also studied a unit of "Computer Systems". This helped them to understand some of the main components of a computer and how networks are created. This will aid them to pin point on what is required to protect data from 'hackers'.	The principle of encryption to secure data across network connections.	Homework, Teams activities and oral feedback	End of unit assessment
Summer 2	Mobile App Development	Students will be using App Lab from code.org where learners will explore the different stages of an app developer. They will design, develop and create their own mobile app.	Learners will learn computational thinking skills in order to design, project manage and develop their app. They will work in pairs to consider the needs of the user and carry out tests on each others app. This will enable them to then evaluate the success of the project against the needs of the user.	Students will draw from the skills and knowledge they have learnt from Y7 and earlier in the year when working with different programming language/platforms such as Scratch and Python.	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.	Homework, tasks to develop apps and oral feedback	Project based - students will have developed a mobile app