To create, share, and apply knowledge in Computer Science, including in interdisciplinary areas that extend the scope of Computer Science and benefit humanity; to educate pupils to be successful, ethical, and effective problem-solvers and life-long learners who will contribute positively to the economic well-being of our region and nation and who are prepared to tackle complex 21st Century challenges facing the world.

	UDW 1							UDW 2							UDW 3							UDW 4							UDW 5						
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14	Week 15	Week 16	Week 17	Week 18	Week 19	Week 20	Week 21	Week 22	Week 23	Week 24	Week 25	Week 26	Week 27	Week 28	Week 29	Week 30	Week 31	Week 32	Week 33	Week 34	Week 35
Year 7	Learn about the school computer use policy, how to login and create a secure password.	Undertake a baseline assessment to review KS2 prior learning	Be able to create/manage folders in Google Drive and understand the importance of backups.	Using, creating and uploading a Google Doc.	Using, creating and uploading Google Slide presentation.	Using and creating a Google Form to collect data using a range of multiple choice questions.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Use basic file management techniques.	Guidelines for keeping their identity secure on the internet. Understand phishing emails and how to deal with them appropriately.	How to use email effectively.	Use the advanced features of a search engine.	The dangers and drawbacks of social networking sites.	Exam technique - answering short and long questions about computer science, thinking and logic.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Distinguish between hardware and software	Distinguish between main memory and permanent storage devices.	Convert decimal (denary) integers to binary numbers.	Add two binary numbers (each no more than eight binary digits).	Describe how data is stored on a CD.	Exam technique - answering short and long questions about Computer Science, understanding computers.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Introduction to Computer Science including key terms.	Use basic formulae with cell references for calculations in a spreadsheet	CREATE a Financial model to predict the profit on the sale of merchandise	USE the functions SUM, MAX, MIN, and COUNTA in a spreadsheet. Create appropriate charts in a spreadsheet.	USE the functions COUNTIF, AVERAGE, and IF in a spreadsheet	Exam Technique - test understanding of modelling.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Introduction to Vector graphics	Understanding Bitmap graphics	Conveying meaning through graphics	CREATE Effects and enhancements to of graphics	Summative assessment Google Form	Creating your own graphical design	Complete a learning review of the topics covered
	Pupils will know: - how to login, manage my files and folders on the network and Google Classroom; - how to create a presentation using Google Slides; - how to create a word processor using Google Docs; - how to create a quiz using Google Forms; Pupils will be able to: - login, create and manage folders on the network and Google Classroom; - create a presentation using a range of functions in Google Slides; - create a word processor using a range of functions in Google Docs; - create a quiz using a range of functions in Google Forms;							Pupils will know: - the possible dangers of social networking sites; - how to respond to threats on the internet; - how to keep their identity secure on the internet. Pupils will be able to: - create a secure, memorable password; - identify a phishing scam; - identify dangers of social media sites; - respond to threats on the internet.							Pupils will know: - how to distinguish between hardware and software; - how to draw a block diagram of the main components of a computer; - how to convert decimal (denary) integers to binary numbers. Pupils will be able to: - identify input, output and storage devices; - name at least five pieces of software; - add two binary numbers (each no more than eight binary digits); - describe how data is stored on a CD.							Pupils will know: - how set up a basic spreadsheet; - how to use relative and absolute referencing; - how to use a range of functions to calculate profit to meet a given target. Pupils will be able to: - compare different types of computer models; - CONTRASTING a financial model to predict the profit on the sale of merchandise; - USE functions including Max, Min and IF; - USE relative and absolute referencing.							Pupils will know: - how to use and manipulate layers to create a final layer graphic; - how to compare the different characteristics, strengths and uses of vector and bitmap images; - how to change the saturation, brightness and contrast in an image. - how to use care and attention when selecting a font to get a particular message across. Pupils will be able to: - create and manipulate a simple group of objects to form a logo design; - manipulate vector and bitmap images; - create and save a graphic in a format that preserves the layers; - use care and attention when selecting a font to get a particular message across.					
Year 8	Understand how to ask logical questions to solve problems. Understand the common boolean operators.	Learn how logic is used in different situations. Use different logic gates to create truth tables.	Learn how to create a sequence of instructions to achieve a goal and understand how loops are used in algorithms.	Create algorithms to communicate data and understand why compression is needed in the exchange of files.	Be able to create abstractions for different purposes.	Exam technique - answering short and long questions about computation ar thinking and logic.	Complete a learning review of the topics covered. Also complete pupil voice survey.	IDENTIFY common types of computer crime. Learn about different types of email scams and how to recognise the signs of fraudulent emails.	INVESTIGATE the Computer Misuse Act. EXPLORE what is meant by hacking and how research ways to protect yourself from malware and hacking.	DISCUSS the need for various organisations to hold data about you. Be aware of the possibility of identity theft and understand how to minimise the chance of identity theft.	INVESTIGATE Copyright law, what it says and what it means. ASSESS the consequences of illegal copying does to individuals, companies and society and compare copyright infringement with plagiarism.	RESEARCH the Health and Safety law and learn about the need to dispose of computer equipment in a responsible manner.	Exam technique - answering short and long questions about computer crime and Health and Safety.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Understand what a database is including flat file database, record and field. Enter data into a database in order to create queries using multiple criteria to answer questions.	Design a database with one or two tables. Create the database structure using suitable field types. Add validations and lookups to a suitable field. Add records to the table and test the validations.	Note examples of queries on websites. Use complex criteria to select records and create a parameterised query. Sort data on one or two fields and choose which data to display in the results table.	Discuss examples of forms on websites and create a new form from a query or table. Make the form user-friendly, add a title and order tabbing. Add tables using a form and find, enter, modify and delete records using a form.	Create a report based on a query, edit the layout of the report and create a consistent design.	Exam technique - answering short and long questions about Database Development.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Compare how humans and computers understand instructions	Define a variable as a name that refers to data being stored by the computer	Define a condition as an expression that will be evaluated as either true or false	Create condition that use comparison operators (<,>,=)	Summative assessment - Google Form	Define iteration as a group of instructions that are repeatedly executed	Independently design and apply programming constructs to solve a problem	Define a subroutine as a group of instructions that will run when called by the main program or other subroutines	Identify where condition-controlled iteration can be used in program	Evaluate which type of iteration is required in a program	Define a list as a collection of related elements that can be referred to by a single name	Summative assessment Google Forms	Decompose a larger problem into smaller subproblems	Recap learning over two scratch units
	Pupils will know: - how to ask logical questions to solve problems; - how the common boolean operators are used; - how boolean operators work on a Venn diagram; - how to create different situations; - how to use different logic gates. Pupils will be able to: - complete truth tables; - explain the difference between lossy and lossless compression; - use an algorithms to communicate data; - explain why compression is needed in the exchange of files.							Pupils will know: - how to IDENTIFY common types of computer crime; - how to minimise the chance of identity theft; - how to dispose of computer equipment in a responsible manner; - how to create a programming data protection. Pupils will be able to: - RECOGNISE the signs of fraudulent emails; - IDENTIFY common types of computer crime; - DESCRIBE the content on Major Acts concerning computer use; - DESCRIBE how to safely dispose of an old computer.							Pupils will know: - the features of a database; - how databases work to hold and search for data; - how to manipulate data using a database; - how to create a query in an appropriate format. Pupils will be able to: - give examples of databases used by organisations; - create a database using several field with different data types; - create a basic form to input data; - query the database and produce a suitable report.							Pupils will know: - how to pseudocode; - how to perform basic programming using Scratch; - how to use iteration and variables in computer to perform a desired outcome; - how to create a basic algorithm. Pupils will be able to: - predict the outcome of a set of scratch instructions; - produce a sequence of instructions on scratch; - build an algorithm on Scratch; - independently debug an set of errors in a program.							Pupils will know: - how to identify identify how subroutines can be used for decomposition; - how to problem solve using scratch; - how variables and iteration are used in a program; - how to decompose a larger problem into smaller subproblems. Pupils will be able to: - define programming key terms; - identify and solve problems in a scratch sequence; - identify which type of iteration is required in a program; - decompose a larger problem into smaller subproblems.					
Year 9	Introduction to Computer Science including key terms.	Use variable and iteration expressions when writing programs in Python.	Use selection statements when writing programs in Python. Use indentation when creating the programs.	Use iteration when writing programs in Python.	Create more complex programs using nested three control structures (sequence, selection, iteration)	Exam technique - answering short and long questions of data types, selection, iteration, sequence and errors.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Learning review of Python Skills - selection.	Create lists and be able to access the items in the list.	Practise different methods using lists that have been created.	Use iteration (while loops) to control the flow of program execution.	Develop guided mini-projects that will help pupils refine their skills.	Exam technique - answering short and long questions about programming.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Identify what HTML is and the basic HTML tags.	Apply HTML tags to construct a web page using a template from a provided design.	Describe and use CSS to style static web pages.	Create self-formatted HTML documents to include links, text and images.	Explain how search engines work and how the World Wide Web and how they select and rank results.	Exam technique - answering short and long questions about HTML and website development.	Complete a learning review of the topics covered. Also complete pupil voice survey.	Identify control flowchart symbols and use them to describe systems.	Understand why a flowchart is useful, how they are used to describe systems.	Identify common types of sensors used in control systems.	Develop a control flowchart for a system that uses a subroutine	Summative assessment - Google form	Understand what function are and why they are used in control systems.	Understand what a subroutine is and why they are used to control systems.	Analyse what makes a good advertisement.	Create a storyboard and script for a TV advert.	Work in a group to create a TV advert.	Edit a video clip. Review unwanted parts.	Review your video.	Exam technique - answering short and long questions about creating videos.	Complete a learning review of the topics covered. Also complete pupil voice survey.
	Pupils will know: - how to write, save and run a program in Script mode; - how to use the input statement to prompt for input and output; - how to USE indentation correctly to define a block of code. Pupils will be able to: - run a simple Python program in interactive mode using the input and print functions; - APPLY arithmetic using the BIDMAS rule; - write a program involving input, calculation and output; - IDENTIFY different types of program errors.							Pupils will know: - how to use selection to control the flow of a program; - how to CREATE lists; - how to use indentation correctly to define a block of code. Pupils will be able to: - access items from a list in Python; - APPLY arithmetic using the BIDMAS rule; - USE indentation correctly within a block of code; - IDENTIFY different types of program errors.							Pupils will know: - how to USE basic HTML tags using a text editor; - how to RECORD the HTML code and view the changes in a browser; - how to USE CSS to style static web pages. Pupils will be able to: - USE HTML tags to create a simple webpage; - APPLY HTML tags to construct a web page structure from a provided design; - USE CSS to style static web pages; - PRESENT images within a flowchart.							Pupils will know: - how to describe different functions within a Flowd flowchart; - how to identify problems that may arise in control systems; - how to identify the correct flowchart symbols in a flowchart; - how to create a flowchart to produce a decision. Pupils will be able to: - produce a flowchart using different functions; - use variables within a control system; - use decision symbols in a flowchart; - independently produce a flowchart.							Pupils will know: - how to describe the purpose of different video clips; - how to create a storyboard and script for a TV advert; - how to edit a video clip, removing unwanted parts. Pupils will be able to: - identify different filming techniques used in advertisement; - define and identify diegetic and non-diegetic sounds; - work in a group to create a TV advert; - Add text, including any relevant information, to your video frames where necessary.					
Year 10	Introduction to Computer Science including key terms.	Understanding Algorithms.	Technique to improve understanding and understanding. Using programming constructs: input/output and repetition.	Using repetition construct for loops, range function.	Exam technique - answering short and long questions about constants, variables, constructs and algorithms.	Binary addition and representation of unsigned integers.	Review understanding of programming structures.	INVESTIGATE different types of software and OS operating system (OS).	INVESTIGATE how data is represented in binary.	EXPLAIN the purpose and implications of compressing files.	INVESTIGATE the bandwidth in computer networks.	DEVELOP my exam technique by answering short and long questions about lossless compression and run length encoding.	INVESTIGATE the implications of network protocols.	Review understanding of networks and protocols.	INVESTIGATE different types of cyber attack and associated protection techniques including penetration testing and ethical hacking.	INVESTIGATE how data is stored in a relational database and arranged in a structured database.	IDENTITY iteration (for loop) and conditional (if x, variable declarations and commenting in example pseudocode.	EXPLAIN the implications of binary representation of data (sampling frequency, resolution) when constrained by the number of available bits.	DEVELOP my exam technique by answering short and long questions about network security and databases.	EXPLAIN the implications and disadvantages of binary and linear searches. Binary representation of sound and binary shifts.	Review understanding of binary and linear searches, binary representation of sound and binary shifts.	RESEARCH the ethical implications of AI including inclusion and professionalism) on society.	RESEARCH the relationship between open source and proprietary software.	RESEARCH the role of standard network protocols (Ethernet, Wi-Fi, TCP/IP, HTTP, HTTPS, FTP, email (POP3, SMTP, IMAP)	DESIGN and use flowcharts to record data (normal, boundary, erroneous) for a given scenario.	Review understanding of ethical issues, protocols, translators and software modelling.	DEVELOP and PRESENT a report on current different temperature measurements.	ANALYSE a given problem to identify the requirements of the program that will be designed.	CREATE an algorithm which outputs a username based on a student's first name, surname and year of enrolment.	CREATE an algorithm which outputs a username based on a student's first name, surname and year of enrolment.	DEVELOP my exam technique by answering short and long questions about high level languages and algorithms.	ASSESS the suitability of my program meets the requirements for the given task.	EVALUATE the suitability of my program meets the requirements for the given task.	Review understanding of programming skills.
	Pupils will know: - how to ANALYSE standard operators; - how to use the input, print and fix errors in algorithms; - how to create and use algorithms to meet specified requirements. Pupils will be able to: - DEFINE different data types; - ANALYSE the purpose of algorithms in meeting specified requirements; - identify, locate and fix errors in algorithms; - represent numbers as binary.							Pupils will know: - how negative numbers are represented in binary form using two's complement; - how to use indentation correctly to define the advantages and disadvantages of doing so; - how data is transmitted in wired and wireless networks. Pupils will be able to: - EXPLAIN the key functions of an Operating System; - DEMONSTRATE lossy and lossless compression algorithms; - EXPLAIN the role of a range of network protocols.							Pupils will know: - how sound and an analogue signal is represented in binary; - how to use the advantages and disadvantages of a structured database; - how to find the position of a target value within a sorted array. Pupils will be able to: - perform logic and arithmetic binary shifts; - explain the different forms of cyber attack and associated protection techniques; - construct general expressions to solve a range of problems.							Pupils will know: - how technology impacts the environment; - how high level programming languages are translated to machine code; - how software can be used to simulate aspects of the real world. Pupils will be able to: - analyse a range of ethical debates in computer science; - describe the role of protocols in networking; - test a program effectively using normal, boundary and erroneous data.							Pupils will know: - how to calculate different file sizes; - how to DEVELOP and PRESENT a computer program for a given scenario; - how to test my program and make suitable improvements. Pupils will be able to: - DESCRIBE different types of hardware components; - EXPLAIN the role of different computer components; - ANALYSE a given problem and identify the requirements of the program that will be designed.					
Year 11	Course introduction - Computer Science and introduction to Python and programming	Using operators and variables in operations. Understanding of precedence and parenthesis	Using input/output. Input function, int function.	Using repetition construct for loops, range function.	Exam technique - answering short and long questions about constants, variables, constructs and algorithms.	Using and interpreting flowcharts and pseudo-code.	Review understanding of programming structures.	UNDERSTAND the function of and purpose of the main components of programs.	Write programs that use sequence, selection, repetition.	Write programs that store appropriate use of primitive data types.	Write a program in Python that accepts and responds appropriately to user inputs.	Exam technique - answering short and long questions about constants, variables, constructs and algorithms.	Write programs that use appropriate use of variables and constants.	Review understanding of programming structures.	Write programs that manipulate strings.	Write programs that accept and respond appropriately to user input.	Write programs that read from and write to comma separated value text files.	UNDERSTAND the purpose of and be able to write programs that implement validation.	Exam technique - answering short and long questions about constants, variables, constructs and algorithms.	UNDERSTAND the purpose of and be able to write programs that implement authentication.	Review understanding of programming structures.	Write programs that use arithmetic operators.	Write programs that use relational operators.	Write programs that use logical operators.	Write functions that take parameters but must return values.	Exam technique - answering short and long questions about constants, variables, constructs and algorithms.	UNDERSTAND the difference between and be able to write programs that make appropriate use of global and local variables.	Review understanding of programming structures.	FORMULATE an algorithm and testing for the given program.	CREATE pseudo code efficiently for the given program.	COMPILE a program to meet the given requirements.	ASSESS how my program meets the requirement s for the given task.	DEVELOP my exam technique by answering short and long questions about high level languages and algorithms.	EVALUATE how my program meets the requirement s for the given task.	complete a review and REVISE to ASSESS my programming skills.
	Pupils will know: - how to ANALYSE standard operators; - how to identify, locate and fix errors in algorithms; - how to create and use algorithms to meet specified requirements. Pupils will be able to: - DEFINE different programming constructs; - identify, locate and fix errors in algorithms; - ANALYSE the purpose of algorithms in meeting specified requirements; - identify, locate and fix errors in algorithms;							Pupils will know: - how to identify the structural components of programs; - how to use appropriate use of sequencing, selection, repetition; - how to make appropriate use of primitive data types. Pupils will be able to: - write programs that make appropriate use of sequencing, selection, repetition; - write programs that make appropriate use of primitive data types; - write a program in Python that accepts and responds appropriately to user inputs.							Pupils will know: - how to manipulate strings; - how to write programs that accept and respond to user inputs; - how to write programs that read and write to CSV files. Pupils will be able to: - write programs that manipulate strings; - write a program that accepts and responds appropriately to user input; - write a program that reads from and writes to comma separated value text files.							Pupils will know: - how to use arithmetic operators; - how to use relational operators; - how to use logical operators; - how to use functions in a program. Pupils will be able to: - write programs that use arithmetic operators; - write programs that use relational operators; - write programs that use logical operators; - write programs that use functions.							Pupils will know: - how to assess the requirements needed to create a computer program for a client; - how to create a computer program for a given task; - how to extend my program to meet all of the clients requirements. Pupils will be able to: - CREATE a computer program for given requirements; - ASSESS if my program meets the clients needs; - EVALUATE my program against the clients needs.					
Year 12 - Unit 1	Categorise hardware and software and input out and storage devices.	Features and and software and input. Understand the purpose of computer architecture.	Features and implications of computer systems for data processing	The role of the kernel, special and general registers.	The factors affecting the choice and use of different user interfaces. UCW1 assessment.	Use and interpretation of different types of systems used in computer systems.	The implications of using codes to represent character sets. How bitmap/raster images are stored. Learning review.	The purpose, features and use of internal components.	The hardware used in computer systems.	Data storage and recovery systems.	The role of the operating system in controlling and managing system components and tasks.	Factors affecting the choice and use of user interfaces. UCW2 assessment.	The concept of utility and application software and the distinction between the two.	The use, features and implications of computer systems for data processing. Learning review.	INVESTIGATE the implications and implications of data compression.	RESEARCH common viruses and other forms of malware and their effects.	DESCRIBE the methods of connecting devices and transmitting data across and between computer systems.	Learn how bitmap/raster image data are stored and represented in a computer system.	DEVELOP my exam technique by answering short and long style exam questions on computer security and software.	IDENTIFY the protocols used to govern and control data transmission.	Complete a learning review on ASSESS the understanding of computer security and software.	Learn about searching and sorting algorithms.	DEMONSTRATE how sign and magnitude and 1's and 2's complement methods.	Learn how bitmap/raster image data is stored and represented in a computer system.	INVESTIGATE the applications and implications of data compression.	DEVELOP my exam technique by answering short and long style exam questions on data compression, data storage and software.	DESCRIBE the factors that affect the choice of hardware, utility and application software.	Compare a learning review on ASSESS the transmission.	DEVELOP my exam technique by answering short and long style exam questions on the use and choice of hardware, the features and implications of embedded and mobile CPU architecture, the advantages and disadvantages of non-volatile and floating point numbers.	DEVELOP my exam technique by answering short and long style exam questions on the types of communication and the factors affecting the choice of connecting devices and transmitting data across and between computer systems, the advantages and disadvantages of methods to fulfil specific tasks and functions.	DEVELOP my exam technique by answering short and long style exam questions on the choice and use of user interface, the use of library subroutines to perform calculations.	DEVELOP my exam technique by answering short and long style exam questions on the use, application software, and exemplar to in flow charts.	EXPLAIN the implications of websites for unit 15.	EXPLAIN the principles of website design for unit 15.	ANALYSE the principles of website design for unit 15.
	Pupils will know: - how to distinguish between different types of software; - how the hardware and software components of a computer system work; - how to create and use a computer system. Pupils will be able to: - EXPLAIN the purpose of the components of a computer system; - distinguish between different types of software; - EXPLAIN how different hardware and software are used; - DESCRIBE how data is stored and controlled in a computer system.							Pupils will know: - the purpose, features and use of internal components; - The different hardware used within a computer system; - The different types of data storage and recovery systems within a computer system. Pupils will be able to: - describe the role of the kernel; - explain the factors affecting the choice of user interfaces; - explain the features and implications of data processing.							Pupils will know: - how data is transmitted across computer systems; - how protocols are used to govern and control data; - how data is represented. Pupils will be able to: - describe the methods of transmitting data; - explain the features of application and utility software.							Pupils will know: - how computer software is used within a computer; - how image data is stored in a computer system; - how sign and magnitude are represented. Pupils will be able to: - IDENTIFY the purpose and uses of different computer software; - ANALYSE how images are stored and represented; - EXPLAIN different types of computer architecture.							Pupils will know: - how binary is represented as negative and floating point numbers; - how data is transmitted across computer systems; - how the factors affecting the choice of connecting devices are used with different computer devices. Pupils will be able to: - DEVELOP my exam technique by answering short exam style questions; - DEVELOP my exam technique by answering long exam style questions.					
Year 12 - Unit 2	Introduction to Computer Science including key terms.	Identify different threats to data and the IT systems of organisations.	Research a report based on the research for assessment criteria A.P1	Introduction to the principles of information security when protecting the IT systems of organisations.	Produce a report based on the research for assessment criteria A.P2	Research and report on organisations must adhere to legal requirements when protecting IT systems security.	Produce a report based on the research for assessment criteria A.P3	Research and report on different methods of authentication and applications.	Research and report on different methods of authentication and applications.	Summarise symmetric and asymmetric key encryption methods.	Produce a report based on the research for assessment criteria B.M2.	Research and report on how organisations can develop disaster recovery plans to cover a range of threats.	Research and report on how organisations can develop business continuity plans to help defend an organisation from security threats.	Research and report on techniques used to help defend an organisation from security threats.	Produce a report based on the research for assessment criteria C.P5.	CREATE a plan to protect an IT system that meets legal and legislative requirements.	Produce a report based on the research for assessment criteria C.P6.	Configure anti-malware scanning schedules.	Learn how to configure firewalls to control network connections that are allowed and prevent all other unauthorised connections.	ANALYSE firewall events and log entries.	Produce a report based on the research for assessment criteria D.P8.	ANALYSE and interpret activity logs.	IDENTIFY the choice of protocols for connections used to defend the IT systems of an organisation, showing how the IT systems protect against attacks, security threats.	Produce a report based on the research for assessment criteria C.M3.	Evaluate the effectiveness of the choice of protocols against the requirements given in my security plan.	Produce a report based on the research for assessment criteria D.M4.	Carry out, my plan and evaluate the effectiveness of the choice of protocols against requirements.	Produce a report based on the research for assessment criteria C.D3 C.D4.	Summarise responsibility and effective self-management in the planning and protection of an IT system.	Produce a report based on the research for assessment criteria D.D5.	DEVELOP my exam technique by answering short exam style questions.	DEVELOP my exam technique by answering short exam style questions.
	Pupils will know: - how to identify different security threats; - how to USE different cryptography methods; - how different protection techniques can stop security threats. Pupils will be able to: - DEVELOP a disaster recovery plan to protect an organisation; - USE different encryption methods; - identify different types of security threats; - DEVELOP security protection techniques to defend an organisation.							Pupils will know: - how to identify different security threats; - how to USE different cryptography methods; - how different protection techniques can stop security threats. Pupils will be able to: - DEVELOP a disaster recovery plan to protect an organisation; - USE different encryption methods; - identify different types of security threats; - DEVELOP security protection techniques to defend an organisation.							Pupils will know: - how to produce a plan to protect an IT system; - how to install anti-virus software and a firewall; - how to configure security settings to prevent unauthorised access. Pupils will be able to: - install protection on to a computer system; - ANALYSE and interpret security events; - to configure a wireless router's security settings.							Pupils will know: - how to produce a plan to protect an IT system; - how to install anti-virus software and a firewall; - how to configure security settings to prevent unauthorised access. Pupils will be able to: - install protection on to a computer system; - ANALYSE and interpret security events.							Pupils will know: - how to produce a plan to protect an IT system; - how to install anti-virus software and a firewall; - how to configure security settings to prevent unauthorised access. Pupils will be able to: - install protection on to a computer system; - ANALYSE and interpret security events.					
Year 13 - Unit 1	Introduction to Computer Science including key terms.	Develop flowcharts using appropriate symbols. Computer Society (BCS) symbols to describe a system or solution.	Interpret common data-handling techniques as they are produced within programming languages to process data.	Interpret common functions provided within programming languages to perform specific tasks to process data.	Interpret common data structures within a computer program to store and process data.	The use of code written using object oriented programming paradigm.	Review the issues and implications of translating code into machine programming languages.	Produce, APPLY and interpret pseudo code statements to describe a system or solution, processes and solve problems.	USE and interpret common functions provided within programming languages to perform specific tasks to process data.	USE and interpret common functions provided within programming languages to perform specific tasks to process data.	USE and interpret common data structures within a computer program to store and process data.	DEVELOP my exam technique by answering short and long style exam questions on programming languages.	Interpret, ANALYSE and EVALUATE the use of code written using object oriented programming paradigms.	Complete a learning review on ASSESS my understanding of programming languages.	Interpret, ANALYSE and EVALUATE the use of code written for web languages.	DEVELOP my exam technique by answering short and long style exam questions on topics covered in unit 1.	DEVELOP my exam technique by answering short and long style exam questions on topics covered in unit 2.	DESCRIBE the purpose and principles of website products.	EXPLAIN the requirements of websites.	EXPLAIN the principles of website design.	IDENTITY media and objects in a website.	EXPLAIN factors affecting website performance.	DEVELOP HTML and CSS skills.	CREATE moodboards and storyboards to store ready made and create original assets for a website.	DEVELOP HTML and CSS skills.	Create a website for a specific need.	Create a website for a specific need.	Create a website for a specific need.	Course Evaluation					
	Pupils will know: - how to DEVELOP flowcharts to describe a system or solution; - how to interpret common functions to perform specific tasks; - how to ANALYSE code written in different programming languages. Pupils will be able to: - DEVELOP pseudocode statements to solve problems; - DEVELOP flow charts to describe a system or solution; - USE data-handling techniques to perform specific techniques; - improve the accuracy and validity of data.							Pupils will know: - how to DEVELOP flowcharts to describe a system or solution; - how to interpret common functions to perform specific tasks; - how to ANALYSE code written in different programming languages. Pupils will be able to: - DEVELOP pseudocode statements to solve problems; - DEVELOP flow charts to describe a system or solution; - USE data-handling techniques to perform specific techniques; - improve the accuracy and validity of data.							Pupils will know: - how to ANALYSE code written in different programming languages; - how to EXPLAIN the purpose and principles of a website; - how to EXPLAIN factors affecting website performance. Pupils will be able to: - EXPLAIN the purpose and principles of a website; - EXPLAIN factors affecting website performance; - IDENTITY media and objects in a website.							Pupils will know: - how to PLAN a website; - how to CREATE a mood board; - how to CREATE a storyboard. Pupils will be able to: - PLAN a website; - CREATE a website in HTML; - manipulate a website using CSS.							Results					