Year 10	Lesson Name	Substantive Knowledge	Disciplinary Knowledge	Tier 3 Vocabulary
Algorithms				
Half Term 1	**Computational Thinking**	Understand the principles of Computational Thinking, specifically, abstraction, decomposition, algorithmic thinking	Students will be able to produce structure diagrams to show, the structure of a problem and the subsections and their links to other subsections.	Computational thinking, abstraction, decomposition, algorithmic thinking, inputs, processes, outputs, structure diagrams, pseudocode, flowcharts, reference language, trace tables, syntax error, logical error, algorithm, decision, terminal, sub program, process, binary search, linear search, bubble sort, merge sort, insertion sort, variables, constants, operators, assignments, sequence, selection, iteration, Boolean operators, arithmetic operators, modulus, quotient, exponentiation
	Searching Algorithms	Students will understand different types of searching algorithms and their purpose: Binary Search, Linear Search	Students will be able to perform a Binary Search and a Linear Search.	
	Sorting Algorithms	Students will understand different types of sorting algorithms and their purpose: Bubble sort Insertion Merge Identify positive and negative interactions online	Students will be able to perform a Bubble sort, and an Insertion Merge	

Developing algorithms using flowcharts	Understand the different flow chart symbols	Create, interpret, correct, complete and refine algorithms using flowcharts	

	Understand arithmetic operators and variables		
Developing algorithms using pseudocode	Define the data types integer, real, Boolean, character, string Understand the different Boolean operators	Write algorithms in pseudocode involving sequence, selection and iteration Be able to use Boolean operators	
Interpret correct complete algorithms	Understanding the purpose of an algorithm Understand how to determine the output of an algorithm Understand different errors in an algorithm	Determine the output of an algorithm Correct errors in an algorithm Create and complete a trace table	

Lesson 10 Unit Assessment			
Reteach	TBC		
Systems Architecture			

	Architecture of the CPU	Understand the purpose of the CPU	Explain the purpose and how the fetch-execute cycle works Explain the purpose of the following registers: MAR MDR PC ACC	Fetch-execute, CPU, ALU (Arithmetic Logic Unit), CU (control unit), cache, registers, Von Neumann architecture, MAR (Memory Address Register), MDR (Memory Data Register), Program Counter, Accumulator, clock speed, cache size, cores, embedded
			Explain what the following components do: ALU CU Cache	systems, memory address, Primary storage, RAM, ROM, virtual memory, volatile, non-volatile, secondary storage, optical, magnetic, solid state, drive, disk, hard disk, floppy disk, tape drive, Blu-ray, DVD, CD, capacity, speed, portability, durability, reliability, cost, storage device, storage media
	14 Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.

	CPU Performance	Understand the function of cache within the CPU Name the following characteristics that affect performance: Clock Speed Cache size Number of cores	Explain how the characteristics affect performance Explain the purpose of embedded systems	
	Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.
Half term 2	**Memory**	Understand the difference between RAM & ROM Understand virtual memory	Describe the difference between RAM & ROM Explain the need for virtual memory	

Secondary Storage	Name the different secondary storage devices: optical, magnetic, solid state Name the characteristics of each storage device: Capacity, speed, portability, durability, reliability, cost	Explain why secondary storage is required Evaluate the different storage	
4 + 5 Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.

6 Revision Lesson	TBC – Based on formative assessments in lessons, homework, quizizz and smart revise as to what the students require		

	7 Unit Assessment			
	8 Reteach	TBC		
	9 Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.
	Programming Fundamentals			
	10 + 11 + 12 Programming Fundamentals	Understand and describe different data types: Integer, real/float, Boolean character, string	Understand and describe different data types: Integer, real/float, Boolean character, string Understand the difference between constants and variables Understand how to assign variables and constants	Variables, constants, operators, inputs, outputs, assignment, sequence, selection, iteration, arithmetic operators, Boolean operators, AND, OR, NOT, ==, !=, <, <=, >, >=, +, -, *, /, MOD, DIV, ^, exponentiation, data types, integer,

		Understand the difference between constants and variables Understand how to assign variables and constants Understand the difference between MOD and DIV	Understand the difference between MOD and DIV	real, Boolean, character, string, casting, string manipulation, file handling, open, read, write, close, records, SQL, arrays, onedimensional array, two-dimensional array, sub program/subroutine, functions, procedures, random numbers, concatenation, slicing, SQL, SELECT, FROM, WHERE.
	13 + 14 Sequence & Selection	Understand what selection is Understand what nested selection is Understand the terms required to create Boolean expressions Understanding the need for random number generator	Use selection Use nested selection Use Boolean expressions Use random number generator	
Half Term 3 (21)	**1 + 2 Iteration**	Understand iteration in an algorithm	Use iteration in an algorithm	

	3 + 4 Array	Understand what a 2d and 3d array are Understand that Spreadsheets have	Be able to use and explain what a 2d and 3d are used for	
		formulas and how to write them		
	5 + 6 Procedures & Functions	Understand the concepts of subroutines Understand parameters to pass data to procedures and functions Understand that subroutines can use local variables	Write simple subroutines (procedures and functions) Use parameters to pass data to procedures and functions Explain why local variables are good practise Use local variables Explain the advantages of functions and procedures	
	7 + 8 Records & files	Understand basic file handling operations: Open, read, write, close	Use basic file handling operations: Open, read, write, close	

	9 Introduction to SQL	Understand the need for SQL and how it works Understand the key criteria and words AND, OR, LIKE, SELECT, FROM WHERE, ORDER BY, *(WILDCARD)	Use SQL including all the criteria and key words	
	10 Revision Lesson	TBC – Based on formative assessments in lessons, homework, quizizz and smart		
		revise as to what the students require		
	11 Assessment			
	12 Reteach	TBC		
	Logic & Language			
	13 + 14 Logic Diagrams & Truth Tables	Understand the purpose of a truth table Understand Logic Gates: AND, OR, NOT	Construct truth tables Interpret the results of truth tables Create, modify and interpret logic circuit diagrams	Defensive design, anticipating misuse, authentication, validation, maintainability, sub programs, naming conventions, indentation, commenting, testing, iterative testing, final/terminal testing,

	15 +16 Defensive Design	Understand how to make maintainable programs including: The use of sub programs, Naming conventions, Indentation, Commenting Describe defensive design considerations: Input validation Anticipating misuse Authentication	Write programs that use: sub programs, Naming conventions, Indentation, Commenting	syntax, syntax error, logic error, test data, normal, boundary, invalid, erroneous, test plan, AND, OR, NOT, truth table, logical operators, logic gates, logic diagrams, conjunction, disjunction, negation, high-level language, low-level language, translators, compiler, interpreter, compiler, interpreter, Integrated Development Environment (IDE), editors, error diagnostics, run-time environment.
	17 + 18 Errors & Testing	Understand the purpose of iterative and final testing	Correct syntax and logic errors Use correct testing data for Normal, Boundary, Invalid, Erroneous	

		Understand the difference between syntax and logic errors Understand different types of test data including: Normal, Boundary, Invalid, Erroneous		
Half Term 4	**1 Revision Lesson**	TBC – Based on formative assessments in lessons, homework, quizizz and smart revise as to what the students require		
	2 Unit Assessment			
	3 Reteach	TBC		
	Data Representation			

4 Units & Binary Numbers	Define the terms bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte and petabyte Understand that data needs to be converted into binary so it can be processed by a computer	Convert positive denary whole numbers (0-255) into 8-bit binary numbers and vice versa	Bit, nibble, byte, kilo, mega, giga, tera, peta, binary, bit depth, sample rate, colour depth, pixel, bit per character, binary shift, shift left, shift right, most significant bit, least significant bit, character set, ASCII, Unicode, metadata, hertz, compression, lossy, lossless
5 Binary arithmetic and hexadecimal	Understand the need to convert whole numbers to hexadecimal and vice versa Understand the need to convert binary to denary and to hexadecimal Understand and explain how overflow errors occur	Convert whole numbers to hexadecimal and vice versa Convert binary to denary and to hexadecimal Add binary numbers	
6 Programming Lesson	To develop python skills	Learn how to write structured programs	

7 + 8 Characters	Understand the use of binary codes to represent characters Understand the term 'character set' Understand why characters sets are required and how	Explain the need for character sets Explain the relationship between the number of bits per character in a character set, and the number of characters that can be represented using: ASCII, Extended ASCII, Unicode	
9 + 10 Images	Understand how bitmap graphics are made up of pixels	Explain how each pixel is represented in binary Explain the need for image metadata	

	Understand the term pixels Understand that the number of bits per pixel determines the number of available colours for an image	Explain the relationship between file size and image resolution	
11 + 12 Sound	Understand how sound is sampled and stored in digital form Understand the terms Sampling, resolution sample rate, and bit depth	Explain how sampling intervals and resolution affect the size of a sound file using the terms: Sample rate Bit depth Explain the trade-off between file size and the quality of playback Be able to represent a short sound file in binary	

	13 Programming Lesson	To develop python skills	Learn how to write structured programs	
	14 + 15 Compression	Understand the terms: compression, lossy, lossless	Explain the need for compression Describe the difference between lossy and lossless compression	
Half Term 5 (18)	**1 Revision Lesson**	TBC – Based on formative assessments in lessons, homework, quizizz and smart revise as to what the students require		
	2 Unit Assessment			
	3	Re-Teach TBC		

	4 Programming Lesson	To develop python skills	Learn how to write structured programs	
	5 Programming Lesson	To develop python skills	Learn how to write structured programs	
	6 Programming Lesson	To develop python skills	Learn how to write structured programs	
	7 Programming Lesson	To develop python skills	Learn how to write structured programs	
	8 Programming Lesson	To develop python skills	Learn how to write structured programs	
	9 Programming Lesson	To develop python skills	Learn how to write structured programs	

	10 Programming Lesson	To develop python skills	Learn how to write structured programs	
	11 Programming Lesson	To develop python skills	Learn how to write structured programs	
	12 Programming Lesson	To develop python skills	Learn how to write structured programs	
	13 Programming Lesson	To develop python skills	Learn how to write structured programs	
	14 Programming Lesson	To develop python skills	Learn how to write structured programs	
	15 Programming Lesson	To develop python skills	Learn how to write structured programs	
	16 Programming Lesson	To develop python skills	Learn how to write structured programs	
	17	Revision lesson for end of year assessment - TBC		
	18	Revision lesson for end of year assessment - TBC		
	Impacts of Digital Technology			

Half Term 6	**1 + 2 + 3 Ethical and cultural issues**	Understand the terms: Ethical Cultural	Discuss the impacts of digital technology on the wider society with specific reference to Ethical and cultural issues	Ethical, cultural, environmental, legislation, manufacture, disposal, upgrade, replace, e-waste, privacy, legal, data protection, computer misuse, copyright, copyright designs and patents act, open source, proprietary, software licence
	4 + 5 + 6 Environmental Issues	Understand the impacts that digital technology has on the environment Understand the term environment	Discuss the impacts of digital technology on the environment including: The impact of manufacture and disposal The impact of upgrading or replacing The impact of e-waste	
	7 Programming Lesson	To develop python skills	Learn how to write structured programs	
	8 Programming Lesson	To develop python skills	Learn how to write structured programs	
	9 Programming Lesson	To develop python skills	Learn how to write structured programs	

	10 + 11 + 12 Legislation & privacy issues	Understand the legislation: Data protection Act 2018 Computer Misuse Act 1990 Copyright Designs & Patents Act 1988 Understand the difference between open source and proprietary software	Discuss the impacts of digital technology on wider society including: Legal issues Privacy issues Describe legislation relevant to Computer Science: The Data Protection Act 2018 Computer Misuse Act 1990 Copyright Designs and Patents Act 1988 Describe the different types of software licenses including open source and proprietary	
	11	How to answer an 8 mark question		
	12	Revision lesson TBC		
	13	**Unit Assessment**		
	14	Re-teach TBC		
	15	Last weeks of term to be planned when details of activities are announced. GCSE RS exam and mocks will have taken place this half term and along with contingency for any missed lessons from the past year.		
	16			
	17			
	18			
	19			
	20			
	21			

Year 11	Lesson Name	Substantive Knowledge	Disciplinary Knowledge	Tier 3 Vocabulary
Networks Connections & Protocols				
Half Term 1	**The Internet and WANs**	Define a WAN, the internet, IP address, DNS, NIC, MAC addressing, packet switching	Describe what the internet is Explain the need for IP addressing of resources on the Internet and explain how this can be facilitated by the role of DNS services Explain the need for Network Interface Cards and the uses of MAC addressing Explain packet switching	LAN, Local Area Network, WAN, Wide Area Network, bandwidth, latency, Wireless access points, routers, switches, NIC, Network Interface Controller/Card, Transmission media, DNS, Domain Name Server, Hosting, The Cloud, Web servers and clients, star network, mesh network, topology, IP address, web server, file server, wired network, wireless network, Ethernet, Wi-Fi, Bluetooth, encryption, IP addressing, MAC
	Programming Lesson	To develop python skills	Learn how to write structured programs	

	Local Area Network	Describe the difference between a Local Area Network and a Wide Area Network Describe star and mesh network topologies Describe routers and switches needed to connect stand-alone computers into a Local Area Network Understand that there are different protocols that are in place to send data across a wired and wireless network	Identify different types of networks and explain why they are either a LAN or WAN Explain why routers and switches/hubs are required to connect a network together Explain the use of Ethernet standards to transmit data over a wired network Explain the concept of virtual networks	addressing, TCP/IP, Transmission Control Protocol/Internet Protocol, FTP, File Transfer Protocol, POP, Post Office Protocol, IMAP, Internet Message Access Protocol, SMTP, Simple Mail Transfer Protocol, layers, IPv4, IPv6, MAC address.

		Understand the term virtual networks		
	Wireless Networking	Understand the term wireless and how the two different connection types: Wi-Fi and Bluetooth Understand the term encryption	Explain how Wi-Fi and Bluetooth work Explain why Wireless Access Points (WAPs) are created Explain the need for encryption in given scenarios	
	Programming Lesson	To develop python skills	Learn how to write structured programs	

Clientserver and peer-topeer networks	Understand the terms clientserver and peer-to-peer networks, Hosting, The Cloud, transmission media Describe what network performance is	Explain the advantages and disadvantages of client-server and peer-to-peer networks Explain the advantages and disadvantages of various transmission media Explain the factors that affect network performance	
Lesson 11 Unit Assessment			
Reteach	TBC		
Network Security & System Software			
Network Threats	Understand the different forms of attacks on networks Understand the threats that are posed by the following attacks: Malware Phishing Social engineering Brute force attacks	Explain each attack Explain the threat and damaged that can be caused by each attack and the potential consequences	Malware, virus, Trojan horse, worm, social engineering, phishing, bruteforce attack, denial of service attack, data interception and theft, SQL injection, penetration testing, anti-malware software, anti-virus software, firewalls, user access levels, passwords, encryption, physical security, operating system,

		Denial of service attacks Data interception and theft SQL injection		user interface, graphical user interface (GUI), command line interface (CLI), memory management, multitasking, peripheral management, drivers, user management, file management, utility software, encryption software, defragmentation, data compression
	Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.
	Preventing Vulnerabilities	Be able to identify the following preventions against network attacks: penetration testing anti-malware software firewalls user access levels passwords encryption physical security	Be able to explain how each prevention method can help protect against different forms of attacks and be able to use the correct prevention for the corresponding attack	

	Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.
Half term 2	**OS**	Understand what an OS is Name different OS	Explain the need for the following functions of an operating system: User interface Memory management and multitasking Peripheral management and drivers User management File management	
	Utility Software	Understand the term Utility software Name the different types of utility software: Encryption, Defragmentation, Compression	Explain the purpose of Encryption, Defragmentation, Compression and why they are required	

Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.

Revision Lesson	TBC – Based on formative assessments in lessons, homework, quizizz and smart revise as to what the students require		
9 Unit Assessment			
Reteach	TBC		

	Programming Lesson	To develop python skills	Learn how to write structured programs	Python Objective 00 & 01 – TIME (Try, Investigate, Make, Evaluate), Students work at own pace through the different workbooks with guidance from the teacher, who tracks each students progress via an excel sheet. Support is given on an individual basis.
	Impacts of Digital Technology			
	Ethical and cultural issues	Understand the terms: Ethical Cultural	Discuss the impacts of digital technology on the wider society with specific reference to Ethical and cultural issues	Ethical, cultural, environmental, legislation, manufacture, disposal, upgrade, replace, e-waste, privacy, legal, data protection, computer

	Environmental Issues	Understand the impacts that digital technology has on the environment Understand the term environment	Discuss the impacts of digital technology on the environment including: The impact of manufacture and disposal The impact of upgrading or replacing The impact of e-waste	misuse, copyright, copyright designs and patents act, open source, proprietary, software licence
	Programming Lesson	To develop python skills	Learn how to write structured programs	

	Legislation & privacy issues	Understand the legislation: Data protection Act 2018 Computer Misuse Act 1990 Copyright Designs & Patents Act 1988 Understand the difference between open source and proprietary software	Discuss the impacts of digital technology on wider society including: Legal issues Privacy issues Describe legislation relevant to Computer Science: The Data Protection Act 2018 Computer Misuse Act 1990 Copyright Designs and Patents Act 1988 Describe the different types of software licences including open source and proprietary	
		How to answer an 8 mark question		
		Revision lesson TBC		
Half term 3	**1**	**Unit Assessment**		
		Reteach tbc		
	Programming Lesson	To develop python skills	Learn how to write structured programs	
		Revision for Mocks		
		Re-teach from Mocks		
	Programming Lesson	To develop python skills	Learn how to write structured programs	
	Trace Tables	Understand the purpose of trace tables	To successfully understand and complete a (numbers of) trace tables	
	Data types	Understand the purpose of data types	To successfully give the correct type of data to given scenarios	

	Test Data	Understand the purpose of test data	To successfully give the correct type of test data and data types to given scenarios	
	Programming Lesson	To develop python skills	Learn how to write structured programs	
	Spot the bug!	To develop coding skills and error finding	To be able to successfully find different types of errors in programs To identify the correct error type and fix the error	
	Refining algorithms	To develop coding skills and exam technique	To be able to successfully be given an algorithm and refine it to create a more slimline version e.g. adding in iteration or repetition	
Half terms 4 + 5 will be revision based on what is required. A new document will be written to cater for the needs of the students after the mocks.				