GCSE EDEXCEL Computer Science – Scheme of Learning

LONG-TERM OVERVIEW

YEAR 10				YEAR 11		
Term	Topics	Assessment		Term	Topics	Assessment

1	• Introduction to the course
	• SLR 1 Systems architecture and storage
	    o 7 lessons
	• SLR 2 System software
	    o 5 lessons
	• Plus 4 dedicated programming lessons
		SLR 1 Student workbook
		SLR 1 End-of-topic test
				1	• SLR 7 Basic programming concepts
					    o 7 lessons
					• SLR 8 Advanced programming concepts
					    o 7 lessons
					• Plus 3 dedicated paper 2 revision lessons
						SLR 7 Student workbook
						SLR 7 End-of-topic test

2	• SLR 2 System software
	    o 2 lessons
	• SLR 3 Data representation – part 1
	    o 9 lessons
	• Plus 6 dedicated programming lessons
		SLR 2 Student workbook
		SLR 2 End-of-topic test
		SLR 3 Student workbook – part 1
		SLR 3 End-of-topic test – part 1
				2	• SLR 8 Advanced programming concepts
					    o 2 lessons
					• SLR 9 Robust and secure programming
					    o 8 lessons
					• Plus 7 dedicated paper 2 revision lessons
						SLR 8 Student workbook
						SLR 8 End-of-topic test
						SLR 9 Student workbook
						SLR 9 End-of-topic test

3	• SLR 3 Data representation – part 2
	    o 7 lessons
	• SLR 4 Computer networks, protocols and layers
	    o 5 lessons
	• Plus 3 dedicated programming lessons
		SLR 3 Student workbook – part 2
		SLR 3 End-of-topic test – part 2
				3	• SLR 10 Algorithms and computational logic
					    o 15 lessons
					• Opportunities to practice programming algorithms throughout the topic.
						SLR 10 End-of-topic test – part 1

4	• SLR 4 Computer networks, protocols and layers
	    o 8 lessons
	• Plus 7 dedicated programming lessons
		SLR 4 Student workbook
		SLR 4 End-of-topic test
				4	• SLR 10 Algorithms and computational logic
					    o 5 lessons
					• SLR 11 Classification of programming languages
					    o 5 lessons
					• Plus 5 dedicated paper 2 revision lessons
						SLR 10 Student workbook
						SLR 10 End-of-topic test – part 2
						SLR 11 Student workbook
						SLR 11 End-of-topic test

5	• SLR 5 Network and cyber security o 9 lessons • Plus 6 dedicated programming lessons	SLR 5 Student workbook SLR 5 End-of-topic test		5	This final term after Easter has been set aside for you to use as you see fit for your students. See SoL below for details of suggested activities.	Past papers Smart Revise app
6	• SLR 6 Ethical, legal and environmental issues o 8 lessons • Plus 10 dedicated programming lessons	SLR 6 Student workbook SLR 6 End-of-topic test				

SHORT-TERM SCHEME OF LEARNING

1. This lesson-by-lesson breakdown is based on the two-week calendar.
2. Programming is taught in standalone lessons but as it is a key skill representing 50% of the marks (Paper 2) this is also delivered alongside other learning and shown in green This should of course be reinforced by regular practice in the student's own time. The final teaching in Year 11
3. Dedicated lesson time for assessment and students responding to feedback/making improvements are shown in blue.
4. **Along** with each lesson's learning outcomes there is also a key question that each lesson addresses. These key questions appear in red.
5. Each topic in this SoL is presented as part of a workbook so that students can track their learning against the components of each exam paper to support independent learning and revision. This will be accessible through Google Classroom.

Resources:

Lesson overview PowerPoints
One for each lesson to be displayed at the front of the classroom to provide structure to the lesson and shared on Google Classroom for students.

End of topic test
Written using exam style questions.
All tests are out of 20 so easy comparisons can be made between different topics.
Full answers provided to assist students with identifying lost marks and areas for improvement.

Student workbook
This is the main work students complete as they progress through a topic. Includes an opportunity for assessment and feedback. Includes a RAG rating self-assessment page.
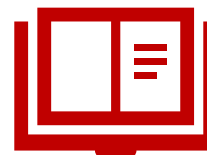
Model workbook (answers)
A grade 9 model answer version of the blank student workbook to be shared with students.
Doubles up as a knowledge organiser to hand out to students when needed.

▶ YouTube Assessment with Craig 'n' Dave - (GCSE) – Videos cover each element and allow for flipped learning in home learning activities Videos are available through YouTube or can be shared on Classroom

Home Learning – Watch 2 videos and make notes about it for class learning and discussion. These are generally 5-15 mins long. Spend 20 mins on a set programming task. Up to 60 mins per week of home learning.

YEAR 10 - TERM 1

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
1	1.0 Introduction Lesson - About the course	• Understand the course structure and how you will be taught and assessed on this subject. • Understand what the flipped classroom approach to learning is and what it means for homework. • Understand what a computer is.	What is a computer?	Introductory activities (Slides 8-11) Watching introduction to programming concepts video as a class and allow students to ask questions.	Video(s) from: "Topic 3A Hardware" • 🔗 Common CPU components and their function • 🔗 Von Neumann architecture	
	SLR 1 Systems architecture					
2	1.1 Lesson 1 - The von Neumann architecture	• Understand what the CPU of a computer does. • Know what major components of the CPU are. • Begin learning to program in Python. KEY QUESTION: What is the "architecture" of a CPU?	Who was John von Neumann? What are the characteristics of the von Neumann computer architecture?	SLR 1 Workbook Complete slides 2 and 3 Complete the exercise 1.1 Programming introduction	Video(s) from "Topic 3A Hardware": • 🔗 The fetch-execute cycle	**Slides 75-90** Von Neumann architecture, RAM, CPU, Control unit, ALU, Register, Clock, Address bus, Data bus, Control bus, Fetch-decode-execute cycle, Secondary storage, Magnetic storage, Optical storage, Solid state storage, Embedded system
3	1.2 Lesson 2 - Fetch-execute cycle	• Understand what the CPU of a computer does. • Know what major components of the CPU are. • Know the stages of the fetch, execute cycle. • Begin learning to program in Python. KEY QUESTION: How does the fetch-decode-execute cycle work?	A train on a child's toy railway takes one second to travel on each segment of track. How long does it take the train to travel around the inside track? How long does it take the train to travel around the outside track? What is the relevance of this illustration to caching? To what extent is this illustration of caching accurate?	SLR 1 Workbook Complete slides 3-7 Every programming language uses **"keywords"**. Certain words which have a special meaning in that language. 1. Use the internet to find a list of keywords for the programming	Video(s) from "Topic 3A Hardware": • 🔗 The need for secondary storage • 🔗 Common types of storage	

				language you will be taught. 2. Use the site wordclouds.com to create a word cloud out of the keywords. **Extension task** 3. Can you find out what some of these keywords do?	
4	1.3 Lesson 3 - Common types of storage	• Understand the need for secondary storage. • Know the common types of storage. • Know the characteristics of storage devices. • Understand how to program in Python. KEY QUESTION: What are the differences between secondary storage devices?	A digital camera uses an SD card to store images. How much would it cost me to upgrade my memory card from 32GB? A friend has a Nintendo Switch console. They need a memory card to store more games. What card would you recommend and how much would it cost?	SLR 1 Workbook Complete slides 8-11 Begin programming in Python: • Complete the tasks. • Try the challenges.	Video(s) from "Topic 3A Hardware": • 🔗 Suitable storage devices and storage media
5	1.4 Lesson 4 - Applications of storage	• Understand the suitability of storage devices for given applications. • Understand the advantages and disadvantages of devices based on their characteristics. • Understand how to program in Python. KEY QUESTION: What features of secondary storage make devices suitable for different situations?	What storage media would you choose for:	SLR 1 Workbook Complete slides 12-15 Begin programming in Python: • Complete the tasks. • Try the challenges.	Video(s) from "Topic 3A Hardware": • 🔗 Embedded systems
6	1.5 Lesson 5 - Embedded systems	• Know what is meant by the term "embedded system." • Know several examples of embedded systems.	In what ways is this washing machine a computer?	SLR 1 Workbook Complete slide 16	Revise what you have learnt in this topic

		• Know what a microcontroller is • Understand how to program in Python. KEY QUESTION: What are embedded systems, and what are their characteristics?		Begin programming in Python: • Complete the tasks. • Try the challenges.		
7 to 10	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various		
11	SLR 1 – End-of-topic test	End-of-topic test		SLR 1 - Test		
12	SLR 1 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	Video(s) from "Topic 3B Software and programming languages": • 🔗 The purpose and functionality of operating systems	
SLR 2 System software						
13	2.1 Lesson 1 - The purpose and functionality of operating systems	• Understand the need for operating systems. • Understand the functions of operating systems. • Understand how to program in Python. KEY QUESTION: Why does your computer need an operating system?	How many operating systems can you name?	SLR 2 Workbook Complete slides 2 and 3 Programming tasks and challenges	Video(s) from "Topic 3B Software and programming languages": • 🔗 Operating systems part 1	Slides 91-102 Operating system, File management, Process management, Peripheral management, User management, Utility software, File repair, Backup, Data compression, Disc defragmentation, Anti-malware
14	2.2 Lesson 2 - Operating systems part 1 (process	• Understand how the operating system handles the management of the processor(s).	Below is an illustration of the RAM of a computer. Each square is a page of memory that can hold a	SLR 2 Workbook Complete slides 4-7 Programming		

	and peripheral management)	• Understand how the operating system handles. the management of the memory. • Understand how scheduling works. • Understand what a device driver is. • Understand how to program in Python. KEY QUESTION: How does a computer manage having lots of programs open and running at the same time?	fragment of a program. The memory always fills up left to right, but fragments are never moved. Show the state of the memory after this sequence of processes:		
15	2.3 Lesson 3 - Operating systems part 2 (virtual memory)	• Understand the need for virtual memory. • Understand how to program. KEY QUESTION: How does virtual memory work?	Why would the RAM be faster than the hard disk?	SLR 2 Workbook Complete slides 8-12 Programming	Video(s) from "Topic 3B Software and programming languages": • 🔗 Operating systems part 2
16	2.4 Lesson 4 - Operating systems part 3 (file and user management)	• Understand how the operating system handles file management. • Understand how the operating system handles user permissions. • Understand how the operating system handles user management. • Understand how to program in Python. KEY QUESTION: What features does an operating system give users?	Find out what is meant by the term "roaming profile" in the context of an operating system. What are the advantages and disadvantages of roaming profiles?	SLR 2 Workbook Complete slides 13-17 Programming	Video(s) from "Topic 3B Software and programming languages": • 🔗 Utility software
17	2.5 Lesson 5 - Utility system software	• Understand anti-malware utilities. • Understand defragmentation utilities. • Understand data compression utilities. • Understand backup utilities. KEY QUESTION: What is the purpose of utility software?	What are these catchphrases?	SLR 2 Workbook Complete slide 18 Complete any other areas of the workbook not yet completed.	Revise what you have learnt in this topic

YEAR 10 - TERM 2						
	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
18	SLR 2 – End-of-topic test	End-of-topic test		SLR 2 - Test		
19	SLR 2 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings		
20 to 22	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.	Various		Video(s) from "Topic 2A Binary": • 🔗 Bit patterns	
SLR 3 Data representation – part 1						
23	3.1 Lesson 1 - How binary represents all data and instructions	• Understand why computers systems use binary to store data. • Understand that computers use binary to represent many different types of data and instructions. • Understand how to program in Python. KEY QUESTION: Why is it impossible to say what '100101101' represents without more information?	Hard drives are considered old technology today as people replace their HDD with SSD, but could they instead actually be the future technology for mass storage of data? Research: "single atom magnets" What is your conclusion?	SLR 3 Part 1 Workbook Complete slides 2 and 3 Programming	Video(s) from "Topic 2A Binary": • 🔗 Number bases • 🔗 Why hexadecimal is used in computer science	Slides 47-54 and 65-71 Base-2 binary, bit pattern, Unsigned integers, Two's complement signed integers, Base-10 denary, Binary shifts, Overflow, Base-16 hexadecimal, Bit, Nibble, Byte, Kibibyte, Mebibyte, Gibibyte, Tebibyte
24	3.2 Lesson 2 - Number bases	• Understand the following number bases: o Denary (base 10) o Binary (base 2) o Hexadecimal (base 16) • Explain why hexadecimal is often used in computer science. • Understand how binary can be used to represent whole numbers.	Throughout history civilisations have used all sorts of different base number systems. Using the internet see what you can find.	SLR 3 Part 1 Workbook Complete slides 4-7 Programming	Video(s) from "Topic 2A Binary": • 🔗 Signed integers using two's complement	

		• Under how hexadecimal can be used to represent whole numbers. KEY QUESTION: What is meant by the term 'number-base'?			
25	3.3 Lesson 3 - Two's complement representation	• Understand how Two's complement can be used to represent both positive and negative integers in binary. • Understand how to program in Python. KEY QUESTION: How can we represent negative whole numbers in binary?	Two's complement is the method you need to know for representing negative numbers in binary. However, it is not the only method. Do some quick research and see what other methods you can find!	SLR 3 Part 1 Workbook Complete slide 8 Programming	Video(s) from "Topic 2A Binary": • 🔗 Converting between denary and 8-bit binary • 🔗 Converting between denary and two-digit hexadecimal
26	3.4 Lesson 4 - Base number conversions	• Know how to convert denary whole numbers (0 to 255 and -128 to +127) into 8-bit binary numbers and vice versa. • Know how to convert binary whole numbers (0 to 255 and -128 to +127) into hexadecimal. • Know how to convert denary whole numbers (0 to 255 and -128 to +127) into hexadecimal. • Understand how to program in Python. KEY QUESTION: Why do we use hexadecimal to represent information?	Have you ever thought about why our number system is base-10? What would be your answer?	SLR 3 Part 1 Workbook Complete slides 9-12 Programming	Video(s) from "Topic 2A Binary": • 🔗 Adding two 8-bit binary integers • 🔗 Binary shifts
27	3.5 Lesson 5 - Binary addition and shifts	• Know how to add together up to two 8-bit binary numbers. • Know how to perform a left and right logical binary shift. • Know how to perform a left and right arithmetic binary shift.	What are the uses of hexadecimal in computer science? Can you research three?	SLR 3 Part 1 Workbook Complete slides 13-15 Programming	

		• Understand what binary shift achieves and their limitations. What can happen to the most significant bit when you add two binary numbers together?			
28	3.6 Lesson 6 - Catch-up lesson	• Complete any outstanding work. KEY QUESTION: What is the relationship between denary, binary and hexadecimal?	We are all familiar with terms like: • Thousands • Millions • Billions • Trillions But are you aware there are official terms for much bigger numbers? See what you can find!	Complete any outstanding work Programming	Video(s) from "Topic 2B Binary": • 🔗 The units of data storage
29	3.7 Lesson 7 - Units	• Understand what is meant by the terms bit, nibble, byte, kibibyte, mebibyte, gibibyte and tebibyte. • Know how to represent the capacity of data storage using these units and be able to convert between them. KEY QUESTION: Why is data stored in binary?	A Blu-Ray of 1 minute at HDTV quality requires 540 mebibytes of storage. How many gibibytes would be required to store a two-hour film?	SLR 3 Part 1 Workbook Complete slides 16-19 Programming	Revise what you have learnt in this unit.
30 to 32	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various	Revise what you have learnt so far in this unit.
33	SLR 3 Part 1 – End-of-topic test	End-of-topic test		SLR 3 Part 1 - Test	
34	SLR 3 Part 1 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	Video(s) from "Topic 2B Data representation and compression": • 🔗 Representing characters and character sets

YEAR 10 - TERM 3

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
	SLR 3 Data representation – part 2					
35	3.8 Lesson 8 - Character sets	• Understand that all data must be represented in binary numbers, including text. • Know what is meant by the term "character set". • Understand the relationship between the number of bits in the character set and the number of characters that can be represented. • Know two common character sets: ASCII and Unicode. KEY QUESTION: How does a computer store characters, and what are the implications for the number of bits used?	How many binary combinations are there with: 2 bits? 4 bits? 8 bits?	SLR 3 Part 2 Workbook Complete slides 2-4 Programming	Video(s) from "Topic 2B Data representation and compression": • 🔗 Representing images	Slides 55-64 and 72-74 7-bit ASCII, Bitmap, Pixel, Resolution, Colour depth, Analogue sound, Amplitude, Sample rate, Bit depth, Sample interval, Data compression, Lossless compression, Lossy compression
36	3.9 Lesson 9 - Bitmaps	• Understand how an image is represented as a series of pixels represented in binary. • Explain what the terms "ppi" and "colour depth" mean. • Explain how to calculate the physical size of a bitmap image. • Understand the effect of colour depth and pixel dimensions on the size of an image file. • Be able to covert binary data into a simple image and vice versa. KEY QUESTION: How does a computer store graphics and what are the implications for image size and resolution?	Find out what colours these hexadecimal numbers represent: FF0000 00FF00 0000FF What does that tell you about what all colours on a computer are made up of?	SLR 3 Part 2 Workbook Complete slides 5-9 Programming		

37	3.10 Lesson 10 - Catch-up lesson	• Complete any outstanding work. KEY QUESTION: How do you calculate the size of a bitmap image?	Televisions are made up of pixels just like bitmap images. 4K TV's are now quite common with 8K already available. How many pixels are there in a 4K and 8K TV image?	Complete any outstanding work Programming	Video(s) from "Topic 2B Data representation and compression": • 🔗 Representing sound • 🔗 Limitations of binary representation
38	3.11 Lesson 11 - Sound	• Understand how sound can be sampled and stored in digital form. • Understand the difference between analogue and digital sound. • Understand how sampling rate and bit depth affect the size of a sound file and the quality of its playback. • Understand the limitations of binary represent for character, image and sound files. • Understand how to construct expressions for calculating file sizes. KEY QUESTION: How do computers store sound and what are the implications for sample rate, duration and bit depth?	Describe what is happening in this illustration:	SLR 3 Part 2 Workbook Complete slides 10-15 Programming	Video(s) from "Topic 2B Data representation and compression": • 🔗 Compression
39	3.12 Lesson 12 - Compression	• Know why data is often compressed for transfer and storage. • Understand the difference between lossy and lossless compression. • Know why some types of data are only suitable for one type of compression. KEY QUESTION: Where is compression used and why?	Using the metadata and data below, what is the paragraph of text?	SLR 3 Part 2 Workbook Complete slides 16-19 Programming	Revise what you have learnt this unit ready for Test 2.
40 to 42	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various	Revise what you have learnt in this unit.
43	SLR 3 Part 2 – End-of-topic test	End-of-topic test		SLR 3 Part 2 - Test	

44	SLR 3 Part 2 – Action	Action/Response lessons	Chance for students to respond to feedback, improve workshop, correct misunderstandings	Video(s) from "Topic 4 Networks and network security": • 🔗 Types of networks	
SLR 4 Computer networks, protocols and layers					
45	4.1 Lesson 1 - Types of networks	• Know what is meant by a 'computer network'. • Know the different types of networks: LAN and WAN. • Understand the advantages of networking. • Understand the disadvantages of networking. • Understand how to program in Python. **KEY QUESTION:** What are the characteristics of LANs and WANs?	Wide area networks have enabled email and social networking to become a major part of our society today. To what extent have these tools changed our society for the better?	SLR 4 Workbook Complete slides 2 and 3 Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 How the internet is structured	**Slides 109-139** Network, LAN, WAN, The internet, IP address, Router, Wired network, Wireless network, speed, range, latency, bandwidth, Network speed, Protocol, Ethernet, Wi-Fi, TCP/IP, HTTP, HTTPS, FTP, POP3, SMTP, IMAP, Application layer, Transport layer, internet layer, Link layer, Network topology, Bus network, Star network, Mesh network
46	4.2 Lesson 2 - The internet	• Understand what the internet is. • Understand what IP address is. • Understand what routers are and why they are important. • Understand what packets are and what sort of information packet headers contain. • Understand how to program in Python. **KEY QUESTION:** How does the internet work?	Where did "the internet" come from?	SLR 4 Workbook Complete slides 4 and 5 Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 Wired and wireless networks and performance	
47	4.3 Lesson 3 - Wired and wireless networks	• Understand that Ethernet is a wired method of connection. • Understand that Wi-Fi, Bluetooth, RFID/NFC and ZigBee are wireless methods of connection.	A small business is moving into a new premises. They are going to have a small client-server local area network,	SLR 4 Workbook Complete slides 6 and 7 Programming		

		• Learn how to program in Python. KEY QUESTION: Which is better, a wired or wireless network?	wired and wireless connected to the internet. List 6 items of hardware they will need.		
48	4.4 Lesson 4 - Wired and wireless networks	• Understand the benefits and drawbacks of wired versus wireless connections. • Be able to recommend a connection type for a given scenario. • Learn how to program in Python. KEY QUESTION: What is the difference between Wi-Fi, Bluetooth, NFC and ZigBee?	How is the internet connected across an ocean? What media would be best to use?	SLR 4 Workbook Complete slides 8 and 9 Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 Network speeds
49	4.5 Lesson 5 - Network speeds	• Know what factors affect the performance of networks. • Understand how to construct expressions involving file size, transmission speed and time. • Understand how to program. KEY QUESTION: What can affect the performance of a network?	The school network also includes several CCTV cameras that record video images to the file server. The Headteacher wants to add more cameras on the outside of the school building for security. What should the network manager consider?	SLR 4 Workbook Complete slides 10 and 11 Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 Common protocols

YEAR 10 - TERM 4

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
SLR 4 Computer networks, protocols and layers						
50	4.6 Lesson 6 - Common protocols	• Understand the importance of protocols. • Understand the 9 common protocols and what they are used for. • Understand how to program. KEY QUESTION: What are protocols and why are they important?	Using the letters below, identify all the networking protocol acronyms.	SLR 4 Workbook Complete slide 12 Programming		**Slides 109-139** Network, LAN, WAN, The internet, IP address, Router, Wired network, Wireless network, speed, range, latency, bandwidth, Network speed, Protocol, Ethernet, Wi-Fi, TCP/IP, HTTP, HTTPS, FTP, POP3, SMTP, IMAP, Application layer, Transport layer, internet layer, Link layer, Network topology, Bus network, Star network, Mesh network
51	4.7 Lesson 7 – Catch-up lesson	• Complete any outstanding work to this point. KEY QUESTION: What hardware is required to connect to the internet?	What is the difference between a switch and a router?	Complete any outstanding work Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 TCP-IP protocol model	
52	4.8 Lesson 8 - The four-layer TCP IP protocol model	• Know why protocols are layered. • Understand the four layers that make up the TCP/IP protocol model. • Understand which common network protocols from last Lesson operate at each layer. • Understand how to program in Python. KEY QUESTION: What are the benefits of layering protocols?	Using the fillings below, create two different burgers using one item from each category: Category 1: sesame seed bun, plain bun, ciabatta bun Category 2: beef, vegetarian beef, chicken Category 3: Cheddar cheese, red Leicester, Roquefort Category 4: Bacon, egg, chorizo, tomato Category 5: Lettuce, onions Category 6: Tomato sauce, mayonnaise, BBQ sauce	SLR 4 Workbook Complete slide 13 Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 Network topologies	
53	4.9 Lesson 9 - Star and bus network topologies	• Know what a star network is. • Know what a bus network is. • Know the advantages and disadvantages of star and bus networks.	Using the internet, identify the 3 types of address shown here: 180.17.255.1 2001:0DB8:AC10:FE01:0000:0000:0000:0000 00-15-E9-2B-99-3C	SLR 4 Workbook Complete slides 14-16 Programming		

		• Understand how to program in Python. KEY QUESTION: Why is a star network better than a bus network?			
54	4.10 Lesson 10 - Mesh network topologies	• Know what a mesh network is. • Know the advantages and disadvantages of a mesh network. • Understand the difference between a full and partial mesh network. • Understand how to program in Python. KEY QUESTION: Why is a mesh network better than a star network?	Research: 802.11b, 802.11g, 802.11n and 802.11ac • When were these protocols adopted? • What is the maximum data transfer speed? • Which band do they operate in? • Are there any other advances in technology from previous protocols?	SLR 4 Workbook Complete slides 17-20 Programming	
55	4.11 Lesson 11 – Catch-up lesson	• Complete any outstanding work to this point. KEY QUESTION: Is the internet and the Word Wide Web the same thing?	The internet is not owned or governed by any single country or government. Instead it is governed by several internal organisations, see if you can find out what some of their names are.	Complete any outstanding work Programming	
56 & 57	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various	
58	SLR 4 – End-of-topic test, part 1 and 2	End-of-topic test		SLR 4 Part 1and2 - Test	Revise what you have learnt in this topic.
59	SLR 4 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	

| 60 to 64 | Independent programming | Gain experience in practical programming
Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials. | Various | Video(s) from "Topic 4 Networks and network security":
• 🔗 Importance of network security | |

GCSE EDEXCEL Computer Science – Scheme of Learning

YEAR 10 - TERM 5

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
SLR 5 Network and cyber security						
65	5.1 Lesson 1 - Network security	• Define the term network security. • Describe the main purpose of network security. • Understand how to program in Python. KEY QUESTION: What are the threats to devices and computers?	ILOVEYOU was the most damaging malware event of all time gaining an entry into the Guinness World Records in 2000. What was it? What did it do? How are people a 'weak point' in this example?	SLR 5 Workbook Complete slides 2 and 3 Programming	Video(s) from "Topic 4 Networks and network security": • 🔗 Ways of identifying network vulnerabilities • 🔗 Methods of protecting networks	**Slides 140-146, 167-177** Network security, Network vulnerabilities, Penetration testing, Ethical hacking, Access control, Physical security, Firewall, Malware, Viruses, Worms, Trojans, Ransomware, Key loggers, Social engineering, Cyber security, Encryption, Acceptable use policy, Backup and recovery procedures
66	5.2 Lesson 2 - Methods of protecting networks	• Explain what penetration testing is and what it is used for. • Explain the difference between black-box and white-box penetration testing. • Understand what an "ethical hacker" is. • Understand how to program in Python. KEY QUESTION: What is the difference between white box and black box penetration testing?	Use the internet to find out what the difference been a white-hat, grey-hat and black-hat hacker is.	SLR 5 Workbook Complete slides 4 and 5 Programming	Video(s) from "Topic 5 Networks and network security": • 🔗 Malware • 🔗 How hackers exploit technical vulnerabilities	
67	5.3 Lesson 3 - Malware (viruses, worms, trojans)	• Define the term malware. • Describe what malware is and how it can be protected against. • Describe the following form of malware: Virus, Worms, Trojans. • Understand how to program in Python. KEY QUESTION: What effect do different malware attacks have on your computer?	Identify all the potential threats to network security in this image.	SLR 5 Workbook Complete slides 6-10 Programming		

68	5.4 Lesson 4 - Malware (ransomware, keyloggers)	• Define the term malware. • Describe what malware is and how it can be protected against. • Describe the following form of malware: Ransomware, Keylogger. • Understand how to program in Python. KEY QUESTION: What effect do different malware attacks have on your computer?	Much of America's internet was brought down on 21st October 2016. This included Netflix, Twitter, Spotify, Reddit, CNN, PayPal, Pinterest, Fox News, The Guardian, The New York Times and the Wall Street Journal. What happened?	SLR 5 Workbook Complete slides 11 and 12 Programming	Video(s) from "Topic 5 Networks and network security": • 🔗 Social engineering
69	5.5 Lesson 5 - Social engineering (pretexting, phishing)	• Define the term social engineering. • Describe what social engineering is and how it can be protected against. • Explain the following form of social engineering: Pretexting, Phishing. • Understand how to program in Python. KEY QUESTION: How is a phishing and pretexting attack carried out?	How can you tell this email is phishing?	SLR 5 Workbook Complete slides 13 and 14 Programming	
70	5.6 Lesson 6 - Social engineering (baiting, quid pro quo)	• Define the term social engineering. • Describe what social engineering is and how it can be protected against. • Explain the following form of social engineering: Baiting, Quid pro quo • Understand how to program in Python. KEY QUESTION: What is meant by shouldering or "shoulder surfing"?	Imagine you bank with TrustedBank, What should you do if you receive this email?	SLR 5 Workbook Complete slides 15-17 Programming	Video(s) from "Topic 5 Networks and network security": • 🔗 Protecting digital systems and data

71	5.7 Lesson 7 - Methods of protecting digital systems and data	• Understand and be able to explain a range of security measures, including:
		  1. Anti-malware
		  2. Encryption
		  3. Acceptable use policies
		  4. Backup and recovery procedures
			How long would it take a computer to brute force crack your password?
			Check it at:
			howsecureismypassword.net
				SLR 5 Workbook Complete slides 18
				
				Complete any other areas of your hacker's	

		• Understand how to program in Python. KEY QUESTION: What are some of the security measures which can be used to protect digital systems and data?	How could you improve your password strength?	handbook you have not had a chance to finish Programming	
72 to 74	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various	Revise what you have learnt in this topic.
75	SLR 5 – End-of-topic test	End-of-topic test		SLR 5 - Test	
76	SLR 5 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	
77 to 79	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various	Video(s) from "Topic 5 Issues and impact": • 🔗 Environmental impact of computer science

GCSE EDEXCEL Computer Science – Scheme of Learning

YEAR 10 - TERM 6

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
SLR 6 Ethical, legal and environmental issues						

80	6.1 Lesson 1 - Environmental issues	• Understand the environmental impact of computers in terms of: Manufacturing
		  • Use
		  • Disposal
		  • Recycling
		KEY QUESTION:
		What is the environmental impact of computing?
			Which 20 elements are used in the manufacturing of computers?
			How many can you guess correctly?
			Highlight the ones most hazardous to humans for double points.
				SLR 6 Workbook Complete slides 2-3
				
				Spend 10-15 minutes playing the REVISE-IT game
					Video(s) from "Topic 5 Issues and impact":
					• 🔗 Ethical and legal issues with collection of personal data
						Slides 147-166
						Environmental issues, Ethical issues, Legal issues, Personal data, Privacy issues, Ownership, Consent, Data Protection Act (2018), Computer Misuse Act (1990), Artificial intelligence, Machine learning, Robotics, Accountability, Algorithmic bias, Legal liability, Intellectual property, Copyright, Patents, Trademarks, Licencing

81	6.2 Lesson 2 - Ethical and legal privacy issues (general)	• Know a range of things to consider beyond development when implementing new computer systems. • Understand at least one ethical issue of digital technology. KEY QUESTION: What are the ethical issues of computing?	To what extent is copying software, music and motion picture files really a form of stealing? Make your own mind up, and then look at the comments in this online debate: www.debate.org/opinions/ should-piracy-be-legal What are the arguments for and against piracy?	SLR 6 Workbook Complete slides 4-7 Spend 10-15 minutes playing the REVISE-IT game		

82	6.3 Lesson 3 - Ethical and legal privacy issues (legislation)	• Know the principles of the Acts of Parliament:
		  • Data Protection Act 2018
		  • Computer Misuse Act 1990
		  • Privacy and Electronics Communication Regulation 2003
		• Understand some of the legal impacts of computer science and its related technologies.
		KEY QUESTION:
			1. I know a teacher username and password. I use this to access the school information system to change the grades on my report. Am I breaking the law?
			2. I copy a picture from the internet to use in my new book that I intend to sell
				SLR 6 Workbook Complete slides 8-10
				
				Spend 10-15 minutes playing the REVISE-IT game
					Video(s) from "Topic 5 Issues and impact":
					• 🔗 Other ethical and legal issues	

		What does the legislation for computing prohibit?	online. Am I breaking the law? 3. I want to know what the latest traffic improvement scheme in town cost to put in place. Can I find out?		

83	6.4 Lesson 4 - AI, machine learning and robotics	• Understand the difference between:
		  • Artificial intelligence
		  • Machine learning
		  • Robotics
		• Understand the concepts of Accountability, Safety, Algorithmic bias and Legal liability in relation to the above topics.
		KEY QUESTION:
		Are artificial intelligence, machine learning and robotics the same thing?
			To what extent can you maintain your privacy on social networking?
			What are the potential problems of public profiles?
				SLR 6 Workbook Complete slides 11-16
				
				Spend 10-15 minutes playing the REVISE-IT game
					Video(s) from "Topic 5 Issues and impact":
					• 🔗 Intellectual property protection

84	6.5 Lesson 5 - Intellectual property protection	• Understand the term intellectual property.
		• Understand the following methods of protecting intellectual property:
		  • Copyright
		  • Patents
		  • Licensing
		  • Trademarks
		• Understand the difference between open source and proprietary software
		KEY QUESTION:
		What methods are available to help you protect your intellectual property?
			To prepare for a school parents evening, students take a blank appointment sheet to each teacher, and book a time slot for their parent/carer to see a teacher. Once complete, they give this sheet to their parent/carer.
			The school is going to introduce an online booking system instead, where parents can book their own appointments.
			Identify 3 impacts for each of the 3 stakeholders: students, parents and teachers.
				SLR 6 Workbook Complete slides 17-18
				
				Spend 10-15 minutes playing the REVISE-IT game	

85	6.6 Lesson 6 – Catch-up lesson	• Reinforce and consolidate your understanding of the ethical, legal and environmental issues and risks of digital technology.	An electricity supplier is changing their practice of reading customer usage meters.	SLR 6 Workbook Finish slides 2-18	

		KEY QUESTION: What are some of the moral and legal considerations behind autonomous self-drive cars?	Instead of estimating bills and sending someone to read the meter at houses once a year, the company are investing in "smart meters". These meters send the usage data via the internet to the electricity supplier automatically once a month. Identify the key stakeholders and state the impact on them.	Spend 10-15 minutes playing the REVISE-IT game	
86 & 87	Independent programming	Gain experience in practical programming Use our T.I.M.E workbooks, Programming challenges and Defold games tutorials.		Various	Revise what you have learnt in this topic.
88	SLR 6 – End-of-topic test	End-of-topic test		SLR 6 - Test	
89	SLR 6 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	
90 91 92 93 94 95 96 97	Text adventure game All the resources needed for this section are in the folder "Text-based adventure game (Telium)" These resources can be used in many ways. If you are following our delivery plan, we have set aside these dedicated lessons in the final term of year 10 for your students to attempt an extended text-based space adventure game. By this point, students should have a fair amount of experience in programming – this is a nice exercise to bring all the concepts together in an extended exercise. There is a detailed teacher notes file in the folder above named "+ Telium – Teacher notes (README)". We have also provided a PDF workbook for students to use throughout this section. All the coded solutions are also provided for your reference.				

GCSE EDEXCEL Computer Science – Scheme of Learning

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
YEAR 11 - TERM 1						
SLR 7 Basic programming concepts						
					Video(s) from "Topic 6B Programming fundamentals": • 🔗 The use of data types and casting "Topic 1A Computational thinking and algorithms": • 🔗 Data structures and arrays	
1	7.1 Lesson 1 - Data types, structures and casting	• Know the different variable data types. • Know how to convert from one data type to another. • Understand the difference between arrays and records. • Know how to create, append to, and delete from a list. KEY QUESTION: Why are numbers sometimes stored as strings?	Code breaker. What is the code to open the safe?	SLR 7 Workbook Complete slides 2-6 Continue working on any programs you have yet to get working. Or improve/extend any of your previous programs.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 The common arithmetic and comparison operators • 🔗 The common Boolean operators "Topic 6B Programming fundamentals": • 🔗 Using arithmetic, comparison and logic operators in programs	Slides 3, 5-36 Subprogram, Flowchart, Pseudocode, Sequence, Selection, Count-controlled repetition, Condition-controlled repetition, Program inputs, Program processes, Program outputs, Variables, Constants, One-dimensional data structures, Two-dimensional data structures, Strings, Records, Arrays, +, -, *, /, //, %, **, ==, !=, <, >, <=, >=, AND, OR, NOT
2	7.2 Lesson 2 - Operators	• Know the arithmetic operators. • Know the logical operators. • Know the comparison operators. KEY QUESTION: Why are logical operators so useful when writing selection and iteration statements?	Solve the logic puzzle presented to you on paper. Average time to solve this puzzle: Two minutes.	SLR 7 Workbook Complete slides 7 and 8 Continue working on any programs you have yet to get working. Or improve/extend any of your previous programs.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 The use of variables, constants and assignments • 🔗 User input and display output • 🔗 Formatting strings	

3	7.3 Lesson 3 - variables, constants, inputs, outputs and assignments	• Understand what a variable is and when to use one. • Understand what a constant is and when to use one. • Understand how to get input from the keyboard. • Understand how to output data to the screen. • Understand what assignment is. • Understand how to customise string output. KEY QUESTION: What terms are associated with programming?	Use the internet to look up some of the first programming languages ever written. How far back can you go?	SLR 7 Workbook Complete slides 9 and 10 Continue working on any programs you have yet to get working. Or improve/extend any of your previous programs.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 The use of the three basic programming constructs	
4	7.4 Lesson 4 - Basic programming constructs	• Know the 3 programming constructs: 1. Sequence 2. Selection 3. Iteration/Repetition • Understand the use of nested selection and iteration structures. KEY QUESTION: What are the three basic programming constructs and how do they differ?	Constants and variables both store data for a program. The concepts are very similar. Why should a programmer use a constant in their code instead of a variable?	SLR 7 Workbook Complete slides 11 and 12 Using code snippets from slide 9, 11 and 12 of the SLR 7 workbook, write a program that: 1. Suggests 3 usernames for a new user 2. Asks for their choice of username. 3. Rejects any username entered that is less than 4 or more than 12 characters long.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Introduction to subprograms	
5	7.5 Lesson 5 - Introduction to subprograms	• Understand that programs can be structured using procedures and functions.	Find the error in this python program.	SLR 7 Workbook Complete slides 13		

		KEY QUESTION: Why are sub programs used?		Continue to work on the username program		
6 & 7	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Revise what you have learnt in this topic.	
8	SLR 7 – End-of-topic test	End-of-topic test		SLR 7 - Test		
9	SLR 7 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings		
10	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Video(s) from "Topic 6B Programming fundamentals": • 🔗 The use of basic string manipulation	
SLR 8 Advanced programming concepts						
11	8.1 Lesson 1 - String manipulation	• Understand and be able to use a range of string manipulation functions. KEY QUESTION: What are some of the ways we can manipulate string variables?	Below is some code presented in Python Basic. What ends up being assigned to the variable x?	SLR 8 Workbook Complete slides 2-3 Practice using the various string manipulation techniques you have learnt in a variety of small programs.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 The Math and Time module	**Slides 183-195, 203-209** Initialisation, Assignment, Parameters, Primitive Data type, Integer, Real, Boolean, Char, String manipulation, File handling: Open, Read, Write, Close, Procedure, Function, Parameters, Parameter passing, Return value, Local variable, Global variable
12	8.2 Lesson 2 - Random, math and time modules	• Understand how to use random number generation. • Understand how to use the math module. • Understand how to use the time module to pause the execution of programs. KEY QUESTION:	"Can computers produce truly random numbers?" What can you find out about this question in the next few minutes?	SLR 8 Workbook Complete slides 4-5 Write the dice roll program that you pseudo-coded on slide 4 of your workbook in Python.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 The Turtle module	

		In what sort of problems might we need to generate a random number or sequence of random numbers?			
13	8.3 Lesson 3 - Turtle module	• Understand how to use the Python turtle module, including use of: 　• forward and back 　• left and right 　• home, setpos and reset 　• hideturtle and showturtle 　• penup and pendown 　• pensize and pencolor KEY QUESTION: How can the turtle module in Python be used to create geometric shapes?	The turtle library in Python is based on a much old language, using the internet, what can you find out about "Logo".	SLR 8 Workbook Complete slide 6 Practice using the turtle commands, combined with other constructs such as selection and iteration to create a range of geometric drawings.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 The use of basic file handling operations
14	8.4 Lesson 4 - File handling	• Understand how to use basic file handling operations: 　• Open files 　• Read from files 　• Write to files 　• Close files KEY QUESTION: What are the steps to using data files with programs?	Examine the following program and data file. The program should output the names of all the countries, but it does not work. Why?	SLR 8 Workbook Complete slides 7 and 8 Using the unscrambled programs from SLR8 workbook sides 7 and 8, enter these into Python as two separate programs and check they work. The data entry will be case sensitive. The two programs are not compatible with each other, re-write the text file and the programs to use dates in ascending order instead.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 How to use pre-existing and user-devised subprogram
15	8.5 Lesson 5 - Procedures and functions	• Understand how to use subprograms. • Explain the advantages of using subprograms.	What does this program output if A=6?	SLR 8 Workbook Complete slide 9	Video(s) from "Topic 6B Programming fundamentals":

		• Understand the difference between a procedure and a function. • Describe the user of parameters to pass data within programs. • Understand that subroutines can return values to the calling routine. KEY QUESTION: What is meant by "parameter passing" when talking about subroutines?		Write a program to output a random bingo ticket. These objectives get progressively more difficult. See how far you can get.	• 🔗 Local and global variables	
16	8.6 Lesson 6 - Local and global variables	• Understand how to use subprograms. • Understand how to use local variables. • Explain the importance of using local variables. • Understand how to use global variables. • Understand how to pass variables between parts of a programs. KEY QUESTION: What are local and global variables and how are they different?	Using the internet, you have five minutes to list as many computing data structures as you can. Are you able to find any obscure ones which no one else in the class can find?	SLR 8 Workbook Complete slide 10 Continue working on the bingo from last Lesson. Remember the objectives get progressively more difficult. See how far you can get.		
17	8.7 Lesson 7 – Catch-up lesson	• Complete various programming tasks and challenges you have been presented with during the programming workbooks. KEY QUESTION: What is a 'return value'?	Where in this program are 'parameters' being passed and where are they being returned?	Complete any outstanding workbook pages. Complete any outstanding programs: 1. Lift off 2. Random dice 3. Turtle programs 4. Bingo ticket	Revise what you have learnt in this topic.	

GCSE EDEXCEL Computer Science – Scheme of Learning

YEAR 11 - TERM 2

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
SLR 8 Advanced programming concepts						
18	SLR 8 – End-of-topic test	End-of-topic test		SLR 8 - Test		
19	SLR 8 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings		
20 to 22	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Identifying errors and suggesting fixes	
SLR 9 Robust and secure programming						
23	9.1 Lesson 1 - Syntax, logic and runtime errors	• Know four reasons why a program should be tested. • Know what a syntax error is. • Know what a logic error is. • Know what a runtime error is. • Know how to identify and fix errors. KEY QUESTION: What are the different types of errors that can occur in a program?	The following program should output the factorial of a number input. E.g. 5! = 5*4*3*2*1 = 120, but it contains syntax and logic errors. Where are they?	SLR 9 Workbook Complete slides 2-7 Spend 10-15 minutes playing the whole class activity "spot the bug" with your teacher.	Video(s) from "Topic 3B Software and programming languages": • 🔗 Importance of robust software	Slides 38-40, 96, 178-182, 196-201 Syntax error, Logic error, Runtime error, Robust software, Maintainable programs, Indentation, Comments, Meaningful identifiers, White space, Data validation, Length check, Presence check, Range check, Pattern check, Authentication routines
24	9.2 Lesson 2 - Importance of robust software	• Understand what we mean by robust software. • Understand the purpose of an audit trail. • Understand the purpose of a code review. KEY QUESTION: What methods can be used to help identify vulnerabilities in software code?	Using the internet see what you can find out about the famous "Heartbleed" vulnerability.	SLR 9 Workbook Complete slide 8 Continue working on any unfinished programs you may have.	Video(s) from "Topic 6A Developing code": • 🔗 Program maintenance techniques	

25	9.3 Lesson 3 - Maintainability	• Know why creating easy to read code is important with large projects. • Understand what programmers can do to make their code more readable. • Understand how to refine algorithms to make them more robust. KEY QUESTION: What does code maintainability mean?	Research on the internet: "best practices for writing super readable code." What are 7 deadly sins for creating unreadable code?	SLR 9 Workbook Complete slide 9 Using the code on from slide 9 of workbook as a starting point, adapt it so that it will only accept valid integers.	Video(s) from "Topic 6B Programming fundamentals": • 🔗 Data validation	
26	9.4 Lesson 4 - Data validation	• Understand why data validation is necessary. • Know a range of validation techniques that can be used to write simple data validation routines. KEY QUESTION: What issues should a programmer consider to ensure a program can handle all likely input values?	What input validation could be performed on this sign-up form?	SLR 9 Workbook Complete slides 10 and 11 Write a program that asks the user to enter a date in the format dd/mm/yyyy The program should validate the data in the following ways, and output which type of check was failed or that the date is valid:	Video(s) from "Topic 6B Programming fundamentals": • 🔗 Simple authentication routines	
27	9.5 Lesson 5 - Authentication routines	• Understand some authentication techniques a programmer may choose to use to protect their program from misuse. • Be able to write simple authentication routines. KEY QUESTION: What do we mean by the term authentication?	Why do online forms often have this prompt? What else is ReCaptcha used for?	SLR 9 Workbook Complete slides 12 Write a simple authentication routine that uses a username and password. SUPER CHALLENGE: Research what makes a strong password. Create a program to validate a secure password.		

28	9.6 Lesson 6 – Catch-up lesson	• Complete any outstanding pages of your workbook. KEY QUESTION: What makes a robust program?	What does it mean to "produce a robust program"? Create a list of points you would include in an answer to a question worth 12 marks.	Complete any outstanding workbook pages. Complete any unfinished program from this topic. SUPER CHALLENGE: Write a program to convert a decimal into a fraction. E.g. 0.5 = ½, 0.75 = ¾.	
29 to 32	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Revise what you have learnt in this topic.
33	SLR 9 – End-of-topic test	End-of-topic test		SLR 9 - Test	
34	SLR 9 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	

YEAR 11 - TERM 3

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
SLR 10 Algorithms and computational logic						
					Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Abstraction	
35	10.1 Lesson 1 - Algorithms and abstraction	• Be able to define the term 'algorithm'. • Know what is meant by the term 'abstraction'. • Know some examples of abstraction. KEY QUESTION: What is an algorithm?	The picture below shows Perrygrove railway. In what ways is this an example of abstraction?	SLR 10 Workbook Complete slides 2 and 3 Start slide 4. Can you extend your icon set to include more animals? What features have you included in your icons so they are recognisable as part of the same icon set?	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Decomposition "Topic 6A Developing code": • 🔗 Using abstraction and decomposition to solve problems	Slides 1-2, 37, 41-46 Abstraction, Decomposition, Trace table, Bubble sort, Merge sort, Linear search, Binary search, Algorithm efficiency, Truth table
36	10.2 Lesson 2 – Abstraction and decomposition	• Know what is meant by problem decomposition. • Know the advantages of decomposition when applied to programming. • Know an example of problem decomposition. KEY QUESTION: What is the purpose of abstraction and decomposition and how can producing structure diagrams help with this process?	You and a friend decide to go to the cinema on Saturday. How would you apply problem decomposition to this task?	SLR 10 Workbook Complete slides 5-7 Complete the programming challenge on slide 7	Video(s) from "Topic 6A Developing code": • 🔗 6A Algorithmic thinking	
37	10.3 Lesson 3 - Algorithmic thinking	• Understand how to solve computational problems by a systematic approach to problem solving. KEY QUESTION:	A storage unit is in the shape of a hemisphere on top of a cylinder. The surface of the storage unit is to be painted. Calculate the area to be painted.	SLR 10 Workbook Start slide 8 Write the programs on slide 8.		

		What do we mean by a "systematic approach to problem solving and algorithmic thinking"?			
38	10.4 Lesson 4 - Algorithmic thinking	• Understand how to solve computational problems by applying a systematic approach to problem solving. KEY QUESTION: What do we mean by a "systematic approach to problem solving and algorithmic thinking"?	What does this algorithm output for each of the following values of a and b? a = 3 b = 4 What does this algorithm do?	SLR 10 Workbook Complete slide 9 Continue working on problems on either slide 8 or slide 9.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 How to produce algorithm
39	10.5 Lesson 5 - How to produce algorithms	• Know the flow diagram symbols. • Know that flow diagrams are also called flowcharts. • Know how to make a flow diagram. • Understand how to construct a program from a flow diagram. • Know what is meant by the term pseudocode. • Understand how to write pseudocode. KEY QUESTION: How can algorithms be described without ambiguity?	Can you write a program for this algorithm?	SLR 10 Workbook Complete slides 10-12 Complete the program on slide 12.	
40	10.6 Lesson 6 - How to produce algorithms	• Know how to make a flow diagram. • Understand how to write pseudocode. • Understand how to write a program from a flow diagram and pseudocode. KEY QUESTION: How can algorithms be described without ambiguity?	A role-playing game (RPG) often requires dice with a different number of sides to be rolled. This Python program should ask the user how many sides the dice has, and how many rolls to perform, before outputting the result of the dice rolls, but it does not work. Where is the bug?	SLR 10 Workbook Complete slides 13-15 Write the program described on slide 14.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Inputs, processes and outputs • 🔗 How to identify syntax and logic errors

41	10.7 Lesson 7 - Identifying inputs, processing and outputs	• Understand how to interpret algorithms. • Understand how to correct algorithms. • Be able to identity inputs, processing and outputs from an algorithm. KEY QUESTION: What do we mean when we talk about a problem's inputs, processes and outputs?	A role-playing game (RPG) often requires dice with a different number of sides to be rolled. This Python program should ask the user how many sides the dice has, and how many rolls to perform, before outputting the result of the dice rolls, but it does not work. Where is the bug?	SLR 10 Workbook Complete slides 16-18 Write the program described on slide 16 or 18.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Using trace tables to determine the purpose of algorithms	

42	10.8 Lesson 8 - Visual inspection and trace tables	• Understand how to use visual inspection to determine how simple algorithms work and what their purpose is.
		• Understand how to use trace tables to determine how simple algorithms work and what their purpose is.
		KEY QUESTION:
		How and why do programmers use a trace table?
			What does this algorithm output for each of the following values of n?
			    n = 2
			    n = 3
			    n = 4
			What does this algorithm do?
				SLR 10 Workbook Complete slides 19 and 20
				
				Continue working on problems from either slide 16 or slide 18.
					Video(s) from "Topic 1A Computational thinking and algorithms":
					• 🔗 Linear search	

43	10.9 Lesson 9 - Linear search	• Understand the linear search algorithm. • Understand it is not an efficient algorithm, but it is easier to program than alternatives and does not require the items to be in any order. KEY QUESTION: How does a linear search work?	Consider this list of mountains. With a linear search, how many conditions will be executed to find each of the mountains below? • Manaslu • Annapurna • Broad Peak	SLR 10 Workbook Complete slide 21 Write a program to perform a linear search to output the latitude and longitude of a given capital city.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Binary search	
44	10.10 Lesson 10 - Binary search	• Understand the binary search algorithm. • Know the special condition of the list of items for the binary search to work. • Understand which searching algorithm is quicker.	Given this list of numbers, can you illustrate the binary search as a picture of a tree? 2, 4, 6, 8, 10, 12, 14	SLR 10 Workbook Complete slide 22 Write a program to perform a binary search on a list of items.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Merge sort	

		KEY QUESTION: How does a binary search work?			
45	10.11 Lesson 11 - Merge sort	• Understand the merge sort algorithm. KEY QUESTION: How does a merge sort work?	Class is divided into two groups. Group 1 are demonstrating a **merge** sort. Group 2 are demonstrating an **insertion** sort. To demonstrate this, each person in the team lines up and holds an A4 sheet with a number in front of them. According to the algorithm one student physically moves at a time to show how the sort works.	SLR 10 Workbook Complete slide 23 Continue working on programs you have already started in this unit. There is no need to learn how to program the merge sort as it requires some A Level knowledge.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Bubble sort
46	10.12 Lesson 12 - Bubble sort	• Understand the bubble sort algorithm. KEY QUESTION: How does a bubble sort work?	What is the value of n after running this code? What does this code do?	SLR 10 Workbook Complete slide 24 Continue working on programs you have already started in this unit. There is no need to learn how to program the bubble sort as it requires some A Level knowledge.	Video(s) from "Topic 1A Computational thinking and algorithms": • 🔗 Algorithm efficiency
47	10.13 Lesson 13 - Algorithm efficiency	• Understand that more than one algorithm can be used to solve the same problem. • Understand that certain algorithms can be more efficient (quicker) at solving a problem than another. KEY QUESTION: What do we mean by algorithm efficiency?	Use the internet to find out what a "link analysis" algorithm is and how it is used.	SLR 10 Workbook Complete slides 27-29 Continue working on programs you have already started in this unit.	

48	10.14 Lesson 14 - Catch-up lesson	• Use this lesson to catch up on any outstanding work so far from this topic. KEY QUESTION: Which sorting algorithm is better, merge or quick sort?	A role-playing game (RPG) often requires dice with a different number of sides to be rolled. This Python program should ask the user how many sides the dice has, and how many rolls to perform, before outputting the result of the dice rolls, but it does not work. Where is the bug?	Complete any unfinished pages of your workbook so far Continue working on programs you have already started in this unit.	Video(s) from "Topic 6A Developing code": • 🔗 Evaluating a program's fitness for purpose and efficiency
49	10.15 Lesson 15 - Evaluating a programs fitness for purpose	• Know how to work out the best-case and worse-case scenarios for a linear search and binary search. • Know how algorithm code could be improved to make it more efficient. • Know what is meant by in-place algorithms. • Know what is meant by out-of-place algorithms. KEY QUESTION: What do we mean by best-case and worst-case when talking about algorithms?	Consider the interface of a sat-nav device. What are the necessary details for the input? What are the necessary details for the output?	SLR 10 Workbook Complete slides 30-32 Continue working on programs you have already started in this unit.	Video(s) from "Topic 1B Computational logic": • 🔗 Simple logic diagrams • 🔗 Truth table

YEAR 11 - TERM 4

	Focus for Lesson	Learning Outcomes	Starter	Activities	HW for next Lesson	Key Terms
				SLR 10 Algorithms and computational logic		
50	10.16 Lesson 16 - Truth tables for simple logic gates and diagrams	• Understand how to construct truth tables from logic gates NOT AND OR. • Be able to interpret the results of simple truth tables. KEY QUESTION: What are the symbols used in logic diagrams and how do they relate to truth tables?	Use http://logic.ly/demo/ to make this circuit: What internal component do you think it could be part of in a computer system?	SLR 10 Workbook Complete slides 33-39 Just for fun – this circuit is part of the arithmetic logic unit in the CPU. It adds two binary digits: 0+0=0, 0+1=1, 1+0=1, 1+1=0 carry 1. Have a go at making it, using https://logic.ly/demo/	Video(s) from "Topic 1B Computational logic": • 🔗 Applying logical operators in truth tables to solve problems	Slides 1-2, 37, 41-46 Abstraction, Decomposition, Trace table, Bubble sort, Merge sort, Linear search, Binary search, Algorithm efficiency, Truth table
51	10.17 Lesson 17 - Using truth tables to solve problems	• Understand how to create, complete or edit truth tables for given scenarios. KEY QUESTION: How do you construct truth tables from real life scenarios?	In our videos we presented a simple way of remembering which of the following logic gates represents, AND, OR and NOT. What was this method? Describe it to the person sitting next to you.	SLR 10 Workbook Complete slides 40-42 Mark your truth tables by drawing the logic diagrams on slides 18-20 using http://logic.ly/demo/ Use toggle switches for inputs and a light bulb for the output. Check that the output matches the inputs given for each of your truth tables.		
52	10.18 Lesson 18 - Catch-up lesson	• Use this lesson to catch up on any outstanding work from this topic. KEY QUESTION:	No starter for this lesson, just crack on with any outstanding theory work or programming exercises from this topic!	Complete any unfinished pages of your workbook		

		What real life situations can you think of where you apply algorithms daily without even realising it?		Continue working on programs you have already started in this unit.	
53 & 54	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Revise what you have learnt in this topic
55	SLR 10 – End-of-topic test	End-of-topic test		SLR 10 – Test (part 1 and 2)	
56	SLR 10 – Action	Action/Response lessons		Chance for students to respond to feedback, improve workshop, correct misunderstandings	
57	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Video(s) from "Topic 3B Software and programming languages": • 🔗 Characteristics and purpose of different levels of programming language
SLR 11 Classification of programming languages					
58	11.1 Lesson 1 - Characteristics of languages	• Know the characteristics of low-level and high-level programming languages. • Explain the main differences between low-level and high-level languages. • Understand the advantages and disadvantages of low-level language programming compared with high-level language programming. KEY QUESTION: What are the differences between high- and low-level languages?	Translate these alphabet hieroglyphics:	SLR 11 Workbook Complete slides 2-3 Complete any outstanding programming challenges or improve any existing programs.		Slides 103-108 Low-level language, High-level language, Interpreter, Compiler, Translator, Machine code

59	11.2 Lesson 2 - Relationship between machine code and assembly	• Understand the relationship between machine code and low-level languages. • Explain the difference between machine code and low-level languages. KEY QUESTION: What is the relationship between machine code and assembly?	What should this program do? What is the problem with it?	SLR 11 Workbook Complete slides 4-5 Complete any outstanding programming challenges or improve any existing programs.	Video(s) from "Topic 3B Software and programming languages": • 🔗 Characteristics of compilers and interpreters
60	11.3 Lesson 3 - Assemblers, compilers and interpreters for translation	• Know what a translator does. • Understand the differences between compilers and interpreters. • Understand when it would be appropriate to use each type of translator. KEY QUESTION: How does code a programmer writes become binary a computer can execute?	Research the family tree of programming languages and put these languages in historical order.	SLR 11 Workbook Complete slides 6-7 Type the two programs to generate the prime numbers between 1 and 100 into Python and a BBC Micro emulator: https://bbc.godbolt.org/ Put a syntax error in line 120. Observe how each language handles the syntax errors differently: Python with a compiler and BBC Basic with an interpreter.	
61 & 62	Dedicated paper 2 exam revision lesson	Gain experience in answering computational thinking, algorithm and programming questions for the one screen exam component (1CP2 – Paper 2) by using our dedicated exam Revision unit.		Progress with activities in the "Paper 2 exam revision unit" folder	Revise what you have learnt in this topic
63	SLR 11 – End-of-topic test	End-of-topic test		SLR 11 - Test	
64	SLR 11 – Action	Action/Response lessons		Chance for students to respond to feedback,	

			improve workshop, correct misunderstandings		

YEAR 11 - TERM 5

	Focus for Lesson	Learning Outcomes and Key Question	Starter	Activities	HW for next Lesson	Key Terms
65	Exam revision	This final term before Easter has been set aside for you to use as you see fit for your students. We have many resources to help with revision:				
66	Exam revision					
67	Exam revision					
68	Exam revision					
69	Exam revision					
70	Exam revision					
71	Exam revision					
72	Exam revision					
73	Exam revision					
74	Exam revision					
75	Exam revision					
76	Exam revision					
77	Exam revision					

For lessons 65–77, the Learning Outcomes column contains:

- Further programming experience with the exam revision unit.
- Additional time to complete the many programming challenges and super challenges presented throughout the SLR theory units and from the dedicated programming resources.
- Our free, dedicated for students with all our videos and other helpful resources: student.craigndave.org
- Our series of videos on exam technique, including how to understand command words and answer extended questions: student.craigndave.org/videos/exam-technique

We also have a dedicated revision tool and course companion called Smart Revise, which has a bank of over 600 questions for the GCSE 1CP2 course. It is *not* simply another MCQ tool – we based the entire design and philosophy of Smart Revise around proven research on how students learn and remember over time. Regular usage of Smart Revise has proven to have marked results on students' ability to recall key information and facts under exam conditions. Smart Revise has a pin-sharp focus on the specification, and every single bullet point is covered.

To find out more about Smart Revise, visit smartrevise.craigndave.org

To get started with a free trial, visit www.smartrevise.online.