

International Advanced Level **Computer Science**

Scheme of Work

Pearson Edexcel International Advanced Subsidiary in Computer Science (XCP01)

First teaching September 2026 | First examination June 2027

First certification from August 2027

Issue 1

Content

How to use this Scheme of Work	3
Unit 1: Principles of Computer Science	6
Unit 2: Practical Programming and Problem Solving	88

How to use this Scheme of Work

This Scheme of Work (SoW) is a suggested approach for a 360-hour course to enable the completion of the International Advanced Level Computer Science (YCP01) qualification content. It provides an approximate teaching time of 325 hours, with approximately 35 hours dedicated to consolidation, summative assessment, feedback, and exam preparation. International Advanced levels have 360 guided learning hours. Each unit is estimated to have 90 guided learning hours. The SoW has been split into one for IAS and one for IA2. This SoW is for IAS (XCP01).

The assumption is made that there are 360 hours of teaching time over 60 weeks and reflects how centres could use the time for practical activities. This Scheme of Work matches the International Advanced Level Computer Science course planner and follows the order of the specification.

IAS Unit 3 Principles of Computer Science

Topic 11 – Computer systems and data representation

Topic 12 – Networks and encryption

Topic 13 – Structuring data

Topic 14 – Problem-solving

Topic 15 – Problem-solving

Topic 16 – Emerging technologies and professional practice

Topic 17 – Problem-solving

Topic 18 – Problem-solving

IAS Unit 4 Practical Programming & Problem Solving

Topic 19 – Programming

Topic 20 – Organising and handling data

Topic 21 – Representing and handling data

Topic 22 – Best practice

Topic 23 – Computational thinking and algorithms

Topic 24 – Algorithms

Topic 25 – Algorithms

Topic 26 – Algorithms

This Scheme of Work is not intended to be prescriptive. This document is editable to allow for any adaptations you may wish to make to best suit your teaching style and learner needs.

The columns in this Scheme of Work indicate:

- An approximate number of week(s) allocated to the course content.
- Which section of the specification this lesson relates to.
- Explicit reference is made to what course content needs covering in lessons.
- Suggested activities that provide teaching ideas and resources that could be used to support the teaching of lessons, including relevant textbook page numbers. All underlined content has an associated hyperlink.
- Skills assessed through the examination and specific examples of questions that do this in the exam.
- Skills that could be acquired through teaching and delivery within lessons.

Why transferable skills?

In recent years, higher education institutions and global employers have consistently flagged the need for students to develop a range of transferable skills to enable them to respond with confidence to the demands of undergraduate study and the world of work. To support the design of our qualifications, we have mapped them to a transferable skills framework. The framework includes cognitive, intrapersonal, and interpersonal skills and each skill has been interpreted for each specification to ensure they are appropriate for the subject. Further information on transferable skills is available on the website. Pearson materials, including this Scheme of Work, will support you in identifying and developing these skills in students.

In the final two columns of this Scheme of Work, we have indicated which transferable skills are explicitly assessed, and where there are opportunities for them to be developed through teaching. Our intention is that teachers can use these columns to increase the opportunities for transferable skills development in students.

Unit 1: Principles of Computer Science

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 1					
Lesson 1					
1.1 Computer Architecture Topic 1.1.1: Stored-program concept Lesson title: Stored-program concept	1.1.1 Stored-program concept Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks	Activities • Discuss how having a stored program differentiates computers from other devices and how switching between different programs allows general-purpose computers to perform a variety of user tasks • Option to consider early computers and the development of computers from early calculating machines to the present day • Option to look inside a computer to identify key hardware components to help students visualise concepts • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: • Students research early computing devices and	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 2–3 Look for videos about ENIAC, EDVAC and Manchester Baby e.g. https://www.youtube.com/watch?v=golNhUyRLcQ	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Initiative

compare them to modern computers

Week 1 Lesson 2

Topic 1.1.2: von Neumann architecture	1.1.2 von Neumann architecture	Activities - Discuss the von Neumann architecture and how it is still in use today - Option to look for vintage CS textbooks and observe how these key principles remain constant - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: - Students complete activity “Thinking like a computer scientist” on p5	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 3–5 Computerphile video: https://www.youtube.com/watch?v=MI3-kVYLNr8 Geeks For Geeks article: https://www.geeksforgeeks.org/computer-organization-architecture/computer-organization-von-neumann-architecture/	Analysis Communication Critical thinking Problem solving	Critical thinking Problem solving
Lesson title: Von Neumann architecture	Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks				

Week 1 Lesson 3					
Topic 1.1.3: Internal components a) CPU b) Memory Lesson title: Internal components part 1	1.1.3 Internal components of a computer system Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of the internal components of a computer system based on von Neumann architecture - Develop an expression for the amount of addressable memory	Activities - Discuss the roles of the CPU and Memory as well as the different types of memory (RAM, ROM, Cache). Note that memory hierarchy is covered in topic 1.1.6. - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks - Teacher demonstrates how to develop an expression for the amount of addressable memory (see worked example on p10) Homework: - Students use Formulator Tarsia to make a matching activity with key terms / definitions and then swap with each other	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 5–7 Code.org video: https://www.youtube.com/watch?v=DKGZlaPIVLY	Analysis Communication Critical thinking Problem solving	Decision making

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 2					
Lesson 1					
Topic 1.1.3: Internal components c) Input/output d) Buses Lesson title: Internal components part 2	1.1.3 Internal components of a computer system Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of the internal components of a computer system based on von Neumann architecture	Activities - Discuss i/o devices and the three system buses (address, data, control) - Option to research i/o devices with assistive technologies - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks - Students answer questions on AdaCS Homework: - Students complete activity 1 on p9	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 7–10 AdaCS notes and questions: https://adacomputerscience.org/concepts/arch_components?topic=architecture	Analysis Communication Critical thinking Problem solving	Self-direction Intellectual interest and curiosity

Week 2 Lesson 2					
Topic 1.1.4: CPU components a) CU b) ALU e) clock Lesson title: CPU components part 1	1.1.4 Components of the CPU Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation Students will be able to: - Interpret and complete a diagrammatic representation of the components of the CPU.	Activities - Discuss the CU, ALU and clock - Students identify key terms and make notes on the definition, function, characteristics and operation of these components as well as why they are needed - Complete activity 2 on p18 Homework: - Students draw and label a diagram showing the components of the CPU.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 10– 12 CrashCourse CS video (ALU): https:// www.youtube.com/ watch? v=1I5ZMmrOfnA&t=26s CrashCourse CS video (clock): https:// www.youtube.com/ watch? v=FZGugFqdr60&t=34 6s	Analysis Communication Critical thinking Problem solving	Analysis Critical thinking Self-direction

Week 2 Lesson 3					
Topic 1.1.4: CPU components c) General-purpose registers d) Specific-purpose registers Lesson title: CPU components part 2	1.1.4 Components of the CPU Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation Students will be able to: - Interpret and complete a diagrammatic representation of the components of the CPU.	Activities - Discuss general-purpose and specific-purpose registers (PC, MAR, MBR, CIR, ACC, SP, SR). Use Appendix 2 to guide use of symbols. - Students identify key terms and make notes on the definition, function, characteristics and operation of registers as well as why they are needed - Look for quizzes about registers on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc. Homework: - Students use Formulator Tarsia to make a matching activity with key terms / definitions and then swap with each other.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 11 CrashCourse CS video: https://www.youtube.com/watch?v=fpnE6UAfbtU&t=481s	Analysis Communication Critical thinking Problem solving	Self-direction Problem solving Productivity Intellectual interest and curiosity

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 3					
Lesson 1					
Topic 1.1.5: FDE cycle a) Stages b) Roles of components Lesson title : FDE cycle part 1	1.1.5 Fetch-decode-execute cycle Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - Interpret and complete a diagrammatic representation of the transfer of data between components during the FDE cycle	Activities - Discuss the stages of the fetch-decode-execute cycle and the roles of the components covered in 1.1.3 and 1.1.4 - Students identify key terms and make notes on the definition, function and operation for the FDE cycle as well as why it is needed Homework: - Students examine the stages of the FDE cycle example on p13–17 to discuss next lesson.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 13–17 UT Austin notes: https://www.cs.utexas.edu/~fussell/courses/cs310h/lectures/Lecture_9-310h.pdf	Analysis Communication Critical thinking Problem solving	Critical thinking Problem solving Analysis Interpretation

Week 3

Lesson 2

Topic 1.1.5: FDE cycle c) Use of read/write control signals Lesson title : FDE cycle part 2	1.1.5 Fetch–decode–execute cycle Students will know and understand: • Definition • Function • Why needed • Operation Students will be able to: • Use register transfer notation to show how data passes between registers during the FDE cycle • Interpret and complete a diagrammatic representation of the transfer of data between components during the FDE cycle	Activities • Discuss FDE cycle example on p13–17 • Discuss the use of read/write control signals • Option to use Little Man Computer to observe the stages in action • Students identify key terms and make notes on the definition, function and operation of read/write control signals during the FDE cycle as well as why they are needed • Teacher demonstrates how register transfer notation can be used to show data passing between registers during the FDE cycle Homework: • Students complete a diagram of data being transferred during the FDE cycle.	Pearson Edexcel International A–level Computer Science Student Book 1: pp. 13–17 Tom Scott video: https://www.youtube.com/watch?v=Z5JC9Ve1sfl	Analysis Communication Critical thinking Problem solving	Critical thinking Problem solving Communication
--	--	---	--	--	--

Week 3

Lesson 3

Topic 1.1.6: Memory hierarchy Lesson title: Memory hierarchy	1.1.6 Memory hierarchy Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of the memory hierarchy	Activities - Discuss memory hierarchy in terms of data transfer speeds and storage capacity (CPU registers, Cache L1/L2/L3, Main memory, Secondary storage) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks of the different types of memory - Option to compare a CPU, RAM sticks, and examples of secondary storage to help students visualise the concepts - Look for quizzes about types of memory on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc Homework: - Students draw and label a diagram representation the memory hierarchy	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 17–18 Geeks for Geeks notes: https://www.geeksforgeeks.org/computer-organization-architecture/memory-organisation-in-computer-architecture/	Analysis Communication Critical thinking Problem solving	Self-direction Productivity Initiative
--	---	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 4					
Lesson 1					
Topic 1.1.7: Data transmission Lesson title: Data transmission	1.1.7 Data transmission Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • Interpret and complete a diagrammatic representation of data transmission	Activities • Discuss serial and parallel data transmission • Option to look at examples of a motherboard, ribbon cable or USB cable to help students visualise the concepts • Teacher demonstrates how diagrams can be used to highlight the differences between serial and parallel data transmission • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks of serial and parallel data transmission as well as why they are needed Homework: • Students complete activity 3 on p21	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 19–21 MrBrown CS video: https://www.youtube.com/watch?v=CGulJriYNSI	Analysis Communication Critical thinking Problem solving	Decision making Self-direction Intellectual interest and curiosity Initiative

Week 4

Lesson 2

1.2 The Operating System	1.2.1 Role of the OS	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 21–24	Analysis Communication Critical thinking Problem solving	Communication Initiative Self-direction Intellectual interest and curiosity
Topic 1.2.1: Role of the OS	Students will know and understand: - Definition - Function - Characteristics - Why needed	- Discuss the role of the OS in user interface, user management, peripheral management, process management and memory management - Option to discuss various OSs that students have experience of through school devices, personal devices or smart phones - Students identify key terms and make notes on definition, function, and characteristics of the OS as well as why these roles are needed - Students use a school device to take screenshots demonstrating each of these roles of the OS - Discuss Case Study 1 on p24	AdaCS notes: https://adacomputerscience.org/topics/operating_systems		
Lesson title: Role of the OS		Homework: Students use their own device to take screenshots demonstrating each of these roles of the OS			

Week 4

Lesson 3

Topic 1.2.2: Multitasking a) Time sharing b) Context switching Lesson title: Multitasking	1.2.2 Multitasking: Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities - Discuss how multitasking can be accomplished using time sharing and context switching - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks of different multitasking management techniques as well as why they are needed - Look for quizzes about multitasking on platforms like Bloocket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc. Homework: - Students make a list (or gather screenshots) of as many different interrupts as they can think of which affect their device usage	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 24 Geeks for Geeks notes (time sharing): https://www.geeksforgeeks.org/operating-systems/time-sharing-operating-system/ Geeks for Geeks notes (context switching): https://www.geeksforgeeks.org/operating-systems/context-switch-in-operating-system/	Analysis Communication Critical thinking Problem solving	Communication Initiative Self-direction Intellectual interest and curiosity Productivity
---	---	--	---	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 5					
Lesson 1					
Topic 1.2.2: Multitasking c) Interrupt handling in multitasking Lesson title: Multitasking	1.2.2 Multitasking Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of the interrupt handling process	Activities - Discuss how an interrupt service routine is used to handle interrupts as part of multitasking - Option to link to exception handling in U2 7.1.1 - Teacher demonstrates the use of a diagram to represent the interrupt handling process (see p25 Fig 1.21) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawback as well as why interrupt handling is needed. Homework: - Students use Formulator Tarsia to make a matching activity with key terms / definitions and then swap with each other.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 24–25 Geeks for Geeks notes: https://www.geeksforgeeks.org/operating-systems/interrupts/ IsaacCS notes: https://isaacomputerscience.org/concepts/sys_os_interrupts?topic=operating_systems	Analysis Communication Critical thinking Problem solving	Productivity Self-direction Intellectual interest and curiosity

Week 5

Lesson 2

Topic 1.2.3: Process management a) Process b) Process states c) Scheduling Lesson title: Process management	1.2.3 Process management Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of process states	Activities - Discuss process management including processes, process states and scheduling - Teacher demonstrates exploring the task manager of a school PC to give students a sense of these concepts in action - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why process management is needed Homework: - Students copy out Fig 1.22 on p26 which is a diagram representing the transitions between process states	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 26–27 CS Lessons video : https://www.youtube.com/watch?v=ocDiT21zRcE	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Interpretation Problem solving Initiative
--	---	--	---	--	---

Week 5

Lesson 3

Topic 1.2.3: Process management d) Scheduling algorithms Lesson title: Process management	1.2.3 Process management Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of a scheduling algorithm	Activities - Discuss scheduling algorithms (first-come-first-served, shortest job first, round robin, multi-level queue) - Option to complete a practical Python challenge on 101computing - Option to carry out a practical activity moving around the room to help students visualise and remember the different algorithms - Teacher demonstrates scheduling algorithms represented as diagrams (figs 1.23, 1.24, 1.25, 1.26 on pp27–30) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why scheduling algorithms are needed. - Complete activity 4 on p31 Homework: - Having completed the discussion for activity 4 on p31 in the lesson, students write up their responses using key terms.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 27–31 101computing notes: https://www.101computing.net/scheduling-algorithms-python-challenge/	Analysis Communication Critical thinking Problem solving	Critical thinking Initiative Self-direction Communication
--	---	---	---	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 6					
Lesson 1					
Topic 1.2.4: Memory management a) Operations b) Allocation techniques Lesson title: Memory management	1.2.4 Memory management Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of memory usage	Activities - Discuss the operations needed for memory management (tracking memory allocation status, determining storage requirements, controlling memory usage) - Introduce allocation techniques and discuss paging (segmentation and virtual memory to follow next lesson) - Teacher demonstrates diagrammatic representation of memory usage (fig 1.27 and 1.28 on p32) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why memory management is needed. Homework: - Flipped classroom: students research additional memory allocation techniques before learning more next lesson	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 32–36 MrBrown CS video: https://www.youtube.com/watch?v=gTtfq610s48&t=290s	Analysis Communication Critical thinking Problem solving	Initiative Self-direction Intellectual interest and curiosity Productivity

Week 6

Lesson 2

Topic 1.2.4: Memory management b) Allocation techniques Lesson title: Memory management	1.2.4 Memory management: allocation techniques Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of memory usage	Activities - Discuss segmentation and virtual memory as memory allocation techniques (having already covered paging last lesson) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why allocation techniques are needed Homework: - Students complete Activity 5 on p36	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 32–36 CS Lessons video : https://www.youtube.com/watch?v=p9yZNLLeOj4s	Analysis Communication Critical thinking Problem solving	Decision making Communication Analysis
--	---	--	---	--	---

Week 6

Lesson 3

Topic 1.2.4: Memory management c) Stack frame Lesson title: Memory management	1.2.4 Memory management: stack frame Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a diagrammatic representation of the contents of a call stack	Activities - Discuss the stack frame as a tool for memory management - Option to link to 6.3.1d IDE (debugger) features and look at the call stack in action while a program is running - Teacher outlines Worked Example on pp. 37–38 - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why the stack frame is needed Homework: - Students complete MCQs	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 37–38 Geeks for Geeks notes: https://www.geeksforgeeks.org/computer-organization-architecture/stack-frame-in-computer-organization/ HW MCQs https://examradar.com/operating-system-main-memory-memory-allocation-process-using-stacks-mcq-type-questions-answers/	Analysis Communication Critical thinking Problem solving	Critical thinking Problem solving Analysis
--	---	--	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 7					
Lesson 1					
Topic 1: Computer Systems Lesson title: Topic summary	Topic 1: Computer Systems topic summary	Activities • Use mind maps or knowledge organisers to summarise concepts • Search for quizzes to test knowledge and understanding • Use flashcards to learn definitions of key terms • Use questions from sample and specimen exam papers to test exam readiness • Complete Checkpoint questions on p38	Pearson Edexcel International A-level Computer Science Student Book 1: p.38 Geeks for Geeks quiz: https://www.geeksforgeeks.org/quizzes/quiz-on-memory-management/	Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity

Week 7 Lesson 2					
2.1 Numbers	2.1.1 Number bases: binary and denary	Activities - Discuss the place values in binary and denary and how they can be used to make a unique representation for each unsigned integer - Teacher demonstrates conversions from binary to denary and vice versa - Students identify key terms and make notes on definition, function, characteristics and why number bases are needed - Complete Activity 1 on p42	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 40–42	Analysis Communication Critical thinking Problem solving	Problem solving Critical thinking Analysis Interpretation
Topic 2.1.1: Number bases a) Binary b) Denary Lesson title: Number bases	Students will know and understand: - Definition - Function - Characteristics - Why needed Students will be able to: - convert unsigned integers between bases. Maths skills: 2.1.1 Converting between number bases	Homework: Students complete questions on IsaacCS	Search for web-based games to practise calculations e.g. Penjee https://games.penjee.com/binary-numbers-game/ IsaacCS https://isaacomputerscience.org/concepts/data_numbases_conversions?topic=number_representation		

Week 7

Lesson 3

Topic 2.1.1: Number bases b) Binary c) Hexadecimal Lesson title: Number bases	2.1.1 Number bases: further binary and hexadecimal Students will know and understand: • Definition • Function • Characteristics • Why needed Students will be able to: • convert unsigned integers between bases. Maths skills: 2.1.1 Converting between number bases	Activities • Recap binary from last lesson • Discuss the place values in hexadecimal and how they can be used to make a unique representation for each unsigned integer • Teacher demonstrates conversions in each direction between hexadecimal, binary and denary • Students identify key terms and make notes on definition, function, characteristics and why number bases are needed • Complete activity 2 on p43 Homework: • Students use Formulator Tarsia to make a matching activity with values in different bases and then swap with each other.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 42–43 AdaCS: https://adacomputerscience.org/concepts/number_conversions_hex?topic=number_representation	Analysis Communication Critical thinking Problem solving	Problem solving Critical thinking Analysis Interpretation
--	---	---	--	---	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 8					
Lesson 1					
Topic 2.1.2: Units of measurement Lesson title: Units of measurement	2.1.2 Units of measurement Students will know and understand: - Definition - Function - Characteristics - Why needed - Benefits - Drawbacks Students will be able to: - use binary units of measurement to express data capacity and file sizes. - use denary units of measurement to denote transmission	Activities - Discuss IEC binary and SI decimal prefixes - Teacher demonstrates how to use binary units of measurement to express data capacity and file sizes - Teacher demonstrates how to calculate the maximum number of states that can be represented by a given number of bits (see p10 for calculations for 1.1.3 addressable memory) - Students identify key terms and make notes on definition, function, characteristics, benefits, drawbacks and why the two systems are needed - Students rank units of measurement in size order and create a diagram showing how to convert between them (see pp. 44–45 for examples) Homework: - Students complete Activity 3 on p46	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 44–46 Craig'n'Dave video : https://www.youtube.com/watch?v=tseF3GYCzqA	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Critical thinking Communication

	speeds. • rank units of measurement in size order. • convert between units of measurement. • interpret and write expressions using units of measurement. Maths skills: 2.1.2 Working with units of measurement				
--	---	--	--	--	--

Week 8

Lesson 2

Topic 2.1.3: Two's complement Lesson title: Two's complement	2.1.3 Two's complement representation of signed numbers Students will know and understand: - Definition - Function - Characteristics - Why needed - Benefits - Drawbacks Students will be able to: - convert signed denary integers into two's complement signed binary and vice versa. - use negation to convert from one sign to another. Maths skills: 2.1.3 Using two's complement	Activities - Discuss how two's complement can be used to represent signed numbers in binary - Teacher demonstrates negation to convert between positive and negative (see worked example 3 on p46) - Option to demonstrate alternative method where MSB represents a negative value - Students identify key terms and make notes on definition, function, characteristics, benefits, drawbacks and why two's complement is needed - Complete Activity 4 on p47 Homework: - Students make up ten signed integers and convert them to two's complement. Also, make up ten 8-bit patterns and convert them to their denary equivalent.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 46–47 Craig'n'Dave video: https://www.youtube.com/watch?v=JTyzCpAmn0k&list=PLCiOXwirraUCo6-YUFGk34mAinrmC7y6j&index=22	Analysis Communication Critical thinking Problem solving	Critical thinking Analysis Problem solving
---	---	---	---	--	---

Week 8

Lesson 3

Topic 2.1.4: Fixed-point representation Lesson title: Fixed-point representation	2.1.4 Fixed-point representation of signed fractional numbers Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation Students will be able to: • convert signed fractional denary numbers to fixed- point signed binary and vice versa. Maths skills: 2.1.4 Using fixed-point representation	Activities • Discuss how fixed-point binary can be used to represent signed fractional numbers • Teacher demonstrates how to convert denary to binary and vice versa (see worked example 4 on p46 and 5 on p47) • Students identify key terms and make notes on definition, function, characteristics and why fixed-point representation is needed • Use AI to generate practice questions to complete Homework: • Students use Formulator Tarsia to make a matching activity with values in denary / binary and then swap with each other.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 47–48 IsaacCS: https://isaaccomputerscience.org/concepts/data_numbases_fixed_point?topic=number_representation	Analysis Communication Critical thinking Problem solving	Analysis Problem solving Critical thinking
---	--	--	--	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 9					
Lesson 1					
2.2 Binary Arithmetic Topic 2.2.1: Addition and subtraction Lesson title: Addition and subtraction in binary	Addition and subtraction Students will know and understand: - Definition - Function - Operation Students will be able to: - perform addition and subtraction on signed and unsigned binary numbers. Maths skills: 2.2.1 Performing binary addition and subtraction	Activities - Discuss how addition and subtraction can be performed by a computer using binary - Teacher demonstrates how to perform addition and subtraction on signed and unsigned binary numbers (see worked examples on pp. 49–51) - Students identify key terms and make notes on definition, function and operation of addition and subtraction Homework: - Students complete the questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 48–52 AdaCS: https://adacomputerscience.org/concepts/number_arithmetic?topic=number_representation	Analysis Communication Critical thinking Problem solving	Analysis Problem solving Critical thinking

Week 9

Lesson 2

Topic 2.2.1: Addition and subtraction Lesson title: Further practice on addition and subtraction in binary	Addition and subtraction Students will know and understand: • Definition • Function • Operation Students will be able to: • perform addition and subtraction on signed and unsigned binary numbers. Maths skills: 2.2.1 Performing binary addition and subtraction	Activities • Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding, including converting between positive and negative values in two's complement (Topic 2.1.3) • Option to extend beyond the spec into multiplication for capable students • Use AI to generate practice questions for students to complete • Students create a video explaining the processes Homework: • Complete Activity 5 on p52		Analysis Communication Critical thinking Problem solving	Critical thinking Analysis Problem solving Communication
--	--	--	--	---	---

Week 9

Lesson 3

Topic 2.2.2: Bitwise manipulation Lesson title: Bitwise manipulation	Bitwise manipulation Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation Students will be able to: - perform binary shifts. - use a mask to check the status of bits in a binary pattern. Maths skills: 2.2.2 Performing bitwise manipulation	Activities - Discuss logical and arithmetic shifts - Teacher demonstrates binary shifts (see worked examples on pp. 52–54) - Students identify key terms and make notes on definition, function, characteristics, operation and why binary shifts are needed - Complete Activity 6 on p54 (to practise binary shifts) - Discuss bitwise masks (AND, OR, XOR) - Teacher demonstrates bitwise masks (see worked examples on pp. 55–56) - Students make notes on definition, function, characteristics, operation and why bitwise masks are needed - Complete Activity 7 on p57 Homework: - Students complete remaining activities from the lesson, or create further examples to practise using AI	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 52–57 Christopher Lum video: https://www.youtube.com/watch?v=4JgtUf5ThqY Geeks for Geeks notes (bit masking): https://www.geeksforgeeks.org/dsa/what-is-bitmasking/	Analysis Communication Critical thinking Problem solving	Critical thinking Problem solving Interpretation Analysis
---	---	---	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 10 Lesson 1					
Topic 2.2.3: Overflow and underflow 2.3 Text 2.3.1 Character encoding systems Lesson title: Overflow and underflow.	Overflow and underflow Students will know and understand: • Definition • Function • Characteristics Character encoding systems Students will know and understand: • Definition • Function • Characteristics • Why needed • Benefits • Drawbacks	Activities • Discuss the causes and effects of overflow and underflow errors • Students identify key terms and make notes on definition, function and characteristics • Use AI to generate some examples of calculations that can give rise to overflow or underflow errors • Discuss the use of character encoding systems (ASCII and UTF-8/Unicode) for representing text in binary • Students identify key terms and make notes on definition, function, characteristics, benefits and drawbacks as well as why these systems are needed Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 57–61 IsaacCS notes: https://isaacomputerscience.org/topics/text_representation	Analysis Communication Critical thinking Problem solving	Interpretation Analysis Communication

Week 10

Lesson 2

Topic 2: Data representation	Topic 2: Data representation topic summary	Activities • Use mind maps or knowledge organisers to summarise concepts • Search for quizzes to test knowledge and understanding • Use flashcards to learn definitions of key terms • Use questions from sample and specimen exam papers to test exam readiness • Complete Checkpoint questions on p61		Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity
Lesson title:					
Topic summary					

Week 10

Lesson 3

3.1 Network Fundamentals Topic 3.1.1: Network components a) NIC b) Switch c) Home router d) Modem e) WAP Lesson title: Network hardware devices	3.1.1 Network components: NIC, switch, home router, modem, WAP Students will know and understand: - Definition - Function - Why needed - Characteristics - Operation - Benefits - Drawbacks	Activities - Discuss the hardware devices involved in the operation of a network (NIC, switch, home router, modem, WAP) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why the components are needed - Option to look at examples of such devices if any are available Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 63–65 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/net_hard_devices?topic=network_hardware	Analysis Communication Critical thinking Problem solving	Self-direction Intellectual interest and curiosity Analysis
Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 11 Lesson 1					
Topic 3.1.1: Network components f) Transmission media	Network components: transmission media Students will know and understand: Definition	Activities - Recap the devices introduced last lesson and discuss the different transmission media that can be used for a network (copper, wireless and fibre) - Option to look at examples, if available	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 66	Analysis Communication Critical thinking Problem solving	Decision making Intellectual interest and curiosity Analysis

Lesson title: Network transmission media	- Function - Why needed - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - interpret and complete a diagrammatic representation of a network.	- Teacher demonstrates a diagram representing the structure of a network (see worked example p68) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits, drawbacks and why different media are needed - Complete Activity 2 on p69 Homework: - Students complete questions on AdaCS	AdaCS notes and questions: https://adacomputerscience.org/concepts/network_wired_wireless?topic=networking		
---	--	--	--	--	--

Week 11

Lesson 2

Topic 3.1.1: Network components g) Firewall Lesson title: Firewalls	Network components: firewall Students will know and understand: - Definition - Function - Why needed - Characteristics - Operation - Benefits - Drawbacks	Activities - Discuss firewalls (both hardware and software) - Option to speak to the centre's IT staff about the structure of the network - Students identify key terms and make notes on definition, function, characteristics, operation, benefits, drawbacks and why firewalls are needed - Complete activity 1 on p67 Homework:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 67 Geeks for Geeks notes: https://www.geeksforgeeks.org/computer-networks/introduction-of-firewall-in-computer-	Analysis Communication Critical thinking Problem solving	Interpretation Intellectual interest and curiosity Analysis
--	--	--	---	--	--

- | | | | | | |
|--|--|--|--|--|--|
| | | <ul style="list-style-type: none">Students complete questions on AdaCS | <p><u>network/</u></p> <p>AdaCS notes and questions:</p> <p><u>https://adacomputerscience.org/concepts/sec_firewalls?topic=security</u></p> | | |
|--|--|--|--|--|--|

Week 11

Lesson 3

Topic 3.1.2: Network metrics A) Bandwidth B) Speed/ transfer rate Lesson title: Network bandwidth and speed	Network metrics: bandwidth, speed/transfer rate Students will know and understand: - Definition - Characteristics Students will be able to: - develop expressions involving file size, transfer rate and time.	Activities - Discuss the measuring of network performance using metrics such as bandwidth and speed/ transfer rate - Option to investigate the bandwidth and speed of students home networks - Teacher demonstrates how to develop an expression work file size, transfer rate and time (see worked example 2 on p71) - Students identify key terms and make notes on the definition and characteristics - Complete Activity 3 on p71 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 68–71 ArvighQ video: https://www.youtube.com/watch?v=A_-L-kn9biw AdaCS notes and questions: https://adacomputerscience.org/concepts/comms_characteristics?topic=communication	Analysis Communication Critical thinking Problem solving	Critical thinking Self-direction Problem solving
---	--	--	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 12 Lesson 1					
Topic 3.1.2: Network metrics c) Latency d) Range Lesson title: Network latency and range	Network metrics: latency, range Students will know and understand: - Definition - Characteristics	Activities - Recap network metrics covered last lesson and introduce latency and range - Option to explore latency in a network using tools like ping, traceroute, etc - Students identify key terms and make notes on definition and characteristics - Complete Activity 4 on p72 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 71–72 TechQuickie video: https://www.youtube.com/watch?v=2LOkI3Xyd_E AdaCS notes and questions: https://adacomputerscience.org/concepts/comms_characteristics?topic=communication	Analysis Communication Critical thinking Problem solving	Analysis Intellectual interest and curiosity Self-direction

Week 12

Lesson 2

Topic 3.1.3: Network addressing Lesson title: Network addressing	Network addressing Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation Students will be able to: • develop expressions involving network addresses.	Activities • Discuss network addressing including IP addresses (v4 and v6) and MAC addresses • Option to recap binary and hexadecimal conversions (2.1.1) • Teacher demonstrates developing expressions involving network addresses • Students identify key terms and make notes on definition, function, characteristics, operation and why network addressing is needed Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 72–74 IsaacCS notes: https://isaacomputerscience.org/concepts/net_internet_ip_addresses?topic=the_internet	Analysis Communication Critical thinking Problem solving	Analysis Problem solving Critical thinking
---	--	---	--	--	---

Week 12

Lesson 3

3.2 Encryption	Encryption fundamentals	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 74–75, 78–79	Analysis Communication Critical thinking Problem solving	Decision making Communication Problem solving
Topic 3.2.1: Encryption fundamentals	Students will know and understand: • Definition • Function • Characteristics • Why needed • Benefits • Drawbacks	• Discuss what encryption is as an overview • Students identify key terms and make notes on definition, function, characteristics, benefits and drawbacks as well as why encryption is needed. • Students complete Activity 5 on p78 • Discuss the monoalphabetic substitution cipher (as the first of several approaches) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks • Teacher demonstrates how to encrypt/decrypt messages with a monoalphabetic substitution cipher (see worked example on p78)	AdaCS notes and questions: https://adacomputerscience.org/concepts/encrypt_fundamentals?topic=encryption		
Topic 3.2.3: Monoalphabetic substitution cipher	Monoalphabetic substitution cipher Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks Students will be able to: • encrypt a plaintext message/decrypt a ciphertext message using a monoalphabetic substitution cipher.	Homework: • Students complete Activity 6 on p79			
Lesson title: Encryption fundamentals. Monoalphabetic substitution cipher					

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 13 Lesson 1					
Topic 3.2.4: Vernam cipher algorithm Lesson title: Vernam cipher algorithm	Vernam cipher algorithm Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - encrypt a plaintext message/ decrypt a ciphertext message using a Vernam cipher.	Activities - Teacher demonstrates how to encrypt a message with a Vernam cipher (see worked example on p78) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks - Option to explore websites like DCode or BlackChamber which will allow more practical examples of encryption and decryption with many different encryption approaches Homework: - Students complete Activity 7 on p81	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 79–81 MrBrown CS video: https://www.youtube.com/watch?v=cpqwp2H0SNo	Analysis Communication Critical thinking Problem solving	Communication Problem solving Analysis

Week 13

Lesson 2

Topic 3.2.2: Methods of encryption a) Symmetric Lesson title: Symmetric encryption	Methods of encryption: symmetric Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - interpret and complete a diagrammatic representation of the encryption/ decryption process.	Activities - Discuss symmetric encryption using a known shared key or key exchange - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why symmetric encryption is needed Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 75–76 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/net_sec_encryption?topic=security	Analysis Communication Critical thinking Problem solving	Analysis Problem solving Communication
---	--	---	--	--	---

Week 13

Lesson 3

Topic 3.2.2: Methods of encryption b) Asymmetric Lesson title: Asymmetric encryption	Methods of encryption: asymmetric Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - interpret and complete a diagrammatic representation of the encryption/decryption process.	Activities - Discuss how a public/private key pair is used in asymmetric encryption - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why asymmetric encryption is needed Homework: - Students complete questions from IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 76–78 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/net_sec_encryption?topic=security Computerphile video: https://www.youtube.com/watch?v=GSIDS_lvRv4	Analysis Communication Critical thinking Problem solving	Analysis Problem solving Communication
---	--	---	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 14 Lesson 1					
Topic 3: Networks and Encryption Lesson title: Topic summary	Topic 3: Networks and Encryption topic summary	Activities • Use mind maps or knowledge organisers to summarise concepts • Search for quizzes to test knowledge and understanding • Use flashcards to learn definitions of key terms • Use questions from sample and specimen exam papers to test exam readiness • Complete Checkpoint questions on 081		Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity

Week 14

Lesson 2

4.1 Relational Databases Topic 4.1.1: Entity-relationship model a) Entity b) Attribute Lesson title: Entity-relationship model	Entity-relationship model: entity, attribute Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation	Activities - Discuss how entities and attributes are used in entity-relationship modelling - Students identify key terms and make notes on definition, function, characteristics, operation and why entity-relationship modelling is needed - Students discuss in pairs the entities and attributes that they can think of in various scenarios for example a school, hotel or supermarket Homework: - Students write up the entities and attributes they identified in one scenario during discussion in class	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 83–85 Geeks for Geeks notes: https://www.geeksforgeeks.org/dbms/introduction-of-er-model/ AdaCS notes and questions: https://adacomputerscience.org/concepts/dbs_entity?topic=databases	Analysis Communication Critical thinking Problem solving	Analysis Critical thinking Interpretation
---	--	---	---	--	--

Week 14

Lesson 3

Topic 4.1.1: Entity-relationship model c) Key Lesson title: Entity-relationship model	Entity-relationship model: key Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation	Activities • Discuss primary, foreign and composite keys • Students identify key terms and make notes on definition, function, characteristics, operation and why keys are needed • Complete Activity 1 on p87 Homework: • Students complete questions from IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 85–87 IsaacCS notes: https://isaacomputerscience.org/concepts/data_dbs_concepts_advanced?topic=databases	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation
--	---	---	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 15 Lesson 1					
Topic 4.1.1: Entity-relationship model d) Relationship cardinality Lesson title: Entity-relationship model	Entity-relationship model: relationship cardinality Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation	Activities - Discuss the cardinality of relationships (one-to-one/1:1, one-to-many/1:M, many-to-many/M:M) - Students identify key terms and make notes on definition, function, characteristics, operation and why cardinality is needed - Complete activity 3 on p91 Homework: - Students identify relationships that exist in the scenario they chose two lessons ago and determine the cardinality	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 87–88 Geeks for Geeks notes: https://www.geeksforgeeks.org/dbms/introduction-of-er-model/	Analysis Communication Critical thinking Problem solving	Critical thinking Analysis Interpretation

Week 15

Lesson 2

Topic 4.1.1: Entity-relationship model e) Ways of representing Lesson title: Entity-relationship model	Entity-relationship model: ways of representing Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation Students will be able to: - interpret, complete, find and correct errors in an ERD. - interpret, complete, find and correct errors in a schema expressed in parenthetical notation.	Activities - Discuss the user of entity-relationship diagrams and parenthetical notation to represent ERM - Teacher demonstrates ERD notation (see worked example p90) - Teacher demonstrates parenthetical notation (see worked example p92) - Students identify key terms and make notes - Complete Activity 2 on p90 Homework: - Students complete questions from AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 88–92 AdaCS notes and questions: https://adacomputerscience.org/concepts/dbs_entity?topic=databases	Analysis Communication Critical thinking Problem solving	Critical thinking Communication Analysis
--	---	--	--	--	---

Week 15

Lesson 3

Topic 4.1.2: Logical schemas a) Table / relation b) Column / attribute c) Row / tuple d) Data types Lesson title: Logical schemas	Logical schemas: table/relation, column/attribute, row/tuple, data types Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation	Activities • Discuss how logical schemas can represent the design of a database (including table/relation, column/attribute, row/tuple, data types) • Students identify key terms and make notes on definition, function, characteristics, operation and why logical schemas are needed Homework: • Students make flashcards covering all the definitions introduced in topic 4.1	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 92–93 Geeks for Geeks notes: https://www.geeksforgeeks.org/dbms/database-schemas/	Analysis Communication Critical thinking Problem solving	Analysis Problem solving Communication
---	---	--	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 16					
Lesson 1					
Topic 4.1.2: Logical schemas e) Integrity constraints Lesson title: Logical schemas	Logical schemas: integrity constraints Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation	Activities • Discuss integrity constraints (entity, referential and domain) • Students identify key terms and make notes on definition, function, characteristics, operation and why integrity constraints are needed • Discuss examples of constraints that might be needed to ensure integrity in one of the examples covered earlier in this topic Homework: • Students write up constraints on their chosen example and classify the type of constraint	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 92–93 Geeks for Geeks notes: https://www.geeksforgeeks.org/dbms/difference-between-entity-constraints-referential-constraints-and-semantic-constraints/	Analysis Communication Critical thinking Problem solving	Interpretation Communication Intellectual interest and curiosity

Week 16

Lesson 2

4.2 Data Structures Topic 4.2.1: Data structures b) List d) Record Lesson title: Lists and records	Data structures: List, record Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - hand trace operations on one- and two-dimensional data structures. - show the contents of a data structure after operations have been performed.	Activities - Discuss the need for data structures with different properties to represent data (starting with list and record) - Option to link to programming work in Unit 2 Topic 8.1.2 - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 93–94 IsaacCS notes and questions (lists): https://isaacomputerscience.org/concepts/dsa_datastruct_list?topic=data_structures IsaacCS notes and questions (records): https://isaacomputerscience.org/concepts/dsa_datastruct_record?topic=data_structures	Analysis Communication Critical thinking Problem solving	Analysis Decision making Problem solving
---	---	--	---	--	---

Week 16

Lesson 3

Topic 4.2.1: Data structures C) Dictionary E) Tuple Lesson title: Dictionaries and tuples	Data structures: Tuple, dictionary Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks Students will be able to: • hand trace operations on one- and two-dimensional data structures. • show the contents of a data structure after operations have been performed.	Activities • Discuss dictionary and tuple data structures • Option to link to programming work in Unit 2 Topic 8.2.4 • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: • Students answer questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 94–97 IsaacCS notes and questions (dictionaries): https://isaacomputerscience.org/concepts/dsa_datastruct_dictionary?topic=data_structures IsaacCS notes and questions (tuples): https://isaacomputerscience.org/concepts/dsa_datastruct_tuple?topic=data_structures	Analysis Communication Critical thinking Problem solving	Analysis Decision making Problem solving
---	--	--	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 17					
Lesson 1					
Topic 4.2.1: Data structures A) Array F) Set Lesson title: Arrays and sets	Data structures: Array, set Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks Students will be able to: • hand trace operations on one- and two-dimensional data structures. • show the contents of a data structure after operations have been performed. • perform common operations on sets.	Activities • Discuss array and set data structures • Option to link to programming work in Unit 2 Topic 8.2.7 • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks • Complete Activity 4 on p97 Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 97–99 IsaacCS notes and questions (sets): https://isaacomputerscience.org/concepts/dsa_datastruct_array?topic=data_structures	Analysis Communication Critical thinking Problem solving	Decision making Analysis Problem solving

Week 17					
Lesson 2					
Topic 4.2.1: Data structures Lesson title: Data structures overview	Further practice Comparing and choosing between data structures	Activities - Discuss the data structures covered so far, and which would be appropriate in which circumstances - Use mind maps or knowledge organisers to link concepts - Use questions from sample and specimen exam papers to test exam readiness Homework: - Students use Formulator Tarsia to make a matching activity with data structures concepts and then swap with each other.		Analysis Communication Critical thinking Problem solving	Decision making Interpretation Analysis Critical thinking
Week 17					
Lesson 3					
4.3 Abstract Data Types Topic 4.3.1: Stack A) Structure B) Operations Lesson title: Stacks introduction	Stack Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks	Activities - Discuss the structure and operations (insert, delete, search) of a stack - Option to link to programming work in Unit 2 Topic 8.2.5 - Teacher demonstrates tracing the operations (see worked example on p101) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: - Students complete Q1 from Activity 5 on p107	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 100–103 CS Dojo video: https://www.youtube.com/watch?v=A3ZUpyrnCbM	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Decision making

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 18					
Lesson 1					
Topic 4.3.1: Stack Lesson title: Further practice using stacks	Stack Students will be able to: - hand trace operations on a stack. - show the contents of a stack after operations have been performed.	Activities - Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding - Use AI to generate practice questions for students to complete - Students create a video explaining the processes - Students complete questions on AdaCS	AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_stack?topic=data_structures	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Productivity Communication
Week 18					
Lesson 2					
Topic 4.3.2: Queue a) Structure b) Operations	Queue Students will know and understand: - Definition - Function - Characteristics - Operation	Activities - Discuss the structure and operations (insert, delete, search) of a queue - Option to link to programming work in Unit 2 Topic 8.2.6 - Teacher demonstrates tracing the operations (see worked example on p105) - Students identify key terms and make notes	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 104–109 Computer Science Lessons video:	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Decision making

Lesson title: Queue introduction	• Benefits • Drawbacks	on definition, function, characteristics, operation, benefits and drawbacks Homework: • Students complete Q2 from Activity 5 on p107	https://www.youtube.com/watch?v=jXMqVpAVyMY		
Week 18 Lesson 3					
Topic 4.3.2: Queue Lesson title: Further practice using queues	Queue Students will be able to: • hand trace operations on a queue. • show the contents of a queue after operations have been performed.	Activities • Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding • Use AI to generate practice questions for students to complete • Students create a video explaining the processes • Students complete questions on AdaCS	AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_queue?topic=data_structures	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Productivity Communication

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 19					
Lesson 1					
Topic 4.3.1 and 4.3.2: Stacks and Queues Lesson title: Comparing stacks and queues	Stacks and queues Students will be able to: - hand trace operations on a queue or stack. - show the contents of a queue or stack after operations have been performed.	Activities - Discuss the differences in usage of stacks and queues - Option to carry out programming tasks to illustrate theoretical concepts - Use questions from sample and specimen exam papers to test exam readiness		Analysis Communication Critical thinking Problem solving	Decision making Interpretation Analysis Communication
Week 19					
Lesson 2					
Topic 4: Structuring data Lesson title: Topic summary	Topic 4: Structuring data topic summary	Activities - Use mind maps or knowledge organisers to summarise concepts - Search for quizzes to test knowledge and understanding - Use flashcards to learn definitions of key terms - Use questions from sample and specimen exam papers to test exam readiness - Complete Checkpoint questions on p108		Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity

Week 19

Lesson 3

5.1 Tools and Techniques	Algorithms	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 110–120	Analysis Communication Critical thinking Problem solving	Problem solving Critical thinking Interpretation
Topic 5.1.1: Algorithms	Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation	• Discuss algorithms as a tool for solving problems • Students identify key terms and make notes on definition, function, characteristics, operation and why algorithms are needed • Complete Activity 1 on p118	MrBrown CS video: https://www.youtube.com/watch?v=kUt0nS0yMtM		
Lesson title: Algorithms	Students will be able to: • interpret, complete and develop algorithms expressed in words, diagrams, flowcharts and code. • analyse the efficiency of algorithms. • find and correct logic errors in algorithms.	Homework: • Students create a flowchart for the steps of a calculation they are familiar with from maths or science	Stuff Made Easy video: https://www.youtube.com/watch?v=Yq1OPs5hCt0 AdaCS notes and questions: https://adacomputerscience.org/concepts/compthink_algorithms?topic=computational_thinking		

--	--	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 20					
Lesson 1					
Topic 5.1.2: Trace table Lesson title: Trace table	Trace table Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation Students will be able to: • complete a trace table for an algorithm.	Activities • Discuss the use of trace tables as a tool for solving problems • Teacher demonstrates how to complete a trace table (see worked example on p120) • Students identify key terms and make notes on definition, function, characteristics, operation and why trace tables are needed Homework: • Students search for examples of flowcharts or Python programs which they can trace	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 120–121 TeachingCS video (easier): https://www.youtube.com/watch?v=DyeVR1zb7Jo&list=P_LzxELyfDjyxLawEb5mlA-pEPATIVIFnmB&index=46 TeachingCS video (harder): https://www.youtube.com/watch?	Analysis Communication Critical thinking Problem solving	Analysis Interpretation Problem solving

			<u>v=ac38IM2S6kw&list=PLzxELyfDjyxLawEb5mlA-pEPATIV1FnmB&index=6</u>		
--	--	--	--	--	--

Week 20 Lesson 2					
Topic 5.1.3: Recursion	Recursion	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 121–124	Analysis Communication Critical thinking Problem solving	Critical thinking Analysis Interpretation
Lesson title: Recursion	Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks	• Discuss recursion as a tool for solving problems • Option to link to programming work in Unit 2 topic 10.4.1 • Teacher demonstrates a recursive algorithm (see worked example on p122) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks • Complete Activity 2 on p123 Homework: • Students complete questions on AdaCS	AdaCS notes and questions: https://adacomputerscience.org/topics/recursion		
Week 20 Lesson 3					
Topic 5.1.3: Recursion	Recursion	Activities:		Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation Communication
Lesson title: Further practice of Recursion	Students will be able to: • hand trace the operation of a recursive algorithm.	• Having introduced the concepts last lesson, students will need more time to practise these topics • Students create a video explaining the processes • Students complete questions on AdaCS			

Topic Sub topic	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this	Which skills could be acquired
--------------------	-------------------------	----------------------	---------------------	-------------------------------	--------------------------------

Lesson title				lesson are explicitly assessed through examination?	through teaching and delivery in this lesson?
Week 21 Lesson 1					
Topic 5.1.2: Trace table Lesson title: Trace table	Trace table Students will be able to: - complete a trace table for an algorithm. - hand trace the operation of a recursive algorithm.	Activities - Having introduced the concepts of recursion and trace tables students will need more time to practise these topics - Use AI (or past exam papers) to generate practice questions for students to complete		Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation Communication

Week 21

Lesson 2

Topic 5.1.2: Linear search algorithm a) Unsorted array Lesson title: Linear search on an unsorted array	Linear search algorithm: unsorted array Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks Students will be able to: • hand trace the operation of a linear search algorithm.	Activities • Discuss the use of a linear search for an unsorted array • Option to link to programming work from Unit 2 Topic 10.3.1 • Teacher demonstrates completing a linear search algorithm (See worked example on p125) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 124–125 TeachingCS video: https://www.youtube.com/watch?v=rq-UxKs6Cyk&list=PLzxELyfDjyxLawEb5mlA-pEPATIVIFnmB&index=47 IsaacCS notes and questions: https://isaaccomputer-science.org/concepts/dsa_search_linear?topic=searching	Analysis Communication Critical thinking Problem solving	Analysis Interpretation Problem solving
--	--	---	--	--	--

Week 21

Lesson 3

Topic 5.2.1: Linear search algorithm b) Sorted array Lesson title: Linear search on a sorted array	Linear search algorithm: sorted array Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks Students will be able to: • hand trace the operation of a linear search algorithm.	Activities • Discuss the use of a linear search for a sorted array • Option to link to programming work from Unit 2 Topic 10.3.1 • Teacher demonstrates completing a linear search algorithm (See worked example on p126) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 125–126 AdaCS notes and questions: https://adacomputerscience.org/concepts/search_linear?topic=searching	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation
---	--	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 22					
Lesson 1					
Topic 5.2.2: Binary search algorithm a) Iterative Lesson title: Iterative binary search algorithm	Binary search algorithm: iterative Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - hand trace the operation of a binary search algorithm.	Activities - Discuss an iterative approach to a binary search algorithm - Option to link to programming work from Unit 2 Topic 10.3.2 - Teacher demonstrates tracing a binary search algorithm (see worked example on p127) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: - Students complete Activity 3 on p127	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 126–127 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_search_binary?topic=searching TeachingCS video: https://www.youtube.com/watch?v=Pn3_Ocjhd8s&list=P_LzxELyfDjyxLawEb5mlA-pEPATIVIFnmB&index=48	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation

Week 22

Lesson 2

Topic 5.2.2: Binary search algorithm b) Recursive Lesson title: Recursive binary search algorithm	Binary search algorithm: recursive <i>Students will know and understand:</i> - Definition - Function - Characteristics - Operation - Benefits - Drawbacks <i>Students will be able to:</i> - hand trace the operation of a binary search algorithm.	Activities - Discuss a recursive approach to a binary search algorithm - Option to link to programming work from Unit 2 Topic 10.3.2 - Teacher demonstrates tracing a recursive binary search - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: - Students complete Activity 3 on p127 again, using a recursive approach	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 128 AdaCS notes and questions: https://adacomputerscience.org/concepts/search_binary?topic=searching TeachingCS video: https://www.youtube.com/watch?v=KdjeZgZGI4A&list=P_LzxELyfDjyxLawEb5mlA-pEPATIVIFnmB&index=49	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation
--	---	--	--	--	--

Week 22

Lesson 3

Topic 5.2.3: Bubble sort algorithm Lesson title: Bubble sort algorithm	Bubble sort algorithm Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks	Activities • Discuss the bubble sort algorithm • Option to link to programming work from Unit 2 Topic 10.3.3 • Teacher demonstrates how to trace a bubble sort (worked example on p129) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: • Students complete Activity 4 on p130	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 128–130 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_search_bubble?topic=sorting	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation
---	--	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 23 Lesson 1					
Topic 5.2.3: Bubble sort algorithm Lesson title: Trace tables for a bubble sort algorithm	Bubble sort algorithm Students will be able to: hand trace the operation of a bubble sort algorithm.	Activities - Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding - Use AI to generate practice questions for students to complete - Students create a video explaining the processes - Students complete questions on AdaCS	TeachingCS video: https://www.youtube.com/watch?v=tDReNV2TB4w&list=PLzxELyfDjyxLawEb5mlA-pEPATIVIFnmB&index=50 AdaCS notes and questions: https://adacomputerscience.org/concepts/sort_bubble?topic=sorting	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation

Week 23 Lesson 2					
Topic 5.2.4: Insertion sort algorithm Lesson title: Insertion sort algorithm	Insertion sort algorithm Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks	Activities - Discuss the insertion sort algorithm - Option to link to programming work from Unit 2 Topic 10.3.4 - Teacher demonstrates how to trace an insertion sort (see worked example on p130) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks Homework: - Students complete Activity 5 on p131	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 130–132 CS50 video: https://www.youtube.com/watch?v=O0VbBkUvril AdaCS notes and questions: https://adacomputerscience.org/concepts/sort_insertion?topic=sorting	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation
Week 23 Lesson 3					
Topic 5.1.2: Trace table Lesson title: Trace tables for insertion sort algorithm	Trace table Students will be able to: hand trace the operation of an insertion sort algorithm.	Activities - Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding - Use AI to generate practice questions for students to complete - Students create a video explaining the processes - Students complete questions on AdaCS - Search for videos comparing the efficiency of different sorting algorithms	Computer Science Lessons video : https://www.youtube.com/watch?v=TPyJ0k8y3Nc AdaCS notes and questions: https://adacomputerscience.org/concepts/	Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation

			sort_insertion?topic=sorting		
Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 24 Lesson 1					
5.3 Boolean Logic Topic 5.3.1: Boolean expressions a) Operators Lesson title: Boolean operators and expressions	Boolean expressions: operators Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - interpret and create Boolean expressions. Maths skills: 5.3.1 Using Boolean expressions	Activities - Discuss Boolean operators (AND, OR, XOR, NOT, NAND, NOR) - Teacher demonstrates how each operator works and its truth table (see table on p133) - Students identify key terms and make notes on definition, function, operation as well as why Boolean operators are needed - Students may also need to practise using the on-screen tools for typing Boolean expressions. - Make sure students are taught the appropriate symbols for Boolean logic (see specification appendix 2) Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 132–133 IsaacCS notes and questions: https://isaaccomputerscience.org/concepts/sys_bool_write_expression?topic=boolean_logic CrashCourseCS video : https://www.youtube.com/watch?v=gl-qXk7XojA&t=80s Firchild video: https://	Analysis Communication Critical thinking Problem solving	Interpretation Communication Intellectual interest and curiosity

			www.youtube.com/watch?v=ATg1lUlxrsY&list=PLCNoTwH49OONTB_ZguAhz0zdjKAVAop4b		
Week 24 Lesson 2					
Topic 5.3.1: Boolean expressions b) Order of precedence Lesson title: Order of precedence of Boolean operators	Boolean expressions: order of precedence Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - interpret and create Boolean expressions. Maths skills: 5.3.1 Using Boolean expressions	Activities - Discuss the correct order of precedence for evaluating Boolean operators in an expression - Students identify key terms and make notes on definition, function, operation and why order of precedence is needed Homework: - Students practise generating similar-looking Boolean expressions and considering whether the addition of brackets changes the meaning	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 133–134 Firchild video: https://www.youtube.com/watch?v=V2ZqhLwi5I AdaCS notes and questions: https://adacomputerscience.org/concepts/boolean_write_expression?topic=boolean_logic	Analysis Communication Critical thinking Problem solving	Analysis Communication Intellectual interest and curiosity

Week 24

Lesson 3

Topic 5.3.2: Truth table Lesson title: Using truth tables	Truth table Students will know and understand: • Definition • Function • Why needed • Operation Students will be able to: • interpret and complete a truth table for a given logical expression. Maths skills: 5.3.2 Using truth tables	Activities • Discuss truth tables as a tool for illustrating the output from every combination of inputs for a Boolean expression • Teacher demonstrates how to complete a truth table for a multi-operator expression (see worked examples on pp. 134–136) • Students identify key terms and make notes on definition, function, operation and why truth tables are needed Homework: • Students complete Activity 6 on p136 • Students discuss Case Study 2 on p136	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 134–136 MrBrownCS video: https://www.youtube.com/watch?v=BnB2m1nXZ84	Analysis Communication Critical thinking Problem solving	Critical thinking Problem solving Analysis Interpretation
--	---	---	--	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 25 Lesson 1					
Topic 5.3.2: Truth table Lesson title: Further practice on truth tables	Truth table Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - interpret and complete a truth table for a given logical expression. Maths skills: 5.3.2 Using truth tables	Activities - Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding - Use AI to generate practice questions for students to complete - Students create a video explaining the processes		Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation Productivity Communication

Week 25 Lesson 2					
Topic 5.3.2: Truth table	Truth table Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - interpret and complete a truth table for a given logical expression. Maths skills: 5.3.2 Using truth tables	Activities - Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding - Use AI to generate practice questions for students to complete - Students create a video explaining the processes		Analysis Communication Critical thinking Problem solving	Problem solving Analysis Interpretation Productivity Communication

Week 25

Lesson 3

Topic 5.4.2: Declarative programming paradigm Lesson title: Declarative programming paradigm	Declarative programming paradigm Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities • Discuss the declarative programming paradigm • Teacher demonstrates similarities between code snippets from different programming languages which use this paradigm • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why the declarative programming paradigm is needed Homework: • Students complete Activity 7 on p140	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 139–141 Geeks For Geeks notes: https://www.geeksforgeeks.org/theory-of-computation/difference-between-imperative-and-declarative-programming/	Analysis Communication Critical thinking Problem solving	Decision making Communication Intellectual interest and curiosity
--	--	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 26					
Lesson 1					
Topic 5: Problem- Solving Lesson title: Topic summary	Topic 5: Problem-solving topic summary	Activities • Use mind maps or knowledge organisers to summarise concepts • Search for quizzes to test knowledge and understanding • Use flashcards to learn definitions of key terms • Use questions from sample and specimen exam papers to test exam readiness • Complete the Checkpoint questions on p141		Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity
Week 26					
Lesson 2					
6.1 Data Science Topic 6.1.1: Big Data a) Five Vs	Big data: Five Vs Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits	Activities • Discuss the Five Vs of big data (volume, velocity, variety, veracity, value) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why the Five Vs of big data are needed Homework:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 143–145 Technology Advice video: https://	Analysis Communication Critical thinking Problem solving	Communication Critical thinking Intellectual interest and curiosity Ethics

Lesson title: The five Vs of big data	Drawbacks	Look for quizzes about big data on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	www.youtube.com/watch?v=jlOUUnQS_qbc		
Week 26 Lesson 3					
Topic 6.1.1: Big data b) Infrastructure Lesson title: Big data infrastructure	Big data: infrastructure Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities - Discuss big data infrastructure (collection, storage, processing, modelling and analysis, networking, security) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why infrastructure is needed Homework: - Students complete a mind map linking together the concepts that have been introduced	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 145–150 Intelligence Gateway video: https://www.youtube.com/watch?v=-bEC0Qo24nk	Analysis Communication Critical thinking Problem solving	Analysis Critical thinking Intellectual interest and curiosity

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 27					
Lesson 1					
Topic 6.1.1: Big data b) Infrastructure Lesson title: Further practice on big data infrastructure	Big data: infrastructure Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • interpret and complete a diagrammatic representation of a Big Data system.	Activities • Having introduced the concepts last lesson, students will need more time to practise these topics to reinforce their understanding • Use AI to generate practice questions for students to complete • Students create a video explaining the processes • Students complete Activity 1 on p150		Analysis Communication Critical thinking Problem solving	Ethics Intellectual interest and curiosity Productivity Self-direction

Week 27					
Lesson 2					
Topic 6.1.2: Tools and techniques of data science Lesson title: Tools and techniques of data science	Tools and techniques of data science Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities • Discuss tools and techniques of data science (collection, storage, web scraping, cleaning, analysis, visualisation) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why these tools and techniques are needed Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 151–152 AdaCS notes and questions: https://adacomputerscience.org/concepts/bigdata_mining?topic=big_data	Analysis Communication Critical thinking Problem solving	Analysis Communication Intellectual interest and curiosity Ethics
Week 27					
Lesson 3					
6.2 Artificial Intelligence (AI) Topic 6.2.1: Categories of AI Lesson title: Categories of AI	Categories of AI Students will know and understand: • Definition • Characteristics • Why needed • Benefits • Drawbacks	Activities • Discuss categories of AI (narrow, general, super) • Students identify key terms and make notes on definition, characteristics, benefits and drawbacks as well as why different categories of AI are needed • Students discuss Case Study 1 on p152 Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 152–154 AdaCS notes and questions: https://adacomputerscience.org/concepts/ai_categories?topic=artificial_intellig	Analysis Communication Critical thinking Problem solving	Ethics Intellectual interest and curiosity Interpretation

Week 28

Lesson 2

Topic 6.2.2: Machine learning (ML) b) Unsupervised Lesson title: Unsupervised machine learning	Machine learning: unsupervised Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - interpret and complete a diagrammatic representation of an unsupervised ML system.	Activities - Discuss unsupervised machine learning (ML) - Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why unsupervised ML is needed Homework: - Students complete Activity 2 on p157	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 155–157 IBM Technology video: https://www.youtube.com/watch?v=W01tIRP_Rqs	Analysis Communication Critical thinking Problem solving	Decision making Ethics Critical thinking Intellectual interest and curiosity
---	---	---	---	--	--

Week 28

Lesson 3

Topic 6.2.3: Large language model (LLM) a) Accuracy b) Knowledge limitations Lesson title: Accuracy and knowledge limitations of large language models	Large language model: context and query accuracy, knowledge limitations Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • interpret and complete a diagrammatic representation of a LLM system.	Activities • Discuss accuracy and knowledge limitations of large language models (LLM) • Option to experiment with available LLM resources • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why LLMs are needed Homework: • Students complete Activity 3 on p160	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 157–158 AdaCS notes and questions: https://adacomputerscience.org/concepts/ai_turing_test?topic=artificial_intelligence	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Ethics Self-direction Critical thinking Communication
--	---	---	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 29					
Lesson 1					
Topic 6.2.3: Large language model c) Bias/ Subject- ivity Lesson title: Bias/ subjectivity of large language models	Large language model: bias/subjectivity Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • interpret and complete a diagrammatic representation of a LLM system.	Activities • Discuss bias/subjectivity of LLMs • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why an understanding of bias/subjectivity of LLMs is needed • Students search for news stories and recent articles on bias/subjectivity in LLMs Homework: • Students write up a case study they have found in their research	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 158–160 IBM Technology video: https://www.youtube.com/watch?v=dAE7OFm9oek	Analysis Communication Critical thinking Problem solving	Ethics Critical thinking Intellectual interest and curiosity Communication

Week 29

Lesson 2

6.3 Tools	Integrated development environment	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 160–163	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Productivity Problem solving
Topic 6.3.1: Integrated development environment (IDE)	Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	• Discuss the use and features of IDEs (text editor, autocomplete, syntax checker, debugger breakpoints and stepping, memory inspection) • Option to link to programming work from Unit 2 Topic 9.3.1 • Teacher demonstrates features of the students' familiar IDE • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why IDEs are needed	AdaCS notes and questions: https://adacomputerscience.org/concepts/soft_ide?topic=software		
Lesson title: Integrated development environment		Homework: • Students complete Activity 4 on p161			

Week 29

Lesson 3

Topic 6.3.2: Version control system (VCS) Lesson title: Version control system	Version control system Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities • Discuss version control systems • Option to demonstrate a version control system (but no direct practical experience is expected) • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why VCSs are needed Homework: • Look for quizzes about machine learning on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 164 AdaCS notes and questions: https://adacomputerscience.org/concepts/soft_ide?topic=software Udacity video: https://www.youtube.com/watch?v=zbKdDsNNOhg	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking
--	--	---	---	--	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 30 Lesson 1					
Topic 6.3.3: Open-source technologies Lesson title: Open-source technologies	Open-source technologies Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities • Discuss open-source technologies • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why open-source technologies are needed Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 165 IsaacCS notes: https://isaacomputerscience.org/concepts/sys_os_open_closed_source?topic=software AdaCS notes and questions: https://adacomputerscience.org/concepts/soft_open_closed_source?topic=software	Analysis Communication Critical thinking Problem solving	Ethics Intellectual interest and curiosity Critical thinking

Week 30

Lesson 2

Topic 6.3.3: Open-source technologies Lesson title: Open-source technologies	Open-source technologies Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities • Discuss open-source technologies • Students identify key terms and make notes on definition, function, characteristics, operation, benefits and drawbacks as well as why open-source technologies are needed Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 165 IsaacCS notes: https://isaacomputerscience.org/concepts/sys_os_open_closed_source?topic=software AdaCS notes and questions: https://adacomputerscience.org/concepts/soft_open_closed_source?topic=software	Analysis Communication Critical thinking Problem solving	Ethics Intellectual interest and curiosity Critical thinking
--	--	---	---	--	---

Week 30

Lesson 3

Topic 6: Enabling technologies Lesson title: Topic summary	Topic 6: Enabling technologies topic summary	Activities • Use mind maps or knowledge organisers to summarise concepts • Search for quizzes to test knowledge and understanding • Use flashcards to learn definitions of key terms • Use questions from sample and specimen		Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest
--	---	--	--	--	---

- | | | | | | |
|--|--|--|--|--|---------------|
| | | exam papers to test exam readiness <ul style="list-style-type: none"> Complete Checkpoint questions on p168 | | | and curiosity |
|--|--|--|--|--|---------------|

Unit 2: Practical Programming and Problem Solving

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 1					
Lesson 1					
Topic 7.1.1: Constructs a) Sequence b) Selection Lesson title: Flow control constructs – Sequence and selection	Students will be able to: 7.1.1 select and use appropriate flow control constructs Maths skills: using appropriate operators	Activities - Discuss how common computing tasks can, at a high level, be broken down into different constructs. Tasks such as sending an email, a robot vacuum cleaner cleaning a room, a simple computer game (space invaders, Pac-man) - Teacher demonstration of the use of the IDLE Python IDE software - Teacher demonstration of the creation of simple Python code using the 'if' construct - Practical classroom activity creating simple Python programs using the 'if' construct with extension activities for more capable learners - Quiz on program constructs, Python IDLE, basic Python syntax and simple if statements Homework:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 170–173 Video explaining if statements: https://www.youtube.com/watch?v=FvMPfrgGeKs	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

- Ask students who have a computer or laptop at home to download and install the Python software on it.

Week 1

Lesson 2

Topic 7.1.1: Constructs b) Selection Lesson title: Flow control constructs Selection, match constructs	Students will be able to: 7.1.1 select and use appropriate flow control constructs Maths skills: using appropriate operators	Activities • Discuss applications students may be familiar with (games, mobile phone apps etc) and where these might require if constructs • Teacher demonstration of more complex 'if' statements (using if..elif..else) • Teacher demonstration of the use of the input function and the match selection construct • Practical classroom activity creating Python programs using more complex 'if' and match constructs with extension activities for more capable learners (see activity 1, student book 1, p173 for some examples) • Class discussion on getting programs to work, what do error messages mean, how to debug a program, common mistakes (e.g. syntax errors) Homework: • Search online for 'GeeksforGeeks' Python if else statement and then read the article on the Geeks for Geeks web site about Python conditional statements	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 173–174 Video explaining match statements: https://www.youtube.com/watch?v=FEaTlAYJAm0 Web page explaining conditional statements: https://www.geeksforgeeks.org/python/python-if-else/	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
--	--	---	--	---	---

Week 1 Lesson 3					
Topic 7.1: Constructs c) Count controlled iteration Lesson title: Flow control constructs – count controlled iteration	Students will be able to: 7.1.1 select and use appropriate flow control constructs Maths skills: using appropriate operators	Activities - Teacher presentation on iteration constructs and the different types of loops - Teacher demonstration of the use of count-controlled iteration in a Python program - Practical classroom activity creating Python programs using count-controlled iteration with extension activities for more capable learners. (see activity 2 in student book 1 page 176 for an example) Homework: - Ask students to create a brief guide (which can be used for revision) or wall poster on the syntax for a Python ‘for’ loop.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 174–176 Video explaining count controlled (for) loops: https://youtu.be/94UHCEmprCY?si=8d9ILZ999porxVUN	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 2 Lesson 1					
Topic 7.1: Constructs d) Condition controlled iteration Lesson title: Flow control constructs – condition controlled iteration	Students will be able to: 7.1.1 select and use appropriate flow control constructs Maths skills: using appropriate operators	Activities - Teacher demonstration of the use of condition-controlled iteration in a Python program - Practical classroom activity creating Python programs using condition-controlled iteration with extension activities for more capable learners. (see activity 3 in student book 1 page 177 for an example) - Class discussion, iteration is a useful construct, but what problems can using iteration cause (e.g. use excessive CPU cycles, endless loops)? How can you avoid these problems? Homework: - Ask student to draw a flowchart describing a simple program that includes both selection and iteration	Pearson Edexcel International A-level Computer Science Student Book: pp. 176–177 Video explaining condition controlled (while) loops: https://www.youtube.com/watch?v=KsiWTrZQFXg	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 2

Lesson 2

Topic 7.1: Constructs e) Exception handling Lesson title: Flow control constructs – Exceptions	Students will be able to: 7.1.1 select and use appropriate flow control constructs	Activities - Teacher presentation on the reason why exceptions occur (demonstrate an exception occurring and ask student what has happened). Ask student why exceptions should be avoided. - Teacher demonstration of how exception handling can be included in a Python program - Practical classroom activity creating and/or modifying Python programs to include exception handling. Students can add exception handling to programs they have previously written. (see activity 4 in student book 1 page 178 for an example) - Class discussion on why exceptions which cause a program to crash should be avoided (user confusion, leaves an impression of poor-quality software, security issue – leads into homework task) Homework: - Research activity: Why can exceptions be a security risk? Can you find any examples of how exceptions can be utilised by hackers?	Pearson Edexcel International A– level Computer Science Student Book 1: pp. 177–179	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
--	--	---	--	---	---

Week 2

Lesson 3

Topic 7.1.2: Variables and constants	Students will be able to: 7.1.2 select and use appropriate variables and constants	Activities - Teacher presentation on variables and constants including naming conventions - Practical class activity writing programs including features covered so far (selection, iteration, exceptions, constants etc.) - Practical class activity, programs provided using features covered so far with syntax and logic errors that students need to find and correct (important skill for the exam) - Quiz covering Python features covered so far (student book 1, Activity 5 p180 could be used for some of the questions)	Pearson Edexcel International A- level Computer Science Student Book 1: pp. 179–180	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
Lesson title: Variables and constants	Maths skills: using appropriate operators	Homework: Ask students to produce a guide or wall display on common syntax errors (e.g. use of == for equals in comparisons, use of colons before conditional and iteration code blocks, indentation errors etc)			

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 3					
Lesson 1					
Topic 7.1.3: Operators Lesson title: Arithmetic and relational operators	Students will be able to: 7.1.3 select and use appropriate operators Maths skills: using appropriate operators	Activities - Teacher demonstration on arithmetic and relational operators (students will have already come across some of these so this will be a recap and covering any that have not yet been used) - Practical classroom activity creating simple Python programs using arithmetic and relational operators - Practical classroom activity creating more complex Python programs using selection and iteration constructs using relational operators - Quiz with students having to correctly identify a true or false outcome from if construct using relational operators. Homework: - Ask students to produce a guide or wall poster listing and giving an example of the arithmetic and relational operators.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 180–182	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 3					
Lesson 2 & 3					
Topic 7.1.3: Operators	Students will be able to: 7.1.3 select and use appropriate operators	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 182–186	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
Lesson title: Boolean and bitwise operators (double lesson)	Maths skills: using appropriate operators	- Teacher demonstration on Boolean and bitwise operators. - Practical classroom activity creating Python programs using Boolean and relational operators combined and bitwise operators with extension activities for more capable learners. (student book 1, Activity 6 p183 gives an example) - Classroom activity, students review worked example 3 on page 187 of student book 1 and discuss: - how the game shown could be developed into a playable game? - Other uses bitwise operators might be put to Homework: - Ask students to search for 'Python bitwise operators' and then read the article on the Real Python web site titled 'Bitwise operators in Python'. This article has some background information about binary and number formats (related to Unit1) and more detail about bitwise operators and their uses including LSB Steganography which might interest more advanced learners	Theory of bitwise operations is covered in Unit 1, 2.2.2 Video explaining relational, Boolean and membership operators: https://youtu.be/IPVke-p4S7s?si=ck__NvOiT8gzoKnh Video explaining bitwise operators: https://youtu.be/F_UIOvfmsew?si=Qxqc3WsachhE2K_E Web page explaining bitwise operators: https://		

			<u>realpython.com/ python-bitwise- operators/</u>		
--	--	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 4					
Lesson 1					
Topic 7.1.4: Subprograms	Students will be able to: 7.1.4 select and use appropriate subprograms	Activities - Teacher demonstration on creating and using subprograms. Explain the meaning of terminology including function, procedure, parameters and arguments. - Practical classroom activity creating Python programs using subprograms and/or rewriting programs previously written to use subprograms - Class discussion. What are the benefits of using subprograms? When should they be used?	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 186–189 Video explaining subprograms (functions): https://youtu.be/KW6qncswzHw?si=EA2CHGHE72grGYhT	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
Lesson title: Using subprograms		Homework: Set students a programming challenge which can be implement using subprograms.			

Week 4 Lesson 2					
Topic 8.1.1 Datatypes Lesson title: Python datatypes	Students will be able to: 8.1.1 select and use appropriate datatypes. Convert between datatypes Maths skills: handling numeric data	Activities - Teacher presentation on datatypes. Datatypes will probably already have been mentioned but this is an opportunity to formally cover all the datatypes and how to convert between them. (this lesson can be combined with the next lesson to give more time for the programming projects if required) - Class activity with students writing code to demonstrate the use of different datatypes and how values can be converted between them. - Class discussion on datatypes. How are different datatypes stored on a computer? Why is getting the correct datatype important? What problems can occur if the wrong datatype is used? Homework: - Ask students to look at some web-based forms they might fill in (such as applying for a bank account, making an online purchase or registering for an online service) and identify what datatypes are likely to be used for each input.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 192–193	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 4

Lesson 3

Topic 7/8 Programming Lesson title: Implementa- tion project (double lesson)	Students will be able to: 7.1.1 select and use appropriate flow control constructs 7.1.2 select and use appropriate variables and constants 7.1.3 select and use appropriate operators 7.1.4 select and use appropriate subprograms 8.1.1 select and use appropriate datatypes. Maths skills: using appropriate operators, handling numeric data	Activities - Teacher presentation on Python features covered so far with a quiz on programming syntax. For example, different Python statements or code snippets could be shown, some with errors and students must identify which ones are correct and which are not (and what the error is) - Practical classroom activity creating some longer 'mini applications' (rather than simple code snippets) implementing the topics covered so far (selection, iteration, exceptions operators, subprograms etc.) - Classroom discussion on progress so far, what have students found most difficult? Hints/tips on avoiding syntax or logic errors. Homework: - Ask students to watch this video which talks about matching career goals to programming languages	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 170–193 https://www.youtube.com/watch?v=8BBHhLiOlus	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
---	--	---	--	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 5					
Lesson 1					
Double lesson continued from Week 4					
Week 5					
Lesson 2					
Topic 8.1.2 Data structures Lesson title: Array and list data structures	Students will be able to: 8.1.2 select and use one and two-dimensional data structures. Select and use one and two-dimensional lists as equivalent to one and two-dimensional arrays. Convert between data structures Maths skills: using sets	Activities - Teacher demonstration on arrays and lists - Practical classroom activity creating Python programs utilising the features of lists, including functions which can be used with lists. - Practical classroom activity including programs provided with syntax and logic errors that students need to find and correct (important skill for the exam) Homework: - Set students a programming challenge using lists and other features covered so far	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 193–197 Video explaining Python lists: https://youtu.be/ohCDWZgNIU0?si=8iAuKxSaibOfckUZ	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 5

Lesson 3

Topic 8.1.2	Students will be able to:	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 197–199	Critical thinking	Critical thinking
Data structures	8.1.2 select and use one and two-dimensional data structures.	- Teacher demonstration on records, tuples and sets (note that set are covered in more detail in 8.2.7 – week 11). - Practical classroom activity creating Python programs utilising the features of tuples, demonstrating the difference between lists and converting between structures - Class quiz on different data structures, features, syntax, relevant functions		Problem solving	Problem solving
Lesson title:	Select and use one and two-dimensional lists as equivalent to one and two-dimensional arrays.			Analysis	Analysis
Record, tuple and set data structures	Convert between data structures		Video explaining the difference Python lists and tuples:	Initiative	Initiative
	Maths skills:	Homework:	https://youtu.be/RaRjZcvbUKE?si=rFTQIUyMuaHQn-78	Productivity	Productivity
	using sets	Ask students to search for ‘Python lists vs sets vs tuples’ then read then read the article on the Geeks for Geeks web site about lists, sets and tuples.			

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 6					
Lesson 1					
Topic 8.1.2 Data structures Lesson title: Two-dimensional data structures	Students will be able to: 8.1.2 select and use one and two-dimensional data structures. Select and use one and two-dimensional lists as equivalent to one and two-dimensional arrays. Maths skills: Convert between data structures	Activities - Teacher demonstration on two-dimensional lists - Practical classroom activity creating Python programs utilising 2D lists. (student book 1, Activity 1, p201 gives an example). This should include a task where students need to choose the appropriate data structure for a given requirement. - A quick fun game of 'bingo' with a 2D table with words (such as colours or football players) in it. The teacher flashes up coordinates, and the learners shout out the corresponding word (or vice versa) Homework: - Ask students to produce a guide or wall poster identifying the differences between the various data structures covered.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 199–200 Video explaining 2D data structures: https://youtu.be/Xy6qeQWQwFw?si=TVaOIXmhYMRgEE63	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 6

Lesson 2

Topic 8.2.1 Methods to handle numeric data Lesson title: Dealing with numbers	Students will be able to: 8.2.1 select and use appropriate methods for handling numeric data Maths skills: handling numeric data	Activities - Teacher demonstration on rounding, truncation and randomisation - Practical classroom activity creating Python programs demonstrating the use of rounding, truncation and randomisation (student book 1, Activity 2, p207 gives an example). - Classroom discussion on potential use of random numbers in games. Ask students to come up with some ideas for games that could be implemented using their current skill level. Homework: - Ask students to take one of the ideas for games a little further, perhaps creating a flow chart or storyboard showing the game play	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 202–204 Video explaining creating random numbers: https://www.youtube.com/watch?v=piJc18hcHOY&t=52s	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
---	--	--	---	---	---

Week 6

Lesson 3

Topic 8.2.1 Methods to handle numeric data Lesson title: Using the math library	Students will be able to: 8.2.1 select and use appropriate methods for handling numeric data Maths skills: handling numeric data	Activities - Teacher presentation on maths library functions. This is a good point to show learners the online official python documentation at https://docs.python.org/3/ which explains all the math module functions in detail. - Practical classroom activity creating Python programs demonstrating the use of maths library functions. - Class discussion on the possible applications of some of the functions covered (e.g. square root, absolute value, pi, raise to powers etc) Homework: - Ask students to search for 'Python maths module' and then look at the page on the W3C Schools web site which covers all the functions in the module. - Ask them to pick 2 or 3 functions and investigate what they do and how they can be used. W3C schools is a useful reference for Python, not as comprehensive as the official documentation but a bit easier for beginners to digest. - Note that only those maths library functions listed in the PLS will be used in the exam.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 204 Web page describing features of the Python math module: https://www.w3schools.com/python/module_math.asp	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
---	--	--	---	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 7					
Lesson 1					
Topic 8.2.1 Methods to handle numeric data Lesson title: Using the date/time library	Students will be able to: 8.2.1 select and use appropriate methods for handling numeric data Maths skills: handling numeric data	Activities - Teacher presentation on the date/time library - Practical classroom activity creating Python programs demonstrating the use of date/time library functions (may need some time to complete this in the next lesson) - Classroom discussion, why can dates be difficult to deal with? (e.g. many different formats, dd/mm/yy, mm/dd/yy, month names rather than number, different separators etc.) Homework: - Ask students to create a guide or wall poster on using the date/time library.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 205–6 Video explaining the use of the date and time library: https://www.youtube.com/watch?v=r1lv4d6CO2Q	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 7

Lesson 2

Topic 8.2.1 Methods to handle numeric data	Students will be able to: 8.2.1 select and use appropriate methods for handling numeric data	Activities - Teacher demonstration on converting between datatypes - Practical classroom activity creating Python programs demonstrating the use of date/time library functions (continued) and conversion between datatypes, including the validation of user input of numeric values. - Class discussion about the importance of validating user input. What problems, other than program crashes can be caused by not validating user input?	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 206	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
Lesson title: Datatype conversions	Maths skills: handling numeric data	Homework: Ask students to review some of the programs that that have written so far which accept user input and consider if validating user input would make them more robust			

Week 7

Lesson 3

Topic 8.2.2 Methods to handle string data Lesson title: Handling strings	Students will be able to: 8.2.2 select and use appropriate methods for handling string data	Activities - Teacher demonstration on basic string handling and formatting - Practical classroom activity creating Python programs demonstrating basic string handling and the use of the placeholder to format string output - Class discussion on user interfaces. Are there any applications that still use a command line interface? What additional functions need to be added to a program to create and use a GUI? Homework: - Ask students to review the programs they have written previously to see if the use of formatted strings could improve the output they produce	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 207–210 Video describing basic string handling: https://www.youtube.com/watch?v=Ctqi5Y4X-jA	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
--	---	---	---	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 8					
Lesson 1					
Topic 8.2.2 Methods to handle string data Lesson title: Manipulating strings	Students will be able to: 8.2.2 select and use appropriate methods for handling string data	Activities - Teacher demonstration on string manipulation and functions that can be used to manipulate and format strings. - Practical classroom activity creating Python programs demonstrating the use of string manipulation functions (student book 1, Activity 3 p212 gives an example) - Class discussion, ask learners if they can think of any applications where string manipulation might be needed (word games, natural language processing, data science, text predictions etc.) Homework: - Ask students to search for 'format specification mini language' to find the official Python documentation on string manipulation. The section within the documentation on formatting string describes many ways of formatting output. Ask students to try out a few of the ones not covered in the lesson. Note that only those	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 210–212 Python documentation on the format specification mini language: https://docs.python.org/3/library/string.html#formatspec	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

		string formats listed in the PLS will be used in the exam.			
Week 8 Lesson 2 & 3					
Topic 8.2.2 Methods to handle string data Lesson title: String Manipulation projects (double lesson)	Students will be able to: 8.2.2 select and use appropriate methods for handling string data	Activities - Recap on key topics recently covered including 1– and 2–dimensional data structures, processing of numeric and string data. - Practical classroom activity creating ‘mini applications’ using string manipulation and other features covered so far - Practical classroom activity including programs provided with syntax and logic errors that students need to find and correct (important skill for the exam) Homework: - Ask students to search for ‘Python errors and exception’ and read the official python documentation on syntax error errors and exceptions	Pearson Edexcel International A–level Computer Science Student Book 1: pp. 207–212 Python documentation on errors and exceptions: https://docs.python.org/3/tutorial/errors.html	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 9					
Lesson 1					
Topic 8.2.3 Methods to handle text files Lesson title: Text files – open, write	Students will be able to: 8.2.3 select and use appropriate methods for handling text files	Activities - Teacher presentation of text files and the CSV format and where the CSV format is commonly used. - Teacher demonstration of methods used to open and read data from text files - Practical classroom activity creating Python programs demonstrating opening and writing to text files Homework: - Set students a research task looking into different file formats and why they are used. For example, CSV files, text files, PDF files, binary files. How can a Python program access different file types?	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 212–214 Video explaining writing to text files: https://www.youtube.com/watch?v=1IYrmTTKOol	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 9 Lesson 2					
Topic 8.2.3 Methods to handle text files Lesson title: Text files – read	Students will be able to: 8.2.3 select and use appropriate methods for handling text files	Activities - Teacher demonstration of methods used to write data to text files including creation of CSV format files which can be opened in Excel etc. - Practical classroom activity creating Python programs demonstrating opening and writing to text files. (student book 1, Activity 4 p217 gives an example) - Classroom activity with students shown code snippets, some of which contain syntax errors other contain some poor practice (e.g. variable names not meaningful, lack of input validation, complex if statements where a match statement could be used etc). Ask students to identify corrections or improvements to the code. Homework: - Ask students to do some research to attempt to find some interesting data online in CSV format. This could be weather data, sports data etc. (www.kaggle.com has a huge collection of data sets). This data can then be used in the project in week 9	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 214–216 Video explaining reading and writing to text files: https://www.youtube.com/watch?v=OMKcCHrTo0c	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 9 Lesson 3					
Topic 8.2.4 Methods to handle data structures Lesson title: Data structures – dictionaries	Students will be able to: 8.2.4 select and use appropriate methods for handling data structures	Activities - Teacher demonstration of how to create and use dictionaries - Practical classroom activity creating Python programs demonstrating creating and accessing dictionaries - A fun class activity with a speed programming challenge. Give learners a simple but clear definition of a program to be written. The first student to complete a fully working version of the program raises their hand and everyone stops work. The teacher checks the program, if it is fully working, they have won, if not they are disqualified and the others can continue and so on until a student completes a fully working version. Make sure the definition of ‘fully working’ is clear at the start. Homework: - As mentioned in the student book 1 (p220) Python dictionaries have some similarities to the JSON format. Ask students to research JSON, what is it used for, how it relates to Python dictionaries and how you convert between JSON and Python	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 218–219 Video explaining dictionaries: https://www.youtube.com/watch?v=4t10v2QmTHU	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 10					
Lesson 1					
Topic 8.2.4 Methods to handle data structures Lesson title: Data structures – dictionary subprograms	Students will be able to: 8.2.4 select and use appropriate methods for handling data structures	Activities - Teacher demonstration of subprograms that can be used with dictionaries - Practical classroom activity creating Python programs demonstrating the use of subprograms with dictionaries - Class discussion on when to use lists, tuples and dictionaries and the differences between them (a short quiz could be used here to help remind students of the different features and syntax) Homework: - Ask students to create a revision guide or wall poster on the different data structures, lists, tuples and dictionaries	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 219–220	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 10

Lesson 2 & 3

Topic 8.2.1 Methods to handle numeric data Topic 8.2.3 Methods to handle text files Topic 8.2.4 Methods to handle data structures Lesson title: Project covering handling numbers, text files and data structures (double lesson)	Students will be able to: 8.2.1 select and use appropriate methods for handling numeric data 8.2.3 select and use appropriate methods for handling text files 8.2.4 select and use appropriate methods for handling data structures Maths skills: handling numeric data	Activities - Recap on previous topics covered and introduction to the projects - Practical classroom activity creating some longer 'mini applications' (rather than simple code snippets) implementing the topics covered so far (including math library, dictionaries and text files etc.). One project could involve developing a Python program to process CSV data students have found in a previous homework exercise. - Practical classroom activity including programs provided with syntax and logic errors that students need to find and correct (important skill for the exam) Homework: Ask students to review their progress in learning Python by answering questions such as: - What aspects of python have you found most difficult? - How easy do you find it to write Python code which has no syntax errors? - Can you remember how to create basic constructs such as selection and iteration without referring to notes? - What would help you improve your ability to write Python programs? This can be developed into an action plan to develop skills.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 202–219	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
---	---	---	---	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 11					
Lesson 1					
Topic 8.2.5 Methods to represent and handle a stack implemented as a fixed length list Lesson title: Stacks	Students will be able to: 8.2.5 select and use appropriate methods to handle a stack implemented as a fixed length list	Activities - Teacher presentation on Abstract Data types and the nature of stacks and how they can be implemented in Python - Practical classroom activity creating Python programs demonstrating creating stacks using a list and the Queue library - Classroom discussion about what ADTs are and why they are important in computer science. What are stacks and queues are typically used for (refer to Unit 1 lessons)? Homework: - Ask students to search for 'python stack data structure' and read the Code Academy article on Python stacks, when and why to use them	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 222–3 The theory of stacks is covered in Unit 1, section 4.3.1 Video explaining how stacks work and can be implemented: https://www.youtube.com/watch?v=NKmasqr_Xkw Web page explaining when and why to use a stack:	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

			https:// www.codecademy.co m/article/python- stack-data-structure- when-and-why-to- use-it		
Week 11 Lesson 2					
Topic 8.2.6 Methods to represent and handle a queue implemented as a fixed length list Lesson title: Queues	Students will be able to: 8.2.6 select and use appropriate methods to handle a queue implemented as a fixed length list	Activities - Teacher presentation on the nature of queues and how they can be implemented in Python - Practical classroom activity creating Python programs demonstrating creating a queue using lists and the Queue module (student book 1, Activity 5 p225 gives an example) - Working in pairs ask students to review each other's programs that they have recently written and discuss potential improvements to them. Homework: - Ask students to watch this video: This describes a real-life application of queues and explains how queues are implemented in Python and other languages so it an interesting extension activity	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 225 The theory of queues is covered in Unit 1, section 4.3.2 https:// www.youtube.com/ watch? v=rUUrMGKYwHw	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity

Week 11

Lesson 3

Topic 8.2.7 Methods handle a set Lesson title: Sets	Students will be able to: 8.2.7 select and use appropriate methods to handle a set Maths skills: Using sets	Activities - Teacher presentation on the nature of a sets and how they can be implemented in Python - Practical classroom activity creating Python programs demonstrating creating sets and using set functions - Set students a programming challenge to answer a simple exam style coding question (for example find syntax errors or resolve a bug in a program) in a limited amount of time Homework: - Ask students to read the section in the official Python documentation on data structures.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 225–227 Video explaining sets in Python: https://www.youtube.com/watch?v=t9j8ICUGZXo Python documentation on data structures : https://docs.python.org/3/tutorial/datastructures.html	Critical thinking Problem solving Analysis Initiative Productivity	Critical thinking Problem solving Analysis Initiative Productivity
---	---	---	--	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 12					
Lesson 1 & 2					
Topic 8.2.5 Methods to represent and handle a stack implemented as a fixed length list Topic 8.2.6 Methods to represent and handle a queue implemented as a fixed length list Topic 8.2.7 Methods handle a set Lesson title: Stack, queues and sets project (double lesson)	Students will be able to: 8.2.5 select and use appropriate methods to handle a stack implemented as a fixed length list 8.2.6 select and use appropriate methods to handle a queue implemented as a fixed length list 8.2.7 select and use appropriate methods to handle a set Maths skills: Using sets	Activities - Recap on stacks queue and sets, introduction to the projects - Practical classroom activity creating some longer 'mini-applications' (rather than simple code snippets) implementing the queues, stacks and sets - Class quiz on queues, stacks and sets, the way they work, how they are used and syntax. Homework: - Ask students to review material covered in unit 1 on databases in preparation for topic 8.3 on relational databases	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 222–227	Critical thinking Problem solving Analysis Initiative	Critical thinking Problem solving Analysis Initiative

Week 12 Lesson 3					
Topic 8.3.1 Methods to administer an SQLite database Lesson title: Intro to SQLite	Students will be able to: 8.3.1 select and use appropriate methods to administer an SQLite database Maths skills: Manipulating SQLite databases	Activities - Teacher presentation on SQL (recap on Unit 1 material) and overview of SQLite - Teacher demonstration of connecting to an SQLite database and creating a table using Python - Practical classroom activity creating Python programs demonstrating connecting to SQLite and creating tables (later lessons with add and retrieve records to and from these tables) - Class quiz on relational database terminology and features Homework: - Ask students to read the tutorial section of the official Python documentation on the SQLite interface,	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 228–229 The theory of relational databases is covered in Unit 1, section 4.1 Python tutorial on SQLite interface: https://docs.python.org/3/library/sqlite3.html	Critical thinking Problem solving Analysis Initiative	Critical thinking Problem solving Analysis Initiative

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 13					
Lesson 1 & 2					
Topic 8.3.2 Methods to manipulate an SQLite database Lesson title: Manipulate data in an SQLite database (double lesson)	Students will be able to: 8.3.2 select and use appropriate methods to manipulate data in an SQLite database Maths skills: Manipulating SQLite databases	Activities - Teacher demonstration of using SQL insert to add records to an SQLite table - Practical classroom activity creating Python programs demonstrating inserting records into an SQLite table (student book 1, Activity 7 p233 gives an example) - Class practical activity. To make maintaining and testing SQLite databases easier install a GUI database browser such as DB Browser (open source) and have students use it to check their Python programs have inserted data in the tables as expected. - Class discussion on the benefits and drawback of using a GUI to access SQLite data compared to a Python program. What situations would best suit each method? Homework: - Ask students to read the official SQLite documentation on the insert statement.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 229–233 SQLite documentation on the insert statement: https://sqlite.org/lang_insert.html	Critical thinking Problem solving	Critical thinking Problem solving

Week 13

Lesson 3

Topic 8.3.3 Methods to retrieve an SQLite database Topic 8.3.4 Methods to filter and refine data retrieved an SQLite database Lesson title: Retrieve data from an SQLite database (double lesson)	Students will be able to: 8.3.3 select and use appropriate methods to retrieve data from an SQLite database Maths skills: - Manipulating SQLite databases - Using appropriate operators	Activities - Teacher demonstration of retrieving data from an SQLite table using a 'select' statement - Practical classroom activity creating Python programs demonstrating retrieving records into an SQLite table (student book 1, Activity 8 p233 gives an example). Extension activities using 'Select - Where' statement, using the 'distinct' keyword and using relational and Boolean operators. - Practical classroom activity creating Python programs demonstrating the use of 'Select' statements with 'Like', 'is null', 'group by' and 'order by' as well as aggregate functions (sum, count, avg, min, max) - Class discussion on security issues and SQL. Ask students to research/discuss questions such as: a) Why is security an important issue with SQL? b) What is SQL injection? c) How can injection attacks be avoided? Homework: - Ask students to read the official SQLite documentation on the select statement.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 233–235 SQLite documentation on the select statement: https://sqlite.org/lang_select.html	Critical thinking Problem solving	Critical thinking Problem solving
--	---	---	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 14					
Lesson 1					
Double lesson continued from WK 13					
Week 14					
Lesson 2 & 3					
Topic 8.3.4 Methods to filter and refine data retrieved an SQLite database Lesson title: Using SQL join (double lesson)	Students will be able to: 8.3.4 select and use appropriate methods to filter and refine data retrieved from an SQLite database Maths skills: Manipulating SQLite databases Using appropriate operators	Activities - Teacher recap on related tables (Primary/foreign key, relationship types – one-to-many etc) and SQL join types. - Teacher demonstration of using SQL join statements. - Practical classroom activity creating SQLite databases with related tables. - Practical classroom activity creating Python programs which implement SQL Joins on database previously created. This should include Inner, Left, Right and Full joins. (student book 1, Activity 9 p242 gives an example) - Teacher demonstration on SQLite errors and how to deal with them.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 235–239	Critical thinking Problem solving	Critical thinking Problem solving

		Homework: Ask students to review some of the SQLite programs they have written and add code to catch relevant SQL errors			
--	--	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
------------------------------------	-------------------------	----------------------	---------------------	---	--

Week 15 Lesson 1

Topic 8.3.4 Methods to filter and refine data retrieved an SQLite database Lesson title: Update and delete an SQLite database	Students will be able to: 8.3.4 select and use appropriate methods to filter and refine data retrieved from an SQLite database Maths skills: Manipulating SQLite databases	Activities - Teacher recap on SQLite features covered, any issues or problems that have arisen - Teacher demonstration of using SQLite Update and Delete statements - Practical classroom activity creating Python programs which implement SQL Update and Delete statements using tables previously created - Class quiz on relational database terminology and SQLite statement syntax Homework: - Ask students to read the official SQLite documentation on the update and delete statements.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 239–240 SQLite documentation on the update and delete statements: https://sqlite.org/lang_update.html https://sqlite.org/lang_delete.html	Critical thinking Problem solving	Critical thinking Problem solving
---	--	---	--	--	--

Week 15

Lesson 2 & 3

Topic 8.3.3 Relational databases	Students will be able to:	Activities	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 229–246		
Lesson title:	8.3.1 select and use appropriate methods to administer an SQLite database	- Teacher presentation outlining the project to be completed - Practical classroom activity completing a database project implementing a complete mini database application such as a to-do list or similar - Practical classroom activity including SQLite programs provided with syntax and logic errors that students need to find and correct (important skill for the exam)			
SQLite database project (double lesson)	8.3.2 select and use appropriate methods to manipulate data in an SQLite database				
	8.3.3 select and use appropriate methods to retrieve data from an SQLite database	Homework:			
	8.3.4 select and use appropriate methods to filter and refine data retrieved from an SQLite database	Ask students to create a revision guide for SQLite and Python covering all the features listed in the PLS			
	Maths skills:				
	- Manipulating SQLite databases - Using appropriate operators				

Topic	Specification reference	Suggested Activities	Suggested	Which skills	Which skills could
-------	-------------------------	----------------------	-----------	--------------	--------------------

Sub topic Lesson title			resources	acquired in this lesson are explicitly assessed through examination?	be acquired through teaching and delivery in this lesson?
Week 16 Lesson 1					
Topic 9.1.1 Methods to produce correct outcomes Lesson title: Decomposition and logic	Students will be able to: 9.1.1 select and use appropriate methods to produce correct outcomes	Activities - Teacher presentation on decomposing problems into component parts, giving example of how simple programs (e.g. calculating standard deviation) can be broken down into several steps. - Teacher demonstration of taking a problem, decomposing it, identifying constructs required, defining logic using a flow chart, identifying variables. Constants and data structures - Class discussion. What makes designing software difficult? Homework: - Ask students to research software design techniques. What methods (other than flow charts) are typically used to design software?	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 248–251	Critical thinking Problem solving	Critical thinking Problem solving

Week 16

Lesson 2

Topic 9.1.1 Methods to produce correct outcomes Lesson title: Decomposition and design project	Students will be able to: 9.1.1 select and use appropriate methods to produce correct outcomes	Activities - Teacher presentation, recap on program design techniques, introduction to the project task - Practical class task. Students are provided with a clearly defined application problem. (See activity 1 in student book 1 on page 251 for an example). They need to decompose the problem into component parts, define the logic for the components (for example by drawing flow charts, defining the required constructs and identifying required variables, constants and data structures). - An alternative would be to provide a less well-defined problem, and which requires to students ask questions to better define what is required. This can be done as an individual activity or in small groups. - Class discussion on the design task. How difficult was it to take a problem and produce a meaningful design? Could the solution be coded from the design produced? What else might be needed to improve the design? Homework: - Ask students watch this video which explains the technique of decomposition:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 248–251 https://www.youtube.com/watch?v=yQVTijX437c	Critical thinking Problem solving	Critical thinking Problem solving
--	--	--	--	---	---

Week 16

Lesson 3

Topic 9.1.2 Methods to implement good practice in program design Lesson title: Good practice	Students will be able to: 9.1.2 select and use appropriate methods to design effective code	Activities - Teacher presentation, introduction to the session and the principles of good practice in coding - Class activity. Divide students into 5 groups and give each group a topic to research, topics are: - Separation of concerns - Minimization of side effects - Scope isolation - Fitness for purpose of inputs and output - Efficient solutions (memory usage/execution time) - Group feedback, each group feeds back on the result of their research. - Class discussion on the outcome and formulates guidelines for programming good practice Homework: - Ask students to look at this video which makes some good points about programming good practice and separation of concerns. - Although the code shown is not Python the points made apply to Python and students should be able to see the similarities and make sense of it	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 251–255 https://www.youtube.com/watch?v=ljW0clZJvAk	Critical thinking Problem solving	Critical thinking Problem solving
--	---	---	--	---	---

[illegible]

Week 17

Lesson 3

Topic 9.1.3 Methods to implement modularity in source code Lesson title: Implementing modularity (double lesson)	Students will be able to: 9.1.3 select and use appropriate methods to produce modular code	Activities - Teacher presentation on the different methods to implement modularity in code. Recap on the material previously covered on subprograms (week 3) - Class practical activity. Students are provided with code which does not demonstrate the use of modularity and are asked to refactor the code to implement modularity. - Students are provided with code samples which contain syntax and logic errors which they are asked to find and resolve. A fun alternative would be to flash up code samples some of which are correct and some of which contain errors, students shout out if they are correct or not. Discuss how to correct those with errors. Homework: - Ask students to watch this video, which summarises the benefits of modularization:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 255–256 https://www.youtube.com/watch?v=XpNo8bpYHFI	Critical thinking Problem solving	Critical thinking Problem solving
--	--	---	--	---	---

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 18					
Lesson 1					
Double lesson continued from WK 17					
Week 18					
Lesson 2					
Topic 9.2.1 Consistency Lesson title: Producing consistent code	Students will be able to: 9.2.1 select and use appropriate methods to improve the consistency of code	Activities - Teacher presentation on consistency in code, what it involves, why it is important. - Practical class task. Students are provided with code that does not comply with consistency guidelines and may also contain some syntax and/or logic errors, and they need to refactor the code to improve consistency and resolve any errors. - Class discussion on what makes a 'good' programmer. What technical and non-technical skills are needed? From an employer's point of view what is a 'good' programmer? How can students develop desirable skills? Homework:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 256–258	Critical thinking	Critical thinking

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 19					
Lesson 1					
Topic 9.2.1 Consistency Topic 9.2.1 Readability Lesson title: Consistency/ readability project	Students will be able to: 9.2.1 select and use appropriate methods to improve the consistency of code 9.2.2 select and use appropriate methods to improve the readability of code	Activities - Teacher presentation, recap on consistency and readability guidelines - Practical class task. Students should review some of the larger 'project' programs they have written previously and modify them to comply with as many of the consistency and readability guidelines as possible. - Class discussion on the importance of following 'clean code' and good practice principles. What problems might arise if these principles are not followed? Are there any disadvantages to following these principles? Homework: - Ask students to read the qualification Good Programming Guide (GPG).	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 256–262	Critical thinking Problem solving	Critical thinking Problem solving

Week 19

Lesson 2 &3

Topic 9.2.3 Methods to control the behaviour of program code Lesson title: Control code behaviour (double lesson)	Students will be able to: 9.2.3 select and use appropriate methods to control the behaviour of code	Activities - Teacher introduction to the concept of controlling the behaviour of program code - Class activity. Divide students into 6 groups and give each group a topic to research and identify situations when these methods should be applied. The topics are: - Boundary value analysis - Exception handling - Minimising nested conditionals - Minimising loops - Using guard clauses - Managed conversion of datatypes - Group feedback, each group feeds back on the result of their research. - Practical class task. Students should review some of the larger 'project' programs they have written previously and modify them to improve the way the behaviour of the program is controlled (for example by testing the program using boundary value analysis, adding exception handling and minimising nested conditionals and/or loops) Homework: - Ask students to watch this video on software testing: - It goes a little further than just boundary value analysis but is short and quite well explained.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 262–269 Video on nested conditionals and guard clauses: https://www.youtube.com/watch?v=NCnPg_S_Eu8 https://www.youtube.com/watch?v=P1Hv2sUPKeM	Critical thinking Problem solving	Critical thinking Problem solving
--	--	---	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 20					
Lesson 1					
Topic 9.3.1 Methods to ensure that code is fit for purpose, produces accurate results and meets requirements Lesson title: Test and debug code	Students will be able to: 9.3.1 - Identify errors in program code - Correct errors in program code	Activities - Class discussion on debugging and testing. What bugs do students typically come across? How do they resolve them? Why can debugging be difficult? How should testing be performed? - Teacher demonstration of debugging, using break points and viewing variables. - Practical class task. Students are provided with code that contains logic bugs and use debug to help resolve the issues, so the code works correctly. - Feedback on practical activity. Was using debug effective? Were students able to resolve the issues? Any other methods useful for resolving code bugs? How to test software to find bugs? Homework: - Provide students with further examples of code which contains logic errors and ask them to debug and correct the errors	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 269–277 Video explaining how to use the debug feature in IDLE: https://www.youtube.com/watch?v=HiQZKDI61E4	Critical thinking Problem solving	Critical thinking Problem solving

Week 20

Lesson 2

Topic 9.3.1 Methods to ensure that code is fit for purpose, produces accurate results and meets requirements Lesson title: Review and refine code	Students will be able to: 9.3.1 - Produce accurate outcomes - Refine solutions to meet requirements	Activities - Class discussion. In what ways can a program be reviewed and refined? Who should review a program? What criteria should be used when reviewing a program? - Class practical activity with student work in pairs. Each pair should review some of the longer programs they have written and make constructive suggestions as to how the code might be refined. Particular attention should be paid to the best practice guidelines that have been covered in this topic - Class discussion. Are best practice guidelines worth following? Or are they a waste of time? What are the benefits of coding following the best practice guidelines? Homework: - Ask students to think about how they might create a questionnaire to be used by someone who is going to review one of the programs they have written. What kind of criteria do they think other people might use to review their programs (e.g. ease of use, lack of bugs, correct results etc.)?	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 269–277	Critical thinking Problem solving Analysis	Critical thinking Problem solving Analysis
---	---	--	---	---	---

Week 20

Lesson 3

Topic 9.3.1 Methods to ensure that code is fit for purpose, produces accurate results and meets requirements Lesson title: Test, debug and refine project (double lesson)	Students will be able to: 9.3.1 - Identify errors in program code - Correct errors in program code - Produce accurate outcomes - Refine solutions to meet requirements	Activities - Teacher introduction, recap on topic 9 – Best practice. - Class practical session, student are provided with a partially complete mini-application which contains coding logic bugs. They need to debug the code and then add the required functionality. They should apply the best practice principles when completing the code. Once complete they should test the code to make sure it is fully functional. - Working pairs students should review each other's completed code and suggest constructive refinements that would improve the application. Homework: - Ask students to write a review of their programming skills. How easily can they write a Python program? Are there particular things they struggle with? How can they improve their skill? They can make a table listing their weaknesses, how they will improve on each weakness and a target for completion.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 269–277	Critical thinking Problem solving Analysis	Critical thinking Problem solving Analysis
--	---	--	---	---	---

Week 21

Lesson 3

Topic 10.1.2 Decomposition Lesson title: Computational thinking – decomposition	Students will be able to: 10.1.2 use comment–first coding to describe an algorithm	Activities - Teacher recap on decomposition (first covered in topic 9.1.1 (week 16) - Class practical activity. Students are given an outline of an application that is to be developed, they apply the principle of decomposition and split the application into a series of tasks. Activity 1 on page 282 of student book 1 provides an example. - Class discussion. How can the list of decomposed tasks be developed into code comments using the comment first method? Homework: - Provide students with another application description and ask them to attempt to decompose it into sub–tasks	Pearson Edexcel International A–level Computer Science Student Book 1: pp. 281–282	Critical thinking Problem solving	Critical thinking Problem solving
--	--	--	---	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 22					
Lesson 1					
Topic 10.1.3 Pattern recognition Topic 10.1.4 Generalisation Lesson title: Computational thinking – Pattern recognition and generalisation	Students will be able to: 10.1.3 use the same or similar solution/algorithm/subprogram in a different problem 10.1.4 use a solution/algorithm/subprogram for similar problem by changing input parameters	Activities - Teacher introduction to the concepts of pattern recognition and generalisation - Class activity with students given an application description and asked to identify patterns in the application. - Class practical activity with students asked to write a subprogram which demonstrates the concept of generalisation. For example, a subprogram to convert measurements (e.g. weights, distances etc) into different units. - Class discussion. Why are pattern recognition and generalisation useful in software development? What benefits being able to recognise patterns and generalise code bring? Homework - Ask students to watch this video which gives some interesting examples of pattern recognition:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 283–284 https://www.youtube.com/watch?v=SixLnIDV1yY	Critical thinking Problem solving	Critical thinking Problem solving

Week 22

Lesson 2

Topic 10.2.1 Create programmed solutions from algorithms Lesson title: Create programs from algorithms expressed in words	Students will be able to: 10.2.1 Convert algorithms into code	Activities - Teacher introduction to the topic of algorithms, linking to material covered in Unit 1. - Practical class activity. Students are provided with an algorithm expressed in words and attempt to convert the algorithm into code. - Feedback from the practical activity. How successful were they in converting the algorithm into code? What made the process difficult or easy? What methods helped them in the task? Homework: - Ask students to watch this video which provides some good information on the general topic of algorithms:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 284–287 The theory of algorithms is covered in Unit 1, section 5.1.1 https://www.youtube.com/watch?v=JJkWemM03Lg	Critical thinking Problem solving	Critical thinking Problem solving
--	---	--	--	--	--

Week 22

Lesson 3

Topic 10.2.1 Create programmed solutions from algorithms Lesson title: Create programs from algorithms expressed diagrams/flow charts	Students will be able to: 10.2.1 Convert algorithms into code	Activities - Teacher introduction to the lesson, recapping on the use of flowcharts to describe an algorithm. - Practical class activity. Students are provided with an algorithm expressed as a flowchart and attempt to convert the algorithm into code. - Feedback from the practical activity. How successful were they in converting the algorithm into code? Was task easier using the flowchart compared with the algorithm expressed in words? How closely did the code produced match the flowchart? Homework: - Give students another example of an algorithm expressed in a flowchart and ask them to develop the Python code to implement the algorithm	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 284–287	Critical thinking Problem solving	Critical thinking Problem solving
--	--	---	---	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 23					
Lesson 1					
Topic 10.3.1 Linear search algorithm Lesson title: Linear search	Students will be able to: 10.3.1 implement and use an appropriate linear search algorithm	Activities - Teacher introduction to the lesson, recapping on the linear search algorithm (covers in Unit 1). - Practical class activity. Students are given a flow chart of the linear search algorithm and write code based on the algorithm. - Class discussion or small group activity. How to represent the linear search algorithm using a flow chart. This is a reverse of the tasks in previous lesson where flowcharts are converted to code but should help students understand the relationships between algorithms and code. Homework: - Ask students to do some research into other search algorithms. The specification only covers linear and binary search but there are other algorithms such as jump search. Ask learners to find out how jump search works and as an extension activity they can practice their python skills by implementing a simple version of it.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 288–289 The theory of linear search algorithms is covered in Unit 1, section 5.2.1 page 521 Video explaining linear search: https://www.youtube.com/watch?v=UldZOLylez4	Critical thinking Problem solving	Critical thinking Problem solving

Week 23

Lesson 2

Topic 10.3.1 Linear search algorithm Lesson title: Linear search improved	Students will be able to: 10.3.1 implement and use an appropriate linear search algorithm	Activities - Class discussion on the linear search algorithm, its inefficiencies and ways it might be improved. Several improvements should be identified including taking an early exit when the target is found and sorting the list before searching. Sorting is the next topic covered. - Practical class activity. Students modify the linear search programs developed in the previous lesson to create versions which implement the improvements. - Class discussion about the linear search. How effective would it be with very large number of values to search through? Is there are more efficient method of searching? Homework: - Ask students create a version of the linear search with a fairly large number of values to search through and add timestamps at the start and end of the search so the time taken can be found. Several different search should be timed, looking for values towards the start, middle and end of the list. A simpler alternative is to just count the number of iterations needed to find the search target.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 288–289	Critical thinking Problem solving	Critical thinking Problem solving
--	---	---	---	--	--

Week 23

Lesson 3

Topic 10.3.2 Binary search algorithm Lesson title: Binary search (iterative)	Students will be able to: 10.3.2 implement and use an appropriate binary search algorithm	Activities - Teacher introduction to the lesson, recapping on the binary search algorithm (covered in Unit 1). Note that this lesson covers the iterative binary search, recursive binary search is covered later in the lesson on recursion. - Practical class activity. Students are given a flow chart of the iterative binary search algorithm and write code based on the algorithm. - Class discussion on the linear search vs the binary search. What are the advantages of each and disadvantages of each? Homework: - Ask students to create a timed version (or a version that counts the iterations) of the binary search (as described in the previous lesson). Compare the performance of the binary search with the linear search, which is faster and by how much?	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 289–290 The theory of binary search algorithms is covered in Unit 1, section 5.2.2 page 126 Video explaining linear search: https://www.youtube.com/watch?v=DE-yeOt0oxE	Critical thinking Problem solving	Critical thinking Problem solving
---	---	---	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 24					
Lesson 1					
Topic 10.3.3 bubble sort algorithm Lesson title: Bubble sort worse case	Students will be able to: 10.3.3 implement and use an appropriate bubble sort algorithm	Activities - Teacher introduction to the lesson. Linking the need for sorting with the binary search covered previously. Recap on the 'worse case' bubble sort algorithm (covered in Unit 1). - Practical class activity. Students are given a flow chart or written description of the bubble sort algorithm and write code based on the algorithm - Class discussion on how the 'worse case' bubble sort could be improved to make it more efficient. Ask students what makes the bubble sort inefficient in terms of CPU usage? Homework: - Ask students to do some research into the selection sort algorithm. This algorithm is not in the specification but it's a simple sort algorithm that students may find interesting and can practice their python skills by writing a program to implement it.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 290–291 The theory of bubble sort algorithm is covered in Unit 1, section 5.2.3 page 128	Critical thinking Problem solving	Critical thinking Problem solving

Week 24

Lesson 2

Topic 10.3.3 bubble sort algorithm Lesson title: Bubble sort improvements	Students will be able to: 10.3.3 implement and use an appropriate bubble sort algorithm	Activities - Teacher introduction to the lesson, recapping the previous lesson and the inefficiencies of the bubble sort. Describe the two possible improvements, exiting the loop when no swaps occur and reducing the loop upper bound on each pass. - Practical class activity. Students modify their bubble sort programs to try the improved versions. Student should keep all the different versions of their bubble sort programs as they are needed in a later lesson. - Class discussion on sorting and the bubble sort. Is the bubble sort algorithm efficient? Are there other ways that sorting can be done? Homework: - Ask students to read the Geek for geeks article on sorting and the linked pages on introduction to sorting and applications of sorting	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 291–293 This video explains the bubble sort algorithm https:// www.youtube.com/ watch?v=Vca808JTbI8 Web page discussing different sorting algorithms: https:// www.geeksforgeeks.o rg/dsa/sorting- algorithms/	Critical thinking Problem solving	Critical thinking Problem solving
--	---	---	--	--	--

Week 24

Lesson 3

Topic 10.3.4 In-place insertion sort algorithm Lesson title: Insertion Sort	Students will be able to: 10.3.4 implement and use an in-place insertion sort algorithm	Activities - Teacher introduction to the lesson, recapping on the insertion sort algorithm (covered in unit 1) - Practical class activity. Students are given a flow chart or written description of the insertion sort algorithm and write code based on the algorithm - Class quiz on searching and sorting algorithms. Homework: - Ask students to create a revision guide or wall chart covering different search and sort algorithms.	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 293–294 The theory of insertion sort algorithm is covered in Unit 1, section 5.2.4 page 130 Video explaining insertion sort https://www.youtube.com/watch?v=jRsm6FmHp7M	Critical thinking Problem solving	Critical thinking Problem solving
--	---	--	--	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 25 Lesson 1					
Topic 10.3 Search and sort algorithms Lesson title: Sort comparisons	Students will be able to: 10.3.3 implement and use an appropriate bubble sort algorithm 10.3.4 implement and use an in-place insertion sort algorithm	Activities - Teacher introduction to the lesson, recapping on different versions of the bubble sort and the insertion sort algorithm and how their performance will be compared - Practical class activity. Students will research into the speed of the different sort algorithms by creating versions which use a large list of numbers (created using the random library) to sort and time the execution of the sorting. - Class discussion on the outcome. How much faster was the fastest algorithm compared to the slowest? Why is speed important? What kind of applications requires searching and sorting? Homework: - Ask students to research other sorting algorithms (some of these will be covered in Unit 4). Which sorting algorithm provides the fastest sort? Why are there so many different sorting algorithms? Which is the 'best' (or what type of application are they best suited	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 289–294	Critical thinking Problem solving	Critical thinking Problem solving

		to)?			
Week 25 Lesson 2					
Topic 10.4.1 Recursive algorithms Lesson title: Recursion	Students will be able to: 10.4.1 implement and use recursive algorithms	Activities - Teacher introduction to the lesson, recapping on the concept of recursion (covered in Unit 1). - Teacher demonstration of how recursion can be used in Python - Practical class activity. Students write programs demonstrating the use of recursion. - Class discussion on recursion. What are the benefits and drawbacks of using recursion? What sort of situations is recursion best suited to? Homework: - Ask students to read the Real Python article about recursion in Python:	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 294–295 The theory of recursion is covered in Unit 1, section 5.1.3 page 121 Real Python article on recursion: https://realpython.com/python-recursion/	Critical thinking Problem solving	Critical thinking Problem solving

Week 25

Lesson 3

Topic 10.3.2 Binary search algorithm Lesson title: Binary search (recursive) (double lesson)	Students will be able to: 10.3.2 implement and use an appropriate binary search algorithm 10.4.1 implement and use recursive algorithms	Activities - Teacher recap on binary search (covered in week 23) - Teacher explanation of how binary search can be implemented recursively. - Classroom practical. Students write programs implementing the recursive version of the binary search - Classroom practical with students writing a complete mini application which implements both a sort and search algorithms. For example, they could allow a user to enter some data, sort the data and then allow to use to search for data items - Class quiz on algorithms and recursion Homework: - Ask student to review and refine the mini-application and identify potential improvements to its efficiency (e.g. can the sort code be improved?), features (e.g. could input validation be added or improved? and user interface	Pearson Edexcel International A-level Computer Science Student Book 1: pp. 288 – 295		
--	--	---	---	--	--

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 26					
Lesson 1					
Double lesson continued from WK 25					
Week 26					
Lesson 2					
Topic – All Lesson title: Final project (triple lesson)		Activities - Teacher introduction to the lesson, explaining the project to be completed. - Classroom practical. Students are given a project specification describing the application and the functionality required. Students will need to: - Use computational thinking techniques to decompose the overall application into sub-tasks - Create algorithms for each sub-tasks, described using words or diagrams (flowcharts) - Write code to implement the algorithms following best practice and clean code principles. - Test the completed application to make sure it functions as it should - Working in pairs students should review each other's completed applications and identify		Critical thinking Problem solving	Critical thinking Problem solving

		possible refinements			
		Homework: Ask students to review their knowledge of Python. What aspects of python do they feel they understand well? What topics covered in this unit do they feel less confident with? Using this information they should make a revision scheme, focussing on those areas that need development			
Week 26					
Lesson 3					
Final Project					

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 27					
Lesson 1					
Final Project					
Week 27					
Lesson 2					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 27					
Lesson 3					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 28					
Lesson 1					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 28					
Lesson 2					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 28					
Lesson 3					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 29					
Lesson 1					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 29					
Lesson 2					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 29					
Lesson 3					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			

Topic Sub topic Lesson title	Specification reference	Suggested Activities	Suggested resources	Which skills acquired in this lesson are explicitly assessed through examination?	Which skills could be acquired through teaching and delivery in this lesson?
Week 30					
Lesson 1					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 30					
Lesson 2					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			
Week 30					
Lesson 3					
		These lessons are allocated for revision and exam time but can also be used flexibly if you want to spend more time on certain topics or embed more frequent assessments throughout the year.			