

International Advanced Level **Computer Science**

Scheme of Work

Pearson Edexcel International Advanced Level in Computer Science (YCP01)

First teaching September 2026 | First examination June 2028

First certification from August 2028

Issue 1

Content

How to use this Scheme of Work	3
Unit 3: Advanced Principles of Computer Science	5
Unit 4: Advanced 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 of IA2. **This SoW is for IA2 (YCP01).**

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.

IA2 Unit 1 Principles of Computer Science

Topic 11 – Computer systems and

Topic 2 – Data representation

Topic 12 – Networks and

encryption

Topic 13 – Programming data

Topic 5 – Problem-solving

Topic 14 – Emerging technologies

Topic 15 – Problem-solving

Topic 16 – Emerging technologies

IAS Unit 2 Practical Programming IA2 Unit 4 Advanced Practical & Problem Solving

Topic 7 – Programming

Topic 17 – Programming

Topic 8 – Organising and handling

Topic 18 – Representing and

handling data

Topic 9 – Best practice

Topic 10 – Best practice

Topic 10 – Computational thinking

Topic 20 – Additional

programming paradigms

Topic 21 – 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 3: Advanced 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					
11.1 Computer Architecture	11.1.1 Harvard architecture	Activities: - Discuss Harvard architecture (and contrast with von Neumann from Unit 1 Topic 1.1.2) - Teacher demonstrates a diagrammatic representation (see p3) - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 2–5 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/sys_arch_architecture?examBoard=aqa&stage=a_level&topic=architecture	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis
Topic 11.1.1: Harvard architecture Lesson title: Harvard architecture	a) Memory structure b) Difference between Harvard and von Neumann architecture Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks	Homework: Students complete questions from IsaacCS			

	Students will be able to: interpret and complete a diagrammatic representation of the internal components of a computer system based on the Harvard architecture.				
Week 1 Lesson 2					
Topic 11.1.2: CISC and RISC Lesson title: CISC and RISC	11.1.2 Reduced instruction set computer (RISC): a) Architecture b) Difference between RISC and CISC Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks	Activities: - Discuss RISC and CISC architectures - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks - Complete activity 1 on p7 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 5–7 AdaCS Notes and questions: https://adacomputerscience.org/concepts/arch_processor_types?topic=architecture	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis

Week 1

Lesson 3

Topic 11.1.3: Five-stage instruction pipelining Lesson title: Instruction pipelining	11.1.3 Five stage instruction pipelining: a) Stages: • Instruction fetch (IF) • Instruction decode (ID) • Execute (EX) • Memory Access (MEM) • Register write back (WB) b) Branch hazards Students will know and understand: • Definition • Function • Characteristics • Operation • Benefits • Drawbacks Students will be able to: • interpret and complete a diagrammatic representation of the pipelining process.	Activities: • Discuss pipelining as a mechanism to increase throughput • Teacher demonstrates diagram for representing pipelining (see p8) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks Homework: • Students complete activity 2 on p10	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 8–10 Computerphile video on CPU pipeline: https://www.youtube.com/watch?v=BVNx3wtJ9vs	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Decision making Analysis 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 2					
Lesson 1					
Topic 11.1.4: Parallel processing Lesson title: Parallel processing	11.1.4 Parallel processing: a) Multicore processors - Multiple instruction multiple data (MIMD) b) Graphic processing units (GPUs) - Single instruction multiple data (SIMD) Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Identify and complete a diagrammatic representation of parallel processing.	Activities: - Discuss approaches to parallel processing - Teacher demonstrates diagrams of parallel processing p11 & 13 - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why parallel processing is needed Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 10–14 AdaCS notes and questions: https://adacomputerscience.org/concepts/arch_performance#multiple-cores	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis
Week 2					

Lesson 2

Topic 11.1.5: Embedded systems a) Hardware components Lesson title: Hardware components of embedded systems	11.1.5 Embedded systems: a) Hardware components: • Microprocessor • Microcontroller unit (MCU) • Memory • I/O interfaces and ports • Sensors [accelerometers, vibration, sound, temperature] • Actuators Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss the different hardware components in an embedded system • Option to look at some examples of embedded systems • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why embedded systems are needed Homework: • Students complete activity 3 on p16	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 14–19 MrBrownCS video: https://www.youtube.com/watch?v=WxcosGVshOY	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Self-direction Teamwork
--	--	---	---	--	---

Week 2 Lesson 3					
Topic 11.1.5: Embedded systems a) Hardware components Lesson title: More hardware components of embedded systems	11.1.6 Embedded systems: a) Hardware components: - Analogue-digital Converter (ADC/DAC): [sampling, quantisation, encoding] - Power supply Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - Interpret and complete a design for an embedded system. - Interpret and complete a diagrammatic representation of the ADC/DAC process	Activities: - Discuss additional hardware devices: ADC and power supply - Teacher demonstrates a diagram representing the ADC/DAC process (see p19) - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why embedded systems are needed Homework: - Students complete activity 4 on p21	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 18–21 AdaCS notes and questions: https://adacomputerscience.org/concepts/arch_embed?topic=architecture	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Creativity Innovation Reasoning

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 11.1.5: Embedded systems b) Software components Lesson title: Software components of embedded systems	11.1.5 Embedded systems b) Software components: • Firmware • Operating system (OS) • Application software • Programming languages used to develop software for embedded systems Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss the software components of an embedded system • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why embedded systems are needed Homework: • Students research the popularity of different programming languages for embedded systems	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 21–22	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis

Week 3

Lesson 2

Topic 11.1.5: Embedded systems Lesson title: Recap of embedded systems	11.1.5 Embedded systems Students will be able to: Interpret and complete a design for an embedded system.	Activities: - Discuss how all the components of an embedded system link together - Students revisit activity 4 on p21 to consider the software aspects Homework: - Students draw mind map to consolidate their learning	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 21–22	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Productivity Creativity Innovation Reasoning
--	---	--	---	---	---

Week 3

Lesson 3

11.2 The Operating System Topic 11.2.1: I/O device management a) Role of device drivers b) BIOS Lesson title: I/O device management in the OS	11.2.1 Input/output (I/O) device management: a) Role of device drivers b) Basic I/O system (BIOS) 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 I/O in a computer system	Activities: • Discuss device drivers and the BIOS • Option to investigate the device drivers and BIOS on a school device • Teacher demonstrates a diagram representing I/O in a computer system (see p22) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why the operating system is needed Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 22–24 AdaCS notes and questions: https://isaaccomputerscience.org/topics/operating_systems	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Self-direction
---	--	--	---	---	--

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 11.2.1: I/O device management c) Role of buffering d) Role of caching e) Interrupt handling in device management Lesson title: More I/O device management in the OS	11.2.1 Input/output (I/O) device management: c) Role of buffering d) Role of caching e) Interrupt handling in device management Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss buffering, caching and interrupt handling • Teacher demonstrates buffer and cache sizes changing (or refer to Networknutsdotnet video) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why the operating is needed Homework: • Students complete activity 5 on p25	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 24–25 Networknutsdotnet video: https://www.youtube.com/watch?v=Xy_rLc-E4Xg	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Reasoning

Week 4 Lesson 2					
Topic 11.2.2: Virtualisation a) Containerisation Lesson title : Containerisation	11.2.2 Virtualisation: a) Containerisation 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 containerised system.	Activities: - Discuss containerisation - Teacher demonstrates a diagram of a containerised system (see p26) - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why containerisation is needed Homework: - Students investigate other videos and add to their notes with examples of modern technologies	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 25–26 IBM Technology video on containerisation: https://www.youtube.com/watch?v=0qotVMX-J5s IBM Technology video on containerisation vs virtualisation: https://www.youtube.com/watch?v=cjXI-yxqGTI	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Self-direction

Week 4 Lesson 3					
Topic 11.2.2: Virtualisation b) Virtual machines Lesson title: Virtual machines	11.2.2 Virtualisation: b) Virtual machines 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 virtual machine.	Activities: • Discuss virtual machines • Teacher demonstrates a diagram of a virtual machine (see p27) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why virtual machines are needed Homework: • Students complete activity 6 on p28	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 27–28 AdaCS notes: https://adacomputerscience.org/concepts/soft_virtual_machines	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Decision making Reasoning

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					
11.3 Data representation 11.3.1 Floating-point representation Lesson title: Floating-point representation	11.3.1 Floating-point representation (IEEE-754) Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - convert fractional numbers into binary and vice versa using 16-bit half precision IEEE-754 standard (1-bit sign, 5-bit exponent (biased), and 10-bit significant). Maths skills: Using floating-point representation	Activities: - Discuss floating-point representation - Option to recap fixed-point representation from Unit 1 Topic 2.1.4 - Option to link to floating-point programming in Unit 4 Topic 18.1.1 - Teacher demonstrates worked examples 1–3 on p30–32 - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why floating-point representation is needed - Students make up examples of positive and negative integers to convert to binary floating-point representation Homework: - Students make up examples of 16-bit binary pattern to convert to denary	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 28–34 Computer Science Lessons video : https://www.youtube.com/watch?v=RuKkePyo9zk (focus on “half” precision only)	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Analysis Problem solving Communication

Week 5 Lesson 2					
11.3.1 Floating-point representation Lesson title: Further practice on floating-point representation	Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • convert fractional numbers into binary and vice versa using 16-bit half precision IEEE-754 standard (1-bit sign, 5-bit exponent (biased), and 10-bit significand). Maths skills: Using floating-point representation	Activities: • Students make up examples of positive and negative real numbers to convert to binary floating-point representation • Complete activity 7 on p 34 • Use AI to make up additional questions • Option to consider the range of values that can be represented, and the accuracy	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 28–34	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Analysis Problem solving Communication

Week 5 Lesson 3					
Topic 11: Computer Systems and Data Representation Lesson title: Topic summary	Topic 11: Computer systems and 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 p34	Pearson Edexcel International A-level Computer Science Student Book 2: p34	Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis 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 6					
Lesson 1					
12.1 Networking Topic 12.1.1: Open Systems Interconnection (OSI) Lesson title: The OSI model	12.1.1 Open systems interconnection (OSI) reference model: a) Layers b) encapsulation/decapsulation Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation Students will be able to: - Interpret and complete a diagrammatic representation of encapsulation/decapsulation of packets flowing through the OSI protocol stack	Activities: - Discuss the OSI model - Teacher demonstrates diagram of the layers (see p37) - Students identify key terms and make notes on definition, function, characteristics, operation, as well as why OSI is needed Homework: - Students use Formulator Tarsia to make a matching activity with layer names and functions and then swap with each other.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 36– 40 ByteByteGo video: https://www.youtube.com/watch?v=0y6FtKsg6J4&t=16s	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Interpretation Productivity

Week 6 Lesson 2					
Topic 12.1.2: Protocol stacks Lesson title: Protocol stacks	12.1.2 Protocol stacks: a) TCP/IP b) SSL/TLS Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation Students will be able to: • Interpret and complete a diagrammatic representation of encapsulation/decapsulation of packets flowing through the TCP/IP protocol stack. • Interpret and complete a diagrammatic representation of the SSL/TLS handshake.	Activities: • Discuss both TCP/IP and SSL/TLS protocol stacks • Teacher demonstrates diagrams for these protocol stacks (See p41–42) • Students identify key terms and make notes on definition, function, characteristics, operation, as well as why protocol stacks are needed • Complete activity 1 on p43 Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 40–43 AdaCS notes and questions: https://adacomputerscience.org/concepts/network_protocols?topic=networking	Analysis Communication Interpretation	Intellectual interest and curiosity Critical thinking Analysis Interpretation

Week 6 Lesson 3					
Topic 12.1.3: Packet switching Lesson title: Packet switching	12.1.3 Packet switching: a) Packet components b) Internet router c) Protocols Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss packet switching • Option to use tools like Traceroute to examine the paths taken by real packets • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why packet switching is needed • Complete activity 2 on p45 Homework: • Students complete questions from AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 43–46 AdaCS notes and questions: https://adacomputerscience.org/concepts/internet_structure	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Teamwork Critical thinking Analysis Self-direction

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 12.1.4: Domain name system (DNS) Lesson title: DNS	12.1.4 Domain name system (DNS) 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 how the DNS works.	Activities: • Discuss DNS • Teacher demonstrates a diagram of the process by which DNS converts a domain name into an IP address (see p47) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why DNS is needed • Look for quizzes about DNS on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc. Homework: • Students make up quiz questions to demonstrate their understanding	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 46–47 Computerphile video: https://www.youtube.com/watch?v=uOfonONTluk	Analysis Communication Interpretation	Intellectual interest and curiosity Critical thinking Analysis Interpretation

Week 7

Lesson 2

12.2 The internet of things (IoT) Topic 12.2.1: The IoT stack a) Device hardware b) Device software Lesson title: Hardware and software of the IoT	12.2.1 The IoT stack: a) Device hardware b) Device software: - Operating system - Applications <i>Students will know and understand:</i> - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	<i>Activities:</i> - Discuss the hardware and software used in the Internet of Things - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why the IoT stack is needed - Complete Activity 3 on p49 <i>Homework:</i> - Students research IoT devices that are readily available	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 48–50 Geeks For Geeks notes: https://www.geeksforgeeks.org/computer-networks/introduction-to-internet-of-things-iot-set-1/	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Ethics Problem solving Self-direction
--	--	--	---	--	--

Week 7 Lesson 3

Topic 12.2.1: The IoT stack c)Connectivity d)Cloud platform Lesson title: IoT connectivity and cloud platform	12.2.1 The IoT stack: c) Connectivity: • Networks • Protocols d) Cloud platform: • Data collection • Analytics • APIs 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 IoT system	Activities: • Discuss connectivity and cloud platforms in the IoT • Teacher demonstrates a diagram of an IoT system (see p49) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why connectivity and cloud platform is needed • Complete Activity 4 on p52 Homework: • Students research current issues and challenges in IoT	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 50–52	Analysis Communication Problem solving Interpretation	Intellectual interest and curiosity Critical thinking Analysis Ethics 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 12.2.1: The IoT stack e) Cloud applications Lesson title: IoT cloud applications	12.2.1 The IoT stack: e) Cloud applications 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 IoT system	Activities: - Discuss cloud applications in the context of IoT - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why cloud applications are needed Homework: - Students research another case study to compare with the one on p51	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 51–52	Analysis Communication Interpretation	Intellectual interest and curiosity Critical thinking Analysis Interpretation

Week 8 Lesson 2					
12.3 Cybersecurity Topic 12.3.1: CIA triad model Lesson title: The CIA triad model	12.3.1 CIA triad model Students will know and understand: • Definition • Characteristics • Why needed	Activities: • Discuss the confidentiality, integrity, availability (CIA) triad model • Students identify key terms and make notes on definition, characteristics, as well as why the CIA triad mode is needed Homework: • Look for quizzes about the CIA triad model on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socratic/ etc.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 52–53 Geeks For Geeks notes: https://www.geeksforgeeks.org/computer-networks/the-cia-triad-in-cryptography/	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis

Week 8 Lesson 3					
Topic 12.3.2: RSA encryption algorithm a) Prime numbers b) Key exchange Lesson title: RSA encryption	12.3.2 RSA encryption algorithm: a) Prime numbers b) Key exchange 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 RSA encryption	Activities: • Discuss how prime numbers and key exchange are used in RSA encryption • Teacher demonstrates a diagram representing RSA encryption (see p55) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why RSA encryption is needed • Follow the worked example on p54 Homework: • Students work through another example using a different choice of prime numbers	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 53–55 Computerphile video: https://www.youtube.com/watch?v=-ShwJqAalOk	Analysis Communication Problem solving Interpretation	Intellectual interest and curiosity 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 9					
Lesson 1					
Topic 12.3.2 RSA encryption algorithm c) Vulnerabilities Lesson title: Vulnerabilities of RSA encryption	12.3.2 RSA encryption algorithm: c) Vulnerabilities <i>Students will know and understand:</i> - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss vulnerabilities in the RSA encryption algorithm - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why RSA encryption is needed Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 55–56 Computerphile video: https://www.youtube.com/watch?v=-ShwJqAalOk IsaacCS notes and questions: https://isaaccomputerscience.org/concepts/net_sec_encryption?topic=encryption	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis

Week 9

Lesson 2

Topic 12.3.3: Error detection a) Parity bit b) 2-D parity block Lesson title: Parity checking	12.3.3 Error detection: a) Parity bit b) Two-dimensional parity block Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - calculate the parity bit for a given set of bits. - calculate the parity bits for a given two-dimensional parity block. Maths skills: Calculating parity bit/block	Activities: - Discuss parity bits and blocks - Teacher demonstrates calculating parity bits and blocks (see worked examples on pp.56–57) - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why error detection is needed Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 56–59 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/net_comm_error_checking	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Analysis Problem solving
--	---	---	---	--	---

Week 9 Lesson 3					
Topic 12.3.3: Error detection c) Checksum Lesson title: Checksums	12.3.3 Error detection: c) Checksum <i>Students will know and understand:</i> - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks <i>Students will be able to:</i> - calculate the checksum for a given block of data. <i>Maths skills:</i> Calculating checksums	Activities: - Discuss checksums - Teacher demonstrates calculating a checksum (see worked example on p59) - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why error detection is needed - Complete Activity 5 on p60 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 59–60 AdaCS notes and questions: https://adacomputerscience.org/concepts/comms_error_checking	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Analysis 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 10					
Lesson 1					
12.4 Computing paradigms Topic 12.4.1: Cloud computing Lesson title: Cloud computing	12.4.1 Cloud computing: a) Infrastructure b) Services Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss cloud computing - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why cloud computing is needed Homework: - Look for quizzes about cloud computing on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 60–62 ECourse Review video: https://www.youtube.com/watch?v=36zducUXl6w Geeks For Geeks notes: https://www.geeksforgeeks.org/software-engineering/difference-between-cloud-computing-and-traditional-computing/	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis

Week 10

Lesson 2

Topic 12.4.2: Edge computing Lesson title: Edge computing	12.4.2 Edge computing: a) Infrastructure b) Services Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss edge computing • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why edge computing is needed • Complete activity 6 on p64 Homework: • Students choose one scenario from activity 6 to write up in detail	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 62–65	Analysis Communication Critical thinking Problem solving	Intellectual interest and curiosity Critical thinking Analysis Communication Decision making Reasoning
---	--	---	---	---	---

Week 10

Lesson 3

Topic 12: Networks and Cybersecurity Lesson title: Topic summary	Topic 12: Networks and cybersecurity 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 p65		Analysis Communication Critical thinking Problem solving	Critical thinking Productivity Initiative Self-direction Analysis 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 11					
Lesson 1					
13.1 Types of programming languages Topic 13.1.1: Object-oriented programming a) Terminology Lesson title: OOP terminology	13.1.1 Object-oriented programming (OOP) paradigm: a) Terminology Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss object-oriented programming • Option to link to object-oriented programming in Unit 4 Topic 20.1.1 • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why OOP is needed • Complete activity 1 on p68 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 2: pp. 67–68 IsaacCS notes and questions: https://isaacomputerscience.org/topics/object_oriented_programming	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity

Week 11 Lesson 2					
Topic 13.1.1: Object-oriented programming b) Concepts Lesson title: OOP concepts	13.1.1 Object-oriented programming (OOP) paradigm: b) Concepts: - Abstraction - Encapsulation - Composition <i>Students will know and understand:</i> - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss abstraction, encapsulation and composition - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why OOP is needed Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 68–74 GeeksForGeeks notes: https://www.geeksforgeeks.org/python/python-oops-concepts/	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity

Week 11

Lesson 3

Topic 13.1.1: Object-oriented programming b) Concepts Lesson title: More OOP concepts	13.1.1 Object-oriented programming (OOP) paradigm: b) Concepts: - Inheritance - Polymorphism Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss inheritance and polymorphism - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why OOP is needed Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 72–73 AdaCS notes and questions: https://adacomputerscience.org/topics/object_oriented_programming Computer Science Lessons video: https://www.youtube.com/watch?v=m_MQYyJplig	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation 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					
Topic 13.1.1: Object-oriented programming c) Unified modelling language (UML) class diagrams Lesson title: UML class diagrams	13.1.1 Object-oriented programming (OOP) paradigm: c) Unified modelling language (UML) class diagrams <i>Students will know and understand:</i> • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks <i>Students will be able to:</i> • Interpret and complete UML diagrams	Activities: • Discuss UML class diagrams • Option to explore a tool like draw.io or Visio for drawing UML class diagrams • Teacher demonstrates exemplar diagram on p76 • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why UML is needed • Start activity 2 on p77 Homework: • Students use Formulator Tarsia to make a matching activity with key terms / symbols and then swap with each other.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 74–77 AdaCS notes and questions: https://adacomputerscience.org/concepts/oop_class_diagrams?topic=object_oriented_programming	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity Problem solving

Week 12

Lesson 2

Topic 13.1.1: Object-oriented programming c) Unified modelling language (UML) class diagrams Lesson title: Further practice on UML class diagrams	13.1.1 Object-oriented programming (OOP) paradigm: c) Unified modelling language (UML) class diagrams Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • Interpret and complete UML diagrams • Identify OOP features implemented in code	Activities: • Discuss UML diagrams again • Teacher demonstrates identifying OOP features in code (see example on p79) • Complete activity 2 on p77 Homework: • Look for quizzes about OOP and UML on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 79–80 Lucid Software video: https://www.youtube.com/watch?v=6XrL5jXmTwM	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity Problem solving
--	---	--	--	---	---

Week 12

Lesson 3

Topic 13.1.2: Functional programming paradigm a) Terminology Lesson title: Functional programming terminology	13.1.2 Functional programming paradigm: a) Terminology Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss functional programming • Option to link to functional programming in Unit 4 Topic 20.1.2 • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why functional programming 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 2: pp. 80–81 IsaacCS notes and questions: https://isaacomputerscience.org/topics/functional_programmin g	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation 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 13					
Lesson 1					
Topic 13.1.2: Functional programming paradigm a) Concepts Lesson title: Functional programming concepts	13.1.2 Functional programming paradigm: b) Concepts: - Pure functions - First-class functions - Higher-order functions <i>Students will know and understand:</i> - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss pure functions, first-class functions and higher-order functions - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why functional programming is needed - Complete activity 3 on p82 Homework: - Students create examples of pure/impure functions, first-class and higher-order functions	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 81–82 Craig'n'Dave video : https://www.youtube.com/watch?v=xjMmrOx-ZA4&list=PLCiOXwirraUDUYF_qDYcZV8Hce8dsE_Ho&index=175&t=9s	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity Critical thinking Reasoning

Week 13

Lesson 2

Topic 13.1.2: Functional programming paradigm Terminology b) Concepts Lesson title: More functional programming concepts	13.1.2 Functional programming paradigm: b) Concepts: • Recursion • Immutable variables • Avoid side-effects Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Maths skills: Using higher-order functions in calculations	Activities: • Discuss • Option to • Teacher demonstrates • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why functional programming is needed Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 82–83 AdaCS notes and questions: https://adacomputerscience.org/topics/functional_programming	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity
---	---	---	---	---	--

Week 13

Lesson 3

Topic 13.1.2: Functional programming paradigm b) Concepts Lesson title: Further practice on functional programming concepts	13.1.2 Functional programming paradigm: b) Concepts: - Pure functions - First-class functions - Higher-order functions - Recursion - Immutable variables - Avoid side-effects Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Maths skills: Using higher-order functions in calculations	Activities: - Work through examples of side-effects in William Grey video Homework: - Look for quizzes about functional programming on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 82–83 William Grey video: https://www.youtube.com/watch?v=eE08t2CnCG8	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation 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 14 Lesson 1					
Topic 13.1.3: Assembly language a) Mnemonics b) Instructions Lesson title: Assembly language mnemonics and instructions	13.1.3 Assembly language: a) Mnemonics b) Instructions: • Opcode • Operand Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss assembly language programming • Teacher demonstrates identifying the opcode and operands in an instruction (see examples on p87) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why assembly language is needed Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 84–87 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/sys_proglang_low_level?topic=programming_languages	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity

Week 14

Lesson 2

Topic 13.1.3: Assembly language c) Addressing modes Lesson title: Assembly language addressing modes	13.1.3 Assembly language: c)Addressing modes: • Immediate • Direct Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss immediate and direct addressing modes • Option to consider why additional addressing modes exist (beyond the spec) • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why addressing modes are needed Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 85–86 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/sys_proglang_low_level?topic=programming_languages	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity
---	--	---	---	---	--

Week 14

Lesson 3

Topic 13.1.3: Assembly language d) Operations Lesson title: Assembly language operations	13.1.3 Assembly language: d)Operations: - Arithmetic - Bitwise - Branch - Data movement - Halt Students will know and understand: - Operation Students will be able to: - read and write simple Assembly language code.	Activities: - Discuss the ARMLite instruction set - Teacher demonstrates worked examples on pp.90–91 - Option to use ARMLite simulator for writing assembly language programs - Students identify key terms and make notes on operation Homework: - Students complete activity 4 on p91	Pearson Edexcel International A–level Computer Science Student Book 2: pp. 86–89 AdaCS notes and questions: https://adacomputerscience.org/concepts/proglang_low_level?topic=programming_languages Peter Higginson's ARMLite simulator: https://peterhigginson.co.uk/ARMLite/	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity 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 15					
Lesson 1					
Topic 13.1.3: Assembly language e) Comments Lesson title: Assembly language comments	13.1.3 Assembly language: e) Comments Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks Students will be able to: - trace the execution of a simple Assembly language program.	Activities: - Discuss comments in assembly language programs - Teacher demonstrates worked example of tracing the execution of an assembly language program on p92 - Option to use ARMLite simulator for writing and tracing assembly language programs - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why comments are needed Homework: - Students complete activity 5 on p93	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 90–93 William Grey video (on ARMLite simulator): https://www.youtube.com/watch?v=lwt669Kogco Peter Higginson's ARMLite simulator: https://peterhigginson.co.uk/ARMLite/	Analysis Critical thinking Problem solving Interpretation	Analysis Communication Interpretation Productivity Decision making

Week 15

Lesson 2

Topic 13: Programming languages Lesson title: Topic summary	Topic 13: Programming languages 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 • Students complete activity 5 on p93 • Complete Checkpoint questions on p94		Analysis Critical thinking Problem solving Interpretation	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity Decision making
---	--	--	--	--	---

Week 15

Lesson 3

14.1 Abstract data types Topic 14.1.1: Linked list a) Structure Lesson title: Linked list structure	14.1.1 Linked list: a) Structure Students will know and understand: • Definition • Function • Characteristics • Benefits • Drawbacks	Activities: • Discuss the linked list data structure • Option to link to linked list programming in Unit 4 Topic 18.2.2 • Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: p 96 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_datastruct_linked_list?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication 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 16					
Lesson 1					
Topic 14.1.1: Linked list b) Ways to represent Lesson title: Linked list representation	14.1.1 Linked list: b) Ways to represent: - Classes - References Students will know and understand: - Definition - Function - Characteristics - Benefits - Drawbacks	Activities: - Discuss how to represent linked lists with classes and references - Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: p 98 AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_linked_list?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Productivity Intellectual interest and curiosity

Week 16

Lesson 2

Topic 14.1.1: Linked list c) Operations Lesson title: Linked list operations	14.1.1 Linked list: c) Operations: - Insert [prepend, append, in-order] - Delete - Search - Traverse Students will know and understand: - Operation Students will be able to: - Hand trace operations on a linked list	Activities: - Discuss linked list operations - Teacher demonstrates worked examples of tracing operations on a linked list (p100–101) - Students identify key terms and make notes on operation - Complete activity 1 on p101–102 Homework: - Students make up questions to pose to each other	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 98–103 HackerRank video: https://www.youtube.com/watch?v=njTh_OwMljA	Analysis Critical thinking Problem solving	Analysis Communication Productivity Intellectual interest and curiosity Critical thinking Problem solving
--	--	---	--	--	--

Week 16

Lesson 3

Topic 14.1.2: Hash table a) Structure Lesson title: Hash table structure	14.1.2 Hash table: a) Structure Students will know and understand: - Definition - Function - Characteristics - Benefits - Drawbacks	Activities: - Discuss the hash table data structure - Option to link to hash table programming in Unit 4 Topic 18.2.3 - Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: p 103 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_datastruct_hash_table?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication 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 17					
Lesson 1					
Topic 14.1.2: Hash Table b) Ways to represent Lesson title: Hash table representation	14.1.2 Hash table: b) Ways to represent: - Classes - References Students will know and understand: - Definition - Function - Characteristics - Benefits - Drawbacks	Activities: - Discuss how to represent a hash table with classes and references - Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks 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 2: p103 CS50 video: https://www.youtube.com/watch?v=nvzVHwrrubO	Analysis Critical thinking Problem solving	Analysis Communication Productivity Intellectual interest and curiosity

Week 17

Lesson 2

Topic 14.1.2: Hash Table c) Operations Lesson title: Hash table operations	14.1.2 Hash table: c) Operations: • Insert • Delete • Search • Traverse • Calculate hash values Students will know and understand: • Operation Students will be able to: • hand trace operations on a hash table. • show the contents of a hash table after operations have been performed. • calculate hash values and handle collisions. Maths skills: Calculating hash values	Activities: • Discuss operations on a hash table • Teacher demonstrates worked example 3 on p104 for calculating a hash value and inserting a value into the has table • Option to link to hashing algorithm programming in Unit 4 Topic 182,4, 18.2.5 • Students identify key terms and make notes on operation • Complete activity 2 on p104 Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: p104 AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_hash_table?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Productivity Problem solving Intellectual interest and curiosity Collaboration
---	---	---	--	---	---

Week 17

Lesson 3

Topic 14.1.2: Hash Table c) Operations Lesson title: More hash table operations	14.1.2 Hash table: c) Operations: - Handle collisions - Load factor and rehashing Students will know and understand: - Operation Students will be able to: - hand trace operations on a hash table. - show the contents of a hash table after operations have been performed. - calculate hash values and handle collisions.	Activities: - Discuss collision handling, load factor and rehashing - Teacher demonstrates worked examples 4 and 5 on p105–106 for handling a collision - Students identify key terms and make notes on operation Homework: Students complete activity 3 on p106–107	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 104–110 Computer Science Lessons video : https://www.youtube.com/watch?v=KyUTuwz_b7Q	Analysis Critical thinking Problem solving	Analysis Communication Problem solving 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 18					
Lesson 1					
Topic 14.1.3: Graph a) Structure b) Ways to represent Lesson title: Graph structure and representation	14.1.3 Graph: a) Structure b) Ways to represent: - Adjacency matrix - Adjacency list <i>Students will know and understand:</i> - Definition - Function - Characteristics - Benefits - Drawbacks	Activities: - Discuss the graph data structure and how it can be represented with an adjacency matrix or adjacency list - Option to link to graph programming in Unit 4 Topic 18.2.6 - Teacher demonstrates worked examples 7 & 8 on p112–113 showing adjacency lists and worked examples 9 & 10 on p114–115 showing adjacency matrices - Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks Homework: - Students complete activity 4 on p115	Pearson Edexcel International A–level Computer Science Student Book 2: pp. 110–115 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_datastruct_graph_implementation?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Productivity Intellectual interest and curiosity Problem solving

Week 18

Lesson 2

Topic 14.1.3: Graph c) Types Lesson title: Graph types	14.1.3 Graph: c) Types: • Undirected • Directed • Weighted Students will know and understand: • Definition • Function • Characteristics • Benefits • Drawbacks	Activities: • Discuss undirected, directed and weighted graphs • Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks • Students discuss examples of graphs as an abstraction of real-life non-linear relationships between objects Homework: • Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 111 IsaacCS notes and questions: https://isaaccomputerscience.org/concepts/dsa_datastruct_graph?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Intellectual interest and curiosity
---	--	--	---	---	---

Week 18 Lesson 3					
Topic 14.1.3: Graph d) Operations Lesson title: Graph operations	14.1.4 Graph: d) Operations: - Insert - Delete - Search Students will know and understand: - Operation Students will be able to: - hand trace operations on a graph. - show the contents of a graph after operations have been performed.	Activities: - Discuss operations to insert, delete and search the graph - Teacher demonstrates worked examples on p116–118 - Students identify key terms and make notes on operation Homework: Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 115–118 AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_graph_implementation?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Problem solving 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 19					
Lesson 1					
Topic 14.1.3: Graph d) Operations Lesson title: More graph operations	14.1.3 Graph: d) Operations: • Traverse [breadth, depth] <i>Students will know and understand:</i> • Operation <i>Students will be able to:</i> • hand trace operations on a graph. • show the contents of a graph after operations have been performed.	Activities: • Discuss operations to traverse a graph (breadth-first and depth-first) • Teacher demonstrates worked examples on p118–120 • Students identify key terms and make notes on operation • Complete activity 5 on p120 Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 118–122 AdaCS notes and questions: https://adacomputerscience.org/concepts/path_dfs_bfs?topic=data_structures HackerRank video: https://www.youtube.com/watch?v=zaBhtODELOW	Analysis Critical thinking Problem solving	Analysis Communication Problem solving Intellectual interest and curiosity

Week 19

Lesson 2

Topic 14.1.4: Tree a) Structure b) Types Lesson title: Tree structure and types	14.1.4 Tree: a) Structure b) Types: - Binary tree - Binary search tree Students will know and understand: - Definition - Function - Characteristics - Benefits - Drawbacks	Activities: - Discuss the structure and types of a tree data structure - Option to link to tree programming in Unit 4 Topic 18.2.7 - Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks Homework: Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 122 & 125 IsaacCS notes and questions: https://isaaccomputerscience.org/concepts/dsa_datastruct_tree?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Productivity Intellectual interest and curiosity
--	--	---	---	---	---

Week 19 Lesson 3					
Topic 14.1.4: Tree c) Operations Lesson title: Tree operations	14.1.5 Tree: c) Operations: - Insert - Delete - Search Students will know and understand: - Operation Students will be able to: - hand trace operations on a tree. - show the contents of a tree after operations have been performed.	Activities: - Discuss the operations of a tree data structure - Teacher demonstrates worked examples on p123–128 - Students identify key terms and make notes on operation - Complete activity 6 on p126 Homework: - Students complete activity 7 on p129	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 123–129 Colleen Lewis video: https://www.youtube.com/watch?v=wclRPqTR3Kc	Analysis Critical thinking Problem solving	Analysis Communication Problem solving 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 20					
Lesson 1					
Topic 14.1.4: Tree c) Operations Lesson title: More tree operations	14.1.4 Tree: c) Operations: • Traverse [pre-order, in-order, post-order] Students will know and understand: • Operation Students will be able to: • hand trace operations on a tree. • show the contents of a tree after operations have been performed.	Activities: • Discuss algorithms to traverse tree data structures (pre-order, in-order and post-order) • Teacher demonstrates worked examples using a video • Option to link tree traversal programming in Unit 4 Topic 18.2.8 • Students identify key terms and make notes on operation • Complete activity 8 on p130–131 Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 129–132 Techdose video: https://www.youtube.com/watch?v=WLvU5EQVZqY AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_tree_traversals?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Problem solving Intellectual interest and curiosity

Week 20

Lesson 2

Topic 14.1.5: Circular queue a) Structure Lesson title: Circular queue structure	14.1.6 Circular queue: a) Structure Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks	Activities: - Discuss the structure of a circular queue data structure - Option to link to circular queue programming in Unit 4 Topic 18.2.1 - Students identify key terms and make notes on definition, function, characteristics, benefit and drawbacks Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: p 132 IsaacCS notes and questions: https://isaaccomputerscience.org/concepts/dsa_datastruct_queue?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Productivity Intellectual interest and curiosity
---	---	--	---	---	---

Week 20

Lesson 3

Topic 14.1.5: Circular queue b) Operations Lesson title: Circular queue operations	14.1.7 Circular queue: b) Operations: • Insert • Delete • Search Students will know and understand: • Operation Students will be able to: • hand trace operations on a circular queue. • show the contents of a circular queue after operations have been performed.	Activities: • Discuss operations on a circular queue data structure • Teacher demonstrates the worked example on p133–135 • Students identify key terms and make notes on operation Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 133–136 AdaCS notes and questions: https://adacomputerscience.org/concepts/struct_queue?topic=data_structures	Analysis Critical thinking Problem solving	Analysis Communication Problem solving 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 21					
Lesson 1					
Topic 14: Structuring Data Lesson title: Topic summary	Topic 14: 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 p136		Analysis Critical thinking Problem solving Communication	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity

Week 21

Lesson 2

15.1 Algorithm design Topic 15.1.1: Problem-solving techniques a) Brute force/exhaustive b) Divide and conquer Lesson title: Brute force and divide and conquer	15.1.1 Problem-solving techniques: a) Brute force/exhaustive b) Divide and conquer Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss brute force and divide and conquer as problem-solving techniques • Option to try some of the examples from the videos or textbook • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why these techniques are needed Homework: • Students research well-known problems which suit these problem-solving approaches	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 138–140 Nikhil Lohia video (brute force): https://www.youtube.com/watch?v=kdTpUjd7IG8 Nikhil Lohia video (divide and conquer): https://www.youtube.com/watch?v=yePqwt8wSi8	Analysis Critical thinking Problem solving	Analysis Problem solving Intellectual interest and curiosity Self-direction
---	---	---	---	---	---

Week 21

Lesson 3

Topic 15.1.1: Problem-solving techniques c) Greedy algorithms d) Heuristics Lesson title: Greedy algorithms and heuristics	15.1.1 Problem-solving techniques: c) Greedy algorithms d) Heuristics <i>Students will know and understand:</i> • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	<i>Activities:</i> • Discuss greedy algorithms and heuristics and problem-solving techniques • Option to try some of the examples from the videos or textbook • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why these techniques are needed <i>Homework:</i> • Students research well-known problems which suit these problem-solving approaches	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 140–141 Nikhil Lohia video (greedy algorithms): https://www.youtube.com/watch?v=3H2G3KuEiRU Headless Professor video (heuristics): https://www.youtube.com/watch?v=jJ3CV3yhajM	Analysis Critical thinking Problem solving	Analysis Problem solving Intellectual interest and curiosity Self-direction
--	--	--	---	---	---

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 15.1.1: Problem-solving techniques e) Backtracking f) Recursion Lesson title: Backtracking and recursion	15.1.1 Problem-solving techniques: e) Backtracking f) Recursion <i>Students will know and understand:</i> • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss backtracking and recursion as problem-solving techniques • Option to try some of the examples from the videos or textbook • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why these techniques are needed Homework: • Students research well-known problems which suit these problem-solving approaches	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 142–143 Nikhil Lohia video (backtracking): https://www.youtube.com/watch?v=51Zy1ULau1s Nikhil Lohia video (recursion): https://www.youtube.com/watch?v=FTTHkmnvzIM	Analysis Critical thinking Problem solving	Analysis Problem solving Intellectual interest and curiosity Self-direction

Week 22

Lesson 2

Topic 15.1.1: Problem-solving techniques g) Tracing algorithms Lesson title: Tracing algorithms	15.1.1 Problem-solving techniques: g) Tracing algorithms Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss tracing algorithms as a problem-solving technique • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why these techniques are needed • Complete activity 1 on p144 Homework: • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 144 AdaCS notes and questions: https://adacomputerscience.org/concepts/design_pseudocode?topic=program_design	Analysis Critical thinking Problem solving	Analysis Problem solving Communication Interpretation
--	--	--	---	---	---

Week 22

Lesson 3

15.2 Algorithm	15.2.1 Merge sort algorithm	Activities:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 145–147	Analysis Critical thinking Problem solving	Analysis Problem solving Interpretation
Topic 15.2.1: Merge sort algorithm	Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - Hand trace the operation of a merge sort algorithm	- Discuss the merge sort algorithm - Teacher demonstrates worked example on p146 - Option to link to merge sort programming in Unit 4 Topic 21.1.1 - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks - Complete activity 2 on p147 Homework: - Students complete questions on IsaacCS	IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_search_merge?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 23					
Lesson 1					
Topic 15.2.2: Quick sort algorithm Lesson title: Quick sort algorithm	15.2.2 Quick sort algorithm Students will know and understand: - Definition - Function - Characteristics - Operation - Benefits - Drawbacks Students will be able to: - Hand trace the operation of a quick sort algorithm	Activities: - Discuss the quick sort algorithm - Teacher demonstrates worked example on p148 - Option to link to quick sort programming in Unit 4 Topic 21.1.2 - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks - Complete activity 3 on p149 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 147–149 AdaCS notes and questions: https://adacomputerscience.org/concepts/sort_quick?topic=sorting	Analysis Critical thinking Problem solving	Analysis Problem solving Interpretation

Week 23

Lesson 2

Topic 15.2.3: Dijkstra's shortest path algorithm Lesson title: Dijkstra's shortest path algorithm	15.2.3 Dijkstra's shortest path algorithm <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 Dijkstra's shortest path algorithm	<i>Activities:</i> • Discuss Dijkstra's shortest path algorithm • Teacher demonstrates worked example on pp. 149–153 • Option to link to Dijkstra's shortest path algorithm programming in Unit 4 Topic 21.2.1 • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks • Complete activity 4 on p155 <i>Homework:</i> • Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 149–156 AdaCS notes and questions: https://adacomputerscience.org/concepts/path_dijkstra?topic=pathfinding	Analysis Critical thinking Problem solving	Analysis Problem solving Intellectual interest and curiosity Productivity
--	---	--	---	---	---

Week 23

Lesson 3

Topic 15.2.3: Dijkstra's shortest path algorithm Lesson title: Further practice on Dijkstra's shortest path algorithm	15.2.3 Dijkstra's shortest path algorithm Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks Students will be able to: • Hand trace the operation of Dijkstra's shortest path algorithm	Activities: • Students complete questions on IsaacCS Homework: • Look for quizzes about functional programming on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_search_dijkstra?topic=pathfinding	Analysis Critical thinking Problem solving	Analysis Problem solving 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 24 Lesson 1					
15.3 Algorithmic efficiency Topic 15.3.1: Big O notation Lesson title: Time complexity	15.3.1 Big O notation: a) Time complexity: - Constant - Logarithmic - Linear - Linearithmic - Polynomial Students will know and understand: - Definition - Function - Why needed Students will be able to: - Analyse algorithms to determine best and worst Big O. - Give the best and worst Big O time complexity for common algorithms. - Determine the Big O time complexity for given code.	Activities: - Discuss Big O time complexity and the relative order of different complexities - Students identify key terms and make notes on definition, function, and why Big O notation is needed as well - Complete activity 5 on p164 Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 156–161 IsaacCS notes and questions: https://isaacomputerscience.org/concepts/dsa_cmplx_big_o?topic=complexity	Analysis Critical thinking Problem solving	Analysis Critical thinking Intellectual interest and curiosity Communication Decision making Reasoning

	Maths skills: Determining Big O complexity				
Week 24 Lesson 2					
Topic 15.3.1: Big O notation Lesson title: Further practice on time complexity	As above	Activities: - Discuss how to analyse an algorithm or code to determine best- and worst-case Big O - Teacher demonstrates worked example on p161–163 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 161–164 AdaCS notes and questions: https://adacomputerscience.org/concepts/complex_big_o?topic=complexity	Analysis Critical thinking Problem solving	Analysis Critical thinking Intellectual interest and curiosity Communication

Week 24

Lesson 3

15.4 Boolean logic Topic 15.4.1: Boolean algebra a) Commutative b) Associative c) Absorption d) Distributive f) Double negation Lesson title: Boolean algebra laws	15.4.1 Boolean algebra: a) Commutative b) Associative c) Absorption d) Distributive f) Double negation Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - Use Boolean algebra to manipulate Boolean expressions. Maths skills: Using Boolean algebra	Activities: - Discuss laws of Boolean algebra (commutative, associative, absorptive, distributive, double negative) - Teacher demonstrates examples of manipulating Boolean expressions - Option to practise using the Pearson tools if students will be taking the exam onscreen - Students identify key terms and make notes on definition, function, operation, as well as why Boolean algebra is needed Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 165–168 IsaacCS notes and questions: https://isaaccomputerscience.org/concepts/sys_bool_boolean_algebra?topic=boolean_logic	Analysis Critical thinking Problem solving Communication	Analysis Critical thinking Intellectual interest and curiosity 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 25					
Lesson 1					
Topic 15.4.1 Boolean algebra e) De Morgan's law Lesson title: More Boolean algebra laws	15.4.1 Boolean algebra: e) De Morgan's Students will know and understand: - Definition - Function - Why needed - Operation Students will be able to: - Use Boolean algebra to manipulate Boolean expressions. Maths skills: Using Boolean algebra	Activities: - Discuss de Morgan's law - Teacher demonstrates examples of manipulating Boolean expressions - Option to practise using the Pearson tools if students will be taking the exam onscreen - Students identify key terms and make notes on definition, function, operation, as well as why Boolean algebra is needed Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 167–168 AdaCS notes and questions: https://adacomputerscience.org/concepts/boolean_de_morgans?topic=boolean_logic	Analysis Critical thinking Problem solving Communication	Analysis Critical thinking Intellectual interest and curiosity Problem solving Communication

Week 25

Lesson 2

Topic 15.4.1 Boolean algebra Lesson title: Further practice with Boolean algebra laws	15.4.1 Boolean algebra: a) Commutative b) Associative c) Absorption d) Distributive e) De Morgan's f) Double negation Students will know and understand: • Definition • Function • Why needed • Operation Students will be able to: • Use Boolean algebra to manipulate Boolean expressions. Maths skills: Using Boolean algebra	Activities: • Use questions from sample and specimen exam papers to test exam readiness • Search for videos with worked examples to pause and solve (but look for the appropriate notation) • Option to practise using the Pearson tools if students will be taking the exam onscreen Homework: • Look for quizzes about Boolean algebra on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.		Analysis Critical thinking Problem solving Communication	Analysis Critical thinking Intellectual interest and curiosity Problem solving Communication
--	---	---	--	--	---

Week 25

Lesson 3

Topic 15.4.2: Karnaugh map Lesson title: Karnaugh map	15.4.2 Karnaugh map Students will know and understand: • Definition • Function • Why needed • Operation Students will be able to: • Use Karnaugh maps to simplify Boolean algebraic expressions Maths skills: Using Karnaugh maps	Activities: • Discuss Karnaugh maps as a tool for simplifying Boolean expressions • Teacher demonstrates worked examples on p170–172 • Option to practise using the Pearson tools if students will be taking the exam onscreen • Students identify key terms and make notes on definition, function, operation, as well as why Karnaugh maps are needed Homework: • Students complete activity 6 on p173	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 169–173 AllAboutElectronics video (2– and 3–variable examples): https://www.youtube.com/watch?v=Iw1STgKUpW0&t=120s AllAboutElectronics video (4–variables): https://www.youtube.com/watch?v=v08rGyVYall	Analysis Critical thinking Problem solving Communication	Analysis Critical thinking Intellectual interest and curiosity 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 26					
Lesson 1					
Topic 15.4.2: Karnaugh map Lesson title: Further practice with Karnaugh maps	15.4.2 Karnaugh map <i>Students will know and understand:</i> - Definition - Function - Why needed - Operation <i>Students will be able to:</i> - Use Karnaugh maps to simplify Boolean algebraic expressions <i>Maths skills:</i> Using Karnaugh maps	<i>Activities:</i> - Use questions from sample and specimen exam papers to test exam readiness - Search for videos with worked examples to pause and solve (but look for the appropriate notation) - Option to practise using the Pearson tools if students will be taking the exam onscreen <i>Homework:</i> - Look for quizzes about Karnaugh maps on platforms like Blooket/ Kahoot/ Quizlet/ Quizizz/ Socrative/ etc.	Dragonfly Statistics video: https://www.youtube.com/watch?v=Enb3C4yZ5rw	Analysis Critical thinking Problem solving Communication	Analysis Critical thinking Intellectual interest and curiosity Problem solving Communication

Week 26

Lesson 2

Topic 15: Problem Solving Lesson title: Topic summary	Topic 15: 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 Checkpoint questions on p173		Analysis Critical thinking Problem solving Communication	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity
---	--	--	--	---	--

Week 26

Lesson 3

16.1 Emerging technologies Topic 16.1.1: Encryption a) Digital signatures and certificates Lesson title: Digital signatures and certificates	16.1.1 Encryption: a) Digital signatures and certificates Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss digital signatures and certificates - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why digital certificates and signatures are needed - Complete activity 1 on p177 Homework: - Students complete questions on AdaCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 175–177 Computerphile video: https://www.youtube.com/watch?v=s22eJleVLTU AdaCS notes and questions: https://adacomputerscience.org/concepts/sec_digital_signatures	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics Intellectual interest and curiosity Teamwork
---	--	---	---	--	--

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 16.1.1: Encryption b) Hash digest and use of public/private keys Lesson title: Hash digest and public/private keys	16.1.1 Encryption: b) Hash digest and use of public/private keys Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss hash digests and the use of public/private keys - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why hash digest and public/private keys are needed Homework: - Students complete questions on IsaacCS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 177–178 Computerphile video: https://www.youtube.com/watch?v=b4b8ktEV4Bg IsaacCS notes and questions: https://isaacomputerscience.org/concepts/net_sec_digital_signatures	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics

Week 27

Lesson 2

Topic 16.1.2: Blockchain Lesson title: Blockchain	16.1.2 Blockchain 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 blockchain	Activities: • Discuss blockchain • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why blockchain is needed Homework: • Students investigate an application of blockchain	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 178–181 Computerphile video (bitcoin and blockchain): https://www.youtube.com/watch?v=qcuc3rgwZAE&t=297s Computerphile video (alternative uses of blockchain): https://www.youtube.com/watch?v=qBAOsB6ETrY GeeksForGeeks notes: https://www.geeksforgeeks.org/ethical-hacking/how-does-the-blockchain-work/	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics
--	---	--	---	---	--

Week 27

Lesson 3

Topic 16.1.2: Blockchain Lesson title: Further practice with blockchain	16.1.2 Blockchain 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 blockchain	Activities: Students present their research to the group and discuss the findings		Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics
---	---	--	--	---	--

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					
Topic 16.1.3: Quantum computing a) Qubits b) Principles Lesson title: Quantum computing principles	16.1.3 Quantum computing: a) Qubits b) Principles • Superposition • Entanglement • Decoherence • Interference Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Discuss qubits and principles of quantum computing • Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why quantum computing 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 2: pp. 181–183 GeeksForGeeks notes: https://www.geeksforgeeks.org/blogs/introduction-quantum-computing/	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics Productivity

Week 28

Lesson 2

Topic 16.1.3: Quantum computing c) Threat d) Quantum-safe cryptography Lesson title: Impact of quantum computing	16.1.3 Quantum computing: c) Threat d) Quantum-safe cryptography Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss threats from quantum computing and the need for quantum-safe cryptography - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why quantum-safe cryptography is needed Homework: - Students complete activity 2 on p184	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 183–184 AdaCS notes and questions: https://adacomputerscience.org/concepts/encrypt_quantum?topic=encryption	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics Innovation Intellectual interest and curiosity Teamwork
--	--	---	--	--	--

Week 28

Lesson 3

Topic 16.1.4: Deep learning Lesson title: Deep learning	16.1.4 Deep learning Students will know and understand: - Definition - Function - Characteristics - Why needed - Operation - Benefits - Drawbacks	Activities: - Discuss deep learning - Students identify key terms and make notes on definition, function, characteristics, operation, benefit and drawbacks, as well as why deep learning is needed - Complete activity 3 on p186 Homework: - Students create a presentation from their research in activity 3	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 184–186 AdaCS notes and questions: https://adacomputerscience.org/concepts/ml_engines	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics Innovation Teamwork
--	--	--	--	--	---

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 16.1.4: Deep learning Lesson title: Further practice with deep learning	16.1.4 Deep learning Students will know and understand: • Definition • Function • Characteristics • Why needed • Operation • Benefits • Drawbacks	Activities: • Students present their research to the group and discuss the findings • Explore up-to-date videos about deep learning	3Blue1Brown video: https://www.youtube.com/watch?v=aircAruvnKk Computerphile video: https://www.youtube.com/watch?v=l42lr8AlrHk	Analysis Critical thinking Problem solving Ethics	Analysis Critical thinking Problem solving Communication Ethics

Week 29

Lesson 2

16.2 Professionalism 16.2.1 Responsible practice a) Ethical issues b) Legal frameworks c) Codes of conduct Lesson title: Responsible practice	16.2.1 Responsible practice: a) Ethical issues b) Legal frameworks c) Codes of conduct Students will know and understand: - Definition - Function - Characteristics - Why needed	Activities: - Discuss the need for responsible practice - Students identify key terms and make notes on definition, function, characteristics, as well as why responsible practice is needed - Complete activity 4 on p189 Homework: - Students build a presentation from their research for activity 4	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 187–189 AdaCS notes and questions (ethical issues): https://adacomputerscience.org/topics/impacts_of_tech AdaCS notes and questions (legal frameworks): https://adacomputerscience.org/topics/legislation	Analysis Critical thinking Ethics Communication	Interpretation Critical thinking Communication Ethics Productivity Personal and social responsibility Teamwork
---	--	---	---	--	--

Week 29

Lesson 3

16.2.1 Responsible practice Lesson title: Further practice with responsible conduct	16.2.1 Responsible practice: Students will know and understand: - Definition - Function - Characteristics - Why needed	Activities: - Students present their research to the group and discuss their findings Homework: - Students search for examples of codes of conduct		Analysis Critical thinking Problem solving Ethics	Interpretation Critical thinking Communication Ethics Productivity
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 30 Lesson 1					
Topic 16: Emerging Technologies and Professional Practice Lesson title: Topic summary	Topic 16: Emerging technologies and professional practice 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 p189		Analysis Critical thinking Problem solving Ethics	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity
Week 30 Lesson 2					
Revision				Analysis Critical thinking Problem solving Ethics	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity

Week 30
Lesson 3

Revision				Analysis Critical thinking Problem solving Ethics	Critical thinking Productivity Initiative Self-direction Analysis Intellectual interest and curiosity
----------	--	--	--	--	---

Unit 4: Advanced 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 17.1.1 Select and use appropriate operators and flow control constructs Lesson title: Program constructs	Students will be able to: 17.1.1 select and use appropriate operators and flow control constructs when designing solutions to complex problems Maths skills: Using appropriate operators	Activities: - Teacher introduction to the unit covering an outline of the topics and the assessment method. - Teacher recap on previously covered material on operators and flow control - Class practical activity. Students are given a series of question to answer on operators and selection/iteration constructs. These should include some code snippets which either have errors or are incomplete so students can identify the errors and/or complete the code - Teacher introduction to resources for this units including the specification document, PLS and GPG and the Sample exam paper (and any past exam papers when they are available). Homework: Ask students to read the pages from the specification which cover this unit and the PLS. Note down any issues or concerns they have such as lack of existing underpinning Students will knowledge or terms they don't understand.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 191–193 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 170–174	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 1 Lesson 2					
Topic 17.1.2 Select and use appropriate data types, variables and constants Lesson title: Data types, variables and constants	Students will be able to: 17.1.2 select and use appropriate datatypes, variables and constants when designing solutions to complex problems Maths skills: Handling floating point data	Activities: - Teacher recap on previously covered material on datatypes, variables and constants. - Teacher demonstration of how Python displays float datatype numbers of different sizes (including the use of scientific format and how to control this with the format function) - Class practical activity. Students are given a series of questions to answer on datatypes, variables and constants. These should include some code snippets which either have errors or are incomplete so students can identify the errors and/or complete the code Homework: Ask students to review the pages of student book on which cover the material on operators, flow control constructs, datatypes, variables and constants – pp 170 – 186	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 193–194 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 179–180, 192–193	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 1 Lesson 3					
Topic 17.1.2 Select and use appropriate Data types, variables and constants Lesson title: String manipulation methods	Students will be able to: 17.1.2 select and use appropriate datatypes, variables and constants when designing solutions to complex problems	Activities: - Teacher demonstration showing the use of string manipulation methods such as those which format strings (capitalize(), upper() etc) and those which return Boolean values (startswith(), isnumeric() etc). - Class practical activity with students writing programs to demonstrate the use of string manipulation methods - Class quiz with students shown code snippets utilizing string methods. Students need to identify what the functions return. Could also add questions on other topics covered so far, such as code with errors in it where students need to identify what the error is and how to correct it. Homework: Ask students to read the official Python documentation on string methods but remind them that only those methods listed in the PLS will be used in the exam:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 193–194 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 207–212 https://docs.python.org/3/library/stdtypes.html#string-methods	Critical thinking Problem solving Analysis	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 2					
Lesson 1					
Topic 17.1.2 Select and use appropriate Data types, variables and constants Lesson title: Additional operators	Students will be able to: 17.1.2 select and use appropriate datatypes, variables and constants when designing solutions to complex problems Maths skills: Using appropriate operators	Activities: - Teacher demonstration showing the use of the increment, decrement and membership operators - Class practical with students writing programs to demonstrate the use of additional operators. An example can be found in Student Book 2, Activity 1 page 198 - Class practical with students given code snippets using the features covered so far but having syntax and/or logic errors that they need to correct - Class activity – a ‘quick fire’ fun quiz with true/false questions and/or ‘what does this output’ code snippets Homework: Ask learners to read this article about python membership and identity operators	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 194–198 Video explaining membership operators: https://www.youtube.com/watch?v=OJ5E7VLsZQM Python documentation on membership and identity operators: https://www.geeksforgeeks.org/python/python-membership-identity-operators-not-not/	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 2					
Lesson 2					
Topic 17.1.3 Select and use appropriate one- and two-dimensional data structures Lesson title: One- and two-dimension data structures	Students will be able to: 17.1.3 select and use appropriate one- and two-dimensional data structures when designing solutions to complex problems.	Activities: - Teacher recap on previously covered material on one- and two-dimensional data structures including the differences between lists, tuples, dictionaries etc. - Teacher demonstration of the correct way to create 2D matrix using list comprehension. - Class practical with students writing programs to demonstrate the use of one- and two-dimension data structures. - Class task with students provided with exam style short questions (identify errors, complete code snippet etc.) Homework: Ask students to search for Python lists and read the Real Python in-depth article on lists	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 198–200 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 193–200 Web page discussing Python lists: https://realpython.com/python-list/	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 2 Lesson 3					
Topic 17.2.1 Using computational thinking techniques Lesson title: Computational thinking	Students will be able to: 17.2.1 select and use appropriate computational thinking techniques when designing solutions to complex problems.	Activities: - Teacher recap on previously covered material on computational thinking - Class practical with learners presented with a problem in the form of an application description that they need to apply computational thinking techniques to (activity 2 on page 205 of Student Book 2 provides an example). - Ask students to make a list of the Python features covered so far and classify them into those that they fully understand, partially understand and those that they feel they have a poor understanding of. Review the list and deal with any issues. Homework: Ask students to complete a design (using computational thinking techniques) for an application and identify the basic data structures that the code might require.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 200–206 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 278 – 284		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 3					
Lesson 1					
Topic 17.2.2 Using IDEs and other tools Lesson title: Development Environments	Students will be able to: 17.2.2 select and use appropriate tools when designing solutions to complex problems	Activities: - Teacher introduction to the lesson and a demonstration of 1 or 2 different IDEs that students have not previously used (e.g. VS code) - Class practical with learners working pairs comparing using the IDLE IDE and VS Code (or other IDEs). If it's not possible to get VS code or another IDE installed, then students could review on-line videos of carrying out different tasks in another IDE compared to IDLE. - Class discussion on the benefits and drawback of the different IDEs they have tried. What makes a good IDE? What features do they find most useful? Homework: Ask student to do some research into what IDEs and other development tools are recommended (e.g. search for 'best IDE for Python')	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 206–208 Video showing how to use Python within VS code: https://www.youtube.com/watch?v=D2cwvpJSBX4		

Week 3

Lesson 2 & 3

Topic 17.2.3 Using libraries, files, modules and subprograms Lesson title: Libraries, files, modules and subprograms (double lesson)	Students will be able to: 17.2.2 select and use appropriate libraries, files, modules and subprograms when designing solutions to complex problems	Activities: - Teacher introduction to the lesson and an introduction to different library modules including the operator, random and maths. By this stage a demonstration should not be needed, just a brief overview and lead into the practical session. Ask learners to look up details of the library functions to be used in the Python documentation if they need more information. - Class practical with learners writing programs to demonstrate the use of functions from the operator, random and maths libraries (Activity 4 on page 212 of Student Book 2 provides an example). - Class exercise with learners given various questions to answer. These can include small programming tasks, identifying errors, completing code snippets etc. Homework: Ask student to search for 'Python rock paper scissors' and read the Real Python article on creating a python game using the random module, they can also implement a version of the game.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 209–213 Web page describing a 'rock, paper, scissors' game: https://realpython.com/python-rock-paper-scissors/		
---	---	--	--	--	--

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 & 2					
Topic 17: Programming Lesson title: Project (double lesson)	Students will be able to: 17.1.1 select and use appropriate operators and flow control constructs when designing solutions to complex problems 17.1.2 select and use appropriate datatypes, variables and constants when designing solutions to complex problems 17.1.3 select and use appropriate one- and two-dimensional data structures when designing solutions to complex problems 17.2.1 select and use appropriate computational thinking techniques when designing solutions to complex problems. 17.2.2 select and use appropriate libraries, files, modules and subprograms when designing solutions to complex problems	Activities: - Teacher introduces the lesson and recap on the previous lessons covering this topic. - Class practical with students provided with an application description and they use computational thinking techniques to develop a design for the application and then create the application using the skills covered in this topic. - Students work in pairs to review the programs they have created. What is good about them, what could be improved? Where do their skills need to be improved? Homework: Ask students to prepare for the next session by reviewing their lesson notes from Unit 3 topic 11.3.1 (Student book 2, pages 29–33) and floating-point representation.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 191–213	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 4

Lesson 3

Topic: 18.1.1 Methods to handle floating-point data: Lesson title: Decimal class	Students will be able to: 18.1.1 select and use appropriate methods for handling floating point data Maths skills: Handling floating point data	Activities: - Teacher presentation/demonstration of how floating-point data is represented in Python and the problems this can cause. Demonstrate how the use of the decimal class can resolve the issues with calculations, Rounding and conversions. - Class practical with students writing programs which demonstrate the use of the decimal class to carry out calculations, rounding and datatype conversions. - Classroom activity, with students making a list of 4 things they have learnt this week, do they fully understand them all? Deal with any issues arising. Homework: Ask students to read the article from the Medium web site which explains some of the reasons why the Decimal class is used	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 215–216 Video explaining how the Decimal class can be used and how it solves problems with floating point calculations: https://www.youtube.com/watch?v=-4XI4B39k_U Web page describing the decimal class and its uses (may need to create a free account to read it) https://medium.com/@hadi1812001/decimal-class-in-python-an-intro-and-its-use-cases-28dc84380d87	Critical thinking Problem solving Analysis	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 5					
Lesson 1					
Topic: 18.1.1 Methods to handle floating-point data: Lesson title: Decimal class – compare, exceptions	Students will be able to: 18.1.1 select and use appropriate methods for handling floating point data Maths skills: handling floating point data	Activities: - Teacher recap on previous lesson and demonstration of additional decimal class features including comparing numbers, signals and dealing with exceptions. - Class practical with students writing programs which demonstrate the use of the decimal class to carry out comparisons, signals and dealing with exceptions (Activity 1 on page 219 of Student Book 2 provides an example) - Classroom discussion on the type of application where the decimal class might be useful. What problems might occur if the decimal class is not used. Ask students to research 'UK post office horizon problems'. What happened? Homework: Ask students to read first few pages of the official Python documentation on the Decimal class. This also contains tutorial which may be of interest to more able students but remind students that only those features of the Decimal class listed in the PLS can appear in the exam. The tutorial covers much more than this.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 216–218	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 5					
Lesson 2					

Topic: 18.1.2 Methods to handle regular expressions Lesson title: Introduction to regular expressions	Students will be able to: 18.1.2 select and use appropriate methods for handling regular expressions	Activities: - Teacher presentation on the nature and purpose of regular expressions and the re module. - Teacher demonstration of the research function, match objects and the start() and span() methods. - Class practical with students writing programs which demonstrate the use of the research function, match objects and the start() and span() methods. - Class discussion on applications where the re module and regular expressions might be useful. Homework: Ask students to read the page on the Geek for Geeks web site about the use of regex search	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 219–221 Web page about the use of regex search: https://www.geeksforgeeks.org/python/re-search-in-python/	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 5 Lesson 3					
Topic: 18.1.2 Methods to handle regular expressions Lesson title: Regular expressions – functions	Students will be able to: 18.1.2 select and use appropriate methods for handling regular expressions	Activities: - Teacher demonstration of the re findall, sub and split functions. - Class practical with students writing programs which demonstrate the use of the re findall, sub and split functions. - Class quiz with students shown code snippets using re functions already covered and they have to identify what the function will return. Homework: Ask students to complete a programming challenge using the regex functions covered so far	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 221–222 Video explaining the use of re used in this lesson: https://www.youtube.com/watch?v=EzeeypMKx7o		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 6					
Lesson 1					
Topic: 18.1.2 Methods to handle regular expressions Lesson title: Regex special characters and quantifiers	Students will be able to: 18.1.2 select and use appropriate methods for handling regular expressions	Activities: - Teacher explanation of regex special characters and quantifiers and a demonstration of their use. - Class practical with students writing programs which demonstrate the use of regex special characters and quantifiers. - Teacher demonstration of the regex101 web site at https://regex101.com/ Showing them how to use this site to develop regex codes. Homework: Ask students to create regex validation codes for a number of text and number combinations	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 222–224	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 6

Lesson 2

Topic: 18.1.2 Methods to handle regular expressions Lesson title: Regex examples	Students will be able to: 18.1.2 select and use appropriate methods for handling regular expressions	Activities: - Teacher demonstration of the use of regex character sets and matching ranges of numbers - Class practical with students writing programs which demonstrate the creation of regex codes to match some various examples of strings, including those containing just number ranges (Activity 2 on page 226 provides an example) - Classroom discussion on regex and any difficulties students have with understanding how to use them. Include some quiz type questions, such as showing a regex code and asking what string would match or showing a string and asking how to create a regex to match it. Homework: Ask students to read the first few pages of the official Python documentation on the re module. This is a lengthy and comprehensive document so students should also check the PLS to see which aspects of regex are covered the exam and only read about other aspects if they are interested.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 224–226 Python documentation covering the re module: https://docs.python.org/3/library/re.html	Critical thinking Problem solving Analysis	Critical thinking Problem solving
--	--	---	--	--	--------------------------------------

Week 6 Lesson 3					
Topic: 18.1.2 Methods to handle regular expressions Lesson title: Regex project (double lesson)	Students will be able to: 18.1.2 select and use appropriate methods for handling regular expressions	Activities: - Teacher introduces the lesson and recap on the previous lessons covering this topic. - Class practical with students provided with an application description that requires the use of a regex and they develop the application to implement the application. - Hold a regex competition with students given various tasks using the re library with the first student to correctly complete the regex code winning. Homework: Ask students to do some research into what third party libraries are commonly used in python development. What features to these libraries provide and what kind of applications are they typically used in. This research will lead into the next set of lessons on the NumPy and pandas libraries.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 215–226	Critical thinking Problem solving Analysis	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 7					
Lesson 1					
Double lesson continued from WK 6					
Week 7					
Lesson 2					
Topic: 18.1.3 Methods to handle numerical arrays (NumPy) Lesson title: NumPy introduction	Students will be able to: 18.1.2 select and use appropriate methods for handling numerical arrays Maths skills: Handling numerical arrays	Activities: - Teacher introduces NumPy library and outlines its features and purpose - Teacher explanation of how to create NumPy 2D arrays, and how to slice and filter data from arrays - Class practical with students writing program to demonstrate creating, slicing and filtering NumPy arrays - Provide learners with some code samples using features recently covered but containing syntax and/or logic errors and ask them to correct the code. Homework: Ask students to read the beginners introduction to NumPy on the official NumPy web site.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 226–227 This video provides a brief introduction to NumPy: https://www.youtube.com/watch?v=Gc-dwloxNFE Beginners introduction to NumPy https://numpy.org/doc/stable/user/absolute_beginners.html	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 7

Lesson 3

Topic: 18.1.3 Methods to handle numerical arrays (NumPy)	Students will be able to: 18.1.3 select and use appropriate methods for handling numerical arrays	Activities: - Teacher explanation of how to use conditional filters (where function) and arithmetic functions - Class practical with students writing program to demonstrate conditional filters and arithmetic functions used with NumPy arrays - Class quiz with students given code snippets using NumPy conditional filters and where() functions and asked to identify what the results would be (or vice-versa)	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 227–228	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Lesson title: NumPy arithmetic functions	Maths skills: Handling numerical arrays	Homework: Ask students to read the official NumPy documentation on creating arrays	Video explaining the use of conditional filters in NumPy: https://www.youtube.com/watch?v=L-oNZ23HCb8 NumPy documentation on creating arrays https://numpy.org/doc/stable/user/basics.creation.html		

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: 18.1.3 Methods to handle numerical arrays (NumPy) Lesson title: NumPy shape, size and reshape	Students will be able to: 18.1.3 select and use appropriate methods for handling numerical arrays Maths skills: Handling numerical arrays	Activities: - Teacher explanation of how to use NumPy shape, size and reshape functions - Class practical with students writing programs to demonstrate the use of shape, size and reshape functions used with NumPy arrays - Ask learners to review their progress in learning Python. What aspects of Python do they feel confident with? What aspects do they struggle with? What can they do to improve their weak areas? Ask them to make a learning action plan to deal with their weaknesses, Homework: Ask students to research maths functions such as exponents (powers of) arithmetic mean and standard deviation. What are these functions and what kinds of applications might require them. This is a lead into the next lesson.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 228–229 Video explaining reshaping numpy arrays: https://www.youtube.com/watch?v=JTO2ED5VrOQ	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 8 Lesson 2					
Topic: 18.1.3 Methods to handle numerical arrays (numpy) Lesson title: Numpy Datatypes and maths functions	Students will be able to: 18.1.3 select and use appropriate methods for handling numerical arrays Maths skills: Handling numerical arrays	Activities: - Teacher explanation of how to use NumPy datatypes and maths functions including averages - Class practical with students writing programs to demonstrate the use of datatypes and maths functions used with NumPy arrays - Class discussion on NumPy and the types of applications where it might be used. How is NumPy different from built in Python data structures such as lists and dictionaries? Can they think of an application that would benefit from the use of NumPy? Homework: Ask students to read the official NumPy documentation on datatypes.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 229–231 NumPy documentation on datatypes: https://numpy.org/doc/stable/user/basics.creation.html	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 8 Lesson 3					
Topic: 18.1.3 Methods to handle numerical arrays (NumPy) Lesson title: NumPy Project (double lesson)	Students will be able to: 18.1.3 select and use appropriate methods for handling numerical arrays Maths skills: Handling numerical arrays	Activities: - Teacher introduces the lesson and recaps on the previous lessons covering this topic. - Class practical with students provided with an application description that requires the use of the NumPy library, and they develop the application (Activity 3 on page 231 of Student Book 2 provides an example). - Students work in pairs reviewing the code they have written for the project. Does it work correctly? Could it be improved? - Class quiz on NumPy features. Homework: Ask students to make a revision guide covering the NumPy features listed in the PLS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 215–231	Critical thinking Problem solving Analysis	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 9					
Lesson 1					
Double lesson continued from WK 8					
Week 9					
Lesson 2					
Topic:18.1.4 Methods to handle data analysis (Pandas) Lesson title: Pandas introduction	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis Maths skills: Handling numerical arrays	Activities: - Teacher introduces the Pandas library and outlines its features and purpose - Teacher explanation of how to create pandas series and apply arithmetic operations - Class practical with students writing programs to demonstrate creating pandas series and applying arithmetic operations to them - Class activity looking at code using some of the recently covered features, but which contains syntax and/or logic error or inefficiencies (e.g. poor use of loops, deeply nested conditionals, lack of input validation or exception handling) and ask student to identify issues and suggest corrections/improvements Homework: Ask students to read the introduction to Pandas in the official documentation.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 232–233 Video explaining how to use Pandas (quite long (30 mins) so might want to watch relevant sections over several lessons): https://www.youtube.com/watch?v=EXIglIBu4EU Pandas documentation introduction : https://pandas.pydata.org/docs/user_guide/10min.html	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 9					

Lesson 3

Topic:18.1.4 Methods to handle data analysis (Pandas) Lesson title: Pandas data frames	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher explanation of how to create pandas data frames - Class practical with students writing program to demonstrate creating Pandas data frames - Class discussion on pandas. What kind of application might require pandas? Why use pandas and a Python program rather than using Excel? Homework: Ask students to read the section on data frames in the official Pandas documentation.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 233–234 Pandas documentation on data frames : https://pandas.pydata.org/docs/user_guide/dsintro.html#dataframe	Critical thinking Problem solving Analysis	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 10					
Lesson 1					
Topic:18.1.4 Methods to handle data analysis (pandas) Lesson title: Pandas functions	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher explanation of commonly used Pandas functions - Class practical with students writing program to demonstrate the use of common Pandas functions - Class quiz on Pandas functions and other features covered so far. Homework: Ask students to read the article from the Medium web site on useful Pandas functions.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 234–235 Web page describing useful Pandas functions: https://medium.com/learning-data/a-gentle-introduction-to-pythons-pandas-library-the-first-5-functions-you-need-to-Students-will-know-fc045e24f3c8	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 10

Lesson 2

Topic:18.1.4 Methods to handle data analysis (pandas) Lesson title: Pandas CSV import	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher explanation of how to import CSV files with Pandas. - Class practical with students writing programs to demonstrate how to import CSV file with Pandas. - Class discussion on CSV file import. Why is this feature important? How is the pandas CSV import different to the built in method of important CSV files in Python? What applications might want to import a CSV file? Homework: Ask student to research the concept of central tendency and data cleaning in data science. What are these concepts how are they used, why are they important? This leads into the next lesson.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 235–236 Video which explains importing files using Pandas: https://www.youtube.com/watch?v=dUpyC40cF6Q&t=58s	Critical thinking Problem solving Analysis	Critical thinking Problem solving
---	---	--	---	---	---

Week 10

Lesson 3

Topic: 18.1.4 Methods to handle data analysis (pandas) Lesson title: Pandas – central tendency and data cleaning	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher explanation of the measures of central tendency included with the Pandas library and data cleaning functions, dropna() and duplicated(). - Class practical with students writing programs to demonstrate the use of measures of central tendency and data cleaning with pandas (Activity 4 on page 242 of Student Book 2 provides an example). - Class activity with students collecting data that they can later analyse using pandas. This can be live data from the students themselves or data from the internet. The web site Kaggle.com provides a wide range of datasets. Homework: Ask students to read the section in the official pandas documentation on the read CSV function.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 237–239 Pandas documentation on the read CSV function: https://pandas.pydata.org/docs/reference/api/pandas.read_csv.html#pandas.read_csv	Critical thinking Problem solving Analysis	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 11					
Lesson 1					
Topic:18.1.4 Methods to handle data analysis (pandas) Lesson title: Pandas – sort and select	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher explanation of the sort and select functions available in the pandas library - Class practical with students writing programs to demonstrate the use of the sort_values() function and conditional select. - Set students a data analysis task using the data they have previously identified. For example, they could apply the measures of central tendency to a data set, sort the data set and select values which match a given criterion. Homework: Ask students to read the official pandas documentation on the sort_values() function.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 239–240 Pandas documentation on the sort_values() function https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.sort_values.html#pandas.DataFrame.sort_values	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 11 Lesson 2					
Topic:18.1.4 Methods to handle data analysis (Pandas) Lesson title: Pandas – data conversions, date and time	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher explanation of the pandas functions for data conversion and date/time - Class practical with students writing programs to demonstrate the use of pandas functions for dates and times and converting data - Class activity with students completing some shorter versions of exam style questions, for example code with syntax and/or logic errors, code that needs completing etc. Homework: Ask students to read the guide to working with dates and times in pandas from the Medium website	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 240–243 Web page on the Medium web site explaining how to work with dates and times in Pandas https://medium.com/@noorfatimaafzalbutt/working-with-dates-and-times-in-pandas-a-comprehensive-guide-fda47929ace4	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 11 Lesson 3					
Topic:18.1.4 Methods to handle data analysis (Pandas) Lesson title: pandas – project (double lesson)	Students will be able to: 18.1.4 select and use appropriate methods for handling data analysis	Activities: - Teacher introduces the lesson and recaps on the previous lessons covering this topic. - Class practical with students provided with an application description that requires the use of the Pandas library and they develop the application. - Students work in pairs reviewing the code they have written for the project. Does it work correctly? Could it be improved? - Class quiz on Pandas features. Homework: Ask students to make a revision guide covering the pandas features listed in the PLS	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 232–243	Critical thinking Problem solving Analysis	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 12					
Lesson 1					
Double lesson continued from WK11					
Week 12					
Lesson 2 & 3					
Topic: 18.2.1 Methods to represent and handle a circular queue, implemented as a fixed-length list. Lesson title: Circular Queues (Double Lesson)	Students will be able to: 18.2.1 select and use appropriate methods to handle a circular queue, implemented as a fixed-length list.	Activities: - Teacher introduces the topic of ADTs, reminding learners what was covered in Unit 2. - Teacher explanation of the nature of a circular queue (the theory of circular queues is covered in Unit 3, topic 14.1.5) - Practical activity with students writing a program to implement an example of a circular queue including functions to: - Create a queue - Enqueue items - Dequeue items - Identify if the queue is full or empty (isempty and isfull) - Class discussion on queues and circular queues. What are the differences? How are they used (links to Unit 1 and 3)? Homework: Ask students to read the article on circular queues from the Geek for Geeks web site.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 244–247 (practical implementation) and 132–136 (theory) Web page on the Geek for Geeks site covering circular queues: https://www.geeksforgeeks.org/dsa/circular-queue-in-python/	Critical thinking Problem solving Analysis	Critical thinking Problem solving
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 13 Lesson 1 & 2					
Topic: 18.2.2 Methods to represent and handle a linked list 18.2.3 Methods to represent and handle a hash table Lesson title: Hash tables (Double Lesson)	Students will be able to: 18.2.3 select and use appropriate methods to represent a hash table Maths skill: Calculating hash values, implementing hash algorithms	Activities: - Teacher reminds learners of the linked list ADT (covered in unit 3, topic 14.1.1) and explains that they are typically implement using OOP techniques and so are covered later in the unit. - Teacher reminds learners of the hash table ADT (covered in unit 3, topic 14.1.2) and the theory involved. - Teacher explains how a hash table can be implemented in Python - Practical activity with students writing a program to implement a simple version of a hash table (Student book 2 has an example, Activity 6 on page 256) Homework: Ask students to read the first few pages of the article on Python hash tables from the Real Python site, at least up to the end of the section on hash tables vs dictionaries:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 253–256 (practical implementation) and 103–104 (theory) Web page from the Real Python site covering hash tables and dictionaries: https://realpython.com/python-hash-table/	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 13

Lesson 3

Topic: 18.2.4 18.2.4 Hash algorithm 18.2.5 Methods to handle collisions in a hash table Lesson title: Hash algorithms and collisions (Double Lesson)	Students will be able to: 18.2.3 Select and use appropriate methods to represent a hash table 18.2.4 Select and use an appropriate hash algorithm. 18.2.5 Select and use appropriate methods to handle collisions in a hash table. Maths skill: Calculating hash values, implementing hash algorithms	Activities: - Teacher explains hashing algorithms and how they can be used to create a hash table using string values as a key. - Teacher explains the issue of collisions and the different methods to deal with them (Separate chaining, linear probing and Rehashing) - Practical activity with students writing programs to implement a hashing algorithm using a string and to demonstrate the different methods of handling collisions (Activity 7 on page 260 of Student book 2 provides an example). - Class activity with student presented with code snippets that contain syntax or logic errors, and they have to identify what the errors are and how to correct them. Homework: Ask students to write revision notes covering all the different ADTs covered so far (lists, queue, hash tables), their nature and methods of implementation	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 256–260 (practical implementation) and 105–110 (theory)	Critical thinking Problem solving Analysis	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: 18.2.3 Methods to represent and handle a hash table 18.2.4 Hash algorithm 18.2.5 Methods to handle collisions in a hash table Lesson title: Hash table project (Double Lesson)	Students will be able to: 18.2.3 Select and use appropriate methods to represent a hash table 18.2.4 Select and use an appropriate hash algorithm. 18.2.5 Select and use appropriate methods to handle collisions in a hash table. Maths skill: Calculating hash values, implementing hash algorithms	Activities: - Teacher introduces the lesson and recaps on the previous lessons covering this topic. - Class practical with students provided with an application description that requires the use of a hash table with appropriate hashing algorithm and method of handling collisions. - Students work in pairs reviewing the code they have written for the project. Does it work correctly? Could it be improved? - Class quiz on hash table features. Homework: Ask students to revise class notes form Unit 3 on graph ADTs, topic 14.1.3 and read the theory section in the student book 2, pages 110–122	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 253–260 (practical implementation) and 103–110 (theory)	Critical thinking Problem solving Analysis	Critical thinking Problem solving
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 15 Lesson 1					
Topic: 18.2.6 Methods to represent a graph Lesson title: Graphs	Students will be able to: Select and use appropriate methods to represent a graph	Activities: - Teacher reminds learners of the graph ADT (covered in unit 3, topic 14.1.3) - Teacher explains how a graph can be implemented in Python using an adjacency matrix. (The use of an adjacency list is covered in the next lesson and if preferred these two lessons can be combined into a double lesson on graphs) - Practical activity with students writing a program to implement a graph using an adjacency matrix. - Class activity, ask learner to list the key features and terminology for graph data structures Homework: Ask learners to read the first part of the article on implementing a graph in Python: They do not need to read the section on implementing graphs using classes as OOP has not yet been covered.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 260–262 (practical implementation) and 110–122 (theory) Video which explains graph theory covering both adjacency matrix and list: https://www.youtube.com/watch?v=-VgHk7UMPP4 Web page with a description of how to implement graphs: https://www.w3schools.com/dsa/dsa_data_graphs_implementation.php	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 15 Lesson 2					

Topic: 18.2.6 Methods to represent a graph Lesson title: Graphs – adjacency list	Students will be able to: Select and use appropriate methods to represent a graph	Activities: - Teacher explains an alternative method used to represent a graph using an adjacency list and how this differs from the matrix previously covered - Practical activity with students writing a program to implement a graph using an adjacency list (Activity 8 on page 263 of Student Book 2 provides an example) - Class quiz on features of the ADTs covered in this section (graphs, queues, linked lists, hash tables) Homework: Ask learners to complete a programming challenge involving some of the Python features covered in the last few lessons	Pearson Edexcel International A-level Computer Science Student Book 2: pp 262–263	Critical thinking Problem solving Analysis	Critical thinking Problem solving
--	--	--	---	---	--

Week 15

Lesson 3

Topic: 19.1.1 Methods to produce correct outcomes Lesson title: Correct outcomes	Students will be able to: Select and use appropriate methods to produce correct outcomes to complex problems.	Activities: - Teacher introduces the topic (previously covered in Unit2, topic 9.1.1). Discusses techniques to decompose a problem, identify appropriate logic, identify variables and data structures and select suitable program constructs - Class activity with learners given an application description and asked to design a programmed solution, using a technique such as flowcharts to describe the logic and required constructs and to list required variables and data structures - Working pairs students review each other's attempt at decomposing an application description and consider the strengths and weaknesses of the different outcomes. They should also consider the next steps in developing the application. Homework: Ask students to make a list of the things that they would consider good practice when writing a Python program. The list they produce can be reviewed in the next lesson and compared to the list in the specification and the guidance in the GPG.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 270–272 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 248–251	Critical thinking Problem solving Analysis	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 16

Lesson 1

Topic: 19.1.2 Methods to implement good practice in program design Lesson title: Good practice	Students will be able to: Select and use appropriate methods to design effective code.	Activities: - Teacher introduces the topic (previously covered in Unit 2, topic 9.1.2). Discusses various good practice methods that can be used in program design - Class activity. Divide students into 5 groups and give each group a program design method and ask them to identify programs that they have previously written where they either have demonstrated the use of the method or the program could be improve by implementing the method. Program design methods 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. Homework: Ask learner to read the Python PEP-8 guide and ask them how this compares to the list of good practice guidelines they previously made.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 272–273 Pearson Edexcel International A-level Computer Science Student Book 1: pp. 251–255 Python PEP-8 good practice guide: https://peps.python.org/pep-0008/	Critical thinking Problem solving Analysis	Critical thinking Problem solving
---	---	--	--	---	--

Week 16

Lesson 2

Topic: 19.1.2 Methods to implement good practice in program design Lesson title: AI code generation	Students will be able to: Select and use appropriate methods to design effective code.	Activities: - Teacher introduces the topic and explains how AI generated code can be created and discusses the benefits and drawbacks of AI code. - Class practical activity with students investigating how they can create code with AI (Activity 1 on page 272 provides an example) - Class discussion. How has AI changed the job of a programmer? How has the skills that a programmer needs changed? Homework: Ask students to watch some or all of this video on AI in the development process:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 277 (Activity 2) https://www.youtube.com/watch?v=3VQhdXcQ5qI	Critical thinking Problem solving Analysis	Critical thinking Problem solving
--	---	---	--	---	--

Week 16

Lesson 3

Topic: 19.1.3 Methods to implement modularity in source code Lesson title: Modularity	Students will be able to: Select and use appropriate methods to produce modular code.	Activities: - Teacher introduces the topic (previously covered in Unit 2, topic 9.1.3). Discusses various methods that can be make source code modular. - 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. - Class quiz with questions on python syntax, best practice and ADTs Homework: Ask students to read the article on Python code reuse and modularity.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 273–276 Video explaining how to create user packages: https://www.youtube.com/watch?v=P7wIBTLYyrg Web pages from the Geek for Geeks web site about code reuse and modularity: https://www.geeksforgeeks.org/python/understanding-code-reuse-modularity-python-3/	Critical thinking Problem solving Analysis	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 17					
Lesson 1					
Topic: 19.2.1 Consistency 19.2.2 Readability Lesson title: Consistency and readability	Students will be able to: select and use appropriate methods to promote code consistency. Use appropriate methods to promote readability of program code.	Activities - Teacher introduces the topic (previously covered in Unit 2, topic 9.2.1 and 2). Discusses how to write code which is consistent and readable - Class practical activity. Students work in pairs to review code they have previously written, such as code written for one of the project tasks and consider how they could make the code more consistent and readable (See also Activity 2 on page 277 of Student book 2) - Class practical activity with students given a programming challenge to write Python code which solves a given problem and complies with best practice including modularity, consistency and readability (Activity 3 on page 278 of Student book 2 provides an example) Homework: Ask students to create revision notes or make a classroom poster on best practice to make code consistent and readable	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 276–279	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 17

Lesson 2

Topic: 19.2.3 Methods to control behaviour of program code Lesson title: Control code behaviour	Students will be able to: select and use appropriate methods to control behaviour of code.	Activities: - Teacher introduces the topic (previously covered in Unit 2, topic 9.2.3). Discusses the methods that can be used to control the behaviour of program code. - Class activity. Divide students into 6 groups and give each group a program control method and ask them to identify programs that they have previously written where they either have demonstrated the use of the method or the program could be improved by implementing the method 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. Homework: Ask students to read the article on exception handling in Python	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 280–281 Web page covering exception handling: https://realpython.com/python-exceptions/	Critical thinking Problem solving Analysis	Critical thinking Problem solving
--	---	---	--	---	---

Week 17

Lesson 3

Topic: 19.3.1 Methods to ensure that code is fit for purpose, produces accurate results and meets requirements Lesson title: Testing and debugging (Double Lesson)	Students will be able to: - identify errors in program code - correct errors in program code - produce accurate outcomes - refine solutions to meet requirements	Activities: - Teacher introduces the topic of testing and debugging and explains the different types of errors that can occur. Discussion on appropriate methods to test programs - Teacher demonstrates the use of debugging tools, setting breakpoints and viewing variables. - Class practical activity with programs that contain syntax and logic errors that students need to identify and correct (Activity 4 on page 282 of Student Book 2 provides an example) - Students work in pairs to review some programs they have written previously and decide the best way to test them to ensure they are fully working and robust (for example can they identify invalid user input?) Homework: Ask learners to read the article on debugging Python code	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 281–284 Web page covering debugging Python code: https://realpython.com/debug-python-errors/	Critical thinking Problem solving Analysis	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 WK17					
Week 18					
Lesson 2 & 3					
Topic: 19.1.1 Methods to produce correct outcomes 19.1.2 Methods to implement good practice in program design 19.1.3 Methods to implement modularity in source code 19.2.1 Consistency 19.2.2 Readability 19.2.3 Methods	Students will be able to: Select and use appropriate methods to produce correct outcomes to complex problems Select and use appropriate methods to design effective code. Select and use appropriate methods to produce modular code. Select and use appropriate methods to promote code consistency. Use appropriate methods to promote readability of program code.	Activities: - Teacher introduces the lesson and explains about the project(s) to be completed - Class practical activity with students given one or more application descriptions which they then need to: - Decompose the application description - Create a design, using flowcharts (or other method) for each module applying good practice principles - Write the code to implement the design, applying principles of readability and consistency and using appropriate methods to control the program behaviour. - Debug and test the program to ensure it is fully functional. - Working in pairs students review the designs created and the programs they have written and consider the how well they comply with the principles of good practice and clean code, the strengths and weaknesses of them and possible improvements that could be made.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 270–281	Critical thinking Problem solving Analysis	Critical thinking Problem solving

to control behaviour of program code 19.3.1 Methods to ensure that code is fit for purpose, produces accurate results and meets requirements Lesson title: Best practice project (Triple Lesson)	Select and use appropriate methods to control behaviour of code. Identify errors in program code correct errors in program code produce accurate outcomes refine solutions to meet requirements	Homework: Ask students to research the topic of Object Orientated programming, what is it? How does it differ from the way programs have been written so far?			
--	--	--	--	--	--

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					
Triple lesson continued from WK 18					
Week 19					
Lesson 2					
Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: UML diagrams and OOP theory	Students will be able to: convert Unified Modelling Language (UML) class diagrams into code select appropriate methods for object oriented programming use appropriate methods for object-oriented programming.	Activities: - Teacher introduces the topic of alternative programming paradigms. - Teacher explains the basic principles of OOP and UML class diagrams. Also explains about variable object scope. - Teacher reminds students of UML diagrams, originally covered in Unit 3, topic 13.1.1 - Class practical activity with students creating UML diagrams and writing constructor methods - Class activity with students creating UML diagrams for some given real life and abstract scenarios Homework: Ask students to read the article from the Geek for Geeks site about OOP in Python	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 286–288 Video covering creating classes in Python: https://www.youtube.com/watch?v=ZDa-Z5JzLYM Web page on the Geek for Geeks web site covering OOP in Python: https://www.geeksforgeeks.org/python/python-oops-concepts/	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 19

Lesson 3

Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: Encapsulation	Students will be able to: Convert Unified Modelling Language (UML) class diagrams into code select appropriate methods for object-oriented programming Use appropriate methods for object-oriented programming.	Activities: - Teacher explains the concept of encapsulation in OOP, getter and setter methods and public/private attributes. - Class practical activity with students creating program which demonstrates encapsulation using getter and setter methods (Student book 2 Activity 1 page 290 provides an example) - Class activity with questions about classes, terminology and syntax Homework: Ask students to read the web page article on encapsulation	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 288–290 Video explaining the use of class variables: https://www.youtube.com/watch?v=BJ-VvGyQxho Video explaining encapsulation: https://www.youtube.com/watch?v=GN1LR0UoFI4 Web page describing encapsulation : https://www.datacamp.com/tutorial/encapsulation-in-python-object-oriented-programming	Critical thinking Problem solving Analysis	Critical thinking Problem solving
---	---	---	--	--	--------------------------------------

Topic Sub topic Lesson title	Specification reference	Suggested activities	Suggested resources	Which skills acquired in this lesson are	Which skills could be acquired through teaching
------------------------------------	-------------------------	----------------------	---------------------	--	---

				explicitly assessed through examination?	and delivery in this lesson?
Week 20					
Lesson 1					
Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: Inheritance	Students will be able to: Convert Unified Modelling Language (UML) class diagrams into code Select appropriate methods for object-oriented programming Use appropriate methods for object oriented programming	Activities: - Teacher explains the concept of inheritance in OOP, how this is represented in UML diagrams and in code. - Class practical activity with students creating program which demonstrates inheritance using super, parent and sub classes (Student book 2 Activity 2 page 292 provides an example) - Class activity with students provided with code snippets which contain syntax or logic errors that they have to correct. Homework: Ask students to complete a programming challenge writing a program using OOP principles	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 290–293 Video explaining about class inheritance: https://www.youtube.com/watch?v=RSI87lqOXDE	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 20					
Lesson 2					
Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: Composition	Students will be able to: Convert Unified Modelling Language (UML) class diagrams into code Select appropriate methods for object-oriented programming Use appropriate methods for object-oriented programming.	Activities: - Teacher explains the concept of composition and how it differs from inheritance. - Class practical activity with students creating a program which demonstrates composition. - Class activity, quiz covering OOP concepts and terminology Homework: Ask students to read the web page article on inheritance and composition:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 293–294 Web page on the Geek for Geeks web site covering inheritance and composition: https://www.geeksforgeeks.org/python/inheritance-and-composition-in-	Critical thinking Problem solving Analysis	Critical thinking Problem solving

			python/		
Week 20 Lesson 3					
Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: Aggregation	Students will be able to: convert Unified Modelling Language (UML) class diagrams into code select appropriate methods for object oriented programming use appropriate methods for object-oriented programming.	Activities: - Teacher explains the concept of aggregation and how it differs from composition. - Class practical activity with students creating program which demonstrates aggregation (Student book 2 Activity 3 page 297 provides an example) - A quick fun game of 'aggregation or composition?'. Provide students with various application examples and they have to decide if the relationship between classes is aggregation or composition (See student book 2 page 294 for an example) Homework: Ask students to write revisions notes covering the main OOP concepts of objects, classes, encapsulation and inheritance	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 294–297 Video explaining composition and aggregation: https://www.youtube.com/watch?v=mPFc3JHLnz8	Critical thinking Problem solving Analysis	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 21					
Lesson 1					
Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: Polymorphism	Students will be able to: Convert Unified Modelling Language (UML) class diagrams into code Select appropriate methods for object-oriented programming Use appropriate methods for object-oriented programming.	Activities: - Teacher explains the concept of polymorphism. - Class practical activity with students creating program which demonstrates polymorphism (Student book 2 Activity 4 page 298 provides an example) - Class activity: quiz on OOP Python code syntax Homework: Ask students to revise linked lists by reading the section in the student book (book 2, pages 249–252) in preparation for the next lesson. Also read the theory of linked lists from Unit 3 topic 14.1.1 page 96	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 297–299 Video explaining polymorphism: https://www.youtube.com/watch?v=tHN8I_4Fit8	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 21					
Lesson 2					
Topic: 18.2.2 Methods to represent and handle a linked list Lesson title: Linked lists	Students will be able to: Select and use appropriate methods to represent a linked list Select and use appropriate methods to handle a linked list.	Activities: - Teacher recap on linked lists (theory covered in Unit 3, 14.1.1) and explaining that they are a type of ADT that is typically implemented using OOP (which is why they are covered in this section rather than with the other ADTs) - Class practical activity with students writing a program to implement a linked list using OOP - Class activity with students provided with code snippets which contain syntax or logic errors that they have to correct.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 248–253 Video explaining Python code for linked lists: https://	Critical thinking Problem solving Analysis	Critical thinking Problem solving

		Homework: Ask students to revise hash tables using class notes (weeks 13–14) and reading the section in the student book (book 2, pages 253–260) in preparation for the next lesson	www.youtube.com/watch?v=6sBsF13n5ig		
Week 21 Lesson 3					
Topic: 18.2.3 Methods to represent and handle a hash table: Lesson title: Hash tables using OOP	Students will be able to: Select and use appropriate methods to represent a hash table Select and use appropriate methods to handle a hash table.	Activities: - Teacher recap on hash tables (covered in week 13–14) and explains how they can be implemented using OOP. - Class practical activity creating code which demonstrates hash tables using OOP - Class discussion about the benefits and drawbacks of using OOP to implement a hash table (and other ADTs) compared to a procedural approach used previously Homework: Ask students to revise binary search trees by reading the section in the student book (book 2, pages 263–267 in preparation for the next lesson	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 253–256	Critical thinking Problem solving Analysis	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: 18.2.7 Methods to represent and handle a binary search tree Lesson title: Binary search trees	Students will be able to: Select and use appropriate methods to represent a binary search tree Select and use appropriate methods to handle a binary search tree.	Activities: - Teacher explains how the binary search tree ADT works. - Class practical activity writing code to demonstrate how a binary search tree can be implemented using OOP. - Class activity with students provided with code snippets which contain syntax or logic errors that they have to correct. Homework: Ask students to revise the use of recursion, covered in Unit 1 topic 10.4.1, student book 1 pages 294 - 295	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 263–264	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 22					
Lesson 2					
Topic: 18.2.8 Recursive tree traversal algorithms	Students will be able to: Select and use appropriate methods to represent a binary search tree Select and use appropriate methods to handle a binary search tree.	Activities: - Teacher explains what a recursive tree traversal is and how it can be implemented. - Class practical activity writing code to demonstrate different types of tree traversal using OOP - Class activity with a quiz on ADTs including linked lists, binary trees and hash tables Homework:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 265–267	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Lesson title: Recursive search tree traversal		Ask students to write revision notes describing the main principles of OOP (objects, classes, encapsulation, inheritance etc) including code snippets to demonstrate each principle.			
Week 22 Lesson 3					
Topic: 20.1.1 Methods for object-oriented programming (OOP) Lesson title: OOP project (Double Lesson)	Students will be able to: Convert Unified Modelling Language (UML) class diagrams into code Select appropriate methods for object-oriented programming Use appropriate methods for object-oriented programming.	Activities: - Teacher introduces the lesson, recaps on the topic of OOP and describes the project(s) to be completed. - Class practical activity with students given an application description and creating a program to implement the application using OOP. - Working in pairs ask students to review the programs they have written and consider the strengths and weaknesses of them and possible improvements that could be made. Homework: Ask learners to revise the topic of functional programming, covered in in Unit 3, topic 13.1.2, student book 2 pages 80 – 83	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 286–299	Critical thinking Problem solving Analysis	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					
Double lesson continued from WK 22					
Week 23					
Lesson 2					
Topic: 20.1.2 Methods for functional programming Lesson title: Functional programming introduction	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher explains the basic principles of the functional programming paradigm and describes how a pure function can be implemented. - Class practical activity with learners provided with code that they need to refactor to create pure functions. - Class discussion on programming paradigms, advantages and disadvantage of different paradigms (procedural, OOP, functional), when they their use is most appropriate. Homework: Ask students to read the web page article on pure functions.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 300–301 Video explaining the nature of a pure function: https://www.youtube.com/watch?v=M6k6q88wYCY Web page on the Geek for Geeks web site covering pure functions: https://www.geeksforgeeks.org/c/pure-functions/	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 23

Lesson 3

Topic: 20.1.2 Methods for functional programming Lesson title: 1st class and higher order functions	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher explains what 1st class and higher order functions are and how they can be implemented. - Class practical activity with learners writing code to create 1st class and higher order functions (Student book 2 Activity 5 on page 303 provides an example). - Class activity with students provided with code snippets which contain syntax or logic errors that they must correct. Homework: Ask students to read the web on which has some examples of higher order functions.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 301–303 Video explaining higher order functions: https://www.youtube.com/watch?v=xZtTIm3fpfA Web page on the Geek for Geeks web site covering higher order function with examples: https://www.geeksforgeeks.org/python/higher-order-functions-in-python/	Critical thinking Problem solving Analysis	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: 20.1.2 Methods for functional programming Lesson title: Immutable variables and side effects	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher explains what immutable variable and side effects are. - Class practical activity with learners writing code to demonstrate that variables can be immutable. - Ask students to make a list of 3 things they learnt in this week's lessons and a list of anything they are not sure of or don't understand. Discuss in class, identify any common issues and recap as required Homework: Ask students to do some research into lazy evaluation to find out what is it and how it can be used, in preparation for the next lesson	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 303–304 Video explaining mutable and immutable variables: https://www.youtube.com/watch?v=8EOdCDy_fcs	Critical thinking Problem solving Analysis	Critical thinking Problem solving
Week 24					
Lesson 2					
Topic: 20.1.2 Methods for functional programming Lesson title: Lazy evaluation	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills:	Activities: - Teacher explains what lazy evaluation is and how it can be demonstrated - Class practical activity with learners writing code to demonstrate how lazy evaluation can be implemented - Class discussion on lazy and eager evaluation, the differences between them and advantages/disadvantages of them. Homework:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 304–305 Web page covering lazy evaluation: https://realpython.com/	Critical thinking Problem solving Analysis	Critical thinking Problem solving

	Functional programming	Ask students to read the web page article on Lazy evaluation	python-lazy-evaluation/		
Week 24 Lesson 3					
Topic: 20.1.2 Methods for functional programming Lesson title: Lambda functions	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher explains what a lambda function is and how they can be implemented - Class practical activity with learners writing code to demonstrate how lambda functions can be implemented - Class activity, ask students to write definitions for key functional programming term, such as pure function, 1st class function, immutable variables, side effects, lambda functions, lazy evaluation etc. Ask several students to read out their definition and have the class decide on which is the most appropriate Homework: Ask learners to read at least the first few sections of the web page article on lambda functions.	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 305–306 Video explaining Lambda functions: https://www.youtube.com/watch?v=HQNiSfb795A Web page covering lambda functions: https://realpython.com/python-lambda/	Critical thinking Problem solving Analysis	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: 20.1.2 Methods for functional programming Lesson title: Map and Reduce	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher explains what the map(), filter() and reduce() functions are and how they can be used - Class practical activity with learners writing code to demonstrate the use of the map(), filter() and reduce() functions (Activity 6 on page 307 of Student Book 2 provides an example) - Class activity, a 'backwards' quiz, students are given the answers, and they need to work out what the questions should be Homework: Ask students to read the section in the official Python documentation on map():	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 306–307 Video explaining map(), filter() and reduce(): https://www.youtube.com/watch?v=kj85OY8y8FI Python documentation on the map() function: https://docs.python.org/3/library/functions.html#map	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 25

Lesson 2

Topic: 20.1.2 Methods for functional programming Lesson title: Comprehension and zip	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher explains how comprehension and the zip() can be used - Class practical activity with learners writing code to demonstrate the use of the comprehension and the zip() function. - Class activity – quiz on functional programming features, syntax and code implementation Homework: Ask students to read the official Python documentation on list comprehension:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 307–310 Video explaining list comprehension: https://www.youtube.com/watch?v=YIY2g2xrl6Q Video explaining zip: https://www.youtube.com/watch?v=qj-V2Ep4coY Python documentation on list comprehension : https://docs.python.org/3/tutorial/datastructures.html#list-comprehensions	Critical thinking Problem solving Analysis	Critical thinking Problem solving
---	--	--	--	---	---

Week 25

Lesson 3

Topic: 20.1.2 Methods for functional programming Lesson title: Functional programming project (double lesson)	Students will be able to: Select appropriate methods for functional programming. Use appropriate methods for functional programming. Maths skills: Functional programming	Activities: - Teacher introduces the lesson, recaps on the topic of functional programming and describes the project(s) to be completed. - Class practical activity with students given an application description suitable for implementation using functional programming and they write code to implement the application. - Working in pairs ask students to review the programs they have written and consider the strengths and weaknesses of them and possible improvements that could be made. Homework: Ask students to review class notes on merge sort from Unit 3 topic 15.2.1, and read pages 145–147 in Student book 2	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 300–310	Critical thinking Problem solving Analysis	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 26					
Lesson 1					
Double lesson continued from WK 25					
Week 26					
Lesson 2					
Topic: 21.1.1 Recursive merge sort algorithm Lesson title: Recursive merge sort	Students will be able to: use a recursive merge sort algorithm.	Activities: - Teacher explains how the recursive merge sort works and how it can be implemented - Class practical activity with learners provided with partially complete code for a recursive merge sort that they need to complete - Class activity, learners create a flowchart for the merge sort algorithm Homework: Ask students to review class notes on quick sort from Unit 3 topic 15.2.2, and read page 147–149 in Student book 2	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 145–147 (theory) and 313–315 (practical) Video explaining merge sort: https://www.youtube.com/watch?v=LGiEYu6SkgE	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 26

Lesson 3

Topic: 21.1.2 Recursive quick sort Algorithm Lesson title: Quick sort	Students will be able to: Use a recursive quick sort algorithm.	Activities: - Teacher explains how the quick sort works and how it can be implemented - Class practical activity with learners provided with partially complete code for a quick sort that they need to complete - Class discussion on all the sorting algorithms covered in Unit 2 and 4, identifying key features, benefits and applications of each algorithm. Homework: Ask students to review class notes on Dijkstra's shortest path algorithm from Unit 3 topic 15.2.3, page 149–156 in Student book 2	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 147–149 (theory) and 315–319 (practical) Video explaining the quick sort algorithm: https://www.youtube.com/watch?v=kFeXwkgnQ9U	Critical thinking Problem solving Analysis	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 27					
Lesson 1					
Topic: 21.2.1 Dijkstra's shortest path algorithm Lesson title: Dijkstra's shortest path	Students will be able to: Use the Dijkstra's shortest path algorithm.	Activities: - Teacher explains how Dijkstra's shortest path algorithm works and how it can be implemented - Class practical activity with learners provided with partially complete code implementing the shortest path algorithm that they need to complete (see Activity 3 on page 323 of Student book 2 for an example) - Class activity, a competition style activity with teams of learners rearranging algorithm steps into the correct order Homework: Ask students to read the web page article on Dijkstra's shortest path algorithm:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 149–155 (theory) and 319 – 323 (practical) Video explaining Dijkstra's algorithm: https://www.youtube.com/watch?v=_B5cx-WD5EA+ Web page covering Dijkstra's algorithm: https://www.codecademy.com/article/dijkstras-shortest-path-algorithm	Critical thinking Problem solving Analysis	Critical thinking Problem solving

Week 27

Lesson 2

Topic: 21.2.2 Run-length encoding compression algorithm Lesson title: Run length encoding	Students will be able to: Use a run-length encoding algorithm.	Activities: - Teacher explains how the RLL algorithm works and how it can be implemented - Class practical activity with learners writing code to demonstrate how the RLL algorithm can be implemented - Class quiz on the algorithms covered in this topic Homework: Ask students to read the web page article on run length encoding:	Pearson Edexcel International A-level Computer Science Student Book 2: pp. 323–324 Web page on the Geek for Geeks web site covering run length encoding: https://www.geeksforgeeks.org/python/run-length-encoding-python/	Critical thinking Problem solving Analysis	Critical thinking Problem solving
---	--	--	---	--	--------------------------------------

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	Which skills could be acquired through teaching
------------------------------------	----------------------------	----------------------	------------------------	--	---

				explicitly assessed through examination?	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.			