

INTERNATIONAL GCSE

Computer Science (9-1)

SAMPLE ASSESSMENT MATERIALS

Pearson Edexcel International GCSE in Computer Science (4CP0)

For first teaching September 2017

First examination June 2019

Issue 3



Edexcel, BTEC and LCCI qualifications

Edexcel, BTEC and LCCI qualifications are awarded by Pearson, the UK's largest awarding body offering academic and vocational qualifications that are globally recognised and benchmarked. For further information, please visit our qualifications website at qualifications.pearson.com. Alternatively, you can get in touch with us using the details on our contact us page at qualifications.pearson.com/contactus

About Pearson

Pearson is the world's leading learning company, with 35,000 employees in more than 70 countries working to help people of all ages to make measurable progress in their lives through learning. We put the learner at the centre of everything we do, because wherever learning flourishes, so do people. Find out more about how we can help you and your learners at qualifications.pearson.com

References to third party material made in these sample assessment materials are made in good faith. Pearson does not endorse, approve or accept responsibility for the content of materials, which may be subject to change, or any opinions expressed therein. (Material may include textbooks, journals, magazines and other publications and websites.)

All information in this document is correct at time of publication.

ISBN 978 1 4469 4932 0

All the material in this publication is copyright
© Pearson Education Limited 2019

Summary of Pearson Edexcel International GCSE in Computer Science sample assessment materials Issue 3 changes

Summary of changes made between previous issue and this current issue	Page number
In Unit 2, the sample mark scheme for question 6 has been amended from a levels-based mark scheme to a hybrid mark scheme in order to show specific guidance on how marks should be awarded.	76-77

Earlier issues show previous changes.

If you need further information on these changes or what they mean, contact us via our website at: qualifications.pearson.com/en/support/contact-us.html

Contents

Introduction	1
General marking guidance	3
Paper 1: Principles of Computer Science	5
Paper 1: Mark scheme	27
Paper 2: Application of Computational Thinking	43
Paper 2: Mark scheme	55
Pseudocode reference	81

Introduction

The Pearson Edexcel International GCSE in Computer Science is designed for use in schools and colleges. It is part of a suite of International GCSE qualifications offered by Pearson. These sample assessment materials have been developed to support this qualification and will be used as the benchmark to develop the assessment students will take.

General marking guidance

- All candidates must receive the same treatment. Examiners must mark the last candidate in exactly the same way as they mark the first.
- Mark schemes should be applied positively. Candidates must be rewarded for what they have shown they can do rather than penalised for omissions.
- Examiners should mark according to the mark scheme - not according to their perception of where the grade boundaries may lie.
- All the marks on the mark scheme are designed to be awarded. Examiners should always award full marks if deserved, i.e. if the answer matches the mark scheme.
- Examiners should also be prepared to award zero marks if the candidate's response is not worthy of credit according to the mark scheme.
- Where some judgement is required, mark schemes will provide the principles by which marks will be awarded and exemplification/indicative content will not be exhaustive. However, different examples of responses will be provided at standardisation.
- Where examiners are in doubt regarding the application of the mark scheme to a candidate's response, a senior examiner must be consulted before a mark is given.
- Crossed-out work should be marked **unless** the candidate has replaced it with an alternative response.

Write your name here			
Surname		Other names	
Pearson Edexcel		Centre Number	Candidate Number
Level 1/Level 2		<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>
International GCSE (9–1)			
<h1 style="margin: 0;">Computer Science</h1> <h2 style="margin: 0;">Paper 1: Principles of Computer Science</h2>			
Sample assessment material for first teaching September 2017 Time: 2 hours		Paper Reference 4CP0/01	
You must have: A pseudocode reference			Total Marks

Instructions

- Use **black** ink or ball-point pen.
- **Fill in the boxes** at the top of this page with your name, centre number and candidate number.
- Answer **all** questions.
- Answer the questions in the spaces provided
– *there may be more space than you need.*

Information

- The total mark for this paper is 80.
- The marks for **each** question are shown in brackets
– *use this as a guide as to how much time to spend on each question.*
- You are not allowed to use a calculator.

Advice

- Read each question carefully before you start to answer it.
- Try to answer every question.
- Check your answers if you have time at the end.
- Marks will not be awarded for using product or trade names in answers without giving further explanation.

Turn over ►

S54963A

©2017 Pearson Education Ltd.

1/1



Pearson

Answer ALL questions. Write your answers in the spaces provided.

Some questions must be answered with a cross ☒. If you change your mind about an answer, put a line through the box ☒ and then mark your new answer with a cross ☒.

- 1** Characters, numbers and colours are examples of data that can be represented in a computing device.

(a) Give **one** limitation of the ASCII character set.

(1)

.....

.....

.....

.....

(b) Complete the table to show the results of applying a Caesar cipher.

(3)

Plain text	Shift	Cipher text
WINDOW	+4	
	-3	AOFSB
CACHE		ECEJG

(c) Here is a string of data.

S S S S B W W

Convert the string using a run-length encoding algorithm.

(1)

.....

.....

.....

(d) Identify the 8-bit binary equivalent of the denary number 77.

(1)

- ☐ **A** 01001101
- ☐ **B** 01101100
- ☐ **C** 11000111
- ☐ **D** 10110010

(e) One characteristic of lossy compression is that it makes files smaller.

Give **one** other characteristic of lossy compression.

(1)

(Total for Question 1 = 7 marks)

- 2 Networks are composed of hardware components arranged in topologies. They use protocols and addresses to make sure that data can be exchanged between nodes.

(a) This is an example of an IPv4 address: 192.168.1.253

The expression for the maximum number of IPv4 addresses is $2^8 \times 2^8 \times 2^8 \times 2^8$

An example of an IPv6 address is 2001:2a4c:9d38:6abd:1820:43fc:3f57:febc

Construct an expression for the maximum number of IPv6 addresses.

(2)

(b) An IPv6 address can be represented in hexadecimal and binary.

(i) Identify the binary equivalent of the hexadecimal number C4.

(1)

- ☐ A 11000011
- ☐ B 11010100
- ☐ C 11000100
- ☐ D 11010011

(ii) Identify the result of a logical shift left by 2 when applied to 00011011.

(1)

- ☐ A 01101111
- ☐ B 00011100
- ☐ C 00101100
- ☐ D 01101100

(c) A ring is a type of network topology.

Give **one** reason why there are no collisions on a ring network.

(1)

(d) A simplified TCP/IP model can be represented using four layers.

(i) Three of the layers are Data Link, Transport and Network.

State the order of these layers from the highest to the lowest.

(1)

1

2

3

(ii) A web browser sends a request to a web server.

State the name of the TCP/IP layer that first handles this request.

(1)

.....

.....

(Total for Question 2 = 7 marks)

3 A team of researchers are studying urban wildlife, such as foxes, mice, and birds. The team collect data during the day and night. Sometimes, they work in an office.

(a) The team:

- use a range of hardware, including laptops, tablets, and smartphones
- use different operating systems and applications
- communicate with each other using smartphones or tablets
- store and share data, including images, audio recordings and videos
- work collaboratively on research documents.

Identify the secondary storage medium most suitable for the team and justify why it best meets their needs.

Write your answer on the next page. You do not need to use all of the space.

(6)

Answer to question 3(a)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

- (b) The team communicate using their smartphones.

Identify the type of network they use when making calls via the cellular/mobile phone network.

(1)

- ☐ **A** WAN
- ☐ **B** LAN
- ☐ **C** PAN
- ☐ **D** VPN

- (c) Give **two** drawbacks of using a cellular/mobile phone network such as 3G, 4G, or 5G.

(2)

1

2

- (d) The team analyse the data when they are in the office. They have to log on to the office network using a username and password to access the data.

State the type of network usage model they use.

(1)

.....

.....

- (e) The team use a software application and the data they have collected to create a **model** of urban wildlife in order to understand more about the environment wildlife live in.

Examples of data collected:

- number and type of animals seen each hour
- number of newborn animals
- daily weather conditions
- type and amount of food available.

Describe **one** way the team could use the model to help them to research the urban wildlife environment.

(2)

(Total for Question 3 = 12 marks)

- 4 In a computing device, the software is executed by the hardware.

Figure 1 is a diagram of the inside of a computer.

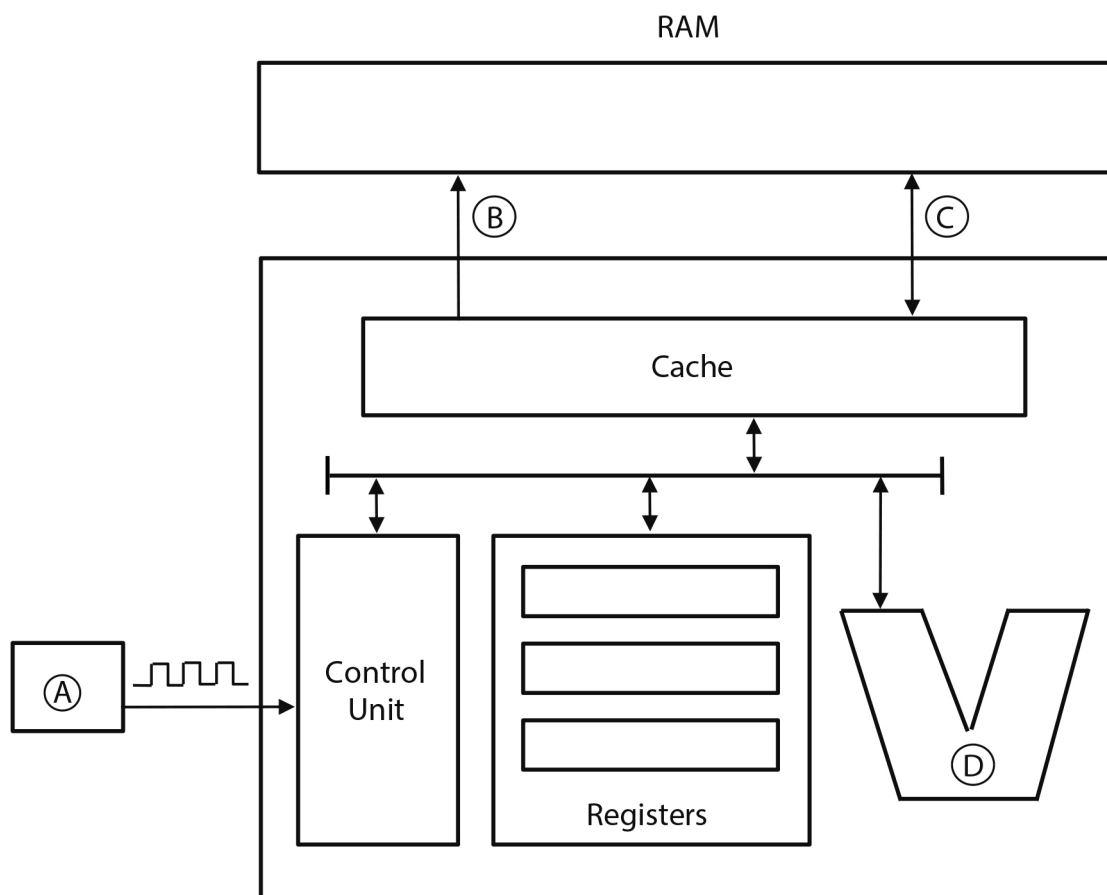


Figure 1

- (a) Use the information in Figure 1 to answer these questions.

- (i) Name the items labelled A, B, C, and D.

(4)

A

B

C

D

(ii) Describe **one** function of the cache.

(2)

(b) Tom has just bought a computer. He opened an executable email attachment from someone he did not know. Since then, all of his email contacts have reported receiving hoax emails from his email address.

State **two** actions that Tom can take to resolve this problem using utility software.

(2)

1

2

(c) The Internet of Things (IoT) is a networking development where everyday objects, such as cars and shipping crates, can send and receive data.

(i) Identify the name for this type of system.

(1)

- ☐ **A** Decomposed
- ☐ **B** Agent-based
- ☐ **C** Embedded
- ☐ **D** Optimised

- (ii) Golf balls can now be part of the IoT. One manufacturer uses an RFID (radio-frequency identification) tag for this purpose.

Describe how the functionality of these new golf balls could be used.

(2)

(Total for Question 4 = 11 marks)

- 5 Some research has shown that the use of technology promotes a sense of belonging. This type of research is usually supported by the use of online questionnaires.

- (a) Describe **one** way in which the use of technology makes us a more inclusive society.

(2)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

- (b) All the 15-year-old students in a school have completed questionnaires about the number of computing devices they can access.

Data is coded **S** for smartphone, **T** for tablet and **L** for laptop.

- (i) Complete the truth table to show the result of each expression.

(2)

S	T	NOT S	NOT S OR T
0	0		
0	1		
1	0		
1	1		

- (ii) Construct a Boolean logic statement to show that a student has a tablet or a smartphone, as well as a laptop.

(1)

- (iii) The questionnaire asks for the number of smartphones a student can access.

One student enters -38 (negative) in error.

Convert -38 to two's complement. Use 8-bit binary. Show your working.

(3)

(Total for Question 5 = 8 marks)

6 Music, photographs and videos are some of the most popular items stored on computing devices.

(a) Photographs are made up of tiny blocks of colour.

Identify the name given to each block of colour.

(1)

- ☐ **A** Resolution
- ☐ **B** Pixel
- ☐ **C** RGB
- ☐ **D** Bitmap

(b) An image has a colour depth of 2 bits.

Describe how a colour depth of 2 bits limits the number of colours that can be represented.

(2)

(c) A 4 mebibytes audio recording is stored on a hard disc.

Construct an expression to show how to convert 4 mebibytes to bits.

(3)

(d) An analogue sound is never fully reproducible in a digital format.

Explain why this statement is true.

(4)

.....

.....

.....

.....

.....

(Total for Question 6 = 10 marks)

- 7 Algorithms can be designed using pseudocode or flowcharts. Then, they need to be translated into code that a computing device can execute.

Figure 2 shows the pseudocode for an algorithm.

```

1  # This is the pseudocode for an algorithm
2  SET inNum TO 0
3  SET result TO 1
4  SET i TO 0
5
6  SEND "Enter a number: " TO DISPLAY
7  RECEIVE inNum FROM (INTEGER) KEYBOARD
8
9  IF (inNum < 0) THEN
10     SEND "Invalid input" TO DISPLAY
11 ELSE
12     IF (inNum = 0) THEN
13         SEND "Answer is 1" TO DISPLAY
14     ELSE
15         FOR i FROM 1 TO inNum DO
16             SET result TO result * i
17         END FOR
18         SEND "The answer is " & result TO DISPLAY
19     ENDIF
20 ENDIF

```

Figure 2

- (a) Use the information in Figure 2 to answer these questions.

- (i) Complete the table to show the output for the given input.

(3)

Input	Output message
0	
-12	
5	

- (ii) State the purpose of this algorithm.

(1)

.....

.....

.....

.....

(b) A bus company sets fares for different groups of passengers.

The fares are:

- a child fare for passengers 15 years old and younger
- a senior fare for passengers 65 years old and older
- a full fare for all other passengers.

Construct a flowchart of an algorithm that will determine the fare for one passenger when an age is input.

No validation of input is required.

(5)

- (c) Complete the table, indicating with a (✓), to show the characteristics of program language translators.

You may select more than one translator per characteristic.

(3)

Characteristic	Translators		
	Compiler	Interpreter	Assembler
An error in the source code is highlighted as soon as it is encountered during execution.			
Translates low-level programming languages.			
Translates high-level programming languages.			
Generates a single executable file.			
One line of source code is translated to one line of machine code.			

(Total for Question 7 = 12 marks)

8 Security is a big concern when using computers. It is also an important consideration as new applications are developed.

(a) Identify the software which protects against unauthorised access to a networked computer.

(1)

- ☐ **A** HTTPS
- ☐ **B** Antivirus
- ☐ **C** Anti-spyware
- ☐ **D** Firewall

(b) Explain why a person attempting to gain unauthorised access to a networked computer can sometimes benefit the owner of the network.

(2)

- (c) Users are forced to change their passwords every 28 days. This requires an algorithm that reports the days in any given month.

The algorithm must report the number of days in a month based on a number entered (e.g. 1 = January, 2 = February etc.).

This pseudocode algorithm does not produce accurate results. These are the test results.

Input	Expected behaviour	Actual behaviour
2	The month is February and it has 28 days.	The month is March and it has 31 days.
13	The month number 13 is not valid.	Potential runtime error: index out of range.
-4	The month number -4 is not valid.	Potential runtime error: index out of range.

Figure 3 shows the errors are on lines 12, 13, and 14.

```

1
2 SET monthNames TO ["January", "February", "March", "April",
3                     "May", "June", "July", "August", "September",
4                     "October", "November", "December"]
5
6 SET monthDays TO [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]
7
8 SEND "Enter the month number. 0 to exit." TO DISPLAY
9 RECEIVE number FROM (INTEGER) KEYBOARD
10
11 WHILE NOT (number = 0) DO
12     IF (number > 1) OR (number < 12) THEN
13         SET month TO monthNames[number]
14         SET days TO monthDays[number]
15
16         SEND "The month is " & month & " and it has " & days & "days."
17     ELSE
18         SEND "The month number: " & number & " is invalid."
19     ENDIF
20
21     SEND "Enter the month number. 0 to exit." TO DISPLAY
22     RECEIVE number FROM (INTEGER) KEYBOARD
23
24 END WHILE

```

Figure 3

Write the corrected replacement codes for lines 12, 13, and 14.

(4)

Line 12

.....

Line 13

.....

Line 14

.....

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

- (d) A group of programmers is developing a software application that will be used over a network. It is very important that the software application has no security vulnerabilities.

Discuss how the programmers could minimise security vulnerabilities during development.

(6)

(Total for Question 8 = 13 marks)

TOTAL FOR PAPER = 80 MARKS

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

REA

Paper 1: Principles of Computer Science

Question number	Answer	Additional guidance	Mark
1(a)	Any one from: <ul style="list-style-type: none"> non-English characters can't be represented 7/8 bits can't represent all characters not enough bits to represent all characters. 		(1)

Question number	Answer	Additional guidance	Mark	
1(b)				
	Plain text	Shift		Cipher text
	WINDOW	+4		AMRHSA (1)
	DRIVE (1)	-3		AOFSB
	CACHE	+2 (1)		ECEJG

Question number	Answer	Additional guidance	Mark
1(c)	4S 1B 2W	Accept S4 B1 W2	(1)

Question number	Answer	Additional guidance	Mark
1(d)	A 0 1 0 0 1 1 0 1		(1)

Question number	Answer	Additional guidance	Mark
1(e)	Any one from: <ul style="list-style-type: none"> data is permanently removed original cannot be reconstructed. 		(1)

Question number	Answer	Additional guidance	Mark
2(a)	<p>1 mark for extending the given pattern in the question. 1 mark for the correct number of patterns.</p> $2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8 \times 2^8$ <p>1 mark for 2^{16}. 1 mark for correct number of patterns.</p> $2^{16} \times 2^{16} \times 2^{16} \times 2^{16} \times 2^{16} \times 2^{16} \times 2^{16} \times 2^{16}$ <p>1 mark for each correct exponent.</p> $(2^8)^{16}$ $(2^{16})^8$ 2^{128}		(2)

Question number	Answer	Additional guidance	Mark
2(b)(i)	C 1 1 0 0 0 1 0 0		(1)

Question number	Answer	Additional guidance	Mark
2(b)(ii)	D 0 1 1 0 1 1 0 0		(1)

Question number	Answer	Additional guidance	Mark
2(c)	<p>Any one from:</p> <ul style="list-style-type: none"> • data packets only travel in a single direction from node to node (so there are no packet collisions) • only one machine at a time can use the network (because it uses a token-passing mechanism) 		(1)

Question number	Answer	Additional guidance	Mark
2(d)(i)	<p>1 mark for the correct order (highest to the lowest).</p> <p>1 Transport 2 Network 3 Data Link</p>		(1)

Question number	Answer	Additional guidance	Mark
2(d)(ii)	Application		(1)

Question number	Answer	Mark
3(a)	<p>Indicative content</p> <p>Cloud storage is most suitable.</p> <p>Accessibility:</p> <ul style="list-style-type: none"> • it is accessible whenever and wherever an internet connection is available • the team's devices probably support mobile data, then they will always have access to their documents • can work with data locally (offline) and upload/re-synchronise documents. <p>Collaboration:</p> <ul style="list-style-type: none"> • the team can work on documents at the same time, which is good for multiple field workers entering/reading data at the same time. <p>File compatibility:</p> <ul style="list-style-type: none"> • applications can be used online instead of having native ones on each device • that way, there is no conversion of file types necessary. <p>Online apps:</p> <ul style="list-style-type: none"> • the online applications can be used to edit online stored data files • there may be cost savings by using online applications instead of buying individual ones for each of the devices. <p>Backups:</p> <ul style="list-style-type: none"> • documents on the cloud will be backed up without the user having to think about it. 	(6)

Level	Mark	Descriptor
	0	No rewardable content.
Level 1	1–2	Basic, independent points are made, showing elements of knowledge and understanding of key concepts/principles of computer science. The discussion will contain information with little linkage between points made.
Level 2	3–4	Demonstrates adequate knowledge and understanding of key concepts/principles of computer science. The discussion shows some linkages and lines of reasoning, with some structure.
Level 3	5–6	Demonstrates comprehensive knowledge and understanding by selecting relevant knowledge and understanding of key concepts/principles of computer science to support the discussion being presented. The discussion shows a well-developed, sustained line of reasoning that is clear, coherent and logically structured.

Question number	Answer	Additional guidance	Mark
3(b)	A WAN		(1)

Question number	Answer	Additional Guidance	Mark
3(c)	Any two from: <ul style="list-style-type: none"> • some areas not covered by required infrastructure • inconsistent connection due to interference • speed of uploading/downloading large files; may be slower than other networks • the cost of uploading/downloading large files may be prohibitive • there is a significant difference in upload and download speeds • there is some concern about the health risks of heavy use of wireless technologies (magnetic waves). 		(2)

Question number	Answer	Additional guidance	Mark
3(d)	Client server (model)		(1)

Question number	Answer	Additional guidance	Mark
3(e)	<p>Any one from:</p> <ul style="list-style-type: none"> the team could use the model to predict/track (1) numbers of wildlife over time (1) the team could research the impact of changing variables (such as the amount of food available) (1) on the numbers of wildlife (1) the team could predict the impact (1) of the increase/decrease of one type of wildlife on other types (1) the team could look for trends (1) in the numbers/types of wildlife (1) the team could investigate the possible impact of changing variables (1) such as amount of foods/temperatures/rainfall (1). 	Accept prediction, analysis of historical data to show trends, the effects of one variable compared to another.	(2)

Question number	Answer	Additional guidance	Mark
4(a)(i)	<p>A = Clock (1) B = Address Bus (1) C = Data Bus (1) D = Arithmetic Logic Unit/ALU (1)</p>		(4)

Question number	Answer	Additional guidance	Mark
4(a)(ii)	Any one from: <ul style="list-style-type: none"> the cache queues instructions/holds data ready for use (1) so that it speeds up the processing (1) the cache speeds up the processing/speed matching (1) by making up for the difference in speed of the memory on the chip and the RAM memory (1). 	Do not accept web cache, hard drive cache.	(2)

Question number	Answer	Additional guidance	Mark
4(b)	Any two from: <ul style="list-style-type: none"> antivirus software should be installed antivirus software should be run/his disc should be scanned to identify the virus a virus removal tool could be used (to remove or quarantine the virus from his machine) antivirus software/signature files should be kept up to date (to make sure any new malware programs are detected). 		(2)

Question number	Answer	Additional guidance	Mark
4(c)(i)	C Embedded		(1)

Question number	Answer	Additional guidance	Mark
4(c)(ii)	Any one from: <ul style="list-style-type: none"> receivers can be placed at intervals down the golf course (1). When the golf ball passes one of the receivers, the receiver can register the distance (1) a hand-held receiver could be carried by a golfer so that if he/she lost a golf ball (1), they could use the receiver to detect the location (1). 		(2)

Question number	Answer	Additional guidance	Mark
5(a)	<p>A description that incorporates a concept, such as:</p> <ul style="list-style-type: none"> the use of technology gives access to a wider range of individuals (local and globally) (1) outside our normal social circumstances (1) the use of technology gives access to individuals with common interests (1) to promote a feeling of belonging (1) the use of technologies provides access to services (e.g. education, government, health, charities) (1) for those with constraints (e.g. geographical time, monetary constraints, disabilities) (1) the use of accessibility/adaptive technologies for disabilities (1) allows people to participate more fully in society (1) (e.g. haptic devices (phone vibrates), communication devices, locked-in syndrome) the use of technologies as alternative infrastructures (1), may provide important services in areas without more conventional infrastructure (1). 	Examples of infrastructure may include service infrastructures and alternatives, e.g. mobile money, non-smartphones, works by SMS messaging, e.g. M-Pesa or physical telecommunication infrastructures, e.g. VOIP (socket stack, smartphone)	(2)

Question number	Answer	Additional guidance	Mark																				
5(b)(i)	<p>1 mark for each correct column.</p> <table border="1"> <thead> <tr> <th>S</th><th>T</th><th>NOT S</th><th>NOT S OR T</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>1</td><td>1</td></tr> <tr> <td>0</td><td>1</td><td>1</td><td>1</td></tr> <tr> <td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr> <td>1</td><td>1</td><td>0</td><td>1</td></tr> </tbody> </table>	S	T	NOT S	NOT S OR T	0	0	1	1	0	1	1	1	1	0	0	0	1	1	0	1		(2)
S	T	NOT S	NOT S OR T																				
0	0	1	1																				
0	1	1	1																				
1	0	0	0																				
1	1	0	1																				

Question number	Answer	Additional guidance	Mark
5(b)(ii)	Any one from: <ul style="list-style-type: none"> (T AND L) OR (S AND L) L AND (T OR S) T OR S AND L 		(1)

Question number	Answer	Additional guidance	Mark
5(b)(iii)	38 To bin: 0010 0110 (1) Flip bits: 1101 1001 (1) Add 1: 0000 0001 -38: 1101 1010 (1) Alternative solution (Subtraction from 2^n where $n=8$ bits) Formula: $2^n - 38$ Substitution: $2^8 - 38$ (1) Calculation: $256 - 38 = 218$ (1) To binary: 1101 1010 (1)	Correct answer only gains 3 marks.	(3)

Question number	Answer	Additional guidance	Mark
6(a)	B Pixel		(1)

Question number	Answer	Additional guidance	Mark
6(b)	There are only four different representations (that can be displayed as 2 bits) (1), which are 00, 01, 10, 11 (1)		(2)

Question number	Answer	Additional guidance	Mark																					
6(c)	<p>Award 1 mark for each stage of the process.</p> <p>(4 mebibytes) $\left(\frac{1024 \text{ kibibytes}}{1 \text{ mebibyte}}\right) \left(\frac{1024 \text{ bytes}}{1 \text{ kibibyte}}\right) \left(\frac{8 \text{ bits}}{\text{byte}}\right)$</p> <table><tr><td></td><td></td><td>(1) ×</td><td></td><td>(1) ×</td><td></td><td>(1) ×</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>4</td><td></td><td>1024</td><td></td><td>1024</td><td></td><td>8</td></tr></table>			(1) ×		(1) ×		(1) ×								4		1024		1024		8	<ul style="list-style-type: none">Ignore attempts to calculate.Units not required.Equivalent expressions awarded.No marks awarded for 4 mebibytes as this is given in the paper.	(3)
		(1) ×		(1) ×		(1) ×																		
4		1024		1024		8																		

Question number	Answer	Additional guidance	Mark
6(d)	<p>An explanation that incorporates the following concepts (to a maximum of 4 marks total):</p> <ul style="list-style-type: none"> the analogue sound is a continuous wave (1) to convert it, the amplitude (1) of the signal is sampled (at different points) (1) the sampled value is converted to binary digits (1) therefore, some parts of the analogue signal will not be represented by samples (in the file) (1). 		(4)

Question number	Answer	Additional guidance	Mark
7(a)(i)	1 mark for each correct output.		

Question number	Answer	Additional guidance	Mark
7(a)(ii)	<p>Any one from:</p> <ul style="list-style-type: none"> multiplies every number between 1 and inNum. calculates factorial of inNum. 		(1)

Question number	Answer	Additional guidance	Mark
7(b)	<p>Indicative content</p> <pre> graph TD Start([Start]) --> EnterAge[Enter age] EnterAge --> D1{age <= 15?} D1 -- Yes --> DisplayChild[Display "Child"] D1 -- No --> D2{age >= 65?} D2 -- Yes --> DisplaySenior[Display "Senior"] D2 -- No --> DisplayFull[Display "Full"] DisplayChild --> Stop([Stop]) DisplaySenior --> Stop DisplayFull --> Stop </pre> <p>Award 1 mark for:</p> <ul style="list-style-type: none"> • start and input of age (1) • decision with correct logic and output for child passenger (1) • decision with correct logic and output for senior passenger (1) • correct logic and output for all other passengers and Stop (1) • correct usage of symbols (1). 	<p>Accept this symbol as input/output</p>	(5)

Question number	Answer	Additional guidance	Mark																												
7(c)	<p>1 mark for each two correct responses up to a maximum of 3 marks.</p> <table> <tr> <th></th><th colspan="3">Translators</th></tr> <tr> <th>Characteristic</th><th>Compiler</th><th>Interpreter</th><th>Assembler</th></tr> <tr> <td>An error in the source code is highlighted as soon as it is encountered during execution.</td><td></td><td>✓</td><td></td></tr> <tr> <td>Translates low-level programming languages.</td><td></td><td></td><td>✓</td></tr> <tr> <td>Translates high-level programming languages.</td><td>✓</td><td>✓</td><td></td></tr> <tr> <td>Generates a single executable file.</td><td>✓</td><td></td><td></td></tr> <tr> <td>One line of source code is translated to one line of machine code.</td><td></td><td></td><td>✓</td></tr> </table>		Translators			Characteristic	Compiler	Interpreter	Assembler	An error in the source code is highlighted as soon as it is encountered during execution.		✓		Translates low-level programming languages.			✓	Translates high-level programming languages.	✓	✓		Generates a single executable file.	✓			One line of source code is translated to one line of machine code.			✓		(3)
	Translators																														
Characteristic	Compiler	Interpreter	Assembler																												
An error in the source code is highlighted as soon as it is encountered during execution.		✓																													
Translates low-level programming languages.			✓																												
Translates high-level programming languages.	✓	✓																													
Generates a single executable file.	✓																														
One line of source code is translated to one line of machine code.			✓																												

Question number	Answer	Additional guidance	Mark
8(a)	D Firewall		(1)

Question number	Answer	Additional guidance	Mark
8(b)	<p>An explanation that makes reference to any one from:</p> <ul style="list-style-type: none"> security vulnerabilities in the network are exposed (1) by someone who is paid/rewarded to do it/ethical hacker (1) identify problems/flaws with the network (1) so that vulnerability can be addressed (1) security vulnerabilities are exposed and addressed (1) before being made public (1). 		(2)

Question number	Answer	Additional guidance	Mark
8(c)	<p>(number >= 1) (1)</p> <p>Boolean operator AND (1)</p> <p>(number <= 12) (1)</p> <p>Both instances of [number -1] (1)</p> <pre> 12 IF (number >= 1) AND (number <= 12) THEN 13 SET month TO monthNames[number - 1] 14 SET days TO monthDays[number - 1] 15 </pre>		(4)

Question number	Answer	Additional guidance	Mark
8(d)	<p>Indicative content</p> <p>Development cycle:</p> <ul style="list-style-type: none"> • security should be considered throughout the development cycle • the requirement for security can be agreed as one of the initial requirements • security should be factored into the design of the application. <p>Programming languages:</p> <ul style="list-style-type: none"> • programming languages, just like other software, may have security loopholes in them • as the application is being developed, the programmers should make sure that any unsecure parts of the language are not used • when they know of language changes, they must incorporate them as quickly as possible • use appropriate settings on language translators. <p>Programming habits:</p> <ul style="list-style-type: none"> • bad programming habits by the programmers themselves can lead to problems • one way around this is special review sessions that focus only on security • bad programming practices need to be addressed and amended to keep the application secure • the simpler the code structure is, the fewer issues that will be exposed once the application is released (Cyclomatic complexity). <p>Tracking:</p> <ul style="list-style-type: none"> • when the application is being developed, it is important to be able to track who made what changes and when • these audit trails should be kept as the code is changed • it means that if a security issue arises, it can be tracked back to the version of the code that first had it. <p>Testing:</p> <ul style="list-style-type: none"> • modular testing is a way to help keep code secure • the smaller the units of testing, the more likely a security issue is to be found • test for boundary conditions and handle exceptions appropriately • use all available code analysis tools. 		(6)

Level	Mark	Descriptor
	0	No rewardable content.
Level 1	1–2	Basic, independent points are made, showing elements of knowledge and understanding of key concepts/principles of computer science. The discussion will contain information with little linkage between points made.
Level 2	3–4	Demonstrates adequate knowledge and understanding of key concepts/principles of computer science. The discussion shows some linkages and lines of reasoning, with some structure.
Level 3	5–6	Demonstrates comprehensive knowledge and understanding by selecting relevant knowledge and understanding of key concepts/principles of computer science to support the discussion being presented. The discussion shows a well-developed, sustained line of reasoning that is clear, coherent and logically structured.

Write your name here			
Surname		Other names	
Pearson Edexcel		Centre Number	Candidate Number
Level 1/Level 2		<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>
International GCSE (9–1)			
<h1 style="margin: 0;">Computer Science</h1> <h2 style="margin: 0;">Paper 2: Application of Computational Thinking</h2>			
Sample assessment material for first teaching September 2017 Time: 3 hours		Paper Reference 4CP0/02	
You must have: A computer workstation with appropriate programming language code editing software and tools, including a code interpreter/compiler, CODES folder containing code and data files, pseudocode reference			Total Marks

Instructions

- Use **black** ink or ball-point pen.
- **Fill in the boxes** at the top of this page with your name, centre number and candidate number.
- Answer **all** questions.
- Answer the questions **requiring a written answer** in the spaces provided – *there may be more space than you need.*
- Only **one** programming language must be used throughout the test.
- Carry out practical tasks on the computer system and save new or amended code using the name given with the appropriate file extension.
- Do **not** overwrite the original code and data files provided to you.
- You must **not** use the internet during the test.

Information

- The total mark for this paper is 80.
- The marks for **each** question are shown in brackets
- This paper covers Python, C# and Java.
- The CODES folder in your user area includes all the code and data files you need.
- The invigilator will tell you where to store your work.

Advice

- Read each question carefully before you start to answer it.
- Save your work regularly.
- Check your answers and work if you have time at the end.

Turn over ►

S54964A

©2017 Pearson Education Ltd.

1/1/1



Answer ALL questions. Write your answers in the spaces provided.

1 Programmers use accepted programming constructs when writing code.

(a) Open the file **Q01a** in the code editor.

Answer these questions about the code.

(i) Identify the name given to a **data structure** in the code.

(1)

(ii) Identify the line number(s) showing **repetition**.

(1)

(iii) Identify the line number(s) showing **selection**.

(1)

(iv) Identify the name of a **variable**.

(1)

(b) Open the file **Q01b** in the code editor.

Answer these questions about the code.

(i) Identify **one** technique that could be used to make the code in **Q01b** more readable.

(1)

(ii) State **one** reason why code should be readable.

(1)

(c) (i) Give a definition of a syntax error.

(1)

(ii) Open the file **Q01c** in the code editor.

Amend the code to correct three program errors.

Save your amended code as **Q01cFINISHED** with the correct file extension for the programming language.

(3)

(Total for Question 1 = 10 marks)

2 A football club uses computer applications.

(a) The club collects this data about visitors:

- country of origin
- number of adults and children in each party.

This pseudocode contains the logic required to do this.

```
6  # Print prompt and take country from user
7  SEND "Enter the country you're visiting from: " TO DISPLAY
8  RECEIVE country FROM (STRING) KEYBOARD
9
10 # Tell the user their country
11 SEND ("You are from: " & country) TO DISPLAY
12
13 # Take number of adults in party
14 SEND "Enter the number of adults in your party: " TO DISPLAY
15 RECEIVE adults FROM (INTEGER) KEYBOARD
16
17 # Take number of children in party
18 SEND "Enter the number of children in your party: " TO DISPLAY
19 RECEIVE children FROM (INTEGER) KEYBOARD
20
21 # Calculate total number in party
22 SET total TO adults + children
23
24 # Tell user the total
25 SEND ("The total in your party is: " & total)
26
```

Write a program to implement the logic in the pseudocode.

Do not add any further functionality.

You **must** use the structure given in file **Q02a** to write the program.

Save your amended code as **Q02aFINISHED** with the correct file extension for the programming language.

(10)

- (b) Each seat in the football stadium has a unique identifier. The identifier is made up of two letters and two numbers, separated by a hyphen.

AB-45, NP-98, ab-46, nP-90 are valid seat identifiers.

Complete the table to show **two** additional validation tests and invalid data.

(4)

Validation test	Invalid data
Is hyphen included?	AB^12

- (c) The football club makes money selling tickets, food and other items. Attendance at matches is also monitored.

At the end of each week, the data collected is processed.

Attendance is considered along with income (money coming in).

Open the file **Q02c** in the code editor.

Amend the code to complete the 'If statement' used to produce the outputs described in the table. Do not add any further functionality.

Save your amended code as **Q02cFINISHED** with the correct file extension for the programming language.

Condition	Output message
Attendance is at least 1500	Sufficient profit made this week
Income is at least 45000	Sufficient profit made this week
Attendance is at least 750; income is at least 22500	Income in line with attendance this week
Attendance is fewer than 500	Attendance is very low this week Contact fan club
All other inputs	Possible accounting error

(4)

(Total for Question 2 = 18 marks)

3 A holiday company has a website. They would like to publish the daily temperatures in their most popular destinations.

(a) Describe **one** benefit of using subprograms.

(2)

(b) The holiday company needs to be able to convert temperatures between Celsius and Fahrenheit.

Open the file **Q03b** in the code editor.

Answer these questions about the code.

(i) Identify the name of a **built-in** subprogram in the code.

(1)

(ii) Identify the name of a **user-defined** subprogram.

(1)

(iii) Identify the name of **one** input parameter used in a subprogram.

(1)

(iv) Identify the name of a subprogram that does **not** use input parameters.

(1)

(v) Identify the name of a **local** variable.

(1)

(vi) Identify the name of a **global** variable.

(1)

(vii) One line in the code is identified as **not** working as expected.

State why this line does **not** work as expected.

(1)

- (c) The holiday company needs to assign numbers to their most popular destinations.

Cities.txt	Numbered.txt
London	1 London
Hong Kong	2 Hong Kong
Delhi	3 Delhi
Istanbul	4 Istanbul
Tokyo	5 Tokyo
Mumbai	6 Mumbai
Mexico City	7 Mexico City
New York City	8 New York City
Rio de Janeiro	9 Rio de Janeiro
Singapore	10 Singapore

The files **Q03c** and **Cities.txt** are provided.

Open the code named **Q03c** in the code editor.

Write a program to implement these requirements.

For all lines in the **Cities.txt** file, the code must:

- read the line
- append a line number and a space to the front
- write the new line to a **Numbered.txt** file
- print the line to the display

You must use the structure given in file **Q03c** to complete the program.

Do not add further functionality.

Save your amended code as **Q03cFINISHED** with the correct file extension for the programming language.

(7)

(Total for Question 3 = 16 marks)

4 A list of numbers is to be sorted using a bubble sort algorithm.

(a) Give a definition of the term 'algorithm'.

(1)

(b) Here is a list of numbers that need to be sorted in **ascending** order.

28	7	26	21	34	18	16	9
----	---	----	----	----	----	----	---

Perform the first pass of a bubble sort.

Use this space for working to help you answer the questions.

(i) Complete the table to show how the list will have changed at the end of the first pass.

(1)

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

(ii) State the number of comparisons made in the first pass.

(1)

(iii) State the number of swaps made in the first pass.

(1)

(c) A bubble sort is only one type of sorting algorithm.

(i) Give **one** reason why a bubble sort is inefficient when sorting a large dataset. (1)

(ii) State the position in a list that will always remain unchanged after the first pass of any ascending order bubble sort. (1)

(Total for Question 4 = 6 marks)

5 Data, stored as numbers, is very easily processed using computer algorithms.

(a) Open the file **Q05a** in the code editor.

Complete the trace table to show the execution of the code.

You may not need to fill in all the rows in the table.

(5)

target	rs	rm	r

(b) Open the file named **Q05b** in the code editor.

Write a program to determine a discount based on an amount entered.

The program should:

- allow the user to input total spend
- display the output message based on the total spend entered.

Total spend	Text output message
More than 300	Discount is 10%
More than 0	No discount
All other input	Invalid input

No validation of input is required.

Save your amended code as **Q05bFINISHED** with the correct file extension for the programming language.

(5)

(Total for Question 5 = 10 marks)

6 Open the file named **Q06** in the code editor.

In file **Q06**, the names and years of birth of artists are stored in a 2-dimensional data structure.

Labels for their work need to be created by joining the first letter of their last name, the first letter of their first name and their year of birth.

For example, a label for ('Andy', 'Warhol', 1928) would be 'WA1928'.

Write a program to:

- process each artist to create a label
- store all the labels in the data structure named 'theLabels'
- display the labels for all the artists
- find and display the name and year of birth of the youngest artist.

Your program should function correctly, even if 'theArtists' data structure has more, fewer, or different artists.

You **must** use the data structures in file **Q06**.

Save your amended code as **Q06FINISHED** with the correct file extension for the programming language.

(Total for Question 6 = 20 marks)

TOTAL FOR PAPER = 80 MARKS

Paper 2: Application of Computational Thinking

Question number	Answer	Additional guidance	Mark
1(a)(i)	myNumbers	Candidates are required to open the file Q01a in the code editor.	(1)

Question number	Answer	Additional guidance	Mark
1(a)(ii)	Python Lines 3–8/3 Java Lines 7–14/7 C# Lines 10–17/10	Candidates are required to open the file Q01a in the code editor.	(1)

Question number	Answer	Additional guidance	Mark
1(a)(iii)	Python Lines 5–8/5 Java Lines 10–14/10 C# Lines 13–16/13	Candidates are required to open the file Q01a in the code editor.	(1)

Question number	Answer	Additional guidance	Mark
1(a)(iv)	myNumbers/i/total	Candidates are required to open the file Q01a in the code editor.	(1)

Question number	Answer	Additional guidance	Mark
1(b)(i)	Any one from: <ul style="list-style-type: none"> • comments/annotations • descriptive variable names. 	Candidates are required to open the file Q01b in the code editor.	(1)

Question number	Answer	Additional guidance	Mark
1(b)(ii)	Any one from: <ul style="list-style-type: none"> • another programmer could understand it • future maintenance would be easier • easier for another programmer to fix bugs/make amendments • less likely to introduce bugs yourself. 		(1)

Question number	Answer	Additional guidance	Mark
1(c)(i)	Any one from: <ul style="list-style-type: none"> • an error in following the rules of the programming language • not following the grammar rules of the programming language • not being able to translate a line of code because of an error in using the language • misspelling command words in the programming language. 		(1)

Question number	Answer	Additional guidance	Mark
1(c)(ii)	<p>Python</p> <ul style="list-style-type: none"> Total is undefined/need to add initialisation for variable total (1). Equals symbol in If statement needs to be replaced with '==' (1). Print ("Odd") needs to be indented (1). <pre> 1 myNumbers = [10, 20, 30, 40 ,50, 60, 70, 80, 90, 100] 2 total = 0 3 for theNumber in myNumbers: 4 total = total + theNumber 5 if(theNumber % 2 == 0): 6 print("Even") 7 else: 8 print("Odd") 9 print(total) </pre>	<p>Candidates are required to open the file Q01c in the code editor. Amended code should be saved as Q01cFINISHED.</p> <p>Do not penalise logic errors such as initialising total inside loop.</p>	(3)

Java

- Total is undefined/need to add initialisation for variable total (1).
- Equals symbol in If statement needs to be replaced with '==' (1).
- Missing closed } after Print ("Odd") (1).

```
5      int total = 0;
6      for (int i=10;i<=100;i+=10)
7      {
8          total = total + i;
9          if(i % 2 == 0)
10             System.out.println("Even");
11         else
12         {
13             System.out.println("Odd");
14         }
15     }
16     System.out.println(total);
```


	<p>C#</p> <ul style="list-style-type: none"> • Total is undefined/need to add initialisation for variable total (1). • Equals symbol in If statement needs to be replaced with '==' (1). • Missing End If after Console.WriteLine ("Odd") (1). <pre> int total = 0; for (int i = 10; i <= 100; i += 10) { total = total + i; if (i % 2 == 0) Console.WriteLine("Even"); else { Console.WriteLine("Odd"); } } Console.WriteLine(total); Console.ReadKey(); </pre>		
--	---	--	--

Question number	Answer	Additional guidance	Mark
2(a)	<p>Award 1 mark for each of:</p> <ul style="list-style-type: none"> attempting to input country and print country (1) printing string plus country (1) attempting to input number of children and number of adults (1) coercion of at least one data type (1) attempting to calculate and print a total (1) calculating a total using the addition operator (1) using two variables (1) printing a string plus an integer (1) compiling without syntax errors (1) executing and producing the correct output (1). <p>Python</p> <pre> 5 # Print prompt and take country from user 6 country = input ("Enter the country you're visiting from: ") 7 8 # Tell the user their country 9 print ("You are from: ", country) 10 11 # Take number of adults in party from user 12 adult = int (input ("Enter the number of adults in your party: ")) 13 14 # Take number of children in party from user 15 children = int (input ("Enter the number of children in your party: ")) 16 17 # Calculate total number in party 18 total = adult + children 19 20 # Tell the user the total number of people in their party 21 print ("The total in your party is: ", total) </pre>	<p>Candidates are required to open the file Q02a in the code editor. Amended code should be saved as Q02aFINISHED.</p> <p>Logic of algorithm must be followed as set out. Alternatives must address each point.</p> <p>Do not penalise candidates who attempt more than the stated requirements.</p>	(10)

Java

```
6 Scanner input = new Scanner(System.in);
7
8 // Print prompt and take country from user
9 System.out.print("Enter the country you are visiting from: ");
10 String country = input.next();
11
12 // Tell the user their country
13 System.out.println("You are from: " + country);
14
15 // Take number of adults in party from user
16 System.out.print("Enter the number of adults in your party: ");
17 int adults = input.nextInt();
18
19 // Take number of children in party from user
20 System.out.print("Enter the number of children in your party: ");
21 int children = input.nextInt();
22
23 // Calculate total number in party
24 int total = adults + children;
25
26 // Tell the user the total number of people in their party
27 System.out.println("The total in your party is: " + total);
28
```

C#

```
8 // Print prompt and take country from user
9 Console.WriteLine("Enter the country you are visiting from: ");
10 String country = Console.ReadLine();
11 Console.WriteLine("You are from: " + country);
12
13 // Tell the user their country
14 Console.WriteLine("Enter the number of adults in your party: ");
15
16 // Take number of adults in party from user
17 int adults = Convert.ToInt32(Console.ReadLine());
18 Console.WriteLine("Enter the number of children in your party: ");
19
20 // Take number of children in party from user
21 int children = Convert.ToInt32(Console.ReadLine());
22
23 // Calculate total number in party
24 int total = adults + children;
25
26 // Tell the user the total number of people in their party
27 Console.WriteLine("The total in your party is: " + total);
28
```

Question number	Answer	Additional guidance	Mark										
2(b)	<p>For up to two tests</p> <p>Award 1 mark for an appropriate validation test and 1 mark for an item of test data that would fail the given test.</p> <table><tr><th>Validation test</th><th>Invalid data</th></tr><tr><td>Is length = 5?</td><td>ABC-123</td></tr><tr><td>Is hyphen in the middle?</td><td>AB1-2</td></tr><tr><td>Are characters 1 and 2 letters?</td><td>12-34</td></tr><tr><td>Are characters 4 and 5 digits?</td><td>AB-MP</td></tr></table>	Validation test	Invalid data	Is length = 5?	ABC-123	Is hyphen in the middle?	AB1-2	Are characters 1 and 2 letters?	12-34	Are characters 4 and 5 digits?	AB-MP		(4)
Validation test	Invalid data												
Is length = 5?	ABC-123												
Is hyphen in the middle?	AB1-2												
Are characters 1 and 2 letters?	12-34												
Are characters 4 and 5 digits?	AB-MP												

Question number	Answer	Additional guidance	Mark																	
2(c)	<p>Award 1 mark for each logical test.</p> <table><thead><tr><th>Condition</th><th>Output message</th><th>Logical test</th></tr></thead><tbody><tr><td>Attendance is at least 1500</td><td>Sufficient profit made this week</td><td rowspan="2">(attendance >= 1500) or (income >= 45000)</td></tr><tr><td>Income is at least 45000</td><td>Sufficient profit made this week</td></tr><tr><td>Attendance is at least 750; income is at least 22500</td><td>Income in line with attendance this week</td><td>(attendance >= 750) and (income >= 22500)</td></tr><tr><td>Attendance is fewer than 500</td><td>Attendance is very low this week Contact fan club</td><td>(attendance < 500)</td></tr><tr><td>All other inputs</td><td>Possible accounting error</td><td>The print statement needs to be in the 'else' block</td></tr></tbody></table> <p>Python</p> <pre>if (attendance >= 1500) or (income >= 45000): print ("Sufficient profit made this week") elif (attendance >= 750) and (income >= 22500): print ("income in line with attendance this week") elif (attendance < 500): print ("Attendance is very low this week. Contact the fan club.") else: print ("Possible accounting error.")</pre> <p>Java</p> <pre>if ((attendance >= 1500) (income >= 45000)) System.out.println("Sufficient profit made this week"); else if ((attendance >= 750) && (income >= 22500)) System.out.println("income in line with attendance this week"); else if (attendance < 500) System.out.println("Attendance is very low this week. Contact the fan club."); else System.out.println("Possible accounting error.");</pre>	Condition	Output message	Logical test	Attendance is at least 1500	Sufficient profit made this week	(attendance >= 1500) or (income >= 45000)	Income is at least 45000	Sufficient profit made this week	Attendance is at least 750; income is at least 22500	Income in line with attendance this week	(attendance >= 750) and (income >= 22500)	Attendance is fewer than 500	Attendance is very low this week Contact fan club	(attendance < 500)	All other inputs	Possible accounting error	The print statement needs to be in the 'else' block	<p>Candidates are required to open the file Q02c in the code editor. Amended code should be saved as Q02cFINISHED.</p> <p>Do not penalise candidates who attempt more than the stated requirements.</p>	(4)
Condition	Output message	Logical test																		
Attendance is at least 1500	Sufficient profit made this week	(attendance >= 1500) or (income >= 45000)																		
Income is at least 45000	Sufficient profit made this week																			
Attendance is at least 750; income is at least 22500	Income in line with attendance this week	(attendance >= 750) and (income >= 22500)																		
Attendance is fewer than 500	Attendance is very low this week Contact fan club	(attendance < 500)																		
All other inputs	Possible accounting error	The print statement needs to be in the 'else' block																		

	C# <pre> if ((attendance >= 1500) (income >= 45000)) Console.WriteLine("Sufficient profit made this week"); else if ((attendance >= 750) && (income >= 22500)) Console.WriteLine("income in line with attendance this week"); else if (attendance < 500) Console.WriteLine("Attendance is very low this week. Contact the fan club."); else Console.WriteLine("Possible accounting error."); </pre>	
--	---	--

Question number	Answer	Additional guidance	Mark
3(a)	Any one from: <ul style="list-style-type: none"> a subprogram can be written once (1) and called many times (1) a subprogram can be debugged once (1) and called many times (1) subprograms can be collected into libraries (1), which can be used by other programs (1). 		(2)

Question number	Answer	Additional guidance	Mark
3(b)(i)	Python time.sleep/print Java Thread.sleep/System.out.println C# System.Threading.Thread.Sleep/Console.WriteLine	Candidates are required to open the file Q03b in the code editor. Accept clear reference to sleep and print built-in subprograms.	(1)

Question number	Answer	Additional guidance	Mark
3(b)(ii)	toCelsius/toFahrenheit/waitTenSeconds/waitTime	<p>Candidates are required to open the file Q03b in the code editor.</p> <p>These are the same across all languages.</p> <p>Accept clear reference to subprogram name.</p>	(1)

Question number	Answer	Additional guidance	Mark
3(b)(iii)	<p>Python inTemp/inSeconds</p> <p>Java and C# inTemp/inMilliseconds</p>	<p>Candidates are required to open the file Q03b in the code editor.</p> <p>Accept clear reference to parameter name.</p>	(1)

Question number	Answer	Additional guidance	Mark
3(b)(iv)	waitTenSeconds	<p>Candidates are required to open the file Q03b in the code editor.</p> <p>This is the same across all languages.</p> <p>Accept clear reference to subprogram name.</p>	(1)

Question number	Answer	Additional guidance	Mark
3(b)(v)	Python inTemp/celsius/fahrenheit/inSeconds Java and C# inTemp/celsius/fahrenheit/inMilliseconds	Candidates are required to open the file Q03b in the code editor. Accept clear reference to variable name.	(1)

Question number	Answer	Additional guidance	Mark
3(b)(vi)	theDate	Candidates are required to open the file Q03b in the code editor. This is the same across all languages. Accept clear reference to variable name.	(1)

Question number	Answer	Additional guidance	Mark
3(b)(vii)	The subprogram (being called on this line) is missing a return statement.	Candidates are required to open the file Q03b in the code editor.	(1)

Question number	Answer	Do not accept	Additional guidance	Mark
3(c)	<p>Award one mark for each of:</p> <ul style="list-style-type: none"> opening 'Cities.txt' for reading only (1) opening/creating 'Numbered.txt' for writing only (1) using for-each/while not EOF to read each line into variable 'theLine' (1) incrementing the count of lines read (1) constructing the output string/coercion as required (1) writing new lines to the output file (1) closing at least one of the text files (1). <p>Python</p> <pre> 5 # Open "Cities.txt" as input 6 theFile = open ("Cities.txt", "r") 7 8 # Open "Numbered.txt" as output 9 outFile = open ("Numbered.txt", "w") 10 11 # Use a for/each loop to read each line of 12 # the input file into a variable named 'theLine' 13 for theLine in theFile: 14 15 # Increment the line count 16 count = count + 1 17 18 # Add the line number to the front of the line followed by a space 19 theLine = str(count) + " " + theLine 20 21 # print the line to the display 22 print (theLine) 23 24 # Write the new line to the output file 25 outFile.writelines (theLine) 26 27 # Close the input file 28 theFile.close() 29 30 # Close the output file 31 outFile.close() </pre>	Data structures such as an array or a list.	Candidates are required to open the file Q03c in the code editor. Amended code should be saved as Q03cFINISHED.	(7)

Java

```
20 // Open "Cities.txt" as input
21 theFile = new Scanner(new BufferedReader(new FileReader("cities.txt")));
22 // Open "Numbered.txt" as output
23 outFile = new PrintWriter("Numbered.txt", "UTF-8");
24 // Use loop to read each line of
25 // the input file into a variable named 'theLine'
26 while (theFile.hasNextLine())
27 {
28     theLine = theFile.nextLine();
29     // Increment the line count variable
30     count++;
31     // Add the line number to the front of the line
32     // followed by a space
33     theLine = (Integer.toString(count) + " " + theLine);
34     // print the line to the display
35     System.out.println(theLine);
36     // Write the new line to the output file
37     outFile.println(theLine);
38 }
39
40 // Close the input file
41 theFile.close();
42 // Close the output file
43 outFile.close();
```

C#

```
13 // open Cities.txt as input
14 System.IO.StreamReader fileReader = new System.IO.StreamReader("Cities.txt");
15 // open Numbered.txt as output
16 System.IO.StreamWriter fileWriter = new System.IO.StreamWriter("Numbered.txt");
17 // use loop to read each line of the input file into a variable named theLine
18 while (fileReader.Peek() >= 0)
19 {
20     theLine = fileReader.ReadLine();
21     // increment the line count variable
22     count = count + 1;
23     // add the line number to the front of the line followed by a space
24     theLine = Convert.ToString(count) + " " + theLine;
25     // print the line to the display
26     Console.WriteLine(theLine);
27     // write new line to the output file
28     fileWriter.WriteLine(theLine);
29 }
30 // close the input file
31 fileReader.Close();
32 // close the output file
33 fileWriter.Close();
34 Console.ReadKey();
```

Question number	Answer	Additional guidance	Mark
4(a)	Any one from: <ul style="list-style-type: none"> a step-by-step procedure (which if followed precisely with a given input produces a predictable output) a list of instructions followed in sequence (to solve a problem) a process or set of rules to be followed (to achieve a predictable result). 		(1)

Question number	Answer								Additional guidance	Mark
4(b)(i)										(1)
	7	26	21	28	18	16	9	34		

Question number	Answer	Additional guidance	Mark
4(b)(ii)	7		(1)

Question number	Answer	Additional guidance	Mark
4(b)(iii)	6		(1)

Question number	Answer	Additional guidance	Mark
4(c)(i)	Any one from: <ul style="list-style-type: none"> requires many passes to complete the sort requires many comparisons/every number is compared every single pass. 		(1)

Question number	Answer	Additional guidance	Mark
4(c)(ii)	Top/highest/right-most/last		(1)

Question number	Answer	Additional guidance	Mark																																																									
5(a)	Award 1 mark for each correct pass of the loop.	Candidates are required to open the file Q05a in the code editor. Penalise each mathematical error once and then follow through.																																																										
	<table><tr><th>target</th><th>rs</th><th>rm</th><th>r</th></tr><tr><td>4</td><td>0</td><td>0</td><td>1</td></tr><tr><td>4</td><td>1</td><td>0</td><td>1</td></tr><tr><td>4</td><td>1</td><td>1</td><td>1</td></tr><tr><td>4</td><td>1</td><td>1</td><td>2</td></tr><tr><td>4</td><td>4</td><td>1</td><td>2</td></tr><tr><td>4</td><td>4</td><td>2</td><td>2</td></tr><tr><td>4</td><td>4</td><td>2</td><td>3</td></tr><tr><td>4</td><td>9</td><td>2</td><td>3</td></tr><tr><td>4</td><td>9</td><td>3</td><td>3</td></tr><tr><td>4</td><td>9</td><td>3</td><td>4</td></tr><tr><td>4</td><td>16</td><td>3</td><td>4</td></tr><tr><td>4</td><td>16</td><td>0</td><td>4</td></tr><tr><td>4</td><td>16</td><td>0</td><td>5</td></tr></table>			target	rs	rm	r	4	0	0	1	4	1	0	1	4	1	1	1	4	1	1	2	4	4	1	2	4	4	2	2	4	4	2	3	4	9	2	3	4	9	3	3	4	9	3	4	4	16	3	4	4	16	0	4	4	16	0	5	(1)
	target			rs	rm	r																																																						
	4			0	0	1																																																						
	4			1	0	1																																																						
	4			1	1	1																																																						
	4			1	1	2																																																						
	4			4	1	2																																																						
	4			4	2	2																																																						
	4			4	2	3																																																						
	4			9	2	3																																																						
	4			9	3	3																																																						
	4			9	3	4																																																						
	4			16	3	4																																																						
	4			16	0	4																																																						
	4			16	0	5																																																						
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
	(1)																																																											
</																																																												

Question number	Answer	Additional guidance	Mark
5(b)	<ul style="list-style-type: none"> Accept user input of total spend (1) Coercion of input (to a numerical data type) (1) Correct logic for totalSpend greater than 300 leads to printing correct output message (1) Correct logic for totalSpend greater than 0 leads to printing correct output message (1) Correct logic for all other input leads to printing correct output message. (1) 	Candidates are required to open the file Q05b in the code editor. Amended code should be saved as Q05bFINISHED.	(5)
	<p>Python</p> <pre> 1 # Write your code below this line 2 totalSpend = int (input ("What is your total spend?")) 3 if (totalSpend > 300): 4 print ("Discount is 10%") 5 elif (totalSpend > 0): 6 print ("No discount") 7 else: 8 print ("Invalid input") </pre>		

Java

```
1 package q05b;
2 import java.util.Scanner;
3 // Write your code here
4 public class Q05b
5 {
6     public static void main(String[] args)
7     {
8         Scanner input = new Scanner(System.in);
9         int totalSpend = 0;
10
11         System.out.println ("What is your total spend?");
12         totalSpend = input.nextInt();
13
14         if (totalSpend > 300)
15             System.out.println ("Discount is 10%");
16         else if (totalSpend > 0)
17             System.out.println ("No discount");
18         else
19             System.out.println ("Invalid input");
20     }
21 }
```


C#

```
1  using System;
2  // Write your code here
3
4  namespace Q05b_cs
5  {
6      class Program
7      {
8          static void Main(string[] args)
9          {
10             int totalSpend = 0;
11
12             Console.WriteLine("What is your total spend?");
13             totalSpend = Convert.ToInt32(Console.ReadLine());
14
15             if (totalSpend > 300)
16                 Console.WriteLine("Discount is 10%");
17             else if (totalSpend > 0)
18                 Console.WriteLine("No discount");
19             else
20                 Console.WriteLine("Invalid input");
21         }
22     }
23 }
24
```

Question number	Answer	Additional guidance	Mark
6	<p>Award one mark for each of the following points up to a maximum of 11 marks:</p> <ul style="list-style-type: none"> • Initialise 'make artist label' loop (1) • Initialise 'find youngest person' loop to cycle through all artists given (1) • Identify / extract initials of artist (1) • Combine/concatenate initials with year of birth to create label (1) • Add the label to the theLabels data structure (1) • Display the label for the artist (1) • Initialise the year of birth variable (for finding youngest artist e.g. maxDate) or initialise oldest person variable (to hold oldest person e.g. maxPerson) (1) • Check date of birth with year of birth variable (1) • Assign younger year to maxDate if necessary (1) • Assign older year to maxPerson if necessary (1) • Display the name and year of birth of artist identified (1) 		(11)

Award up to a maximum of nine marks using the levels-based mark scheme below.				
Band 0	Band 1 (1-3 marks)	Band 2 (4-6 marks)	Band 3 (7-9 marks)	Mark
No rewardable content	Little attempt to decompose the problem into component parts	Some attempt to decompose the problem into component parts	The problem has been decomposed into component parts	(9)
	Some parts of the logic are clear and appropriate to the problem	Most parts of the logic are clear and mostly appropriate to the problem	The logic is clear and appropriate to the problem	
	Some appropriate use and manipulation of data types, variables, data structures and program constructs	The use and manipulation of data types, variables and data structures and program constructs is mostly appropriate	The use and manipulation of data types, variables and data structures and program constructs is appropriate	
	Parts of the code are clear and readable	Code is mostly clear and readable	Code is clear and readable	
	Finished program will not be flexible enough with other data sets or input	Finished program will function with some but not all other data sets or input	Finished program could be used with other data sets or input	
	The program meets some of the given requirements	The program meets most of the given requirements	The program fully meets the given requirements	

Example solutions

Python

```
# Make the artist labels
for person in theArtists:
    newRecord = person[1][0] + person[0][0] + str(person[2])
    theLabels.append (newRecord)
print ("The new userIDs are: ", theLabels)

# Find and print the youngest person and their birthdate
maxDate = 0
for person in theArtists:
    if person[2] > maxDate:
        maxDate = person[2]
        maxPerson = person
print (maxPerson[0], maxPerson[1], "is youngest", str(maxPerson[2]))
```

Java

```
33 // Make the artist labels
34 for (int i = 0; i < theArtists.length; i++) {
35     String newRecord = String.valueOf(theArtists[i][1].charAt(0))
36         + String.valueOf(theArtists[i][0].charAt(0))
37         + theArtists[i][2];
38     theLabels.add (newRecord);
39     System.out.println(newRecord);
40     System.out.println("The new userIDs are: " + theLabels.get(theLabels.size() - 1));
41 }
42 // Find and print the youngest person and their birthdate
43 int maxDate = 0;
44 String maxPerson = "";
45 for (String[] person : theArtists) {
46     if (Integer.parseInt(person[2]) > maxDate) {
47         maxDate = Integer.parseInt(person[2]);
48         maxPerson = person[0] + " " + person[1];
49     }
50 }
51 System.out.println(maxPerson + " is youngest " + maxDate);
```

C#

```
for (int i = 0; i < theLabels.Length; i++)
{
    String newRecord = Convert.ToString(theArtists[i,1][0]) +
        Convert.ToString(theArtists[i,0][0]) +
        theArtists[i,2];
    theLabels[i] = newRecord;
    Console.WriteLine(newRecord);
    Console.WriteLine("The new userIDs are: " + theLabels[i]);
}
// Find and print the youngest person and their birthdate
int maxDate = 0;
String maxPerson = "";
for (int i = 0; i < theLabels.Length; i++)
{
    if (Convert.ToInt32(theArtists[i,2]) > maxDate)
    {
        maxDate = Convert.ToInt32(theArtists[i,2]);
        maxPerson = theArtists[i,0] + " " + theArtists[i,1];
    }
}
Console.WriteLine(maxPerson + " is youngest " + maxDate);
Console.ReadKey();
```


Pseudocode reference

Questions in the written examination that involve code will use this pseudocode for clarity and consistency. However, students may answer questions using any valid method.

Data types

INTEGER
REAL
BOOLEAN
CHARACTER

Type coercion

Type coercion is automatic if indicated by context. For example $3 + 8.25 = 11.25$ (integer + real = real)

Mixed mode arithmetic is coerced like this:

	INTEGER	REAL
INTEGER	INTEGER	REAL
REAL	REAL	REAL

Coercion can be made explicit. For example, RECEIVE age FROM (INTEGER) KEYBOARD assumes that the input from the keyboard is interpreted as an INTEGER, not a STRING.

Constants

The value of constants can only ever be set once. They are identified by the keyword CONST. Two examples of using a constant are shown.

```
CONST REAL PI
SET PI TO 3.14159
SET circumference TO radius * PI * 2
```

Data structures

ARRAY
STRING

Indices start at zero (0) for all data structures.

All data structures have an append operator, indicated by &.

Using & with a STRING and a non-STRING will coerce to STRING. For example, SEND 'Fred' & age TO DISPLAY, will display a single STRING of 'Fred18'.

Identifiers

Identifiers are sequences of letters, digits and `'_'`, starting with a letter, for example: `MyValue`, `myValue`, `My_Value`, `Counter2`

Functions

`LENGTH()`

For data structures consisting of an array or string.

`RANDOM(n)`

This generates a random number from 0 to n.

Comments

Comments are indicated by the `#` symbol, followed by any text.

A comment can be on a line by itself or at the end of a line.

Devices

Use of `KEYBOARD` and `DISPLAY` are suitable for input and output.

Additional devices may be required, but their function will be obvious from the context.

For example, `CARD_READER` and `MOTOR` are two such devices.

Notes

In the following pseudocode, the `< >` indicates where expressions or values need to be supplied. The `< >` symbols are not part of the pseudocode.

Variables and arrays		
Syntax	Explanation of syntax	Example
SET Variable TO <value>	Assigns a value to a variable.	SET Counter TO 0 SET MyString TO 'Hello world'
SET Variable TO <expression>	Computes the value of an expression and assigns to a variable.	SET Sum TO Score + 10 SET Size to LENGTH(Word)
SET Array[index] TO <value>	Assigns a value to an element of a one-dimensional array.	SET ArrayClass[1] TO 'Ann' SET ArrayMarks[3] TO 56
SET Array TO [<value>, ...]	Initialises a one-dimensional array with a set of values.	SET ArrayValues TO [1, 2, 3, 4, 5]
SET Array [RowIndex, ColumnIndex] TO <value>	Assigns a value to an element of a two-dimensional array.	SET ArrayClassMarks[2,4] TO 92

Selection		
Syntax	Explanation of syntax	Example
IF <expression> THEN <command> END IF	If <expression> is true then command is executed.	IF Answer = 10 THEN SET Score TO Score + 1 END IF
IF <expression> THEN <command> ELSE <command> END IF	If <expression> is true then first <command> is executed, otherwise second <command> is executed.	IF Answer = 'correct' THEN SEND 'Well done' TO DISPLAY ELSE SEND 'Try again' TO DISPLAY END IF

Repetition		
Syntax	Explanation of syntax	Example
<pre>WHILE <condition> DO <command> END WHILE</pre>	Pre-conditioned loop. Executes <command> whilst <condition> is true.	<pre>WHILE Flag = 0 DO SEND 'All well' TO DISPLAY END WHILE</pre>
<pre>REPEAT <command> UNTIL <expression></pre>	Post-conditioned loop. Executes <command> until <condition> is true. The loop must execute at least once.	<pre>REPEAT SET Go TO Go + 1 UNTIL Go = 10</pre>
<pre>REPEAT <expression> TIMES <command> END REPEAT</pre>	Count controlled loop. The number of times <command> is executed is determined by the expression.	<pre>REPEAT 100-Number TIMES SEND '*' TO DISPLAY END REPEAT</pre>
<pre>FOR <id> FROM <expression> TO <expression> DO <command> END FOR</pre>	Count controlled loop. Executes <command> a fixed number of times.	<pre>FOR Index FROM 1 TO 10 DO SEND ArrayNumbers[Index] TO DISPLAY END FOR</pre>
<pre>FOR <id> FROM <expression> TO <expression> STEP <expression> DO <command> END FOR</pre>	Count controlled loop using a step.	<pre>FOR Index FROM 1 TO 500 STEP 25 DO SEND Index TO DISPLAY END FOR</pre>
<pre>FOR EACH <id> FROM <expression> DO <command> END FOREACH</pre>	Count controlled loop. Executes for each element of an array.	<pre>SET WordsArray TO ['The', 'Sky', 'is', 'grey'] SET Sentence to "" FOR EACH Word FROM WordsUArray DO SET Sentence TO Sentence & Word & '' END FOREACH</pre>

Input/output		
Syntax	Explanation of syntax	Example
SEND <expression> TO DISPLAY	Sends output to the screen.	SEND 'Have a good day.' TO DISPLAY
RECEIVE <identifier> FROM (type) <device>	Reads input of specified type.	RECEIVE Name FROM (STRING) KEYBOARD RECEIVE LengthOfJourney FROM (INTEGER) CARD_READER RECEIVE YesNo FROM (CHARACTER) CARD_READER

File handling		
Syntax	Explanation of syntax	Example
READ <File> <record>	Reads in a record from a <file> and assigns to a <variable>. Each READ statement reads a record from the file.	READ MyFile.doc Record
WRITE <File> <record>	Writes a record to a file. Each WRITE statement writes a record to the file.	WRITE MyFile.doc Answer1, Answer2, 'xyz 01'

Subprograms		
Syntax	Explanation of syntax	Example
PROCEDURE <id> (<parameter>, ...) BEGIN PROCEDURE <command> END PROCEDURE	Defines a procedure.	PROCEDURE CalculateAverage (Mark1, Mark2, Mark3) BEGIN PROCEDURE SET Avg to (Mark1 + Mark2 + Mark3)/3 END PROCEDURE
FUNCTION <id> (<parameter>, ...) BEGIN FUNCTION <command> RETURN <expression> END FUNCTION	Defines a function.	FUNCTION AddMarks (Mark1, Mark2, Mark3) BEGIN FUNCTION SET Total to (Mark1 + Mark2 + Mark3)/3 RETURN Total END FUNCTION

Subprograms		
Syntax	Explanation of syntax	Example
<id> (<parameter>, ...)	Calls a procedure or a function.	Add (FirstMark, SecondMark)

Arithmetic operators	
Symbol	Description
+	Add
-	Subtract
/	Divide
*	Multiply
^	Exponent
MOD	Modulo
DIV	Integer division

Relational operators	
Symbol	Description
=	equal to
<>	not equal to
>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to

Logical operators	
Symbol	Description
AND	Returns true if both conditions are true.
OR	Returns true if any of the conditions are true.
NOT	Reverses the outcome of the expression; true becomes false, false becomes true.

For information about Edexcel, BTEC or LCCI qualifications
visit qualifications.pearson.com

Edexcel is a registered trademark of Pearson Education Limited

Pearson Education Limited. Registered in England and Wales No. 872828
Registered Office: 80 Strand, London WC2R 0RL
VAT Reg No GB 278 537121

Getty Images: Alex Belmonlinsky