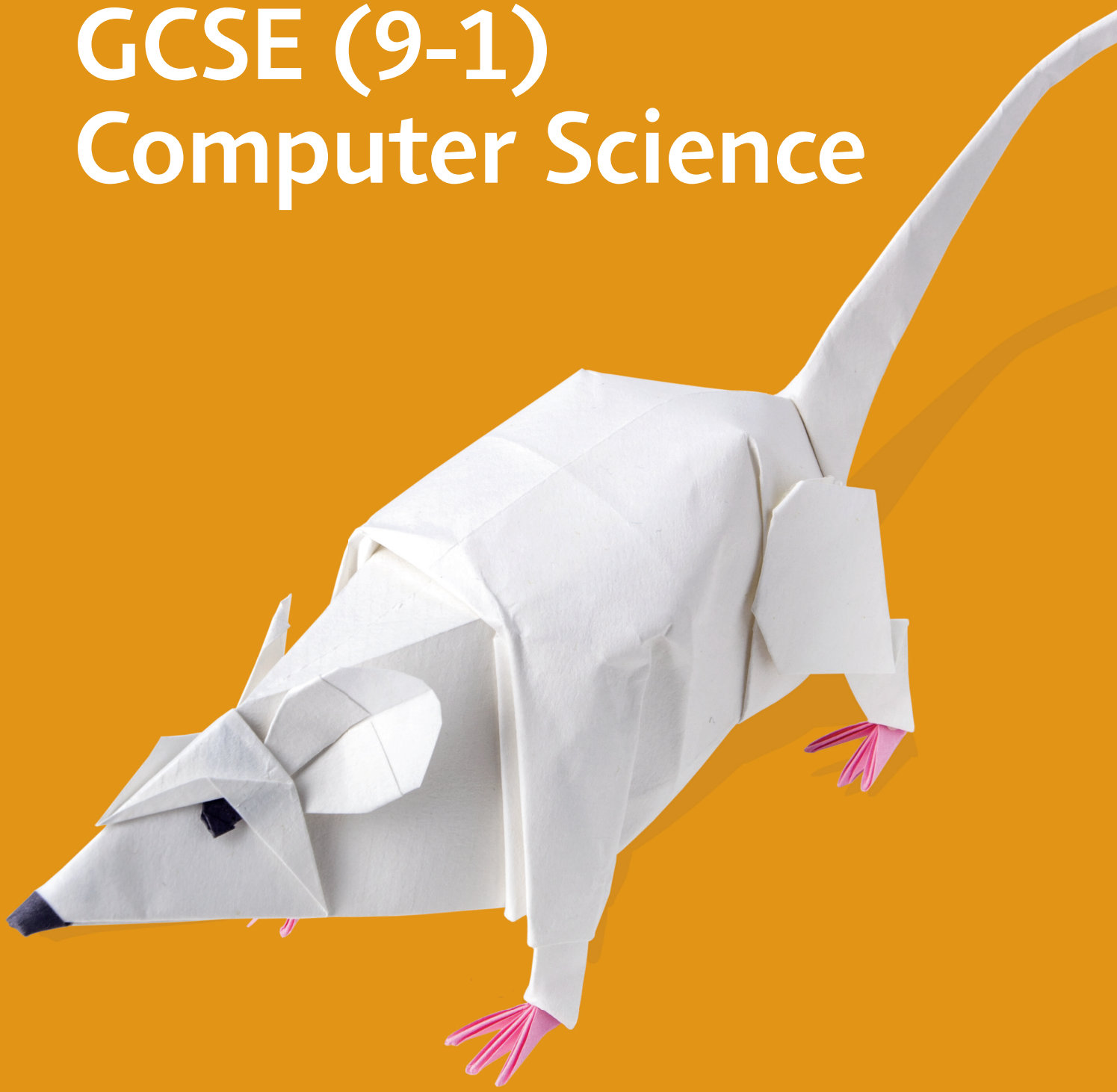
edexcel

GCSE (9-1) Computer Science



Sample Assessment Materials

Pearson Edexcel Level 1/Level 2 GCSE (9-1) in Computer Science (1CP1)

First teaching from September 2016

First certification from 2018

Issue 2

ALWAYS LEARNING

PEARSON

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 qualification websites at www.edexcel.com, www.btec.co.uk or www.lcci.org.uk. 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 40,000 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

These sample assessment materials are Issue 2. We will inform centres of any changes to this issue. The latest issue can be found on the Pearson website: 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.

Original origami artwork: Mark Bolitho

Origami photography: Pearson Education Ltd/Naki Kouyioumtzis

ISBN 978 1 4469 3184 4

Contents

Introduction

The Pearson Edexcel Level 1/Level 2 GCSE (9-1) in Computer Science is designed for use in schools and colleges. It is part of a suite of 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 be 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.
- When 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
Level 1/Level 2 GCSE

Centre Number	Candidate Number

Computer Science

Paper 1: Principles of Computer Science

Sample Assessment Material
Time: 1 hour 40 minutes

Paper Reference
1CP0/01

You will need:
Booklet containing pseudo commands

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.

Turn over ▶

S49766A

S 4 9 7 6 6 A 0 1 1 7

PEARSON

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

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

1 Computing devices are made up of many different types of internal components that hold, manipulate or transmit data.

(a) Name the component that holds instructions and data for programs waiting to be run by the CPU.

(1)

.....

.....

(b) Identify the type of memory used to make up for the difference in speed of two internal components.

(1)

☒ **A** ROM

☒ **B** Cache

☒ **C** Virtual

☒ **D** Non-volatile

(c) The ASCII code for the character 'D' is 68 in denary.

Derive the ASCII code for the character 'J' in denary.

(1)

.....

.....

(d) Storage capacities and network data speeds require measurements.

(i) The capacity of some storage media is measured in bytes.

Calculate how many bytes there are in **three** kilobytes of disc storage.

(2)

.....

.....

.....

.....

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(ii) Calculate how many bits are being transmitted per second for a network described as **three** Mbps.

(2)

(e) Identify a network protocol that is used for emails.

(1)

- ☒ **A** TCP/IP
- ☒ **B** Wi-Fi
- ☒ **C** HTTP
- ☒ **D** IMAP

(f) Describe the process for communication in the client-server network model.

(4)

(g) Identify the reason for using hexadecimal to represent data.

(1)

- ☒ **A** Hexadecimal uses letters instead of digits
- ☒ **B** Hexadecimal takes up less memory in the machine
- ☒ **C** Hexadecimal is easier for humans to read
- ☒ **D** Hexadecimal is quicker for the machine to interpret

(Total for Question 1 = 13 marks)

2 Different types of compression are used for different purposes.

(a) A travel company has designed some brochures that contain images and text in desktop publishing format. The travel company sends the documents electronically to a printing company for them to be printed.

Explain why the travel company uses lossless compression to send the documents.

(2)

.....

.....

.....

.....

(b) Compression normally reduces file size.

State **two** other characteristics of lossy compression.

(2)

1 .....

.....

2 .....

.....

(c) Run length encoding (RLE) is a type of image compression. Some data for an image is shown.

b	b	b	r	g	g	g	g	r	r

Show the result of compressing this data for the image using RLE.

(2)

.....

(Total for Question 2 = 6 marks)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

3 Many different types of software are written by programmers and used in computing devices.

(a) An engineer complains that her disc is too slow. She has been abroad and has used her Wi-Fi in different locations. She has checked that her anti-virus software is up to date.

Explain how different types of utility software could be used to fix the problem.

(3)

(b) Give **two** reasons why high-level programming languages are preferred for some programming tasks.

(2)

1 .....

2 .....

(c) Explain how software can be protected as intellectual property.

(2)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(d) Discuss the benefits of using subprograms when writing software solutions.

(6)

(Total for Question 3 = 13 marks)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

4 The ability to share data via networks has many advantages, but there are also disadvantages.

(a) Describe the role of protocols in a network.

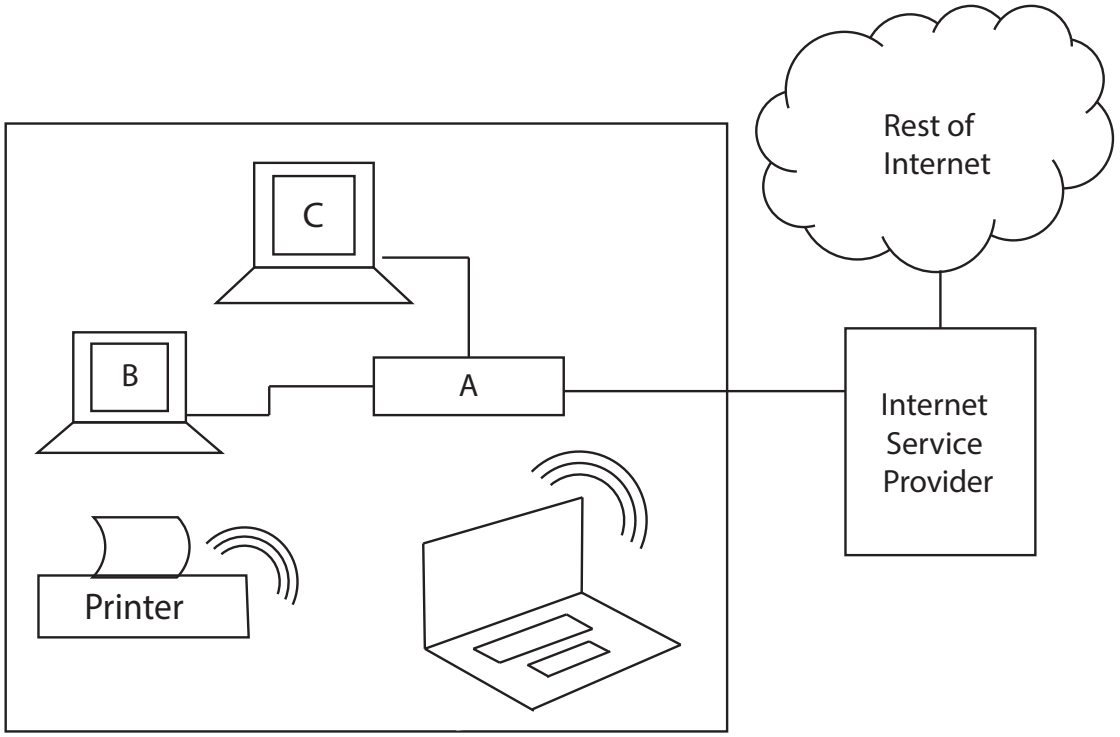
(2)

(b) A patch for an instant messaging application is available.

Explain why users of the application should apply the patch.

(2)

(c) A drawing of a home network is shown.



Describe how data is transmitted from device B to device C.

(2)

(d) The senior leadership team at a school is discussing different storage media. The administrative staff want to use the 'cloud'. The technical team wants to use servers located on the school grounds.

Compare storing data in the 'cloud' with storing data on hard discs connected to the school's servers.

(6)

(Total for Question 4 = 12 marks)

5 Modern society depends on many different embedded systems to function properly.

(a) Explain **one** function of the embedded system in a washing machine.

(2)

(b) Boolean operators are used in embedded systems.

Draw the truth table for the Boolean operator AND.

(1)

(c) Boolean expressions can be used to represent logical behaviours.

- The lights in a room are controlled by an embedded system.
- There is a power (P) switch to control the lights.
- A movement (M) sensor will turn the lights on when there is a person in the room, provided the power switch has been left in the on position.
- An override (O) switch will turn the lights on, regardless of the state of the movement sensor, provided the power switch has been left in the on position.

Construct a logic statement to represent the logic of this behaviour, using the symbols P, M, and O.

(3)

(d) Bit patterns can be used to represent the different states of an embedded system. These bits can be manipulated by several different operations.

Identify the result of applying a logical shift left by 3 to the 8-bit binary number 0101 0101.

(1)

- ☒ **A** 1010 1111
- ☒ **B** 1010 1000
- ☒ **C** 0100 0100
- ☒ **D** 1010 1010

(Total for Question 5 = 7 marks)

6 Algorithms are used to store images, solve problems, encrypt and sort data, and to control devices.

(a) State what is meant by the term 'algorithm'.

(2)

.....

.....

.....

.....

(b) Give **two** reasons why an unauthorised person cannot understand encrypted data.

(2)

1 .....

.....

2 .....

.....

DO NOT WRITE IN THIS AREA DO NOT WRITE IN THIS AREA DO NOT WRITE IN THIS AREA

(c) A Caesar cipher is a simple encryption algorithm based on shifting.

Explain whether a shift of +7 followed by a shift of -2, is more secure than a single shift when applied to the word 'pink'.

You should include a diagram in your answer.

(3)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(d) (i) State what is meant by a 'pixel'.

(1)

(ii) A simple bitmap image has the following characteristics:

4-bit colour depth (bit depth)

100 pixels by 300 pixels.

Calculate the size of this image in bytes.

(2)

(e) A list is made up of the numbers 84, 52, 4, 6, 68, 39, 53, 1.

Show the steps involved when sorting this list of numbers using a merge sort algorithm.

(2)

(f) A programmer is writing software for a new set-top receiver for satellite TV.

Explain why the programmer should use a compiler instead of an interpreter to translate the code.

(2)

(g) State what is meant by the term 'abstracting'.

(2)

(h) Games software involves the use of algorithms and abstractions.

(i) A software game involves the use of two dice. An algorithm in this game is called 'roll'.

State the purpose of the algorithm 'roll'.

(1)

(ii) Give a reason why 'roll' is an abstraction.

(1)

(Total for Question 6 = 16 marks)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

7 Different types of data can be represented digitally.

(a) Negative numbers can be represented digitally.

(i) Complete the table to show how +6 is represented in binary and how -6 is represented in two's complement.

Space for working out.

(2)

+6								
-6								

(ii) Compare **two** features of the two's complement representation of negative numbers with the sign-magnitude representation.

(2)

1 .....

.....

2 .....

.....

(iii) Two 8-bit binary numbers, 1011 0100 and 1001 0001, are added.

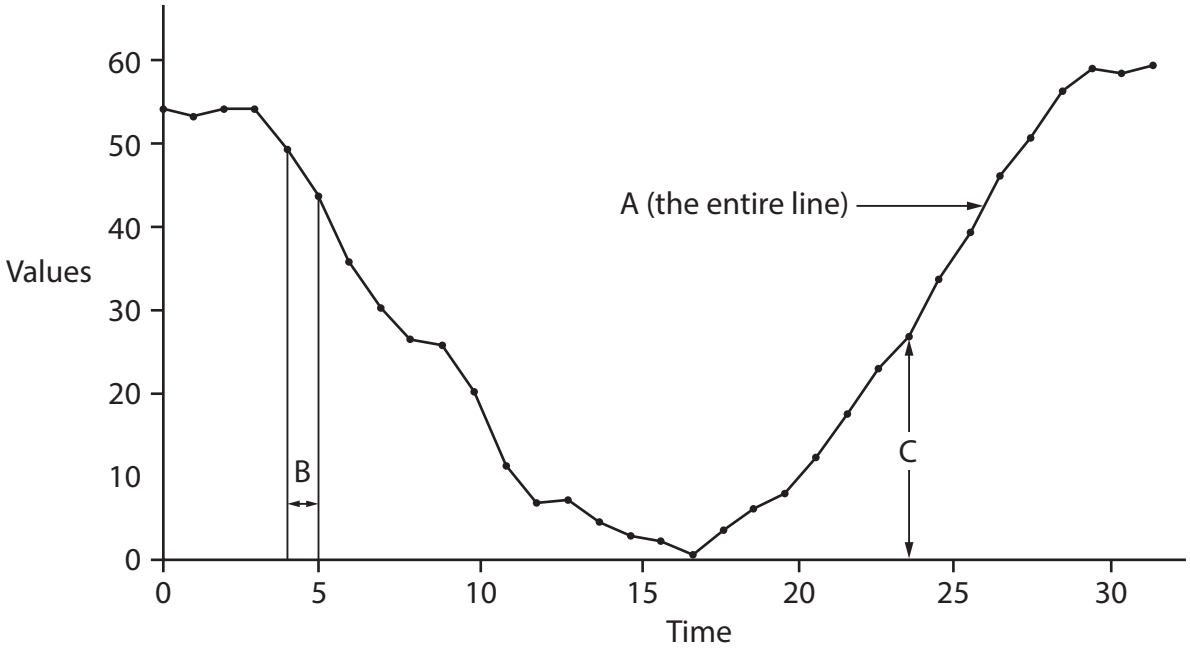
Identify the letter that represents the true statement about the result.

(1)

- ☒ **A** 1100 0101
- ☒ **B** 0100 0101
- ☒ **C** Both A and B are accurate
- ☒ **D** None of the above are accurate

(b) Sound can be stored on a digital device, but only after being converted from its naturally occurring state.

Sound Transformation



(i) Identify each of the items (A, B, C) shown on the image in relation to the conversion required to store sound on a digital device.

(3)

A .....

B .....

C .....

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(ii) Explain the effect of increasing the sampling frequency for this sound conversion.

(3)

.....

.....

.....

.....

(Total for Question 7 = 11 marks)

TOTAL FOR PAPER = 80 MARKS

Pearson Edexcel Level 1/Level 2 GCSE (9–1)

Computer Science

Papers 1, 2 and 3

Sample assessment material for first teaching September 2016
Pseudo-code command set

Paper Reference
1CP1/01
1CP1/02
1CP1/3A-3E

Do not return this resource booklet with the question paper.

Turn over ▶

S50557A

PEARSON

Pseudo-code command set

Questions in the written examination that involve code will use this pseudo-code 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 pseudo-code, the < > indicates where expressions or values need to be supplied. The < > symbols are not part of the pseudo-code.

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**

WHILE <condition> DO
  <command>
END WHILE
	Pre-conditioned loop. Executes <command> whilst <condition> is true.	WHILE Flag = 0 DO
		  SEND 'All well' TO DISPLAY
		END WHILE

REPEAT <command> UNTIL <expression>	Post-conditioned loop. Executes <command> until <condition> is true. The loop must execute at least once.	REPEAT SET Go TO Go + 1 UNTIL Go = 10
REPEAT <expression> TIMES <command> END REPEAT	Count controlled loop. The number of times <command> is executed is determined by the expression.	REPEAT 100-Number TIMES SEND '*' TO DISPLAY END REPEAT

FOR <id> FROM <expression> TO
<expression> DO
  <command>
END FOR
	Count controlled loop. Executes <command> a fixed number of times.	FOR Index FROM 1 TO 10 DO
		  SEND ArrayNumbers[Index] TO DISPLAY
		END FOR

FOR <id> FROM <expression> TO
<expression> STEP <expression> DO
  <command>
END FOR
	Count controlled loop using a step.	FOR Index FROM 1 TO 500 STEP 25 DO
		  SEND Index TO DISPLAY
		END FOR

FOR EACH <id> FROM <expression> DO <command> END FOREACH	Count controlled loop. Executes for each element of an array.	SET WordsArray TO ['The', 'Sky', 'is', 'grey'] SET Sentence to '' FOR EACH Word FROM WordsUArray DO SET Sentence TO Sentence & Word & ' '

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

<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.

Paper 1 Mark scheme

Question Number	Answer	Additional Guidance	Mark
1(a)	The name of the required component: • Main memory/RAM/Random Access Memory (1)	• Do not award memory alone.	**(1)**

Question Number	Answer	Additional Guidance	Mark
1(b)	• B [Cache] (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
1(c)	Derivation of the required ASCII code: • Derive the fact that 74 denary is the ASCII value for the letter 'J' (1) Example: D, E, F, G, H, I, J == 68, 69, 70, 71, 72, 73, 74 J – D = 6; 68 + 6 = 74	/01	**(1)**

Question Number	Answer	Additional Guidance	Mark
1(d)(i)	Calculation: • 1 kilobyte = 1024 bytes (1) • 3 x 1024 or 3072 (1)	• Units are not required. • Award full marks for the correct numerical answer.	**(2)**

Question Number	Answer	Additional Guidance	Mark
1(d)(ii)	Calculation: • 1 Megabit = 1000000 bytes (1) • 3 x 1000000 or 3000000 (1)	• Units are not required. • Award full marks for the correct numerical answer.	**(2)**

Question Number	Answer	Additional Guidance	Mark
1(e)	• D [IMAP] (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
1(f)	A description that makes reference to the following points: • A client uses the address of the server to make a connection (1) • then the client sends a request to the server/client request services from the server (1) • The server obtains the address of the client once that client makes a connection to the server (1) • then the server sends the requested data/services (to the client) (1)	• Do not award examples only, such as a web page and browsers.	**(4)**

Question Number	Answer	Additional Guidance	Mark
1(g)	• C [Hexadecimal is easier for humans to read] (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
2(a)	An explanation that makes reference to the following points: • lossless compression ensures that no data is lost (1) • so that the desktop published file can be restored so that it is exactly the same as the original (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
2(b)	Any two from: • Data is removed during the compression process (1) • The removed data is not normally noticed by a human (sight, sound) (1) • The original contents of the file cannot be restored exactly (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
2(c)	• Letters in correct order (1) • Correct number (1) Either: 3b 1r 4g 2r Or b3 r1 g4 r2	• Disregard spacing. • The single instance of 'r' in position 2, must be denoted by '1r' or 'r1'. The numeric value is necessary.	**(2)**

Question Number	Answer	Additional Guidance	Mark
3(a)	An explanation that makes reference to two of the following points: • Defragmentation software moves file blocks closer together (1) • Compression software reduces file sizes using less space (1) • Use backup software to move files to a secure place and deleting files on the hard disc frees up space (1) and • this speeds up disc access times to the files (which will improve the speed of the disc) (1)	• Allow archive as equivalent to backup and delete.	**(3)**

Question Number	Answer	Additional Guidance	Mark
3(b)	Any two reasons from: • Instructions are close to English/easier for humans to read/write than a low-level language/makes fewer errors (1) • It's quicker to develop code/easier to maintain code (than if it were written in a low-level language) (1) • They are usually available across many platforms/chipsets/operating systems (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
3(c)	An explanation that makes reference to the following point: • Intellectual property can be protected by copyright/patent laws (1) Plus one from: • so that by registering the patent the patent holder has exclusive rights to make/use/sell the software (1) • so that by registering the copyright the source/(object) code is protected (1)	• Do not award marks for examples only. • Do not accept 'prevents theft'.	**(2)**

Question Number	Indicative content	Mark
3(d)	Indicative content: • are written only once/no repeated/copied blocks of code (1) • only needs to be debugged once (1) • can be reused/called from anywhere in the program (1) • make main program more readable/shorter (1) • can reduce the need for global variables (1) • can be put into libraries/reused in other programs/shared with others (1) • can be used to hide complex code/functionality (1) • can be written in low-level languages for special purpose/interact with hardware/optimise performance (1)	**(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 basic 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 which is clear, coherent and logically structured.

Question Number	Answer	Additional Guidance	Mark
4(a)	A description that makes reference to the following points: • the protocol allows the devices on a network to communicate with each other (1) • by the use of a common set of rules (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
4(b)	An explanation that makes reference to the following points: • A patch will fix any known security issues (1) • so the machine running the application is less vulnerable to successful cyberattacks (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
4(c)	A description that makes reference to the following points: • device B puts the address of device C into a packet/message to send (1) • then device A reads that address and sends the packet to device C (1)		**(2)**

Question Number	Indicative content	Mark
4(d)	Indicative content: • The school/technical team will be responsible for backing up/disaster recovery if the data is stored on hard discs (connected to the school's servers) • The storage provider is responsible for backup/disaster recovery if the school stores its data in the 'cloud' • The school's files are available without an internet connection if data is stored on hard discs (connected to the school's servers) • An internet connection is required to access the school's files if they are stored in the 'cloud' • Staff at the school can access their files anywhere by using mobile devices if the school's files are stored in the 'cloud' • The staff can access their files at school only if the data is stored on hard discs (connected to the school's servers) • The school will be responsible for maintaining the servers, replacing them when they age/technology becomes outdated • The storage provider is responsible for updating the storage system in the cloud and providing new technology as it evolves • The school will be responsible for increasing the storage capacity of their servers as demands evolve • The storage provider is able to provide additional storage space on demand.	**(6)**

Level	Mark	Descriptor
	0	No rewardable content.
Level 1	1–2	A comparison may be attempted but with limited application of knowledge and understanding of key concepts of computer science to the theoretical context. The comparison will contain basic information with some attempt made to link knowledge and understanding to the given context.
Level 2	3–4	A comparison will be given with adequate application of knowledge and understanding of key concepts of computer science to the theoretical context. Are occasionally supported through a linkage. The comparison shows some linkages and lines of reasoning with some structure.
Level 3	5–6	A comparison will be given with comprehensive application of knowledge and understanding of key concepts of computer science to the theoretical context. Line(s) of reasoning are supported throughout by sustained application of relevant evidence. The comparison shows a well-developed and sustained lines of reasoning which is clear, coherent and logically structured.

Question Number	Answer	Additional Guidance	Mark
5(a)	An explanation that makes reference to the following points: • monitors weight of load (1) • so that it can adjust water levels for small/large loads (1) **OR** • monitors water temperature (1) • so it can maintain the required value by turning heating element on/off (1) **OR** • monitors soaking button setting (1) • so that it can add additional time at the right place in the programme cycle (1) **OR** • monitors maximum spin switch setting (1) • so that it can set required motor revolutions (1)	• Other appropriate responses can be awarded.	**(2)**

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>5(b)</td><td>A truth table for the function of the operator AND:
• four different input combinations and four different correct output combinations

Examples

<table>
<tr><th>Input 1</th><th>Input 2</th><th>Output</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table>

<table>
<tr><th>Input 1 / Input 2</th><th>0</th><th>1</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
</table>

False AND False = False
False AND True = False
True AND False = False
True AND True = True</td><td>Any format for the truth table is acceptable as long as four different distinct inputs and four different distinct outputs are discernible.</td><td>(1)</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
5(c)	A logic statement that includes: • Correct use of 'OR' (1) • Correct use of 'AND' (1) • Correct use of brackets or order of precedence if brackets not used (1) Examples: • (O OR M) AND P • P AND (M OR O) • (O AND P) OR (P AND M) • O AND P OR P AND M	• Disregard capitalisation. • Overriding order of precedence (NOT, AND, OR) **may** require use of brackets. • Expression or parts of expression may be in different orders. • Any alternative equivalent expression should be awarded.	**(3)**

Question Number	Answer	Additional Guidance	Mark
5(d)	• B [1010 1000] (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
6(a)	A statement that includes: • an algorithm is a sequence of steps/step-by-step instructions (1) • which (if followed exactly) will perform a specific task (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
6(b)	Two reasons: • Encrypted data is encoded/scrambled so that it cannot be understood (1) • The unauthorised person would not have the key to decrypt the data (1)	• Disregard responses indicating 'access'; encryption only makes the data uninterpretable. • Do not award secure/secret alone.	**(2)**

Question Number	Answer	Additional Guidance	Mark
6(c)	• Applying a double shift to 'pink' to obtain unsp (1) • Identifying +5 as the equivalent single shift to obtain unsp (1) • Recognise that both shifts give the same result, so the double shift is no more secure than the single shift (1) | | pink | |---|---| | +7 | wpur | | -2 | unsp | | | pink | |---|---| | +5 | unsp | | | |	Other forms of diagrams are acceptable. A simple statement to say that the double shift is not as secure can score a mark only if explained in full.	**(3)**

Question Number	Answer	Additional Guidance	Mark
6(d)(i)	A statement that includes: • a pixel is the smallest identifiable unit in a graphic image (1)	Do not accept picture element.	**(1)**

Question Number	Answer	Additional Guidance	Mark
6(d)(ii)	• Mathematical expression (1) • 15000 bytes (1) Example: $\frac{300 \times 100 \times 4}{8} = 15000$	• Accept any other equivalent mathematical expression. • Units not required.	**(2)**

Question Number	Answer	Additional Guidance	Mark
6(e)	• Award 1 mark for each row of the process. 84 | 52 | 4 | 6 | 68 | 39 | 53 | 1 52 84 4 6 39 68 1 53 (1) 4 6 52 84 1 39 53 68 (1) This leads to: 1 | 4 | 6 | 39 | 52 | 53 | 68 | 84	• Award 1 mark for each step. • Marks should not be awarded for the last row. • Any notation showing distinct lists at each stage is acceptable.	**(2)**

Question Number	Answer	Additional Guidance	Mark
6(f)	An explanation that makes reference to the following points: • The set-top box must process data quickly (1) • so a compiler is used because compiled code runs faster than interpreted code (1)	• Disregard reference to cross-platforms; question context is a new chip.	**(2)**

Question Number	Answer	Additional Guidance	Mark
6(g)	A statement that includes: • (Abstracting means) removing unnecessary detail/simplifying (1) • so that we can focus on the essence/real part/important part of the problem (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
6(h)(i)	A statement that includes: • The purpose of 'roll' is to generate a single number, chosen randomly, between 1 and 12, inclusive (1) OR • Two numbers between 1 and 6	• Accept between 1 and 12 • Do not award 'simulate a roll of dice' or equivalent.	**(1)**

Question Number	Answer	Additional Guidance	Mark
6(h)(ii)	One of the following reasons: • 'Roll' is an abstraction because it allows the programmer to focus on the result of a roll, rather than on how to implement the roll algorithm (1) **OR** • 'Roll' is an abstraction because it is a model/simulation/representation of a real/physical activity/action (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
7(a)(i)	• One mark for each correct row (2): 0 0 0 0 0 1 1 0 1 1 1 1 1 0 1 0		**(2)**

Question Number	Answer	Additional Guidance	Mark
7(a)(ii)	The comparison should include reference to both two's complement and signed-magnitude numbers for any **two** of the following: • Addition of two's complement will always work (1) • Addition of sign-magnitude numbers does not always work (1) • There is only one way to represent the number 0 in two's complement (1) • There are two ways to represent the number 0 in sign-magnitude numbers (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
7(a)(iii)	D [None of the above are accurate] (1)	• Overflow has occurred so 8-bits are not large enough to hold the result. • 325 = 1 0100 0101	**(1)**

Question Number	Answer	Additional Guidance	Mark
7(b)(i)	Correct identification of the labelled items: A. Analogue signal/sound wave (1) B. Sample interval (1) C. Amplitude (1)	• Do not allow sampling frequency as equivalent to sampling interval.	**(3)**

Question Number	Answer	Additional Guidance	Mark
7(b)(ii)	An explanation that makes reference to the following points: • (Higher sampling frequencies) generate more sampling data (1) • which increases memory/storage requirements (1) • and gives a more accurate digital representation (of the original signal) (1)	• Do not award responses referring to bit rate. • Do not award responses indicating quality of playback.	**(3)**

Write your name here

Surname	Other names

Pearson Edexcel
Level 1/Level 2
GCSE (9–1)

Centre Number | | | | | |

Candidate Number | | | | |

Computer Science

Paper 2: Application of Computational Thinking

Sample assessment material for first teaching
September 2016
Time: 2 hours

Paper Reference
1CP1/02

You will need:
Booklet containing pseudo commands.

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.*
- You are not allowed to use a calculator.

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.*

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.

Turn over ▶

S49768A

S 4 9 7 6 8 A 0 1 2 6

PEARSON

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

Questions in this paper are based on a scenario.

Sparky Autos

Sparky Autos is a family-run business based on a farm. The business offers visitors the opportunity to drive small electric cars around a racetrack on the farm.

There are up to 20 electric cars available each time for a race.

Each car has:

- an electronic tag so its number and position can be tracked using sensors placed around the farm
- a number painted on its side.

1 Cars at Sparky Autos are serviced regularly to check that they are safe and reliable. A service can be triggered by the car being run for 100 hours or if 30 days have passed.

(a) State **two** variables that need to be created to store data for each car.

(2)

1 .....

.....

2 .....

.....

(b) Each car takes at least 30 minutes to service. Older cars take longer to service. Ten minutes is added for each full year of the car's age. The year of purchase is known.

Construct a general expression showing the time needed to service any car.

(2)

.....

.....

.....

.....

(Total for Question 1 = 4 marks)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

2 A computer program can be used to determine information about the cars and the drivers.

(a) Complete the table to show an input, a process and an output, using the following information:

- each race has 5 laps
- there are 10 cars in this race.

(3)

Input	Process	Output
	Calculate overall total time for any car in the race	Sum of lap times
10 car numbers and 10 total times		Car number with quickest total time
10 car numbers with 5 lap times each	Find car with quickest lap time	

(b) The arithmetic operator modulus (MOD) manipulates numbers. It could be used in the calculations required by the processes in the table above.

State the purpose of the MOD function.

(1)

.....

.....

(Total for Question 2 = 4 marks)

3 Sparky Autos gives discounts based on the number of visitors in a group.

The pseudo-code for an algorithm that determines group discounts is shown.

```
1
2   IF ((numAdults > 1) AND (numChildren > 0)) THEN
3       SEND "family discount" TO DISPLAY
4   ELSE
5       IF (numAdults >= 10) THEN
6           SEND "large group discount" TO DISPLAY
7       ELSE
8           IF (numAdults >= 5) THEN
9               SEND "small group discount" TO DISPLAY
10          ELSE
11              SEND "regular pricing" TO DISPLAY
12          ENDIF
13      ENDIF
14  ENDIF
```

(a) Complete the table to show the output of the pseudo-code algorithm, based on the given inputs.

(3)

Input		Output displayed
numAdults	**numChildren**	
8	0	
2	2	
12	0	

(b) The pseudo-code algorithm needs to be tested more thoroughly.

Construct test data to meet the requirements set out in the table.

(2)

Requirements	Input	
	numAdults	**numChildren**
A condition generating 'regular pricing'		0
Smallest group qualifying for 'family discount'		

(Total for Question 3 = 5 marks)

DO NOT WRITE IN THIS AREA DO NOT WRITE IN THIS AREA DO NOT WRITE IN THIS AREA

BLANK PAGE

QUESTION 4 BEGINS ON THE NEXT PAGE.

4 A computer programmer uses a programming language to write program code for Sparky Autos.

(a) State **two** techniques that the programmer could use to make the code easy to read.

(2)

1 .....

.....

2 .....

.....

(b) In the summer, Sparky Autos is open more hours each day. This means additional members of staff are needed.

Pseudo-code that works out the number of days that additional members of staff are needed is shown.

```
# This program is owned by Sparky Autos

SET dates[] TO ["01/06/2014", "02/06/2014", "03/06/2014"]
SET extra[] TO ['Y', 'N', 'Y']
SET visitors[] TO [122, 51, 147]
SET staff[] TO [12, 6, 10]

SET length TO LENGTH (extra)
SET count TO 0
SET index TO 0

REPEAT
    IF extra[index] = 'Y' THEN
        SET count TO count + 1
    END IF
    SET index TO index + 1
UNTIL index = length

```

(i) Complete the table to identify the line number(s) that show **one** example of each of the different programming constructs.

(4)

Programming construct	Line number(s)
A comment	
An integer variable initialisation	
Selection	
Iteration	

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(ii) Complete the trace table showing the execution of the pseudo-code. You may not need to fill in all the rows in the table.

(4)

Length	Count	Index	Extra (index)

(Total for Question 4 = 10 marks)

5 Sparky Autos collects data about many different aspects of the business.

(a) An algorithm had an error as shown.

```
9  TES index TO 0
```

A programmer corrected the error as the code translator displayed this message.

```
TES index TO 0
  ^ Cannot find identifier TES (line 9)
```

Explain this type of error.

(2)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(b) The pseudo-code for an algorithm that determines the maximum number of visitors in a three-day period is shown.

```
2  # This program is owned by Sparky Autos
3
4  SET dates[] TO ["01/06/2014", "02/06/2014", "03/06/2014"]
5  SET extra[] TO ['Y', 'N', 'Y']
6  SET visitors[] TO [122, 51, 147]
7  SET staff[] TO [12, 6, 10]
8
9  SET index TO 0
10 SET maximum TO 100
11 SET length TO LENGTH (visitors)
12
13 FOR index FROM 0 TO length DO
14     IF visitors[index] > maximum THEN
15         SET maximum TO visitors[index]
16     END IF
17 END FOR
18 SEND "Maximum visitors is " & maximum TO DISPLAY
```

Line 10 and line 13 each have a logic error.

Identify the error in each line and construct new lines of code that will correct the errors.

(4)

	Error	**Correction**
Line 10		
Line 13		

(c) Complete the table to give the appropriate data type of a variable to store each item.

(4)

Item	Data type
Gender of individual staff member	
Whether an individual car is still under the manufacturer's warranty	
Mean number of hours needed to recharge the battery in each car	
The number on the individual car	

(d) Each member of staff:

- has a 4-digit ID number, such as 3865 or 4722
- works a whole number of hours on the days they work.
- works no more than 12 hours in a single day.

The business is open 7 days a week.
Draw a diagram of a data structure that shows the hours worked for each day of the week. Include data for at least **two** members of staff.

(3)

(Total for Question 5 = 13 marks)

BLANK PAGE

QUESTION 6 BEGINS ON THE NEXT PAGE.

6 A programmer has written subprograms to help him in his work.

The pseudo-code for a function to determine the amount of time elapsed between the car switch being turned on and turned off is shown.

```
1
2  REAL totalTime
3  SET totalTime TO 28.2946    # Minutes
4
5  CONST REAL TOTALTIME
6  SET TOTALTIME TO 10.25
7
8  REAL newTime
9  SET newTime TO timeRunning (11.4591, 12.1023)
10
11 SEND "Original " & totalTime TO DISPLAY
12
13 FUNCTION timeRunning (timeOn, timeOff)
14 BEGIN FUNCTION
15     REAL totalTime
16     SET totalTime TO timeOff - timeOn
17     SET TOTALTIME TO totalTime
18     RETURN totalTime
19 END FUNCTION
20
```

(a) The underlined item in line 18 has a name when used with the RETURN keyword.

State this name.

(1)

.....

.....

(b) Explain why the variable on line 2, totalTime, is not reset when the pseudo-code on line 16 is executed.

(2)

.....

.....

.....

.....

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

(c) Explain the effect of executing the pseudo-code on line 17.

(2)

.....

.....

.....

.....

(d) Validation is required for the items underlined in line 13.

Construct **two** validation tests and for each test give a piece of invalid test data.

(4)

Validation test	Test data

(Total for Question 6 = 9 marks)

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

7 The pseudo-code for a procedure that repeatedly checks the time is shown. The effect of calling this procedure is that the calling code waits for 5 seconds.

```
1
2  PROCEDURE wait ()
3  BEGIN PROCEDURE
4      SET startTime TO getTime()  # Gets the current time in seconds
5      SET newTime TO getTime ()
6      WHILE (newTime - startTime < 5) DO
7          SET newTime TO getTime ()
8      END WHILE
9  END PROCEDURE
10
```

Amend this procedure so that it will wait for any number of required seconds without user input.

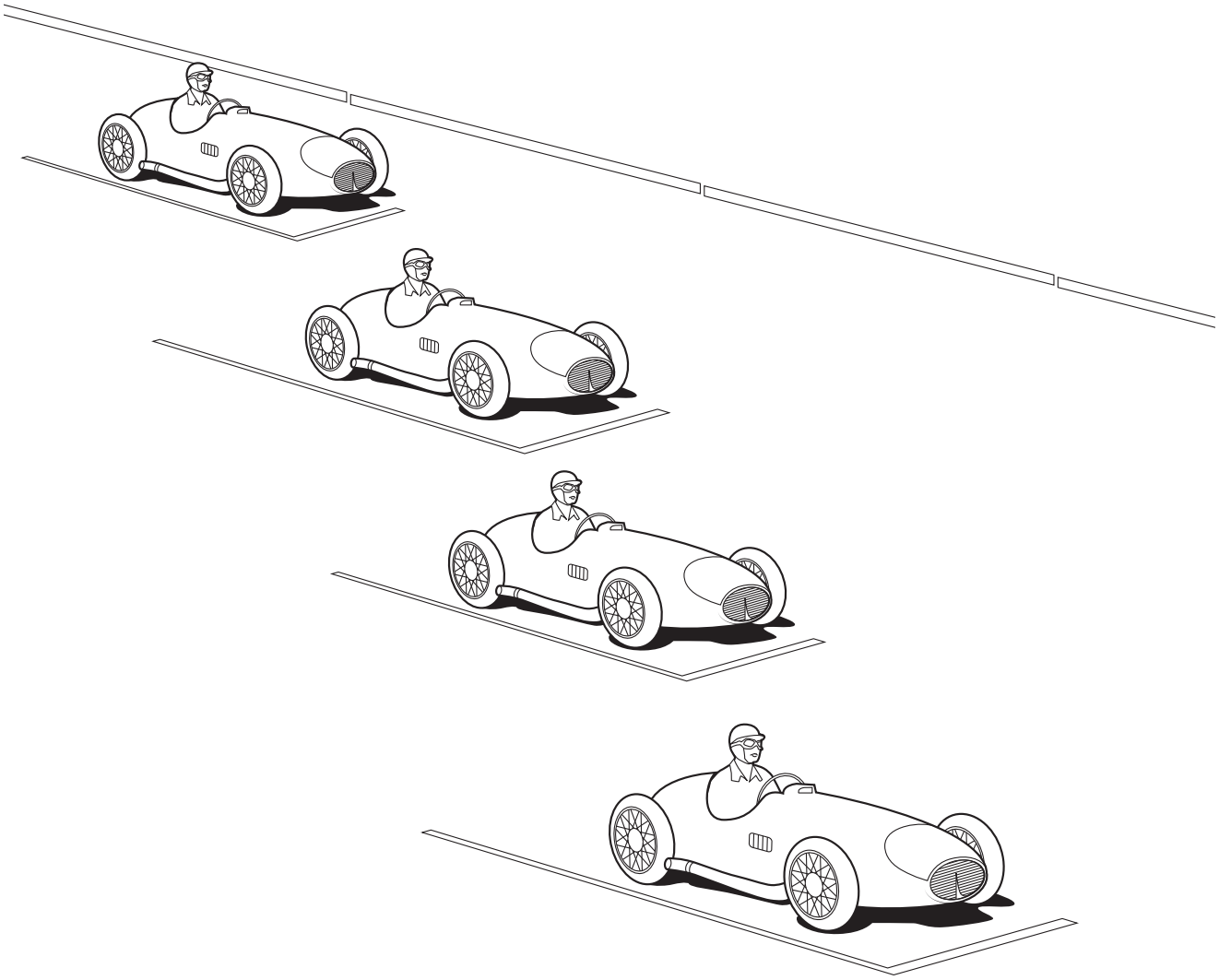
(3)

(Total for Question 7 = 3 marks)

BLANK PAGE

QUESTION 8 BEGINS ON THE NEXT PAGE.

8 At the start of a race, cars are placed on a starting grid in a staggered pattern.



When a car makes a false start it breaks a light beam and this notifies the computer system. The race must be stopped if a car makes a false start.

- There is a maximum of 20 cars in a race.
- The programmer has chosen to use a 1-dimensional array to store this data.

Assess how appropriate a 1-dimensional array is as opposed to using separate variables when storing this data.

(5)

(Total for Question 8 = 5 marks)

9 Sparky Autos collects, stores, and processes data about its customers.

(a) It collects the names and postal addresses of customers. This data is stored in a comma separated value file (CSV) with one record per line. The file contains over 5000 records. Part of the file is shown.

PP12 9AP, White, Winston, 67 Willow Street, Martinsville

SR45 8ZX, Brown, Brenda, 45 Birch Close, Flat Rock

PP08 3TR, Green, Geraldine, 128 Geranium Lane, Auburn

EG24 4KM, Black, Bobby, 132 Boxwood Grove, Galax

Sparky Autos wants to send a newsletter only to customers whose postcode begins with 'PP'.

Explain an improvement that could be made to the data from the file when used for this purpose.

(3)

(b) Before driving the cars, each customer fills in an information sheet. Customers under 16-years old must have the sheet completed by a parent or guardian. The information is entered into a computer program and stored on the hard disc of the computer.

(i) Part of the information sheet is shown.

Name:		
Address:		
Postcode:		
Email:		
☐ Tick if you would like to receive our newsletter.		
Medical		
Height:	Weight:	Do you suffer from seizures?
Ethnic Origin		

Black		**Asian**		**White** ☐	**Other** ☐
African	☐	Chinese	☐		*(please specify)*
Caribbean	☐	Bangladeshi	☐		
Other	☐	Indian	☐		
		Pakistani	☐		

Identify **two** ethical issues of collecting this type of information.

(2)

1 .....

.....

2 .....

.....

(ii) Give a reason why it may be illegal for organisations to share their data with third parties.

(1)

.....

.....

(Total for Question 9 = 6 marks)

DO NOT WRITE IN THIS AREA DO NOT WRITE IN THIS AREA DO NOT WRITE IN THIS AREA

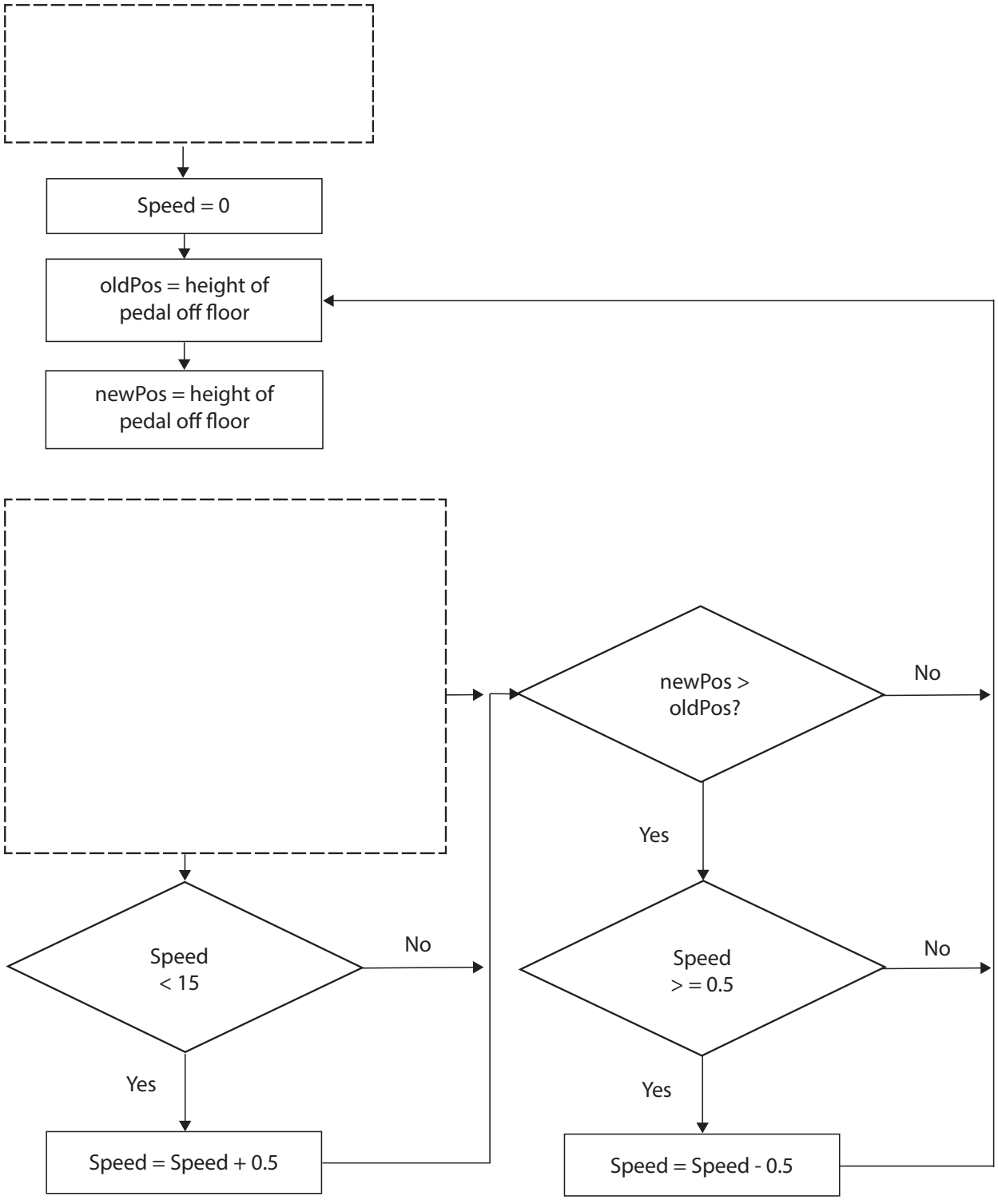
10 Each car at Sparky Autos has a pedal to make the car go forward.

(a) The drivers can make the car go forward by using the pedal.

- The closer the pedal is to the floor, the faster the car goes.
- The further the pedal is from the floor, the slower the car goes.
- Each car has its speed limited to 15 kilometres per hour.

Complete the flow chart to show this process.

(6)



(b) Each car also has direction indicators, which are controlled by an indicator switch that is operated by the driver.

- The car has two indicator lamps, one on the right and one on the left.
- The indicator switch has three positions.
- The middle position is the off position.
- The up position is to indicate a right turn.
- The down position is to indicate a left turn.
- When a lamp is turned on, it flashes (on, off) in 0.5 second intervals, until the switch is moved to the off position.

Construct a flow chart to show this process.

(6)

(Total for Question 10 = 12 marks)

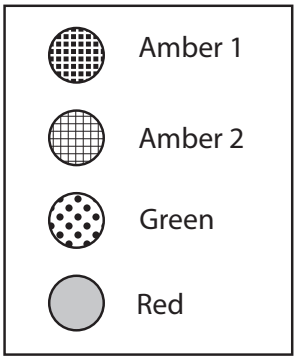
11 A light sequence is used to control the start of the races.

Only the red light is on at the start of the race.

The red light goes off and the first amber light goes on for 1 second.

It goes off and the second amber light goes on for 1 second.

It goes off and the green light goes on to start the race.



The program for controlling the sequence for the lights has two requirements.

- The race controller has to enter 'y' on the keyboard to start the light countdown.
- When a car makes a false start it breaks a light beam. If a car breaks a light beam the race is stopped.

The controlling sequence for the lights has three subprograms that have been written. These are shown in the table.

Subprogram	Purpose
countGridBreaks()	a function that returns the number of cars breaking their light beams
stopRace()	a procedure that resets all lights to their initial state and informs the controller that the race is stopped
wait()	a procedure that suspends execution for 1 second

Write an algorithm to control the sequence for the lights. Use pseudo-code or a programming language with which you are familiar.

(9)

You may continue on the next page.

(Total for Question 11 = 9 marks)

TOTAL FOR PAPER = 80 MARKS

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

DO NOT WRITE IN THIS AREA

Pearson Edexcel Level 1/Level 2 GCSE (9–1)

Computer Science

Papers 1, 2 and 3

Sample assessment material for first teaching September 2016
Pseudo-code command set

Paper Reference
1CP1/01
1CP1/02
1CP1/3A-3E

Do not return this resource booklet with the question paper.

Turn over ▶

S50557A

PEARSON

Pseudo-code command set

Questions in the written examination that involve code will use this pseudo-code 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 pseudo-code, the < > indicates where expressions or values need to be supplied. The < > symbols are not part of the pseudo-code.

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**

WHILE <condition> DO
  <command>
END WHILE
	Pre-conditioned loop. Executes <command> whilst <condition> is true.	WHILE Flag = 0 DO
		  SEND 'All well' TO DISPLAY
		END WHILE

REPEAT <command> UNTIL <expression>	Post-conditioned loop. Executes <command> until <condition> is true. The loop must execute at least once.	REPEAT SET Go TO Go + 1 UNTIL Go = 10
REPEAT <expression> TIMES <command> END REPEAT	Count controlled loop. The number of times <command> is executed is determined by the expression.	REPEAT 100-Number TIMES SEND '*' TO DISPLAY END REPEAT

FOR <id> FROM <expression> TO
<expression> DO
  <command>
END FOR
	Count controlled loop. Executes <command> a fixed number of times.	FOR Index FROM 1 TO 10 DO
		  SEND ArrayNumbers[Index] TO DISPLAY
		END FOR

FOR <id> FROM <expression> TO
<expression> STEP <expression> DO
  <command>
END FOR
	Count controlled loop using a step.	FOR Index FROM 1 TO 500 STEP 25 DO
		  SEND Index TO DISPLAY
		END FOR

FOR EACH <id> FROM <expression> DO <command> END FOREACH	Count controlled loop. Executes for each element of an array.	SET WordsArray TO ['The', 'Sky', 'is', 'grey'] SET Sentence to '' FOR EACH Word FROM WordsUArray DO SET Sentence TO Sentence & Word & ' '

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

<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.

Paper 2 Mark scheme

Question Number	Answer	Additional Guidance	Mark
1(a)	Any two from: • Date/time of last service (1). • Total running time since last service (1) • Details of last service (1) • Overall total of running time since car bought (lifetime) (1) • Car number (1)	• Accept meaningful variable names tied to the scenario, e.g. CarNumber	**(2)**

Question Number	Answer	Additional Guidance	Mark
1(b)	An expression that includes: • Calculation showing age in years (1) • 30 min + (10 min * ageInYears) (1) Example: 30 + (10 * (todayYear – purchaseYear)) (todayYear – purchaseYear) * 10 + 30 30 min + (10 min * (todayYear – purchaseYear))	• Units not required	**(2)**

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>2(a)</td><td>One mark for each point:

<table>
<tr><th>Input</th><th>Process</th><th>Output</th></tr>
<tr><td>(up to) 5 lap times (1)</td><td>Calculate overall total time for any car in the race</td><td>Sum of lap times</td></tr>
<tr><td>10 car numbers and 10 total times</td><td>Find the winner (1)</td><td>Car number with quickest total time</td></tr>
<tr><td>10 car numbers with 5 lap times each</td><td>Find car with quickest lap time</td><td>Car number with quickest lap time (1)</td></tr>
</table>
</td><td>• Award as long as meaning is discernible</td><td>(3)</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
2(b)	One mark for a correct statement: • Returns the integer part of a division		**(1)**

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>3(a)</td><td>One mark for each correct answer in the last column:

<table>
<tr><th colspan="2">Input</th><th>Output displayed</th></tr>
<tr><th>numAdults</th><th>numChildren</th><th></th></tr>
<tr><td>8</td><td>0</td><td>small-group discount (1)</td></tr>
<tr><td>2</td><td>2</td><td>family discount (1)</td></tr>
<tr><td>12</td><td>0</td><td>large-group discount (1)</td></tr>
</table>
</td><td>• In this order only in last column</td><td>(3)</td></tr>
</table>

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>3(b)</td><td>One mark for each correct row:

<table>
<tr><th rowspan="2">Requirements</th><th colspan="2">Input</th></tr>
<tr><th>numAdults</th><th>numChildren</th></tr>
<tr><td>A condition generating ‘regular pricing’</td><td>1/2/3/4</td><td>0</td></tr>
<tr><td>Smallest group qualifying for ‘family discount’</td><td>2</td><td>1</td></tr>
</table>
</td><td></td><td>(2)</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
4(a)	Any two from: • Indentation (1) • White space (1) • Descriptive identifiers (1) • Comments (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
4(b)(i)	One mark for each correct answer: **Programming construct** / **Line number(s)** A comment — 2 (1) An integer variable initialisation — 9/10/11 (1) Selection — 14-16 (1) Iteration — 13-18 (1)	• In this order only in last column of table • Award only blocking lines indicated (14, 16) and (13, 18) • Ignore examples	**(4)**

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>4(b)(ii)</td><td>One mark for each accurately shown sequence (initialisation) or complete pass through loop.

Example:

<table>
<tr><th>Length</th><th>Count</th><th>Index</th><th>Extended [index]</th><th>Marks</th></tr>
<tr><td>3</td><td>0</td><td>0</td><td></td><td>(1)</td></tr>
<tr><td></td><td></td><td></td><td>‘Y’</td><td rowspan="3">(1)</td></tr>
<tr><td></td><td>1</td><td></td><td></td></tr>
<tr><td></td><td></td><td>1</td><td></td></tr>
<tr><td></td><td></td><td></td><td>‘N’</td><td rowspan="2">(1)</td></tr>
<tr><td></td><td></td><td>2</td><td></td></tr>
<tr><td></td><td></td><td></td><td>‘Y’</td><td rowspan="3">(1)</td></tr>
<tr><td></td><td>2</td><td></td><td></td></tr>
<tr><td></td><td></td><td>3</td><td></td></tr>
</table>
</td><td>• Different formats of tracing should be awarded if synchronised change in variables is discernible
• Quote marks not required</td><td>(4)</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
5(a)	An explanation that makes reference to the following point: • This is a syntax error (1) Plus one from: • Because there is an error in using the grammar/rules of the programming language (1) • Because the word SET has been spelled incorrectly as TES/TES is not a recognised command word (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
5(b)	An error and a correction for each line: Line 10 • Error: The variable maximum doesn't get reset to the maximum number of visitors/the variable maximum is initialised too high (1) • Correction: SET maximum TO 0/SET maximum TO –*n* (1) Line 13: • Error: The for loop goes around too many times/the code goes past the end of the visitor array/off-by-one error on array length (1) • Correction: FOR index FROM 0 TO (length-1) DO (1)	• Do not penalise syntax errors as long as discernible • '-*n*' is any negative integer	**(4)**

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>5(c)</td><td>One mark for each correct row:

<table>
<tr><th>Item</th><th>Data type</th></tr>
<tr><td>Gender of individual staff member</td><td>char/character (1)</td></tr>
<tr><td>Whether an individual car is still under the manufacturer’s warranty</td><td>Boolean (1)</td></tr>
<tr><td>Mean number of hours needed to recharge the battery in each car</td><td>Real (1)</td></tr>
<tr><td>The number on the individual car</td><td>Integer (1)</td></tr>
</table>
</td><td></td><td>(4)</td></tr>
</table>

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>5(d)</td><td>
A diagram showing a data structure that includes:
• 2 or more rows (1)
• 8 columns (1)
• Homogenous realistic integer data, between 0 and 12 inclusive (1)

Example (3 marks):
<table>
<tr><th>ID</th><th>Mon</th><th>Tue</th><th>Wed</th><th>Thurs</th><th>Fri</th><th>Sat</th><th>Sun</th></tr>
<tr><td>3865</td><td>10</td><td>8</td><td>5</td><td>5</td><td>8</td><td>0</td><td>0</td></tr>
<tr><td>4722</td><td>0</td><td>0</td><td>8</td><td>8</td><td>8</td><td>6</td><td>6</td></tr>
</table>
Example (3 marks):
<table>
<tr><th>[0]</th><th>[1]</th><th>[2]</th><th>[3]</th><th>[4]</th><th>[5]</th><th>[6]</th><th>[7]</th></tr>
<tr><td>3865</td><td>10</td><td>8</td><td>5</td><td>5</td><td>8</td><td>0</td><td>0</td></tr>
<tr><td>4722</td><td>0</td><td>0</td><td>8</td><td>8</td><td>8</td><td>6</td><td>6</td></tr>
</table>
Example (2 marks):
<table>
<tr><th>Mon</th><th>Tue</th><th>Wed</th><th>Thurs</th><th>Fri</th><th>Sat</th><th>Sun</th></tr>
<tr><td>10</td><td>8</td><td>5</td><td>5</td><td>8</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>8</td><td>8</td><td>8</td><td>6</td><td>6</td></tr>
</table>
Example (2 marks):
<table>
<tr><th>ID</th><th>Mon</th><th>Tue</th><th>Wed</th><th>Thurs</th><th>Fri</th><th>Sat</th><th>Sun</th></tr>
<tr><td>3865</td><td>10</td><td>8</td><td>5</td><td>5</td><td>8</td><td>0</td><td>0</td></tr>
<tr><td>4722</td><td>0</td><td>0</td><td>8</td><td>8</td><td>8</td><td>6</td><td>6</td></tr>
</table>
Example (2 marks):
<table>
<tr><th>[0]</th><th>[1]</th><th>[2]</th><th>[3]</th><th>[4]</th><th>[5]</th><th>[6]</th><th>[7]</th></tr>
<tr><td>3865</td><td>10</td><td>8</td><td>5</td><td>5</td><td>8</td><td>0</td><td>0</td></tr>
<tr><td>4722</td><td>0</td><td>0</td><td>8</td><td>8</td><td>8</td><td>6</td><td>6</td></tr>
</table>
</td><td>
• Data must be homogenous (all the same type)
• At least two rows of data must be included, otherwise it's a 1-D array
• Column/row headings, if included, must be clearly distinguishable from the data
• Columns can be in any order
• Ignore subscripts if provided and clearly not part of the data
• If subscripts appear to be part of the data and they include characters, then the third mark cannot be awarded
</td><td>(3)</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
6(a)	One mark for the correct name: • Return value/return result/result (1)	• Not RETURN alone, which is in the question	**(1)**

Question Number	Answer	Additional Guidance	Mark
6(b)	An explanation that makes reference to the following points: • The variable, totalTime, on line 16 is a local variable, which exists only inside the function (1) • so updating it will not change the value of totalTime on line 2, which is a global variable (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
6(c)	An explanation that makes reference to the following points: • Executing the pseudo-code on line 17 will have no effect on TOTALTIME (1) • because the variable TOTALTIME has been defined on line 5 as a constant, which means its value cannot be changed during execution (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
6(d)	• One mark for each correct validation test (1) • One mark for each appropriate corresponding test data (1) The mark for test data identified must be linked with the associated data validation test. Example:	• Do not penalise syntax • Accept appropriate alternative values for the test data	**(4)**

Validation test	**Test data**
timeOn > 0	timeOn = 0
timeOff >= 0	timeOff = -3

Question Number	Answer	Additional Guidance	Mark
7	An amended procedure showing: • Parameter inserted (1) • Less than (<) test using the new parameter (1) • Descriptive variable name for the parameter relative to context (1) Example: 13 PROCEDURE wait (inSeconds) 14 BEGIN PROCEDURE 15 SET startTime TO getTime() # Gets the current time in seconds 16 SET newTime TO getTime () 17 WHILE (newTime - startTime < inSeconds) DO 18 SET newTime TO getTime () 19 END WHILE 20 END PROCEDURE	• Do not penalise syntax, if discernible	**(3)**

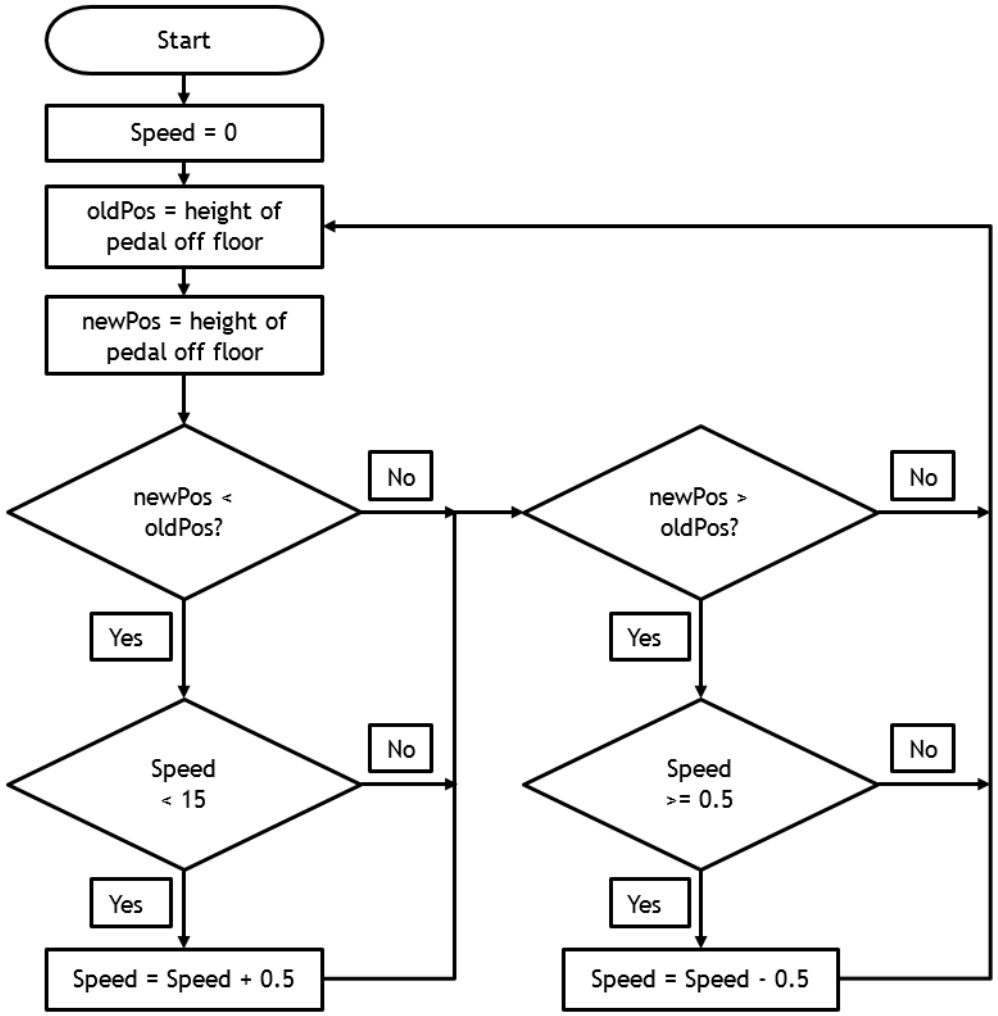
Question Number	Answer	Additional Guidance	Mark
8	An assessment that includes: • Using a 1-dimensional array is the most appropriate choice (1) **and four from:** • 20 variables requires 20 distinct names (1) • An array requires only one name (1) • 20 variables may require complex selection statements (1) • An array may require only one loop (1) • 20 variables are always needed even if fewer cars are in the race (1) • An array can be allocated to the exact number of cars (1)	• Do not award 'more complex to code' alone • Do award 'more complex to code because you have to keep track of more variable names'	**(5)**

Question Number	Answer	Additional Guidance	Mark
9(a)	An explanation that makes reference to the following points: • Sorting the file by postcode would make the searching more efficient (1) • because not every record has to be looked at to find the targeted postcodes (1) • since you can stop looking when you go past where your targeted postcode should be (1)	• Do not award generalities	**(3)**

Question Number	Answer	Additional Guidance	Mark
9(b)(i)	Identified issues that include: • it is unethical to ask for the ethnic origin of a person who will be driving a car (1) • Collected information can be directly tied to an identifiable individual (1)		**(2)**

Question Number	Answer	Additional Guidance	Mark
9(b)(ii)	Any one from: • Redistribution of the data without permission may be illegal (1) • Use of data for purposes other than which it is collected for may be illegal (1) • Organisations may have to ask permission before sharing personal data with a third party (1)	• Accept violates the Data Protection Act (DPA) 2003	**(1)**

Question Number	Answer	Additional Guidance	Mark
10(a)	A completed flow chart that includes: • Correct ellipse symbol for flow chart start (1) • Correct word 'Start' or 'Begin' in symbol (1) • Correct diamond symbol for decision (1) • Correct test 'newPos < oldPos' in symbol (1) • Correct label 'yes' on bottom arrow (1) • Correct label 'no' on right arrow (1)	• Symbol and contents are awarded independently	**(6)**

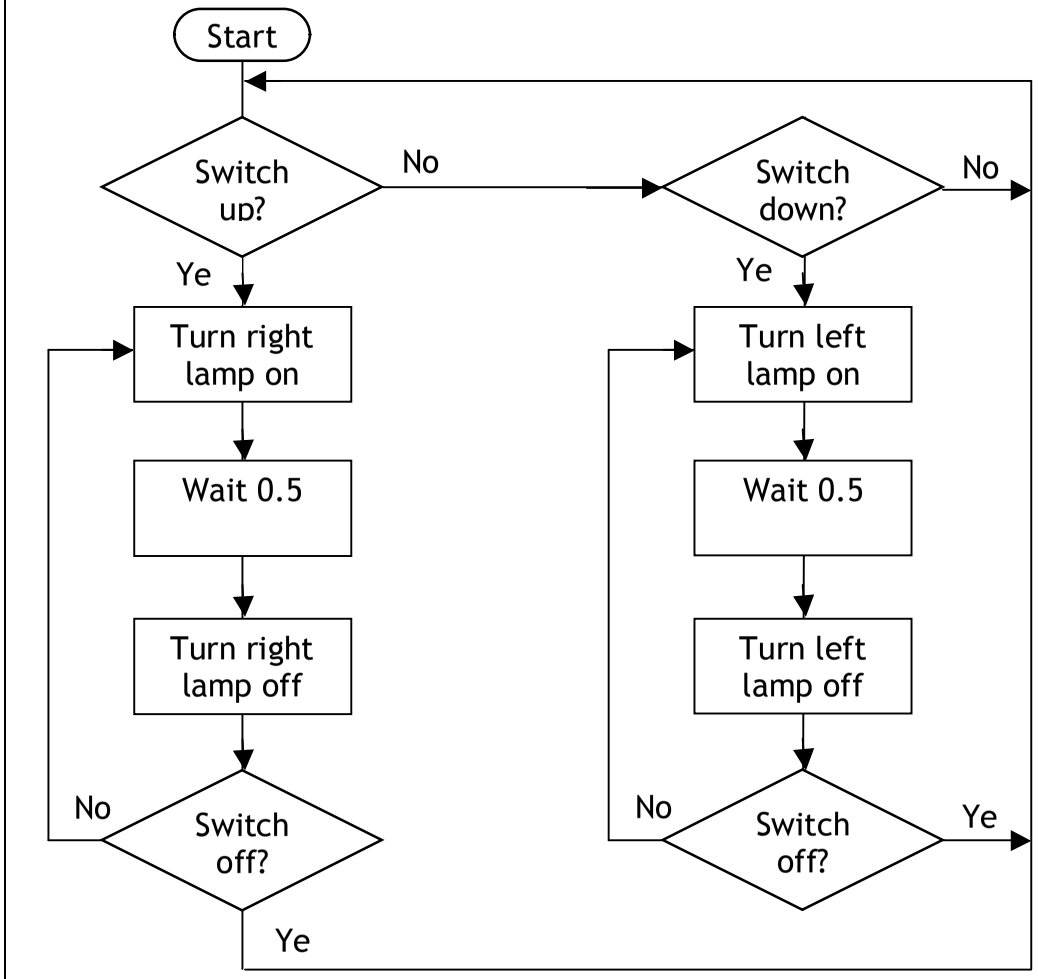
Question Number	Answer	Additional Guidance	Mark
10(a) (continued)			

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>10(b)</td><td>A flowchart that includes the following aspects:

<table>
<tr><th rowspan="2">Aspect of Solution</th><th colspan="4">Marks</th></tr>
<tr><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>Functionality</td><td>No awardable content</td><td>There are significant errors in logic, leading to an overall solution that is non-functional</td><td>There are minor errors in logic, leading to an overall solution that is not completely functional</td><td>There are no errors in logic, leading to an overall solution that is fully functional</td></tr>
<tr><td>Accuracy of Notation</td><td>No awardable content</td><td>Notation follows a broadly unrecognisable convention that is applied inconsistently, although aspects of it are discernible</td><td>Notation follows a recognisable convention which is broadly discernible but is applied inconsistently</td><td>Notation follows a recognisable convention and is applied consistently throughout</td></tr>
</table>

There are a maximum of 3 marks for functionality.
There are a maximum of 3 marks for accuracy of notation.
The marks for functionality and accuracy are awarded independently.

Example:</td><td>• Disregard use of ‘stop’ instead of loop as shown in example
• Do not penalise use of plain rectangle instead of input/output flow chart symbol, if discernible</td><td></td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
10(b) (Continued)			**(6)**

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>11</td><td>Pseudo code that has the following aspects:

<table>
<tr><th rowspan="2">Aspect of Solution</th><th></th><th colspan="3">Marks</th></tr>
<tr><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>Functionality</td><td>No awardable content</td><td>There are significant errors in logic, leading to an overall solution that is non-functional.</td><td>There are minor errors in logic, leading to an overall solution that is not completely functional.</td><td>There are no errors in logic, leading to an overall solution that is fully functional.</td></tr>
<tr><td>Accuracy of Notation</td><td>No awardable content</td><td>Notation follows a broadly unrecognisable convention that is applied inconsistently, although aspects of it are discernible.</td><td>Notation follows a recognisable convention which is broadly discernible but is applied inconsistently.</td><td>Notation follows a recognisable convention and is applied consistently throughout.</td></tr>
<tr><td>Efficiency, Appropriateness, and Accuracy of Solution</td><td>No awardable content</td><td>There are significant errors in the selection and accurate use of appropriate techniques.</td><td>Techniques have been selected and used with some accuracy, although the techniques may not be the most appropriate.</td><td>Techniques have been selected and used accurately and appropriately throughout to demonstrate an efficient solution.</td></tr>
</table>

There are a maximum of 3 marks for functionality.
There are a maximum of 3 marks for accuracy of notation.
There are a maximum of 3 marks for efficiency, appropriateness, and accuracy of solution.
Each row is awarded independently.</td><td></td><td>(9)</td></tr>
</table>

Pearson Edexcel Level 1/Level 2 GCSE (9–1)

Computer Science

Paper 3: Computer Science project

Sample assessment material for first teaching
September 2016
Project brief
Time: 20 hours

Paper Reference
1CP1/3A-3E

Do not return this booklet.

Instructions to teachers

- This paper is valid from November to May
- Students should use one of the following programming languages:
  – (1CP1/3A) Python
  – (1CP1/3B) Java
  – (1CP1/3C) C-derived languages
  – (1CP1/3D) Visual Basic.NET
  – (1CP1/3E) Pascal/Object Pascal.
- The project must be developed under teacher supervision
- Internet access is not allowed. However, students should be given offline help for the programming language they are using. This could be printed.
- The materials submitted for assessment must include:
  – a report on the development of the software
  – the program
  – completed record and authentication sheet – available on the Pearson website.

Information for students

- The total mark for this component is 60.
- The work you submit must be your own work.
- You are reminded of the importance of clear and orderly presentation of your work.

Turn over ▶

S49770A

PEARSON

Weather station

A team of university weather researchers has permission to use computer-controlled drones to monitor the climate of the local area. There are five drones in total that fly over a local area. Each drone records the day and time a reading is taken, temperature, wind speed, and location. This data is sent back to the researchers at the local weather station. The data is stored as a text file.

Sample data file for Drone 1

Day	Time	Temperature (°C)	Wind speed (mph)	Location (six-figure grid reference)
Monday	08:00	6.2	7	207834
Monday	08:15	6.1	7	205834
Monday	08:30	6.8	6	204835
Monday	09:00	7.3	6	204837

Tuesday	08:00	6.3	6	204839
Tuesday	08:15	6.8	7	203840
Tuesday	08:30	7.2	5	202841
Tuesday	09:00	7.5	6	204837

Further data is kept for each day for the rest of the week.

The researchers need to be able to access the temperature and wind-speed data for different time periods and locations. They want to use this data to calculate the average temperature, the average wind speed, and the temperature range over different time periods and locations.

The researchers need a computer program to be written to allow for this to happen. The program will enable them to read and interpret data to provide an accurate understanding of weather conditions in the local area over different time periods.

Your task is to analyse this problem and to design, implement, test and evaluate the solution.

The written report

Your written report is evidence of the work you have done. The report should contain the details mentioned below for each stage of the task. The marks available for each stage are also shown below.

Stage 1 Analysis (6 marks)

You should include a short introduction summarising the overall problem. List each of the requirements of the problem in the form of text or detailed bullet points.

The problem should be broken down into sub-problems. You should give a short description of what each of the sub-problems will do and a short explanation of why you broke down the problem in this way.

Stage 2 Design (18 marks in total)

The algorithm (12 marks)

This section of your report should begin with your algorithm. You should show how the sub-problems link together (if appropriate). You should include inputs, processes and outputs, and the programming constructs that you will use when you produce your program. Finally, you should show how your algorithm meets the requirements.

Test strategy and initial test plan (6 marks)

This section of your report should include your test strategy and the initial test plan that must be completed before you start the implementation stage. It is recommended that you use the following table to produce your initial test plan. Your teacher must see the test plan before you begin to develop your program code.

Test no	Purpose of the test	Test data	Expected result

Stage 3 Implementation (24 marks)

Copy and paste your program code into this part of your report. Include screenshots to show examples of how you debugged your program.

Stage 4 Testing, Refining and Evaluation (12 marks)

Your completed test table should go in this section of your report. You may need to change the program in response to the results that you get from your test plan and add more tests to your table.

Test no	Purpose of the test	Test data	Expected result	Actual result	Action needed/ comments

You might have thought of **refinements** that you added to your program as you were implementing it. These should be documented as additions to your design and described in your program using comments. You should describe the refinements you identified and explain why they were needed. If you implemented these refinements and tested them, then the tests should be included in your table.

Finally, you should evaluate your project by explaining how your project meets each of the requirements that you identified in the analysis stage of your project.

Pearson Edexcel Level 1/Level 2 GCSE (9–1)

Computer Science

Papers 1, 2 and 3

Sample assessment material for first teaching September 2016
Pseudo-code command set

Paper Reference
1CP1/01
1CP1/02
1CP1/3A-3E

Do not return this resource booklet with the question paper.

Turn over ▶

S50557A

PEARSON

Pseudo-code command set

Questions in the written examination that involve code will use this pseudo-code 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 pseudo-code, the < > indicates where expressions or values need to be supplied. The < > symbols are not part of the pseudo-code.

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**

WHILE <condition> DO
  <command>
END WHILE
	Pre-conditioned loop. Executes <command> whilst <condition> is true.	WHILE Flag = 0 DO
		  SEND 'All well' TO DISPLAY
		END WHILE

REPEAT <command> UNTIL <expression>	Post-conditioned loop. Executes <command> until <condition> is true. The loop must execute at least once.	REPEAT SET Go TO Go + 1 UNTIL Go = 10
REPEAT <expression> TIMES <command> END REPEAT	Count controlled loop. The number of times <command> is executed is determined by the expression.	REPEAT 100-Number TIMES SEND '*' TO DISPLAY END REPEAT

FOR <id> FROM <expression> TO
<expression> DO
  <command>
END FOR
	Count controlled loop. Executes <command> a fixed number of times.	FOR Index FROM 1 TO 10 DO
		  SEND ArrayNumbers[Index] TO DISPLAY
		END FOR

FOR <id> FROM <expression> TO
<expression> STEP <expression> DO
  <command>
END FOR
	Count controlled loop using a step.	FOR Index FROM 1 TO 500 STEP 25 DO
		  SEND Index TO DISPLAY
		END FOR

FOR EACH <id> FROM <expression> DO <command> END FOREACH	Count controlled loop. Executes for each element of an array.	SET WordsArray TO ['The', 'Sky', 'is', 'grey'] SET Sentence to '' FOR EACH Word FROM WordsUArray DO SET Sentence TO Sentence & Word & ' '

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

<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.

edexcel

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

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

ISBN 978-1-4469-3184-4