Mark Scheme (Results)

November 2020

Pearson Edexcel GCSE
In Computer Science (1CP1/02)
Paper 2: Application of Computational Thinking

Edexcel and BTEC Qualifications

Edexcel and BTEC qualifications are awarded by Pearson, the UK's largest awarding body. We provide a wide range of qualifications including academic, vocational, occupational and specific programmes for employers. For further information visit our qualifications websites at www.edexcel.com or www.btec.co.uk. Alternatively, you can get in touch with us using the details on our contact us page at www.edexcel.com/contactus.

Pearson: helping people progress, everywhere

Pearson aspires to be the world's leading learning company. Our aim is to help everyone progress in their lives through education. We believe in every kind of learning, for all kinds of people, wherever they are in the world. We've been involved in education for over 150 years, and by working across 70 countries, in 100 languages, we have built an international reputation for our commitment to high standards and raising achievement through innovation in education. Find out more about how we can help you and your students at: www.pearson.com/uk

November 2020
Publications Code 1CP1_02_2011_MS

General Marking Guidance

- All candidates must receive the same treatment. Examiners must mark the first candidate in exactly the same way as they mark the last.
- 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.
- There is no ceiling on achievement. All marks on the mark scheme should be used appropriately.
- 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 may be limited.
- When examiners are in doubt regarding the application of the mark scheme to a candidate's response, the team leader must be consulted.
- Crossed out work should be marked UNLESS the candidate has replaced it with an alternative response.

Question Number	Answer	Additional Guidance	Mark
1(a)(i)	Any two from • departureTime (1) • flightNumber (1) • destination (1) • gateNumber (1) • flightStatus (1) • currentTime (1)	• Accept any equivalent names that are sensible in the context • Accept variable names with spaces	**2**

Question Number	Answer	Additional Guidance	Mark
1(a)(ii)	• In order to relate the name (1) of the variable to its job/role / the data stored (1) • So that it is easier to read the code (1) and follow the programmer's logic (1) / enables multiple programmers to work on the code (1) • So that it is easier to understand (1) how the program works (1) / maintain the program over time (1)	•	**2**

Question Number	Answer	Additional Guidance	Mark
1(a)(iii)	Any one from • time / getTime / currentTime (1) • format (1)	•	**1**

	• string manipulation / concatenation (1) • sort / search • print / display (1)		

Question Number	Answer	Additional Guidance	Mark
1(b)	• (basic_allowance / 30 * plane_type) (1) • + weather_condition (1) Examples: • (basic allowance / 30 * plane type) + weather condition • plane_type * 30 + weather_condition	• Ensure that expression follows the BIDMAS rules of precedence • Allow if basic allowance is replaced by 30	**2**

Question Number	Answer	Additional Guidance	Mark
2(a)	Sequence (1)		**1**

Question Number	Answer	Additional Guidance	Mark
2(b)	One mark for each item in the correct cell • 75 (1) • 25 (1) • 55 (1)		**3**

Question Number	Answer	Additional Guidance	Mark
2(c)	One mark for each item in the correct cell • 4 / 5 / 6 (1) • Any number <= 0 (1)		**2**

Question Number	Answer	Additional Guidance	Mark
3(a)	SENSOR_W (1)	Ignore case and spacing	**1**

Question Number	Answer	Additional Guidance	Mark
3(b)	• The algorithm would allow for two operations to be carried out at the same time (1)		**1**

Question Number	Answer	Additional Guidance	Mark

3(c)	IF (soapRequest) = True THEN
	SET soapStatus TO “ON”
	ELSE (1)
	SET soapStatus TO “OFF” (1)
	END IF
		ELSEIF not accepted	**2**

Question Number	Answer	Additional Guidance	Mark
3(d)	One mark for exclusive use of AND (1) One mark for correct placement of NOT (1) • NOT S AND NOT W AND NOT D Examples: • NOT S AND W AND D (1 mark for AND twice) • NOT (S AND W OR D) (0 marks)		**2**

Question Number	Answer	Additional Guidance	Mark
4(a)(i)	Record (1)	Accept correct language specific structures, e.g. list, dictionary	**1**

Question Number	Answer	Additional Guidance	Mark
4(a)(ii)	It cannot store mixed data types (1)		**1**

Question Number	Answer	Additional Guidance	Mark
4(b)	• Open (1) • Close (1)	Accept Read (1), Write(1) and Amend(1)	**2**

Question Number	Answer	Additional Guidance	Mark
4(c)(i)	Run-time error (1)		**1**

Question Number	Answer	Additional Guidance	Mark
4(c)(ii)	Any two from:		**2**

	• File does not exist (1) • Incorrect filename (1) • Incorrect path (1)		

Question Number	Answer	Additional Guidance	Mark
4(d)	An explanation such as: • More efficient to use a loop / using a loop saves times (1) because • it is not necessary to write out the commands multiple times (1) • the same calculations need to be carried out for each employee / the same process needs to be repeated for each employee (1)		**2**

Question Number	Answer	Additional Guidance	Mark
5(a)	One mark for each item in the correct cell. • Payment (1) • Calculate expiry time / Addition / current time + 2 (1) • Ticket (1)		**3**

Question Number	Answer	Additional Guidance	Mark
5(b)	One mark for each item in the correct cell. • Integer (1) • String (1) • Real / Float / Double (1)		**3**

Question Number	Answer	Additional Guidance	Mark
5(c)(i)	An explanation such as: • Using normal division results in real numbers (1) which when rounded to two decimal places could result in errors / pence that don't add up to 100 / the machine might give incorrect amount of change (1) • Amount has been converted into whole numbers of pence therefore the calculations need to produce a remainder in whole numbers of		**2**

	pence (1) and the correct number of notes and coins (1)		

Question Number	Answer	Additional Guidance	Mark
5(c)(ii)	One mark for each part of the logic illustrated in the overall flow		**5**

pence	tens	fives	ones	fiftyP	twentyP	tenP	fiveP	
1755	1							(1)
755		1						(1)
255			2					
55				1				(1)
5					0			
5						0		(1)
5							1	

0								(1)

Question Number	Answer	Additional Guidance	Mark
6(a)	Flowcharts • To communicate algorithms/ideas (1) in a graphical and understandable format (1) / at a high level of abstraction (1) Pseudocode • To allow focus on the logic of the solution (1) without concern for the syntax of a specific programming language (1) • To provide detail at a lower level of abstraction (1) which is close to actual code (1).		**4**

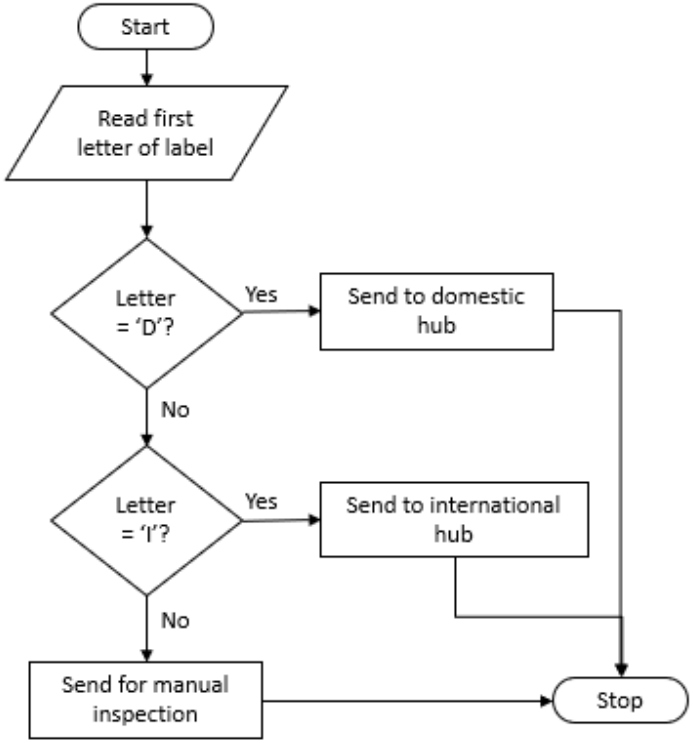
Question Number	Answer	Additional Guidance	Mark
6(b)	Line 10: Error • currWeight = 30 only finds items exactly of weight 30 as too heavy (1) Correction • if (currWeight > 30 / >= 30)(1) Line 12: Error • Adds overweight item to total, which is not right because item can't travel (1) Correction • Replace with nothing / add a comment to the front / line needs to be deleted (1)	• Award any replacement for line 12 that indicates the line should not be executed • Do not award an empty cell as equivalent of deleting line	**4**

Question Number	Answer	Additional Guidance	Mark
7(a)(i)	Look up (1) table	• Do not penalise syntax • Accept appropriate alternative values for the test data	**1**

Question Number	Answer	Additional Guidance	Mark
7(a)(ii)	One mark for values 2 and 4 (1) One mark for slice indicator (1) One mark for format - label [ ] (1) • label[2:4] / label(2,4) / label[2;4]	• Slice indicator cannot be arithmetic operator • Accept values 2 and 5	**3**

Question Number	Answer	Additional Guidance	Mark
7(b)	There is a maximum of 3 marks for functionality. There is a maximum of 3 marks for accuracy of notation for a relevant solution. The marks for functionality and accuracy are awarded independently. Example:	• Use of output box for 'pushing' is acceptable.	**6**

Aspect of Solution	**Marks**			
	0	**1**	**2**	**3**
Functionality	No rewardable content	There are significant errors in logic, leading to an overall solution that is non-functional	There are minor errors in logic, leading to an overall solution that is not completely functional	There are no errors in logic, leading to an overall solution that is fully functional
Accuracy of notation	No rewardable content	Notation follows a broadly unrecognisable convention that is applied inconsistently, although aspects of it are discernible	Notation follows a recognisable convention which is broadly discernible but is applied inconsistently	Notation follows a recognisable convention and is applied consistently throughout


Start
Read first letter of label
Letter = 'D'?
Yes
Send to domestic hub
No
Letter = 'I'?
Yes
Send to international hub
No
Send for manual inspection
Stop

Question Number	Answer	Additional Guidance	Mark
8(a)	One from, 2,4,15,22		**1**

Question Number	Answer	Additional Guidance	Mark
8(b)	**Line 9** FUNCTION**(1)** planeQueue (pRunway, pFlight) both parameters (2) / FUNCTION**(1)** planeQueue (pFlight)one parameter (1) **Line 35** planeQueue **(1)**(runway, flightID) (1) matching order in call (1)		**6**

Question Number	Answer	Additional Guidance	Mark
8(c)	It returns (1) a result to the caller		**1**

Question Number	Answer	Additional Guidance	Mark
8(d)	A comparison to include four from: • Line 12 is a local variable (1), which only exists in memory when the function is called (1) memory is deallocated / becomes available for re-use when the function exits (1) • Line 6 is a global variable (1), which exists in memory for the entire life of the program (1) / is accessible from anywhere in the program (1)		**4**

Question Number	Answer	Additional Guidance	Mark
9	Indicative content: • Assignment of found Boolean • Assignment of index • While loop with **two** conditions, with use of length function • Selection statement • Increment count • Set terminating condition • Else increment index • End loop	Hard-coding length of array to 8 reduces the efficiency of the algorithm and does not meet one of the requirements set out in the scenario Using a 'for' loop processes every item in the unsorted list, which is not necessary. Response should use a 'while' loop and stop processing when found. Hard-coding of values for inBarrier is acceptable	**9**

		Some languages may need a dimension for the LENGTH function.	

Aspect of Solution	**Marks**			
	0	**1**	**2**	**3**
Functionality	No rewardable content	There are significant errors in logic, leading to an overall solution that is non-functional	There are minor errors in logic, leading to an overall solution that is not completely functional	There are no errors in logic, leading to an overall solution that is fully functional
Accuracy of notation	No rewardable content	Notation follows a broadly unrecognisable convention that is applied inconsistently, although aspects of it are discernible	Notation follows a recognisable convention which is broadly discernible but is applied inconsistently	Notation follows a recognisable convention and is applied consistently throughout
Efficiency, Appropriateness, and Accuracy of Solution	No rewardable content	There are significant errors in the selection and accurate use of appropriate techniques.	Techniques have been selected and used with some accuracy, although the techniques may not be the most appropriate.	Techniques have been selected and used accurately and appropriately throughout to demonstrate an efficient solution.
There is a maximum of 3 marks for functionality. There is a maximum of 3 marks for accuracy of notation. There is a maximum of 3 marks for efficiency, appropriateness, and accuracy of solution. Each row is awarded independently.				

```
1
2  ARRAY counts
3  SET counts TO [[2, 0], [8, 0], [5, 0], [4, 0], [1, 0], [3, 0], [6, 0], [7, 0]]
4  INTEGER inBarrier
5
6  BOOLEAN found
7  INTEGER index
8  SET found TO False
9  SET index TO 0
10
11 WHILE (NOT found) AND (index < LENGTH (counts)) DO
12     IF (counts[index][0] = inBarrier) THEN
13         SET counts[index][1] TO counts[index][1] + 1
14         SET found TO True
15     ELSE
16         SET index TO index + 1
17     END IF
18 END WHILE
19
```