OCR
Oxford Cambridge and RSA

Practice Paper

GCSE Computer Science
J276 / 02 Computational thinking, algorithms and programming

MARK SCHEME

Duration: 1 hour 30 minutes

MAXIMUM MARK 80

Version: FINAL
Last updated: 18/12/17

(FOR OFFICE USE ONLY)

MARKING INSTRUCTIONS

PREPARATION FOR MARKING
SCORIS

1. Make sure that you have accessed and completed the relevant training packages for on-screen marking: *scoris assessor Online Training*; *OCR Essential Guide to Marking*.

2. Make sure that you have read and understood the mark scheme and the question paper for this unit. These are posted on the RM Cambridge Assessment Support Portal http://www.rm.com/support/ca

3. Log-in to scoris and mark the **required number** of practice responses ("scripts") and the **number of required** standardisation responses.

   YOU MUST MARK 10 PRACTICE AND 10 STANDARDISATION RESPONSES BEFORE YOU CAN BE APPROVED TO MARK LIVE SCRIPTS.

MARKING

1. Mark strictly to the mark scheme.

2. Marks awarded must relate directly to the marking criteria.

3. The schedule of dates is very important. It is essential that you meet the scoris 50% and 100% (traditional 40% Batch 1 and 100% Batch 2) deadlines. If you experience problems, you must contact your Team Leader (Supervisor) without delay.

4. If you are in any doubt about applying the mark scheme, consult your Team Leader by telephone or the scoris messaging system, or by email.

5. Work crossed out:
   a. where a candidate crosses out an answer and provides an alternative response, the crossed out response is not marked and gains no marks
   b. if a candidate crosses out an answer to a whole question and makes no second attempt, and if the inclusion of the answer does not cause a rubric infringement, the assessor should attempt to mark the crossed out answer and award marks appropriately.

6. Always check the pages (and additional objects if present) at the end of the response in case any answers have been continued there. If the candidate has continued an answer there then add a tick to confirm that the work has been seen.

7. There is a NR (No Response) option. Award NR (No Response)
   - if there is nothing written at all in the answer space
   - OR if there is a comment which does not in any way relate to the question (eg 'can't do', 'don't know')
   - OR if there is a mark (eg a dash, a question mark) which isn't an attempt at the question

   Note: Award 0 marks – for an attempt that earns no credit (including copying out the question)

8. The scoris **comments box** is used by your team leader to explain the marking of the practice responses. Please refer to these comments when checking your practice responses. **Do not use the comments box for any other reason.**
   If you have any questions or comments for your team leader, use the phone, the scoris messaging system, or e-mail.

9. Assistant Examiners will send a brief report on the performance of candidates to their Team Leader (Supervisor) via email by the end of the marking period. The report should contain notes on particular strengths displayed as well as common errors or weaknesses. Constructive criticism of the question paper/mark scheme is also appreciated.

10. For answers marked by levels of response: Not applicable in J276/02
    a. **To determine the level** – start at the highest level and work down until you reach the level that matches the answer
    b. **To determine the mark within the level**, consider the following:

Descriptor	**Award mark**
On the borderline of this level and the one below	At bottom of level
Just enough achievement on balance for this level	Above bottom and either below middle or at middle of level (depending on number of marks available)
Meets the criteria but with some slight inconsistency	Above middle and either below top of level or at middle of level (depending on number of marks available)
Consistently meets the criteria for this level	At top of level

11. **Annotations**

Annotation	Meaning
BP	Blank Page – this annotation **must** be used on all blank pages within an answer booklet (structured or unstructured) and on each page of an additional object where there is no candidate response.
^	Omission mark
BOD	Benefit of doubt
C	Subordinate clause/Consequential error
	Cross
E	Expansion of a point
FT	Follow through
NAQ	Not answered question
NBOD	Benefit of doubt not given
P	Point being made
REP	Repeat
	Slash
	Tick

Question			Answer	Mark	Guidance
1	(a)	(i)	1 mark per bullet to max 2 • Obscuring of unnecessary detail • So the main processes/data can be more easily understood/focused on	2	Do not allow simply "removing detail" or "ignoring sections", etc. For second bullet point, allow answers relating to concentrating on "the important bits".
1	(a)	(ii)	1 mark per bullet to max 2 • To reduce complexity of the student representation… • …to therefore make the implementation simpler / more efficient / quicker	2	
1	(a)	(iii)	1 mark per bullet to max 2. Mark in pairs. E.g. • Teacher would hold details of marks/grades… • …Kitchen system would not need academic information • Kitchen may hold details of food allergies/school meal choices/payment information… • …Teacher would not need information about dietary requirements	2	Allow any sensible answer where a reasonable difference between the two representations is clearly stated, with one mark awarded for each side of the explanation.

1	(b)	(i)	1 mark per bullet to max 4.
			• Function defined called newbalance…
			• ...that allows **two** parameters to be passed in
			• Correct balance calculated using these parameters
			• Value returned
				4	Example pseudocode
					`Function newbalance(balance, cost)`
					`  newcost = balance – cost`
					`  return newcost`
					`End Function`
					Bullet points 3 and 4 may be combined into one line (eg `return balance - cost`). Parameters can have any sensible identifiers. Allow any sensible attempt to return a value for bullet point 4, even if this is not the correct new balance.

1	**(b)**	**(ii)**	1 mark per bullet to max 2. • Data type : Float / Real • Justification : may contain a decimal part / value may include pence as well as pounds / may not a whole number.	2	Allow decimal / currency / single / double for data type.
1	**(c)**	**(i)**	1 mark per bullet to max 6. Max 3 marks per error. • Line number : 03 • Error type : logic error • Correct pseudocode : `if mealcost <= balance` • Line number : 08 • Error type : syntax error • Correct pseudocode : `print("Balance too low")`	6	Do not award error type of corrected pseudocode if wrong line number is identified. For error on line 3, allow any sensible correction which means that a student with the exact balance can pay for a meal (e.g. accept `if mealcost < balance or mealcost == balance` and equivalents)
1	**(c)**	**(ii)**	1 mark per bullet to max 3. • Changing a variable from one data type to another • Used on line 06 / used to concatenate (join) the balance to a printed message • Balance variable cast/converted to a string	*3*	Allow answers for first and third bullet point that discuss the temporary nature of this, i.e. that balance is not permentantly converted to a string but that a string version is created and used at runtime.
2	**(a)**		1 mark per bullet to max 2. • x = 20 • y = 20	2	
2	**(b)**		1 mark per bullet to max 3. • temp = y • y = x • x = temp	3	Note temporary variable can be given any sensible identifier, but to award the 3rd bullet point it must be the same identifier as used in the 1st bullet point. Award marks for any innovative solution which correctly swaps over the values as long as no other code is modified.

<table>
<tr>
<td>2</td><td>(c)</td><td></td>
<td>1 mark per correct row to max 3.
<table>
<tr><th>Construct</th><th>Is used in (b)</th><th>Is not used in (b)</th></tr>
<tr><td>Sequence</td><td>✓</td><td></td></tr>
<tr><td>Selection</td><td></td><td>✓</td></tr>
<tr><td>Iteration</td><td></td><td>✓</td></tr>
</table>
</td>
<td>3</td>
<td>Allow marks other than ticks (e.g. crosses, etc). Do not award mark where both or neither box in a row is ticked.</td>
</tr>
<tr>
<td>3</td><td>(a)</td><td>(i)</td>
<td>1 mark per bullet to max 2.
• Finds the quotient / uses floor division / integer division / division giving an integer answer / division ignoring the remainder
• …of the input value / temp / time in minutes divided by 60
• Finds the number of full hours in a given number of minutes</td>
<td>2</td>
<td>First bullet point awarded for identifying that this finds the quotient or uses floor division (or equivalent term / explanation). DO NOT accept simply “division”

Second bullet point awarded for identification of values used and in correct order i.e. input value divided by 60 (from scenario).</td>
</tr>
<tr>
<td>3</td><td>(a)</td><td>(ii)</td>
<td>1 mark per bullet to max 2.
• Finds the modulus / remainder (after division)
• …of the input value / temp / time in minutes divided by 60
• Finds the minute component of converting a time in minutes into hours and minutes.</td>
<td>2</td>
<td>First bullet point awarded for identifying that this finds the remainder (or equivalent). DO not accept simply “MOD” as this is in the question.

Second bullet point awarded for identification of values used and in correct order i.e. input value divided by 60 (from scenario).</td>
</tr>
<tr>
<td>3</td><td>(b)</td><td></td>
<td>1 mark per bullet to max 2.
• <code>MOD</code>
• <code>else</code></td>
<td>2</td>
<td>Ignore capitalisation.
Allow other keywords / symbols to mean MOD in various languages (eg % in Python)</td>
</tr>
</table>

4	**(a)**	**(i)**	• 0005	1	Correct answer only.
4	**(a)**	**(ii)**	• 0001, 0006	1	Correct answer only.
4	**(b)**		1 mark per bullet to max 3. • `SELECT Name, HighScore` • `FROM Gymnasts` • `WHERE Competition = "Vault"`	3	Ignore capitalisation. Field names for 1st bullet point must have correct spellings and no spaces (e.g. do NOT accept High Score). "Vault" for 3rd bullet point must have correct spelling and be contained in quote marks (single or double). Allow == for comparison on 3rd bullet point. Allow use of LIKE if this is equivalent (i.e. no wildcards).
5	**(a)**	**(i)**	• 3	1	Correct answer only.
5	**(a)**	**(ii)**	1 mark per bullet to max 2. • Maximum number of colours : 16 • Reason : 2^4 is 16 / 0000 to 1111 gives 16 possible colours	2	Accept any sensible explanation of the reason why 4 bits gives 16 possible colours.
5	**(b)**		1 mark per bullet to max 4, mark in pairs. Max 2 per pair. e.g. • Reduce the number of bits used per pixel (to 2 bits)… • …so that the amount of data stored per pixel is smaller • Reduce the dimensions / <u>physical</u> size of the image // reduce height // reduce width… • …so data about fewer pixels is stored • Use <u>lossy</u> compression • … information regarding the pictures is removed / picture quality is reduced / number of colours reduced • Use <u>lossless</u> compression	4	Accept discussion about compression in general for one way of reducing the file size. Discussion regarding specifically lossy and lossless compression could cover both ways.

			• … patterns are identified / an algorithm is applied.		
5	**(c)**		1 mark per bullet to max 2. e.g. • Height / width / resolution / dimensions of image • Colour depth • Resolution • Geotag / location data • Details of hardware/software used • Keyword data • Licensing / usage rights • Date / time created	2	Accept other sensible answers that could feasibly be stored as meta data. 5b – bp3 - Can we allow dimensions rather than physical size?
6	**(a)**		• 8F	2	1 mark for left most nibble = 8, 1 mark for right most nibble = F.
6	**(b)**		• 1110 0010	2	1 mark per nibble
6	**(c)**		1 mark per bullet to max 2. • Computers consist of numerous switches / transistors • 1 is represented as switch / transistor that is on / open // 0 is represented as switch / transistor that is off / closed.	2	Accept equivalent answers relating to storage where alternatives to transistors are used (eg magnetic hard disk, with small sections of the platter magnetised or not magnetised)
6	**(d)**		1 mark per bullet to max 2. • Easy / fast to convert to/from binary… • …4 bits represented by one hex character • Faster / more reliable to communicate / enter / write down / read.	2	Do not accept “hex is smaller” without further clarification of why this is a benefit.

7	**(a)**		1 mark per bullet to max 4.
			• Allows tracking code to be input.
			• Checks if tracking code begins with P or Q
			• Checks if tracking code is 8 characters in length
			• Prints ”VALID” if both of the above are true and invalid if either or both are false.
				4	Example pseudocode
					
					`tcode = input("enter a tracking code")`
					`first = tcode.substring(1,1)`
					`len = tcode.length()`
					`if (first == "P" or first == "Q") and len == 8 then`

					`print ("VALID")`
					`else`
					`print ("INVALID")`
					`end if`
					
					Award bullet points 2 and 3 for checks on the individual requirements even if they do not work together.
					Only award bullet point 4 if VALID or INVALID is printed to meet the requirements of the algorithm.

7	**(b)**		1 mark per bullet to max 3. • Additional digit / by example (eg last digit) • A value is worked out (mathematically) from the other digits… • …which is compared to the check digit • Used to verify that tracking code has been entered correctly.	3	
7	**(c)**		1 mark per bullet to max 2. • Finding defects / problems… • …so that these can be fixed before use. • Ensuring that calculations are done correctly… • …so that that baggage is priced correctly • Ensuring that end results meet user requirements… • …so they system gains user / customer confidence	2	Award any two relevant bullet points. Candidates can gain two marks from either two separate points or one point plus an expansion.
7	**(d)**	**(i)**	• Testing <u>during</u> development	1	
7	**(d)**	**(ii)**	• Testing <u>after</u> (the majority of) development has been completed / testing before release.	1	
7	**(e)**		1 mark per bullet to max 6, mark in pairs. E.g. • Test data : 10kg (any value between 0-24) • Purpose : To test normal values for free luggage	6	Test data given must be specific data (e.g. “10kg”) and NOT a description of data (e.g. do not allow “a value between 0 and 20 kg”)

			• Test data : 25kg • Purpose : To test borderline free baggage • Test data : 30kg (any value between 26-49) • Purpose : To test normal values for paid for luggage • Test data : 49kg / 49.9kg • Purpose : To test borderline paid for allowed luggage • Test data : 50kg • Purpose : To test borderline disallowed luggage • Test data : 60kg (any value over 50kg) • Purpose : To test disallowed free baggage • Test data : Bananas (any textual / non-numeric data) • Purpose : To test erroneous / non-numerical data is rejected. • Test data : -1kg (any value below 0kg) • Purpose : To test erroneous negative values are rejected		

7	**(f)**		1 mark per bullet to max 6.
			
			• Inputs weight from user
			• Checks if weight is bigger than 50…
			• …and displays error message if so.
			• Calculates price correctly as (weight-25) x 10 or equivalent
			• Checks if weight is less than or equal to 50…
			• … and prints out the price if so.
				6	Example pseudocode
					
					`weight = input("enter weight of baggage")`
					`if weight > 50 then`
					`  print ("Error")`
					`else`
					`  price = (weight-25) * 10`
					`  print (price)`
					`end if`
					
					Only award 3rd bullet point if 2nd bullet point is correct.
					Only award 6th bullet point if 5th bullet point is correct.
					2nd and 4th bullet point may be done using IF and ELSE

					as shown in example above. 4th and 6th bullet point may be combined into one statement. Note – above pseudocode is an example, any answer showing understanding that is logically sound should be awarded marks as pre the criteria. Flowcharts or pseudocode are acceptable.