Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/22**

Paper 2 **May/June 2019**

MARK SCHEME

Maximum Mark: 50

Published

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the May/June 2019 series for most Cambridge IGCSE™, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

This syllabus is regulated for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

© UCLES 2019

Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptors for a question. Each question paper and mark scheme will also comply with these marking principles.

GENERIC MARKING PRINCIPLE 1:

Marks must be awarded in line with:

- the specific content of the mark scheme or the generic level descriptors for the question
- the specific skills defined in the mark scheme or in the generic level descriptors for the question
- the standard of response required by a candidate as exemplified by the standardisation scripts.

GENERIC MARKING PRINCIPLE 2:

Marks awarded are always **whole marks** (not half marks, or other fractions).

GENERIC MARKING PRINCIPLE 3:

Marks must be awarded **positively**:

- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can do
- marks are not deducted for errors
- marks are not deducted for omissions
- answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous.

GENERIC MARKING PRINCIPLE 4:

Rules must be applied consistently e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

GENERIC MARKING PRINCIPLE 5:

Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen).

GENERIC MARKING PRINCIPLE 6:

Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

© UCLES 2019

Question	Answer	Marks
	Section A	
1(a)	4 marks for: Data Structure(s) **max 2** • Arrays • Variable(s) for // data entry such as reserve price // index / constant for minimum number of items Further description **max 3** • Data type(s) one or more • Use(s) one or more • Name(s) one or more e.g. `Item_Number`, `NumberBids`, `Description`, `Reserve Price` • Sample data for appropriate arrays e.g. 1234, 0, vase, 20.00 // Sample data for variable or constant e.g. 10 1 mark for: • At least four appropriate named arrays with sample data	**5**
1(b)	• Keep a counter/number • Add one every time a new item is added or • Keep a list of numbers used • Using a loop check number is not already in the list before a new item number is added	**2**

Question	Answer	Marks

1(c)	**Five** from:
	MP1 Enter Buyer Number
	MP2 Check if buyer number valid
	MP3 Enter new bid
	MP4 Check if new bid less than or equal to current highest bid …
	MP5 … if so reject
	MP6 … Otherwise store the bid entered as new highest bid
	MP7 Increase number of bids by one
	MP8 Store Buyer Number
	
	Sample answer
	
	`REPEAT`
	`  PRINT "Enter Buyer Number"`
	`  INPUT BuyerNumber`
	`UNTIL BuyerNumber >= BuyerLow and BuyerNumber <= BuyerHigh`
	`REPEAT`
	`  PRINT "Enter Bid -99 to exit"`
	`  INPUT Bid`
	`  IF Bid > HighestBid(item)`
	`  THEN`
	`  HighestBid(item) ← Bid`
	`  NumberBids(item) ← NumberBids(item) + 1`
	`  ENDIF`
	`UNTIL Bid > HighestBid(item) OR Bid = -99`
	`IF Bid <> -99`
	`  THEN`
	`  BuyerItem(item) ← BuyerNumber`
	`ENDIF`
	
	There are many possible correct answers, this is an example only.
		5

1(d)	**Three** from: MP1 Using index number of item, to find if any bids for item exist MP2 Using IF/UNTIL/WHILE (statement) to compare new bid with highest/latest bid recorded MP3 Use an assignment (statement) to replace current highest bid, if new bid greater than current highest bid MP4 … else reject bid // using a (REPEAT/WHILE) loop ask for bid to be re-entered MP5 If number of bids / highest bid = zero add bid (provided greater than zero)	**3**

Question	Answer	Marks
1(e)	Explanation of how the candidate's program performed the following. **Five** from: MP1 Method to search item arrays MP2 Conditional statement used to check for the successful bids MP3 … equal to / greater than reserve price MP4 Method used to calculate 10% percent of successful bid (for each item) MP5 …and add to auction company total fee MP6 Method used to display total fee with a suitable message All programming statements used must be explained.	**5**

Question	Answer	Marks
	Section B	
2(a)	1 mark for each error identified + suggested correction • `Low ← Count` should be `Low ← Number` • `Number > Low` should be `Number < Low` • `UNTIL Count = 99` should be `UNTIL Count > 99` or `UNTIL Count = 100` or `UNTIL Count >= 100` // `Count ← 1` should be `Count ← 0` • `PRINT "Largest Number is ", Number` should be `PRINT "Largest Number is ", High`	**4**

2(b)	MP1 Add `Total ← 0` // `Total ← Number`
	MP2 Add `Total ← Total + Number`
	MP3 Add `PRINT "Total is ", Total`
	MP4 All positioning explained / seen
	
	`Count ← 1`
	`INPUT Number`
	`High ← Number`
	`Low ← Number`
	`Total ← Number`
	`REPEAT`
	`  INPUT Number`
	`  Total ← Total + Number`
	`  IF Number > High`
	`  THEN`
	`  High ← Number`
	`  ENDIF`
	`  IF Number < Low`
	`  THEN`
	`  Low ← Number`
	`  ENDIF`
	`  Count ← Count + 1`
	`UNTIL Count > 99`
	`PRINT "Largest Number is ", High`
	`PRINT "Smallest Number is ", Low`
	`PRINT "Total is ", Total`
		4

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>3</td><td>
<table>
<tr><th>Total</th><th>Count</th><th>Distinction</th><th>Mark</th><th>OUTPUT</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>50</td><td></td></tr>
<tr><td>50</td><td>1</td><td>0</td><td>70</td><td></td></tr>
<tr><td>120</td><td>2</td><td>0</td><td>65</td><td></td></tr>
<tr><td>185</td><td>3</td><td>0</td><td>30</td><td></td></tr>
<tr><td>215</td><td>4</td><td>0</td><td>95</td><td></td></tr>
<tr><td>310</td><td>5</td><td>1</td><td>50</td><td></td></tr>
<tr><td>360</td><td>6</td><td>1</td><td>55</td><td></td></tr>
<tr><td>415</td><td>7</td><td>1</td><td>85</td><td></td></tr>
<tr><td>500</td><td>8</td><td>2</td><td>65</td><td></td></tr>
<tr><td>565</td><td>9</td><td>2</td><td>35</td><td></td></tr>
<tr><td>600</td><td>10</td><td></td><td>–1</td><td>Number of Distinctions 2</td></tr>
<tr><td></td><td></td><td></td><td></td><td>Average Mark 60</td></tr>
</table>
1 mark for Total and Count columns both correct.
1 mark for each correct column apart from Total and Count.
If no marks awarded allow 1 mark for initialisation of Total, Count and Distinction, set to zero.
</td><td>4</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>4</td><td>
<table>
<tr><th>Statements</th><th>Selection</th><th>Repetition</th></tr>
<tr><td><pre>FOR A ← 1 TO 100
   B ← B + 1
NEXT A</pre></td><td></td><td>✓</td></tr>
<tr><td><pre>CASE A OF
   100: B ← A
   200: C ← A
ENDCASE</pre></td><td>✓</td><td></td></tr>
<tr><td><pre>IF A > 100
   THEN
      B ← A
ENDIF</pre></td><td>✓</td><td></td></tr>
<tr><td><pre>REPEAT
   A ← B * 10
UNTIL A > 100</pre></td><td></td><td>✓</td></tr>
</table>
1 mark for each correct row
</td><td>4</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5</td><td>
Validation
Two from:
• automated checking
• checking that data is reasonable / of a certain type
• checking that data meets certain criteria

Example 1 mark
• range check // length check // type check // check digit etc.

Verification
Two from:
• checking that data has not changed…
• … during input to a computer
• … during transfer between computers / devices

Example 1 mark
• double entry // checking against original // visual check // use of checksum etc.
</td><td>6</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(a)</td><td>Passengers</td><td>1</td></tr>
<tr><td>6(b)</td><td>Explanation
Three from:
• Flight number not displayed
• Passengers displayed when should not be
• Departure time = not >
• "Freight and passengers" flight not excluded

Revised QBE – answers shown are examples only
1 mark per bullet
• correct field and table names (either 3 or 4 columns) must include Notes, Flight number and Departure time
• correct show
• correct time criteria for the candidate’s QBE grid
• use of criteria to select planes with passengers only

<table>
<tr><td>Field:</td><td>Flight number</td><td>Passengers</td><td>Departure time</td><td>Notes</td></tr>
<tr><td>Table:</td><td>FLIGHT</td><td>FLIGHT</td><td>FLIGHT</td><td>FLIGHT</td></tr>
<tr><td>Sort:</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☑</td><td>☐</td><td>☐</td><td>☐</td></tr>
<tr><td>Criteria:</td><td></td><td>=Y</td><td>>10:00</td><td><> "Freight and passengers"</td></tr>
<tr><td>or:</td><td></td><td></td><td></td><td></td></tr>
</table>
OR
<table>
<tr><td>Field:</td><td>Flight number</td><td>Departure time</td><td>Notes</td></tr>
<tr><td>Table:</td><td>FLIGHT</td><td>FLIGHT</td><td>FLIGHT</td></tr>
<tr><td>Sort:</td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☑</td><td>☐</td><td>☐</td></tr>
<tr><td>Criteria:</td><td></td><td>>10:00</td><td>= "Passengers only"</td></tr>
<tr><td>or:</td><td></td><td>>10:00</td><td>="Private passenger flight"</td></tr>
</table>
</td><td>7</td></tr>
</table>

© UCLES 2019