MODULE M PROGRAMMING · M10

Count-controlled loops (FOR): mark scheme

22 marks · OCR J277 spec 2.2.1 (the use of the three basic programming constructs: iteration, count- and condition-controlled loops; variables, operators, inputs, outputs and assignments), with 2.1.2 (trace tables, identifying common errors) for Q3 and Q5

How to use. 1 mark per bullet up to the Max shown. **/** separates alternative words; **//** separates alternative answers for the same mark. Words in (brackets) are not needed for the mark. **...** at the start of a bullet means it depends on the bullet before it. In code answers, ignore capitals and minor syntax slips; repeated slips are penalised once, then follow through. Allow any logically valid method. Credit any other valid answer.

QUESTION	ANSWER	MARKS	GUIDANCE
1(a) J277 2.2.1	A loop that repeats (a block of code) a set / fixed / known / given number of times	**1**	Accept: "a loop where the number of repetitions is known before the loop starts"; "a for loop that runs from a start value to an end value". Do not accept: "a loop that repeats" / "a loop that counts" alone; "repeats until a condition is met" (condition-controlled).
1(b) J277 2.2.1	Print 5 copies: count-controlled Ask for the password until correct: condition-controlled	**2**	1 mark per correct row. A row with both boxes ticked scores 0.
2 J277 2.2.1	`for k = 0 to 6`: 7 `for k = 5 to 20 step 5`: 4 `for k = 9 to 1 step -2`: 5	**3**	1 mark per correct row. Row 1: k takes 0, 1, 2, 3, 4, 5, 6 (both ends included). Do not accept 6. Row 2: k takes 5, 10, 15, 20. Do not accept 3 or 15. Row 3: k takes 9, 7, 5, 3, 1. Do not accept 4 or 8. Accept the answer written in words. Do not accept a list of the values instead of the number of times.

<table>
<tr><th>QUESTION</th><th>ANSWER</th><th>MARKS</th><th>GUIDANCE</th></tr>
<tr>
<td>3
J277 2.2.1
J277 2.1.2</td>
<td>week: 1, 2, 3, 4
sold: 3, 6, 9, 12
stock: 20, 17, 11, 2 (and no further change)
Output: Left: 2 (once, at the end)
Example completed table:
<table>
<tr><th>LINE</th><th>WEEK</th><th>SOLD</th><th>STOCK</th><th>OUTPUT</th></tr>
<tr><td>01</td><td></td><td></td><td>20</td><td></td></tr>
<tr><td>02</td><td>1</td><td></td><td></td><td></td></tr>
<tr><td>03</td><td></td><td>3</td><td></td><td></td></tr>
<tr><td>05</td><td></td><td></td><td>17</td><td></td></tr>
<tr><td>02</td><td>2</td><td></td><td></td><td></td></tr>
<tr><td>03</td><td></td><td>6</td><td></td><td></td></tr>
<tr><td>05</td><td></td><td></td><td>11</td><td></td></tr>
<tr><td>02</td><td>3</td><td></td><td></td><td></td></tr>
<tr><td>03</td><td></td><td>9</td><td></td><td></td></tr>
<tr><td>05</td><td></td><td></td><td>2</td><td></td></tr>
<tr><td>02</td><td>4</td><td></td><td></td><td></td></tr>
<tr><td>03</td><td></td><td>12</td><td></td><td></td></tr>
<tr><td>08</td><td></td><td></td><td></td><td>Left: 2</td></tr>
</table>
</td>
<td>4</td>
<td>1 mark for each correct set of changes to a variable, in the correct order, and 1 mark for the output.
Line numbers are not marked. Values may be repeated on later rows. Several changes may be written on one row.
In week 4, 12 is not less than 2, so line 05 does not run and stock stays 2. Do not award the stock mark if a further value (e.g. −10) is shown.
Ignore a final value of 5 for week.
Output: accept Left: 2 with or without speech marks; do not accept print(...) written around it or 2 alone.</td>
</tr>
<tr>
<td>4
J277 2.2.1</td>
<td>A for loop with start value 60 and end value 0 ...
... and step −10
Outputs the loop variable inside the loop and outputs Lights off once, after the loop
Example answer (ERL):
<pre>for count = 60 to 0 step -10
    print(count)
next count
print("Lights off")</pre>
</td>
<td>3</td>
<td>1 mark per bullet, max 3.
Allow any loop that gives the same output, e.g. for count = 6 to 0 step -1 with print(count * 10) (award bullets 1 and 2 for a correct loop of this kind), or Python for count in range(60, -1, -10):.
Do not award bullet 3 if Lights off is output inside the loop (it would be output 7 times).
Any form is accepted for this algorithm: ERL, a high-level language or a flowchart, as long as a count-controlled loop is used.</td>
</tr>
</table>

QUESTION	ANSWER	MARKS	GUIDANCE
5 J277 2.1.2 J277 2.2.1	Line 02 … `for pots = 1 to 6` Line 03 … `cost = price * pots`	**4**	1 mark for each line number and 1 mark for each matching correction. A correction is only credited with its correct line number. Line 02: the loop must run for `pots` = 1 to 6 inclusive. Do not accept `0 to 5` or `1 to 7` (these output the wrong numbers of pots). Line 03: accept `cost = pots * price`. Do not credit line 04 or 05 (they are correct as written).

6
J277 2.2.1
	Input the number of parcels and store it in a variable
	A `for` loop that runs that number of times (e.g. `1 to number` // `0 to number - 1`)
	Input a weight inside the loop
	Add each weight to a running total inside the loop …
	… output the total after the loop has finished
	Example answer (ERL):
	`parcels = int(input("How many parcels have arrived?"))`
	`total = 0`
	`for count = 1 to parcels`
	`  weight = float(input("Enter weight in kg"))`
	`  total = total + weight`
	`next count`
	`print("Total weight: " + str(total) + " kg")`
		5	1 mark per bullet, max 5.
			Loop: the end value must use the number entered, not a fixed number. Python `for count in range(parcels):` is accepted.
			Total: do not award bullet 4 if the total is reset to 0 inside the loop. An uninitialised total may be taken as 0.
			Output: do not award bullet 5 if the total is output inside the loop (it would be output after every parcel). A comma between the message and the total is allowed.
			Missing casts are not penalised. Any valid high-level language is accepted.

Total: 22 marks