MODULE M PROGRAMMING · M5

Arithmetic operators, including MOD, DIV and exponentiation: mark scheme

20 marks · OCR J277 spec 2.2.1 (the common arithmetic operators: + − * / MOD DIV ^), with 2.1.2 (trace tables)

How to use. 1 mark per bullet up to the Max shown. **/** separates alternative answers. Words in (brackets) are not needed for the mark. **...** means the second point depends on the first. Code answers are marked for logic, not exact syntax: any high-level language or OCR Exam Reference Language is accepted; penalise a repeated syntax slip once, then follow through.

QUESTION	ANSWER	MARKS	GUIDANCE
1(a) J277 2.2.1	`17 MOD 5`: 2 `17 DIV 5`: 3 `3 ^ 4`: 81 `9 / 2`: 4.5 `2 + 3 * 4`: 14	**5**	1 mark per correct row. Correct answer only. Do not accept: 3.4 or "3 r 2" for DIV; 12 for `3 ^ 4`; 4 or "4 r 1" for `9 / 2` (accept 4½); 20 for `2 + 3 * 4` (multiplication is done before addition).

2(a) J277 2.2.1	`minutes = totalSeconds DIV 60`
	`seconds = totalSeconds MOD 60`
	`area = side ^ 2`
		3	1 mark per row. Ignore case. Accept language equivalents: `//` for DIV, `%` for MOD, `**` for ^. Do not accept `/` in row 1 (it gives a decimal, not whole minutes) or `*` in row 3 (`side * 2` is not the area).

3(a) J277 2.2.1	4	**1**	Items 25, 50, 75 and 100 give a remainder of 0. Accept a list of the four messages.
3(b) J277 2.2.1	Inspect item 25	**1**	Ignore quotation marks. Accept "Inspect item25" (spacing). Do not accept "Inspect item 1" or "25" alone.
3(c) J277 2.2.1	24	**1**	Items 1 to 24 give a quotient of 0 (items 25 to 100 give 1 to 4). Correct answer only.

<table>
<tr><th>QUESTION</th><th>ANSWER</th><th>MARKS</th><th>GUIDANCE</th></tr>
<tr><td>4(a)
J277 2.2.1, 2.1.2</td><td>boxes: 3, 6, 0 (in order)
spare: 2, 0, 5 (in order)
Output: 3 boxes, 2 spare / 6 boxes, 0 spare / 0 boxes, 5 spare (all three, in order)
<table><tr><th>DAY</th><th>EGGS</th><th>BOXES</th><th>SPARE</th><th>OUTPUT</th></tr><tr><td>1</td><td>20</td><td>3</td><td>2</td><td>3 boxes, 2 spare</td></tr><tr><td>2</td><td>36</td><td>6</td><td>0</td><td>6 boxes, 0 spare</td></tr><tr><td>3</td><td>5</td><td>0</td><td>5</td><td>0 boxes, 5 spare</td></tr></table></td><td>3</td><td>1 mark per correct column (boxes, spare, Output). The day and eggs columns are not marked. Output: ignore quotation marks and small spacing slips; FT the candidate's own boxes and spare values. Do not accept 0.0 / decimals in boxes (DIV gives a whole number), or a blank for 0 in spare on day 2 or boxes on day 3.</td></tr>
<tr><td>4(b)
J277 2.2.1</td><td>7 boxes, 5 spare</td><td>1</td><td>Ignore quotation marks. Both values needed, the right way round.</td></tr>
<tr><td>5(a)
J277 2.2.1</td><td>1 mark each to max 5:
• Asks for / inputs the number of guests and stores it (as an integer)
• Uses DIV 8 (or equivalent) to find the number of full tables
• Uses MOD 8 (or equivalent) to check whether there are guests left over ...
• ... and adds 1 table only when the remainder is not 0
• Outputs the number of tables
• Calculates the empty seats correctly and outputs them, using either method:
empty = tables × 8 − guests
OR empty = 8 − (guests MOD 8) if guests MOD 8 ≠ 0, otherwise 0
<pre>Example answer (ERL):
guests = int(input("Number of guests: "))
tables = guests DIV 8
if guests MOD 8 != 0 then
    tables = tables + 1
endif
empty = tables * 8 - guests
print("Tables needed: " + str(tables))
print("Empty seats: " + str(empty))</pre></td><td>5</td><td>Allow any logically valid method, e.g. <code>tables = (guests + 7) DIV 8</code> gains bullets 2, 3 and 4; a loop that adds a table while seats < guests gains bullets 2 to 4 if correct. Accept <code>> 0</code> for <code>!= 0</code>. Using / with no rounding up: do not award bullet 2. Adding 1 table every time (so 40 guests give 6 tables): do not award bullet 4. Empty-seat answer that gives 8 instead of 0 when guests is a multiple of 8: do not award the last bullet. Do not accept × in place of * in code. Missing <code>int()</code>/casting: do not penalise. Flowcharts or written English: 0.</td></tr>
</table>

Total: 20 marks