MODULE O ROBUST PROGRAMS · O8

Planning tests with test data: mark scheme

22 marks · Cambridge 0478 syllabus 7.6 suggest and apply suitable test data (normal, abnormal, extreme, boundary); 7.1 testing stage of the program development life cycle; 7.8 identify errors and suggest corrections

How to use. One mark per bullet point unless stated, up to the max shown. **//** separates alternative answers for the same mark. Words in (brackets) are not needed for the mark. **Test data** must be an actual value, not a description ("a number in range" = 0). Use the 0478 definitions: **extreme** = the largest/smallest acceptable value; **boundary** = the largest/smallest acceptable value **and** the corresponding smallest/largest rejected value. Credit any other valid answer.

<table>
<tr><th>QUESTION</th><th>ANSWER</th><th>MARKS</th><th>GUIDANCE</th></tr>
<tr>
<td>1
0478 7.6</td>
<td>One mark for each correct row:
<table>
<tr><th>Test data</th><th>Normal</th><th>Abnormal</th><th>Extreme</th><th>Boundary</th></tr>
<tr><td>25</td><td>✓</td><td></td><td></td><td></td></tr>
<tr><td>40</td><td></td><td></td><td>✓</td><td>✓</td></tr>
<tr><td>9</td><td></td><td>✓</td><td></td><td>✓</td></tr>
<tr><td>forty</td><td></td><td>✓</td><td></td><td></td></tr>
</table>
</td>
<td>4</td>
<td>Each row must have every required tick and no wrong tick.
Row 40: Extreme and Boundary both needed (largest acceptable value). Allow Normal also ticked (40 is accepted data); Abnormal ticked = 0.
Row 9: Abnormal and Boundary both needed (it is the rejected value next to 10). Extreme or Normal ticked = 0.
Rows 25 and forty: one tick only.</td>
</tr>
</table>

QUESTION	ANSWER	MARKS	GUIDANCE
2 0478 7.6	Test 2: 20 // 45 Test 3: Abnormal // Boundary Test 4 data: 19 and 20 // 45 and 46 Test 4 outcome: 19 rejected (`Pressure out of range`) and 20 accepted // 45 accepted and 46 rejected (`Pressure out of range`) Test 5: any value that should be rejected, e.g. `thirty` // 50 // −5 // 18 // 25.5	**5**	One mark per bullet point. Test 2: 20 or 45 only. Test 3: accept abnormal or boundary (46 is the rejected value of the 45/46 boundary pair). Do not accept extreme / normal. Test 4: the two values must be a pair across one edge (one accepted, one rejected). The outcome mark needs the correct outcome for **each** value; FT the outcome for a wrong pair only if each value's outcome is correct for this check. The message does not need to be repeated. Test 5: 46 and 19 are also abnormal; accept them. Do not accept a value from 20 to 45.
3 0478 7.6	A length from 2.0 to 5.0 (e.g. 3.5) … `Fare is 30` A length above 5.0 and up to 12.0 (e.g. 8.0) … `Fare is 55` A length above 12.0 and up to 20.0 (e.g. 15.0) … `Fare is 90`	**3**	One mark per row: the length must be in the correct range **and** the output must match it. Rows in any order; each fare tested once for its mark. Accept 5.0 → 30 and 12.0 → 55. Accept the output as 30 / 55 / 90 alone. Do not accept a length below 2.0 or above 20.0 (it never reaches this algorithm).
4(a) 0478 7.6	Test 2: Rejected // asked to enter the number of seats again Test 3: Rejected // asked to enter the number of seats again Test 4: Accepted // outputs `Booking for 8 seats`	**3**	One mark per row. Test 2 must follow the algorithm as written: 1 > 1 is FALSE, so the loop repeats. "Accepted" for test 2 = 0. Accept "loop repeats" / "prompt shown again" for rejected.

QUESTION	ANSWER	MARKS	GUIDANCE
4(b) 0478 7.8	Line 04 `UNTIL Seats >= 1 AND Seats <= 8` // change `Seats > 1` to `Seats >= 1`	**2**	One mark for the line number, one for a correct correction. Accept `Seats > 0` (whole number of seats) and `1 <= Seats`. Accept a correct description, e.g. "change > to >=". Do not accept ≥ or =>. Do not accept a correction that also lets 0 or 9 through.
4(c) 0478 7.6	4 is normal data // 4 is inside the range, away from the edge, so it is accepted whether the test is > or >= The error only affects the smallest acceptable value // only extreme / boundary data (1) finds it	**1**	Any one point. Do not accept "test 1 passed" alone (NE: needs why).
5 0478 7.6	No extreme data … add a password of exactly 8 // exactly 20 characters, expected to be accepted No boundary data … add a pair such as 7 and 8 characters (rejected, accepted) // 20 and 21 characters (accepted, rejected) No abnormal data that is too long … add a password of more than 20 characters, e.g. 25 characters, expected to be rejected Expected outcome "OK" (test 2) is vague // outcomes do not say what happens … write "Accepted" // "Rejected, error message, re-enter password" Tests 1 and 2 test the same thing (both normal) … replace one with extreme / boundary / long abnormal data	**4**	Mark in pairs: one mark per problem and one for a matching change or extra test, max two pairs. Award a problem on its own if the change is missing or wrong. For a length check, the test data may be given as an actual string or as an exact number of characters (e.g. "a 20-character password"); "a long password" = 0. Do not accept "not enough tests" alone. Do not award the same problem twice.

Total: 22 marks