MODULE L STANDARD ALGORITHMS · L1

Linear search: mark scheme

21 marks · Cambridge 0478 syllabus 7.4 (standard methods: linear search), 7.7 (trace tables), 7.9 (write and amend algorithms)

How to use. One mark per creditworthy point up to the maximum shown. **/** separates alternative words; **//** separates alternative answers. Words in (brackets) are not needed for the mark. Algorithm answers must be in pseudocode; marks are for the features listed, not for one exact algorithm. Credit any other valid answer that shows the same understanding.

QUESTION	ANSWER	MARKS	GUIDANCE
1(a) 0478 7.4	Any **three** from: • starts at the first element // index 1 • compares each element in turn with the value being searched for • if the element matches, the value is found // the search stops // the position is output • if it does not match, moves on to the next element • repeats until the value is found // until the end of the array is reached • if the end of the array is reached with no match, the value is not in the array // "not found" is output	**3**	"It goes through the array" alone: 0. Do not credit swapping elements (bubble sort) or checking the middle element. Credit any other valid step of a linear search.
1(b) 0478 7.4	40	**1**	Accept "all 40". Do not accept "all of them" with no number.
2(a) 0478 7.4	507, 213, 880 (in this order) ... ... then 145, and no further values	**2**	Order must be correct. Any value after 145 listed (e.g. the second 213): max 1.
2(b) 0478 7.4	4	**1**	Do not accept 3 (the array starts at index 1).
3(a) 0478 7.7, 7.4	One mark for each correct column: • Index: 1, 2, 3, 4, 5, 6 • Position: 0, 5 • OUTPUT: Found at 5	**3**	Values must be in the order shown. Ignore repeated values on later rows and the exact row each value is on. Index stops at 6: do not credit a 7 in the Index column. OUTPUT: ignore quotation marks; must be the only output.
3(b) 0478 7.7	Not found	**1**	Ignore quotation marks.

QUESTION	ANSWER	MARKS	GUIDANCE
3(c) 0478 7.7	6	**1**	Every element is compared once before the loop ends with Index > 6.

4(a)
0478 7.4, 7.9
	One mark per bullet point. Max 6:
	• input of a product code into a variable (prompt not needed)
	• index set to 1 // flag set to FALSE before the loop
	• loop through the array from 1 to 200 with the index increased inside the loop
	• comparison of `Code[Index]` with the input code
	• loop stops as soon as the code is found // condition-controlled loop using a flag or the comparison
	• output of `Price[Index]` for the matching index when found
	• suitable message output only when the code is not found (after the whole array has been checked)
	Example answer:
	`OUTPUT "Enter the product code"`
	`INPUT SearchCode`
	`Found ← FALSE`
	`Index ← 1`
	`WHILE Index <= 200 AND Found = FALSE DO`
	`  IF Code[Index] = SearchCode`
	`  THEN`
	`  Found ← TRUE`
	`  ELSE`
	`  Index ← Index + 1`
	`  ENDIF`
	`ENDWHILE`
	`IF Found = TRUE`
	`  THEN`
	`  OUTPUT "Price: ", Price[Index]`
	`  ELSE`
	`  OUTPUT "Product code not found"`
	`ENDIF`
		6	Allow REPEAT ... UNTIL or WHILE ... ENDWHILE. A FOR loop that always runs to 200 cannot get the "stops as soon as found" mark. Not-found message output inside the loop for every non-matching element: no mark for that bullet. Output of the code or the index instead of the price: no mark for that bullet. Program code (not pseudocode): 0. Ignore missing DECLARE statements.

5(a) 0478 7.9, 7.4	Any **three** from: • change the loop so that it always runs to the end of the array // remove `Position <> 0 OR` from line 10 so it is `UNTIL Index > 6` // replace the REPEAT loop with `FOR Index ← 1 TO 6` • output `Index` inside the IF (after line 07), each time a match is found • keep `Position` (or a flag / count) set when any match is found, so "Not found" is output after the loop only if there was no match • remove the ELSE output on line 15, so an index is not output twice	**3**	Each change must be explained, not just written as code. Outputting Index after line 09 (when it has already been increased): do not credit that bullet. Credit any other valid change that meets the requirement.

Total: 21 marks