MODULE L STANDARD ALGORITHMS · L5

Insertion sort: mark scheme

20 marks · OCR J277 spec 2.1.3 (searching and sorting algorithms: insertion sort)

How to use. 1 mark per bullet up to the Max shown. **/** separates alternative answers. Words in (brackets) are not needed for the mark. Candidates do not need to remember the code for insertion sort (J277 2.1.3): credit understanding of the steps, not recall of code. Accept answers that describe swapping the item left one place at a time instead of shifting the larger items right: both are valid ways of inserting the item.

QUESTION	ANSWER	MARKS	GUIDANCE
1(a) J277 2.1.3	A sorted part (at the start / left) **and** an unsorted part (the rest)	**1**	Both parts needed for the mark. Accept sub-list / section / partition for part. Do not accept "left half and right half" (the parts are not halves).
1(b) J277 2.1.3	The first item is treated as sorted / starts as the sorted part Take the next item (from the unsorted part) / start from the second item Compare it with the items in the sorted part (working from right to left / from the end of the sorted part) Move / shift each larger item one place to the right Insert the item into the gap / its correct position (when a smaller or equal item, or the start of the list, is reached) Repeat for every remaining item / until the unsorted part is empty / all items are inserted	**3**	Max 3. Accept "swap it with the item to its left while that item is larger" for the shift and insert points. Do not accept "it sorts the list" / "it puts the items in order" alone (outcome, not steps). Do not accept descriptions of comparing adjacent pairs through the whole list (bubble sort) or splitting the list (merge sort).
2(a) J277 2.1.3	After insertion 1: 18, 31, 42, 7, 25 After insertion 2: 18, 31, 42, 7, 25 After insertion 3: 7, 18, 31, 42, 25 After insertion 4: 7, 18, 25, 31, 42	**4**	1 mark per correct row. Row 2 is unchanged because 42 is already larger than every item in the sorted part. Follow through: if one row is wrong, award a later row if it correctly inserts the next item into the candidate's own previous row. A row that is just the fully sorted list too early (e.g. row 1 = 7, 18, 25, 31, 42) scores 0 for that row.
3(a) J277 2.1.3	Iris	**1**	The first item of the unsorted part.

QUESTION	ANSWER	MARKS	GUIDANCE
3(b) J277 2.1.3	Iris is compared with Mint, which comes after it (alphabetically) so Mint moves one place right / Iris and Mint swap Iris is then compared with Fern, which comes before it, so the comparisons stop Iris is inserted between Fern and Mint / list becomes Aster, Fern, Iris, Mint (Basil, Tulip)	**2**	Max 2. Must refer to the items in this list for full marks; a correct generic description of inserting an item: Max 1. Follow through a wrong answer to 3(a) if the description correctly inserts that item.
3(c) J277 2.1.3	Aster, Basil, Fern, Iris, Mint, Tulip	**1**	All six items in this order. Separators (commas / spaces) not important.
4(a) J277 2.1.3	Insertion sort	**1**	Do not accept bubble sort or merge sort.
4(b) J277 2.1.3	(Line) 09 / 9	**1**	Line number only. Accept the line copied out correctly.
4(c) J277 2.1.3	Loops while the item to the left (`times[j]`) is greater than `current` / and the start of the list has not been passed (j is not below 0) Moves / shifts / copies that item one place to the right Moves left / decreases `j` by 1 to check the next item in the sorted part (So that) a gap is made / found for `current` in its correct position	**2**	Max 2. Do not accept "sorts the list" alone. Do not accept "swaps" pairs of items through the whole list (bubble sort). Do not accept "moves the item to the left" for line 06 (the larger item moves right).
4(d) J277 2.1.3 (J277 2.1.2 trace tables)	`i`: 1, 2, 3 `current`: 35, 52, 29 `j`: 0, −1, 1 (while i is 1 and 2) `j`: 2, 1, 0, −1 (while i is 3) Example rows (i, current, j): (1, 35, 0); (, , −1); (2, 52, 1); (3, 29, 2); (, , 1); (, , 0); (, , −1)	**4**	1 mark per bullet: each variable's values in the correct order. Values repeated on later rows are fine; row positions and line numbers are not marked. 52 is not moved, so j is only 1 while i is 2. Do not accept 0 in place of −1 for the final j values. Ignore a correct extra column showing the contents of times.

Total: 20 marks