MODULE L STANDARD ALGORITHMS · L4

Merge sort: mark scheme

20 marks · OCR J277 spec 2.1.3

How to use. 1 mark per bullet up to the maximum shown. **/** separates alternative answers; **//** separates alternative statements for the same mark. Words in (brackets) are not needed for the mark. **...** means the mark depends on the one before. Credit any other valid answer that shows the same understanding.
Spec points (J277 2.1.3). Standard sorting algorithms: merge sort. Understand the main steps of the algorithm and the segments of code in it; apply the algorithm to a data set; identify an algorithm if given the code or Exam Reference Language for it. Candidates do **not** need to remember the code for merge sort: Q5 gives the code and asks only what it does.

QUESTION	ANSWER	MARKS	GUIDANCE
1(a) J277 2.1.3	Only one item, already sorted: True Data must already be in order: False Joined end to end, then sorted: False	**3**	1 mark per correct row. Two ticks in a row = 0 for that row. Row 2: being in order first is a pre-requisite of binary search, not merge sort. Row 3: the merging itself keeps the items in order; nothing is sorted afterwards.
2(a) J277 2.1.3	(Repeatedly) split / divide the list (in half) ... until each list has one item (Repeatedly) merge pairs of lists (back together) in order // compare the first items of two lists and move the smaller into a new list ... until there is one (sorted) list	**2**	1 mark per stage. Each stage needs its stopping point (one item each // one list) **or** the idea of repeating. Accept "divide" and "conquer" as names only if the description is given. "Split the list" alone = NE. "Merge the lists" alone = NE (must be in order / sorted). Do not accept merging and then sorting the merged list.

QUESTION	ANSWER	MARKS	GUIDANCE
3(a) J277 2.1.3	Split into individual lists of one item each: 9 | 27 | 3 | 14 | 21 | 5 | 11 | 18 Merge into sorted lists of 2: [27, 9] [14, 3] [21, 5] [18, 11] Merge into sorted lists of 4: [27, 14, 9, 3] [21, 18, 11, 5] ... Merge into one list: [27, 21, 18, 14, 11, 9, 5, 3]	**4**	1 mark per bullet. Splitting may be shown in stages (4 + 4, then 2 + 2 + 2 + 2, then 1s) or all at once. If no splitting is shown, assume the list starts as single items and mark BP2 to BP4. Lists must clearly be the right size (boxes, brackets or gaps). Final mark only after a reasonable attempt at the merges; the sorted list on its own = 0. Do not award a merge mark where lists are merged out of order and then sorted. A different algorithm (e.g. bubble sort) = 0. If every merge is correct but in ascending order, penalise once: max 3.
4(a) J277 2.1.3	2: Heron and Crane ... Crane 3: Heron and Fox ... Fox 4: Heron and Stoat ... Heron	**3**	1 mark per row: both items compared **and** the item moved must be correct. The two items may be written in either order. Rule: compare the front item of each list and move the one that comes first alphabetically.
4(b) J277 2.1.3	Badger, Crane, Fox, Heron, Otter, Stoat, Vole, Wren	**1**	All eight in this order. Ignore capitals and the type of separator.
5(a) J277 2.1.3	Merge sort	**1**	Do not accept bubble sort, insertion sort, or "merge" alone without "sort" being clear (BOD "a merge" if the candidate names the sort elsewhere).
5(b) J277 2.1.3	They run when one of the lists has run out / all its items have been moved // the first loop has ended ... they copy / move the remaining items of the other list onto the end of `result` The remaining items are already in order, so no comparisons are needed // they keep their order	**2**	1 mark per bullet, max 2. "Copies the rest of left, then the rest of right" = 1 mark (BP2). "Adds the leftover items" = BP2. "Loops through the lists" alone = 0. Only one of the two loops actually runs; do not penalise a candidate who does not say so.

QUESTION	ANSWER	MARKS	GUIDANCE
5(c)(i) J277 2.1.3	[2, 4, 9, 10, 15]	**1**	All five values in this order. Ignore brackets, quotes and separators.
5(c)(ii) J277 2.1.3	4	**1**	Correct answer only. (4 v 2, 4 v 10, 9 v 10, 15 v 10; then `right` is used up and 15 is copied by lines 16 to 20 without a comparison.)
6(a) J277 2.1.3	Quicker / more efficient (than bubble sort) for a large list // takes fewer comparisons / steps on a large list Time taken grows much more slowly than bubble sort's as the list gets longer Its speed does not depend much on the starting order of the list // always takes about the same time	**1**	Any one. "Faster" alone is BOD here because the question states the list is large. Do not accept "uses less memory" (wrong) or "it sorts the list" (true of both).
6(b) J277 2.1.3	Uses more memory // needs extra space for the split / merged lists // does not sort in place More complex / harder to write (code) // harder to understand / debug Can be slower on a small list or a list that is already (nearly) in order	**1**	Any one. Do not accept "slower" alone, or "slower for large lists" (wrong). Credit any other valid answer.

Total: 20 marks