Merge sort

Algorithms and programming (Paper 2) · revision notes

A merge sort splits a list in half again and again until each part holds one item, then merges the parts back together in order. You need to show the splitting and merging stages on a given list, describe how two lists are merged, and compare merge sort with bubble and insertion sort.

Key facts

- Divide: split the list in half repeatedly until every list has 1 item
- Merge: compare the front items of two lists and move the smaller one
- Keep merging pairs of lists until one sorted list is left
- Con: extra memory, harder to code, no quicker on small or already sorted lists
- A list of one item is already sorted
- When one list runs out, copy the rest of the other list across
- Pro: fast and efficient on large lists

Notes

How a merge sort works

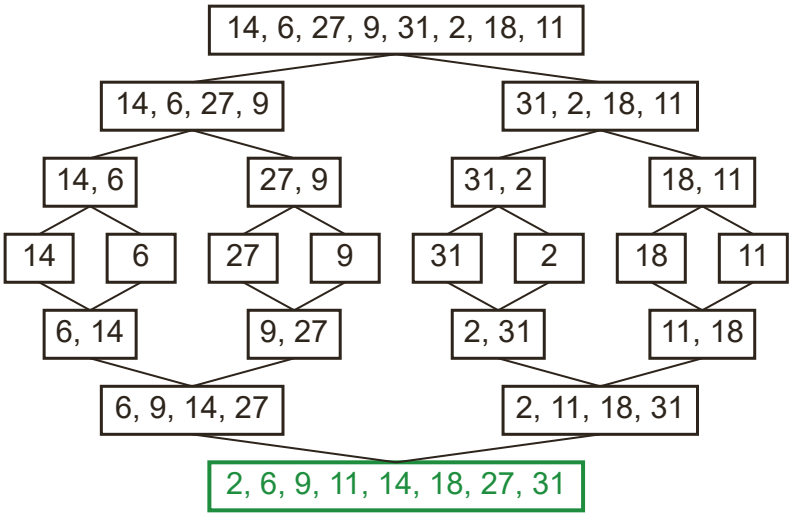
- **Divide**: split the list in half, then split each half in half, and keep going until every sub-list has one item. A list of one item is already sorted.
- **Merge**: merge pairs of sub-lists into sorted lists of two, then fours, and so on, until there is one sorted list.
- If a list has an odd number of items, one half gets one more item than the other. Follow any rule the question gives about which side.
- To recognise it in code: the list is split at a middle position into a left half and a right half, then a loop compares the front items of two lists and adds the smaller to a new list.

How two sorted lists are merged

- Compare the first item of each list. Move the smaller one into the new list.
- Compare the new front items and repeat.
- When one list is empty, copy the rest of the other list across in order (it is already sorted).
- Example: merging 4, 10, 15 with 3, 11, 12, 20 gives 3, 4, 10, 11, 12, 15, 20.

Worked example: 14, 6, 27, 9, 31, 2, 18, 11

- Split: [14, 6, 27, 9] [31, 2, 18, 11], then [14, 6] [27, 9] [31, 2] [18, 11], then eight single items.
- Merge into pairs: [6, 14] [9, 27] [2, 31] [11, 18].
- Merge into fours: [6, 9, 14, 27] [2, 11, 18, 31].
- Final merge: [2, 6, 9, 11, 14, 18, 27, 31].



Merge sort: split down to single items (top half), then merge back in order (bottom half)

Advantages and disadvantages

- Advantage: much more efficient (quicker) than bubble sort or insertion sort for large lists, because it makes far fewer comparisons.
- Advantage: it takes a similar time whatever the starting order of the list.
- Disadvantages: it needs extra memory to hold the sub-lists; it is more complex to code; it still does every split and merge even if the list is already sorted, and for small lists it can be slower than the simpler sorts.

How to answer each type of question

Show the stages of a merge sort 3 to 4 marks **Grade 6**

1. Split the list into halves, then quarters, down to single items. Show each level.
2. Merge neighbouring single items into sorted pairs.
3. Merge the pairs into sorted fours, and so on.
4. Keep each list in square brackets so the examiner can see the groups.

Example. Show how a merge sort would put these names into alphabetical order: Tom, Ann, Raj, Eve, Kim, Bo, Zac, Liv. [4]

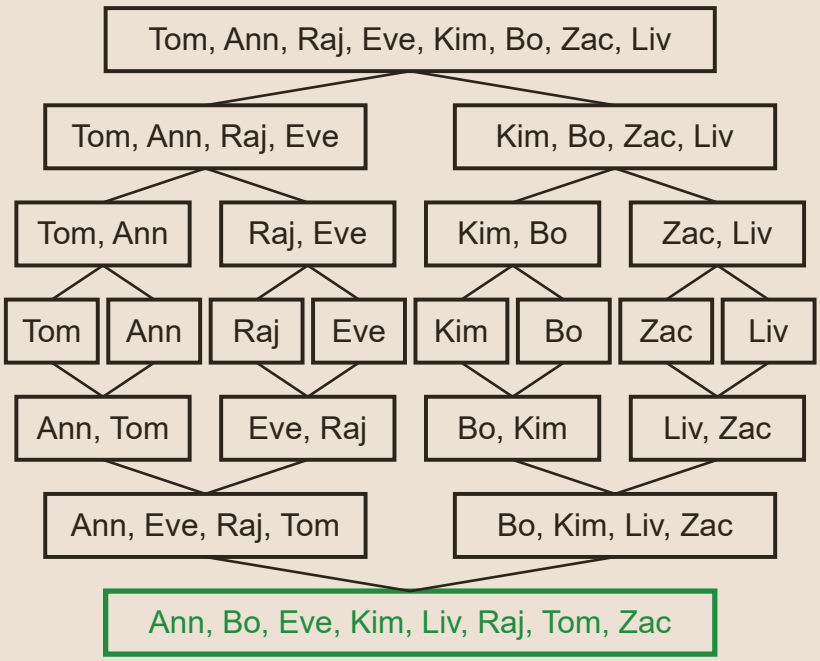
Model answer.

Split: [Tom, Ann, Raj, Eve] [Kim, Bo, Zac, Liv], then [Tom, Ann] [Raj, Eve] [Kim, Bo] [Zac, Liv], then single items (1).

Merge: [Ann, Tom] [Eve, Raj] [Bo, Kim] [Liv, Zac] (1).

[Ann, Eve, Raj, Tom] [Bo, Kim, Liv, Zac] (1).

[Ann, Bo, Eve, Kim, Liv, Raj, Tom, Zac] (1).



Describe how two lists are merged

2 to 3 marks | Grade 5

1. Compare the first item in each list.
2. Move the smaller into the new list and repeat with the next front items.
3. When one list is empty, add the rest of the other list.

Example. During a merge sort, the lists [4, 10, 15] and [3, 11, 12, 20] are merged.
Describe how they are merged and give the merged list. [3]

Model answer. Compare the first item of each list and move the smaller one into the new list (1).
Repeat with the front items until one list is empty, then add the remaining items of the other list (1).
Merged list: 3, 4, 10, 11, 12, 15, 20 (1).

Evaluate merge sort for a scenario

2 to 4 marks | Grade 7

1. Link the size of the data to efficiency: merge sort suits large lists.
2. Give the reason: far fewer comparisons than bubble or insertion sort.
3. If asked for a drawback, mention extra memory or complexity.

Example. An online shop sorts 2 million product prices every night.
Explain why a merge sort is a suitable choice, and give **one** drawback. [3]

Model answer.

Merge sort is much quicker (more efficient) than a bubble sort or an insertion sort on a large list (1) because it needs far fewer comparisons (1).
Drawback: it needs extra memory to store the sub-lists / it is more complex to program (1).

Don’t lose marks

- Merging before the list has been split all the way down to single items.
- Sorting a whole group in one step instead of merging two groups.
- Losing or duplicating an item: count the items at every stage.
- Saying merge sort uses less memory than bubble sort: it uses more.
- Describing a merge as 'join the lists together' without comparing the front items.

More tips

Memory tricks

- Merge sort in two words: split, then merge.
- Sense check: at every merging level, each group must be in order and the total number of items must not change.
- Merging is like two queues of people sorted by height: you always take the shorter of the two people at the front.

Exam technique

- Show every level of splitting and merging on its own line; each level is usually a mark.
- Use brackets to show where each sub-list starts and ends.
- In 'which sort' questions, mention the size of the data and memory, and link them to the scenario.

What each grade needs

What you need to be able to do, from the first marks up to the top grade.

Grade 4

Describe the splitting stage of merge sort

The list is repeatedly divided in half until every sub-list contains one item.

Grade 5

Merge two sorted lists into one

Compare the front item of each list and move the smaller one across, repeatedly.

Grade 6

Show every stage of a merge sort

Show all the splits down to single items, then each round of merging.

Grade 7

Give advantages and disadvantages of merge sort

Much faster on large lists, but uses more memory and is harder to code.

Grade 8

Choose a suitable sort for a scenario

Justify merge sort for large data sets, or another sort when memory or simplicity matters.

Free notes, Full packs, practice papers and mock exams at chhetriacademy.com. Written for this site; not taken from past papers. Stuck? 1-to-1 tutoring with a free 30-minute trial lesson.