

Exercise walk-through

Recursive Searching and Sorting ■ 4.17

eXercise

You must be able to

- describe how **binary search** can be written as a **recursive** method
- trace a recursive binary search halving the range on each call
- describe how **merge sort** 归并排序 splits, sorts, and merges

Method — Recursive binary search

Binary search is naturally recursive: check the **middle**; if the target is smaller, **recurse on the left half**, else recurse on the right — each call halves the range.

Method — Merge sort

- 1 **Split** the collection into two halves.
- 2 **Sort each half** (recursively, using merge sort).
- 3 **Merge** the two sorted halves into one sorted collection.

Practice 2.1 ■ 1 mark

State how binary search can be written recursively.

Solution 2.1

Method: Recursive binary search

Worked steps

it checks the middle element, then calls itself on the half that could contain the target **B1**.

Practice 2.2 ■ 2 marks

State the three steps of merge sort.

Solution 2.2

Method: Merge sort

Worked steps

split into two halves; sort each half; merge them **B1** **B1** *for any two; for all three.*

Practice 2.3 ■ 1 mark

State what merge sort does after sorting the two halves.

Solution 2.3

Method: Merge sort

Worked steps

it merges the two sorted halves into one sorted list **B1**.

Exam-style 3.1 ■ 1 mark

Merge sort works by

- A** checking each element in turn
- B** splitting, sorting each half, and merging
- C** swapping neighbouring elements only
- D** not sorting at all

Solution 3.1

Method: Merge sort

- A checking each element in turn
- B splitting, sorting each half, and merging
- C swapping neighbouring elements only
- D not sorting at all



Worked steps

B.

Exam-style 3.2 ■ 1 mark

A recursive binary search calls itself on

- A** the whole list again
- B** the correct half of the list
- C** a random half
- D** nothing

Solution 3.2

Method: Recursive binary search

- A the whole list again
- B the correct half of the list
- C a random half
- D nothing



Worked steps

B.

Exam-style 3.3 ■ 1 mark

Merge sort is applied to a list.

- (a) State the first step.
- (b) State what happens to each half.
- (c) State the final step.

Solution 3.3

Method: Merge sort

Worked steps

(a) split the list into two halves **B1**. (b) each half is sorted (recursively) **B1**. (c) merge the two sorted halves **B1**.