AP COMPUTER SCIENCE PRINCIPLES • EXERCISE SHEET

3.11 Binary Search

Name: _____ Class: _____ Date: _____ **Total: 8 marks**

Objective

Build the skills to answer exam questions on **binary search**.

You must be able to:

- explain how **binary search** 二分查找 locates a target in a **sorted** list
- describe how each comparison **halves** 减半 the remaining portion
- state the requirement that the data must be **sorted** 已排序
- compare binary search with a linear search

1 Worked examples

Study these first. Each one shows the method for a question type used later.

> ■ **Binary search**
>
> On a **sorted** list, look at the **middle** element. If the target is smaller, search the **left** half; if larger, search the **right** half. Each comparison **halves** the items still to check.
>
> ■ **Requirement**
>
> The list **must be sorted** first, or binary search fails.
>
> ■ **Versus linear search**
>
> A linear search checks each element in turn; binary search is far faster on large sorted lists (about $\log_2 n$ steps instead of n).

2 Practice

2.1 State the requirement a list must meet for binary search to work. [1]

2.2 State what each comparison in a binary search does to the remaining items. [1]

2.3 State one advantage of binary search over linear search. [1]

---

3 Exam-style questions

3.1 Binary search requires the list to be [1]

- **A** sorted
- **B** empty
- **C** reversed
- **D** in random order

3.2 Each step of a binary search [1]

- **A** doubles
- **B** halves
- **C** ignores
- **D** shuffles

the remaining items to search.

3.3 A sorted list has 16 items.

(a) Name the search method that halves the list at each step. [1]

(b) State roughly how many steps it needs in the worst case ($16 \to 8 \to 4 \to 2 \to 1$). [1]

(c) State whether binary search works on an **unsorted** list. [1]

4 Go further

- work through the **3.11 Binary Search** lesson on the **Learn** page;
- read the **Algorithms and Programming** section of the AP Computer Science Principles handout on the **Know** page.

Solutions

2.1 it must be sorted.

2.2 it halves them (discards one half).

2.3 it is much faster on large sorted lists (fewer comparisons).

3.1 A.

3.2 B.

3.3 (a) binary search. (b) about 4 steps. (c) no.