

2.3 ALGORITHMS · 2.3.1(F)

Linear & binary search

How each works, complexity & when to use them. Spec 2.3.1(f).

01 Linear search

How Check each item from the start.

Sorted? No — works on any list.

Worst case Checks all $n \rightarrow O(n)$.

Best for Small or unsorted lists.

02 Binary search

How Halve the range each step (low/mid/high).

Sorted? Yes — required.

Worst case Halves each step $\rightarrow O(\log n)$.

Best for Large sorted lists.

03 Binary search steps

mid $(low + high) \text{ DIV } 2$

=target FOUND.

bigger $low = mid + 1$

smaller $high = mid - 1$

stop found, or $low > high$ (absent).

04 Choosing

Unsorted Linear (or sort first if searched often).

Small Linear is simplest.

Large + sorted Binary wins easily.

Trace Columns: low · high · mid · list[mid].

FINAL PASS BEFORE THE EXAM

Rapid exam tips

Six slips on search questions.

01

Binary search needs a **sorted** list — always state it.

02

bigger → $low=mid+1$; smaller → $high=mid-1$.

03

Linear = $O(n)$; binary = $O(\log n)$. Don't swap.

04

Show the trace columns: **low, high, mid, list[mid]**.

05

Binary search stops when found or $low > high$ (not present).

06

Unsorted/small → linear; large & sorted → binary.