

2.3 ALGORITHMS · 2.3.1(E)

Graph traversals: DFS & BFS

Visiting every node — depth-first vs breadth-first. Spec 2.3.1(e).

01 Depth-first (DFS)

How Go **deep**, then backtrack at dead ends.

Uses A **stack** (or recursion).

Order (A) A B D E F C

For Paths, cycles, mazes, topological sort.

02 Breadth-first (BFS)

How Explore **layer by layer**.

Uses A **queue**.

Order (A) A B C D E F

For **Shortest path** (unweighted), crawlers.

03 Method & key facts

Visited Mark nodes visited — **never revisit**.

Order Take neighbours in the **stated order** (often alphabetical).

Shortest **BFS** finds fewest-edge path (unweighted).

Trees OCR examines tree depth-first as **post-order**.

FINAL PASS BEFORE THE EXAM

Rapid exam tips

Six slips on traversal questions.

01

DFS = **stack** (deep); BFS = **queue** (wide). Don't swap.

02

BFS finds the shortest unweighted path — not DFS.

03

Mark nodes **visited**; never add one twice.

04

Take neighbours in the **order given** for a consistent answer.

05

DFS uses: paths · cycles · mazes (backtracking).

06

Track the **stack/queue** as you go — the order falls out.